

Bug Fixes

ID : 958**Fixed in version :** 3.2.0**Short Description:** Selection of a strike-through formatted text

Full Description: if you select a text which was formatted as strike-through, the strike-through line is displayed in the original font color and the letters are displayed in a selection color.
This behaviour is different to underline formatted text and selection on a MAC. On a MAC the whole selection including the strike-through line is displayed in the original color.

Comments :**ID :** 1193**Fixed in version :** 3.1.6**Short Description:** Strange behaviour loading RTF data

Full Description: We have a situation where we're saving data from an oWrite field into RTF and later on need to load (a possibly modified) copy back into the system.

Works pretty good except when bullet points are used.

Please see attached example and open the window wTest. You'll see some RTF text above and an oWrite field with two bullet points.

Hit the reload button (it gets the RTF and loads the RTF into the oWrite field) and you will see that three things happen:

- 1) the second bullet point loses the bullet
- 2) the first bullet point gets a couple of additional spaces in front
- 3) the font table gets a new entry for the same font

Hit reload a couple of times and you end up with an interesting result.

Comments : We have resolved issues 1 and 2. Issue 3 was caused by specifying an invalid font name for the standard \$font property. This appears to be a problem that appears to only show itself with Studio version 4.3.2.1. This issue does not occur with Version 4.3.2.0 or any version 5 or 6.

ID : 1198**Fixed in version :** 3.1.7**Short Description:** Owrite crashes converting to RTF

Full Description: This may be something inside the document but I have an oWrite document that when I convert it to RTF crashes Omnis.

I've attached a sample library, open wTest and push the button and for me it crashes.

Any idea what could cause this?

P.S. it does not crash with oWrite 3.1.3

Comments : It appears the first cell in the table has \$curalign set to -1. This is not valid and the RTF exporter complains about it. Normally that would not cause too much of a problem, except when it happens inside a table cell, it appears.

We have changed the RTF exporter to ignore invalid alignments and export them as left justified (value 0).

ID : 1200

Fixed in version : 3.1.7

Short Description: OWrite will not load

Full Description: We have identified a issue with the 3.1.5 version of Owrite. The component will not load on our customers 2008 Standard Server. We know this because \$components.OWrite.\$version returns NULL and the Owrite control shows with a black background.

We spent several days troubleshooting the system thinking it was something else causing the problem. Anti-Virus, Backup Software, etc. Nothing worked. On a whim we tried loading Owrite 3.1.3 and it loads fine.

We downloaded a new copy of 3.1.5 to be sure that we did not get a bad copy. Installing it on this system goes back to the problem. We have tested it on our own 2008 server and it works fine. The computer that shows the problem has a Xeon CPU (E1220) while the one that works has a i5 (650). Both have the 64 bit version of Windows 2008.

We have loaded 3.1.3 back on the customers system and it is working.

We have not seen this behaviour on any other computer so far.

Comments : This problem was caused by us using new compilers for building Studio 4.3 components which introduced a reliance on MS Windows distributable DLLs that are not distributed by the Studio 4.x installers. We have now reverted back to our previous compilers for DLLs intended for Studio 4.3.

ID : 1202

Fixed in version : 3.1.8

Short Description: rtf inserting with wrong font

Full Description: if i am inserting a text from rtf, owrite Change the font. In the rtf example the font 1 is Arial. But after Import the selection font is an Adobe font. And this font doesn't exist on this Computer.

Comments : This issue was caused by an unexpected font table structure. Typically, most font table information in RTF provides the font number, family token followed by the font name. The provided RTF was missing the font family token.

We expected this
{\fonttbl{\f0\froman Times New Roman;}{\f1\froman Arial;}}

but encountered this
{\fonttbl{\f0 Times New Roman;}{\f1 Arial;}}

However, as other word processors had no difficulty in reading this information, we have updated OWrite to allow for this.

ID : 1205

Fixed in version : 3.2.0

Short Description: Strange wrap-around behaviour

Full Description: I think we talked about this before a loong time ago but finally managed to make a reproducible example for you.

Just open wTest in the library. In its \$construct it does 4 things:

- 1) loads and converts a large image
- 2) add a line of text to a blank document
- 3) add in the picture on an new line
- 4) add in another line of text on a new line

The picture gets resized automatically to the page width, this problem only happens if the picture is the width of the page.

oWrite wrapping logic runs and wraps the picture as a result leaving a blank space the height of the picture below the picture.

If you have a smaller picture but you add text next to the picture, when that text wraps the same behaviour is experienced, even though the text that is printed on the next line is only half a cm high, the height of the image is used as the line height.

It is as if the line height is only calculated once and then reused as the line gets wrapped instead of re-evaluating the line height of the text that is left over and printed on the next line.

Comments : ~~~11Feb14 12:38 Brainy Data Response~~~

This issue was caused by an image that was the exact width of the row (exact to 1000th of a mm) causing it to find an un-handled condition in the code that determines the wrapping behaviour of OWrite.

ID : 1227

Fixed in version : 3.1.9

Short Description: OWrite crash when inserting kWriCharReturn

Full Description: I have a method with a non-visual OWrite object that we insert text into and save against the database. The method works fine on Windows 7, Omnis Studio 5.2.1 Unicode, OWrite 3.1.5r but causes Omnis to crash on OSX 10.9.1, Omnis Studio 5.2.1 Unicode, OWrite 3.1.5r.

I have a work around that solves it for us, but thought I would alert you to the possible fault. (I say possible since it could be that I have misinterpreted the way \$insert should be used)

Example method:

```
nvOWrite.$insert(kWriObjText,'My text here')
nvOWrite.$insert(kWriObjChar,kWriCharReturn)
nvOWrite.$insert(kWriObjText,'New line of text')
nvOWrite.$savedata(iRow.fieldname,kWriFmtDefault,kTrue)
```

On windows we end up with the expected text in 'fieldname' that can be viewed/printed with OWrite fields, but on OSX a hard crash occurs when it attempts to insert the carriage return. (found by stepping through the code, and then by running the code with all kWriCharReturn lines commented out)

Work around: replace line 2 above with `nvOWrite.$insert(kWriObjText,"{\rtf\par}")`

This results in the correct carriage return being inserted on both platforms.

Comments : ~~~10Feb14 14:31 Brainy Data Response~~~

Thank you for bringing this to our attention.

It appears that inserting a single character using the \$insert method or a user inputting single characters via the keyboard are handled in the same way. When individual characters are inserted, OWrite attempts to create a timer event so that when they are inserted in quick succession they can be combined into a single undo action.

Unfortunately, in the case of a non-visual object, it resulted in a null pointer call to Omnis that was to create a timed event to close the undo action for consecutive single character inserts. On the windows platform, the null pointer causes the request to be ignored, but on Macintosh, this call, eventually,

leads to a crash.

This issue has now been resolved and the fix will be available as part of our next maintenance release.

In the meantime, another work-around may be to change line 2 to
`nvOWrite.$insert(kWriObjText,chr(13))`