

## JSOWrite Bug Fixes

**ID :** 1744**Fixed in version :** 4.3.0**Short Description:** Problem to load a jsowrite document from a second time

**Full Description:** In a pagepane I have a subform that loads a form with jsOwrite. I call it a first time everything is ok. I close the form then reopen it but the jsowrite document does not load, I have to refresh the application.  
I enclose a demonstration video.

**Comments :** The minify process of one of our scripts had removed a vital function because it wasn't declared as an external function in our minify extern declaration. The function was responsible for removing old HTML object references so when the form destructs and a new-one was recreated some parts of our code was using the old HTML object reference.

We have released new scripts version (version 4.3.0.0) for the version 4.3.0.0 general OWrite/JS-OWrite maintenance release. Please make sure you update your jsctempl.html file to prevent the browser from caching the old scripts (see technical hint in the December Newsletter at [https://home.brainydata.com/news/news\\_20181202.pdf](https://home.brainydata.com/news/news_20181202.pdf)).

## OWrite Bug Fixes

**ID :** 1727

**Fixed in version :** 4.2.0.1

**Short Description:** Unable modify page footer

**Full Description:** Using a model of a customer, Omnis freeze when they try to modify the page footer. I attached the model in owr format.

AttachedFile: Modell\_XDomus.owr.zip

**Comments :** There was some corruption present within the formatting information of the footer text. Some text has a zero point size, others has a correctly set point size. We are unsure how this corruption occurred, but it was the cause of an endless loop within OWrite. We have fixed the endless loop but we strongly recommend the user fixes the text by selecting all text within the footer using the 'Select All' edit menu option and re-applying the correct font name and point size.

**ID :** 1729

**Fixed in version :** 4.2.0.2

**Short Description:** Calculated picture field lost in copy

**Full Description:** We uses :  
Do pRefEdit.\$savedata(lBinValue,kWriFmtRTF,kFalse,kWriSaveSelection,kTrue)  
Do pRefEditDest.\$::insert(kWriObjTypeText,utf8tochar(lBinValue),kFalse)

To copy some not evaluated text from a template to another.  
All works fine but Picture Calculated fields are replaced by picture field without any calculation.

I tried using editcopy/editpaste but I got the same result.

Is there a way to keep the calculated picture fields during copy ?

**Comments :** Calculations and all associated data such as click calculations and tooltips were failing to import for calculated pictures. Existing RTF exported should be fine. The data was only lost during importing or pasting RTF from the clipboard.

**ID :** 1740

**Fixed in version :** 4.2.0.3

**Short Description:** Attempt to print multiple pages prints only one page

**Full Description:** We have a problem with oWrite 4.1.3.0 and 4.2.0 in Omnis Studio 8.1.6 and Omnis Studio 8.0.3.5 (on macOS (tested with Mojave) and Windows (tested with Windows 10)):  
When we try to print multiple pages with oWrite, we get only one page printed as result when using an item reference.

The behaviour in different versions:

oWrite 4.0.2:

prints all pages when using an oWrite-item-reference ó> OK

prints all pages when using an oWrite-external-object-variable ó> OK

oWrite 4.0.3:

prints all pages when using an oWrite-item-reference ó> OK

prints all pages when using an oWrite-external-object-variable ó> OK

oWrite 4.1.3.0:

prints all pages when using an oWrite-item-reference ó> OK

prints only one page when using an oWrite-external-object-variable ó> NOK

oWrite 4.2.0:

prints all pages when using an oWrite-item-reference ó> OK

prints only one page when using an oWrite-external-object-variable ó> NOK

I attached an lbs to reproduce the problem.

In our code we use both methods (item-reference and external-object-variable).  
Are we doing something wrong or is it a bug in oWrite 4.1.3.0 and 4.2.0?

AttachedFile: issue1PagePrint.lbs.zip

**Comments :** The issues was caused by an uninitialized variable that was introduced in version 4.1.0 (case 1641).  
The problem is an intermittent problem that effects documents that were loaded using RTF or plain text. Such documents may only be formatted as a single page due to the uninitialized variable.

## OWrite Enhancements

**ID :** 1730

**Implemented in version :** 4.2.1.0

**Short Description:** Selecting a merged field, getting incorrect properties

**Full Description:** Ok weird, one. I have an oWrite field with a calculated field inside. I have some logic that replaces the calculated field with the actual value. I do this so I get accurate sizing information.

To select the calculated field I call:

```
Do owriteobject.$setselection(kWriSelectStart,kWriSelectEnd,objectid)
```

Then I overwrite the field like so:

```
Do owriteobject.$::insert(kWriObjTypeText,data,kWriInsertOver)
```

This used to result in the data being put in place with the correct formatting but we noticed it broke in the current version.

Testing we found out that the properties like \$curtextcolor all have the value of whatever contents is right before the selected field. It is like the start of our selection is slightly too early. As a result inserting the data gives that data the properties of whatever came before.

Any idea if I'm doing something wrong or if this is an issue in oWrite?

**Comments :** The behaviour of \$::insert was changed in version 3.0 at the end of 2012. One of the problems was that if one changed programmatically the style settings at the current insert position (when no text was selected) the \$::insert was loosing those settings. A side effect of this change was that the style of the current selection was ignored if that was different to the surrounding text.

As this change was made so long ago, it is not possible to restore the original behaviour so we had to change the behaviour of the third parameter which amounts to a new feature. Traditionally, the third parameter was a boolean parameter that specified if the inserted text is to be selected following the insert. We have two new parameter constants that can be passed instead.

kWriInsertSelect : inserted text will be selected following the insert. This replaces the original kTrue value for this parameter, although existing code which passes kFalse or kTrue will behave as before.

KWriInsertKeepStyles : if specified this will keep the styles of the current selection and apply it to the inserted text, the behaviour that was lost in version 3.0

To use these in combination you simply add them together.

```
Do ivEdit.$::insert(kWriObjTypeText,"some text",kWriInsertSelect+KWriInsertKeepStyles[,custom parameters])
```

**ID :** 1736

**Implemented in version :** 4.3.0.0

**Short Description:** Links to external web pages

**Full Description:** One of our clients has asked if it is possible to use html type links in the PDFDevice.

At present they are using something like:-

For more info see "[www.adnetltd.co.uk/support.html](http://www.adnetltd.co.uk/support.html)"

which works correctly, however some of their links can be quite long so they would like to use something similar to an html link

i.e

For more information <a href="http://www.adnetltd.co.uk/support.html"> click here</a>

Is this possible in the PDFDevice?

**Comments :** OWrite was already able to produce these kind of links with standard 'http://' urls. We have now added support for links to 'https://' and ensured that the link style and colour are carried through to the PDF file.

To create an active link that displays different text within a PDF file (requires PDFDevice to produce the PDF file), insert a calculated field with the URL as its link calculation.

We have also update OWrite to generate an evObjClick event when the click calculation is not a valid calculation and begins with http:// or https://. Please refer to the documentation regarding the evObjClick event and its parameters. The latest fat-client examples have been updated to handle this event.