

Bug Fixes

ID : 1421**Fixed in version :** 3.8.7**Short Description:** Heap corruption when calling \$getfontlist (was Memory leak in oWrite)

Full Description: Sorry man but I got a hard one for you again. This one is very hard to put our finger on but I've had a few times now that Omnis crashes out with a memory error while I've been messing about in oWrite for awhile.

We had another client encounter this issue on 2 out of a handful of machines but these guys were very willing for us to spend some time on their machines and helping resolve the issue.

We could create the same situation Mark managed to work out previously in that creating a new user profile on a machine with the issue would clear the issue.

A couple of things stood out for the users experiencing the crash:

- they had LOTS of user fonts, I mean thousands
- they had LOTS of different types of font, truetype, suitcase, bmap, bunch of others I've not seen
- they had a number of duplicate fonts

We were able to resolve the issue by trashing all the user fonts and reinstalling them through font book.

Obviously this would have resolved duplicate fonts and may have filtered out some of the fonts with strange formats so it's hard to pinpoint the exact cause.

But maybe it will be one more piece of the puzzle to find out why the crash is happening.

Comments : Although this problem was originally reported to exist in 3.8.0, later testing involved only 3.8.5 and we believe in conjunction to changes we made in regards to the \$getfontlist, an additional factor may have been a change that was made in version 3.8.5 that has the potential to cause memory conditions that may result in heap signal errors, causing Omnis Studio to quit, which is related to case 1515.

Enhancements

ID : 1520

Implemented in version : 3.8.6

Short Description: New border-less printing option

Full Description: We have added a new option when calling `$::print` to print a document border-less. This was originally introduced to work around an issue in Studio 8 printing on Macintosh but may be useful generally. For example when specifying this option when printing to PDFDevice, document content can occupy the entirety of the paper without any content being clipped.

Syntax:

```
Do OWrite.$::print(cDocumentName,bShowJobsetup,bKeepJobOpen,bBorderless)
```

Example use:

```
Do OWrite.$::print('Document Name',kTrue,kFalse,kTrue)
```

Comments :