

Bug Fixes

ID : 1125**Fixed in version :** 3.1.0**Short Description:** Wrong header margin on print**Full Description:** I have a problem with oWrite header margin during print.

Here you are the steps to replicate the error:

- 1) Open the example.lbs
- 2) Select "Templates" from "Document Groups"
- 3) Now you'll see a document with our company logo in the header and a table on two pages
- 4) Print the document using the toolbar button
- 5) Select "Direct to screen" or "Direct to PDF" in print options

You can see in the preview that the table is printed over the header logo on all pages and this is the error. It appears the logo is not positioned correctly.

Comments : When printing to a printer that has a non printable margin, the OWrite header and footer content was pushed into the printable area of the page.

After resolving this issue we found that any content that overlapped the non-printable area as defined by the current printer was clipped. The only way we were able to compensate for this Omnis print manager behaviour was by changing the non-printable margin to zero in the Omnis print job. This will only work when printing OWrite documents directly using the OWrite \$::print method. However, when printing from an Omnis report class, content that overlaps the non-printable margin will continue to be clipped by Omnis.

All Omnis reports appear to effected by this behaviour, even when printing a report to a PDF file via the system PDF output.

ID : 1127**Fixed in version :** 3.1.0**Short Description:** Crash issues**Full Description:** At the start of the week we started using oWrite 3 on our live system (our customers are still on 2 and will be for some time) and we've noticed that oWrite crashes regularly.

I'm going to point the finger at myself here because I suspect this has to do with something dirty we are doing with merging pictures and I'm planning to solve that differently soon. Still I just wanted to make you aware of this just in case you can do something in code to prevent oWrite from crashing.

This is all related with merging in calculated fields that contain both pictures and text.

5MAR2013: Additional information

It was indeed a work-around we implemented to get the pictures in. I've rewritten the code to replace the calculated fields with the actual data and it no longer crashes. When you add a kWriObjCalcText field and have it use a field that contains RTF data and that data has a picture in it, the picture is ignored. It is not merged into the document. The work around that we implemented was to replace the picture with a string, say "^PICTURE^" and then after merging do a search on this string and replace it with the picture. This "worked" in oWrite 2, it crashes in oWrite 3 presumably because an kWriObjCalcText field can not handle having a picture held within the data being merged.

Obviously there is an enhancement request in there for me because I would really like

kWriObjCalcText to be able to handle RTF data that contains a picture. I can't use kWriObjCalcPict because the data I'm trying to merge contains both text and images. For now though we've only ran into this with our email signatures and simply replacing the calculated field with the data works fine for me but I foresee in the not to distant future this limitation might come back to haunt me.

Comments : We have added some extra code that prevents a crash when a document object is deleted more than once due to a left-over reference when objects are accidentally nested in this way.

However, calculated text fields cannot handle RTF data that includes pictures. They were not designed for this and doing so may result in crashes elsewhere.

Enhancements

ID : 1116

Implemented in version : 3.0.6

Short Description: Exporting HREF to HTML now allows raw HTML

Full Description: When exporting documents to HTML, OWrite calls the developer's method `$owrite_export("html","href")`. Prior to OWrite version 3.0.5, this method had to return the 'src' string only. From version 3.0.5 onwards, this can now be the raw src html if the string begins with the open tag character '<'. This allows the inclusion of additional tags such as the target when returning urls.

Valid strings that can be returned by `$owrite_export`:

`'file.html#bookmark1'`

or

`''`

Comments :