13 December 2020

Newsletter

You are receiving this Newsletter because you expressed an interest in our software or are currently licensing one or more of our software components. This newsletter provides up-to-date information on the progress of our current developments, new software releases, and details about future developments.

Please also visit our news page for a summary of developments over the past twelve months.

https://home.brainydata.com/news.htm

NOTE: NEW SUBSCRIBERS will automatically receive a copy of the last newsletter that we circulated and consequently the date shown at the top of the newsletter may not be current.

Holidays/Christmas Office Closure

Our offices will be closed from the 15th of December to the 18th of January inclusively. We understand this is a slightly longer period than is usual and we apologize for any inconvenience this may cause. This extended closure has become necessary due to circumstances beyond our control. However, during this period we will have some limited time to deal with the most urgent cases, should they arise. By most urgent we mean, issues in live systems that cannot be worked around by end users or yourself, the developer. If you require urgent help please continue to use our online support request form at https://support.brainydata.com/request_form.htm. In the meantime

we wish you a ***Merry Christmas*** and a ***Happy New Year***.

Studio 10.2 support

We have now updated nearly all our current product lines to support Studio 10.2. Please note that the current builds only support Apple intel machines. We will build macOS universal binaries for the latest Apple hardware as soon as we receive the universal binary SDK from Omnis Software. The following product versions support Studio 10.2:

jsSignature version 1.1.1

https://support.brainydata.com/jssignature_su.htm

jsoWrite-client version 4.5.1.0

https://support.brainydata.com/jsowrite_su.htm

oWrite-desktop-server version 4.5.1.0

https://support.brainydata.com/owrite_su.htm

pdfDevice version 4.0.0.0 NEW

https://support.brainydata.com/pdfdevice_su.htm

oSpell2 version 3.3.0.0

https://support.brainydata.com/ospell2_su.htm

oGantt version 4.1.2.0

https://support.brainydata.com/ogantt_su.htm

oCal version 2.1.1.0

https://support.brainydata.com/ocal_su.htm

The outstanding ports for oLDAP and PDFWriter we will complete in the New Year.

PDFDevice version 4.0 release

PDFdevice version 4 has seen some major improvements, which include the following

Multi-threaded server

The new PDF server handling creates seperate device instances for each Omnis server stack, greatly simplifying tasks (without impacting other server stacks) such as printing multiple reports to the same PDF data, printing directly to memory or changing device settings.

New memory output

The new *$setmemoryoutput* function sends PDF data directly to your Omnis binary variable. The resulting data can be prefixed with a MIME header suitable for returning directly to the client for embedding using a data-url, or for returning as the result data for ultra-thin client requests. Printing Omnis reports direct to memory is ultra-fast and greatly reduces the tying up of server resources.

New temp file handling

The new function $settempfilename greatly simplifies using PDF device with the client commands “assignpdf” or “showpdf”. Please refer to our technical note TN0024 at https://supportpublic.brainydata.com/tn/tn0024.pdf which has been updated for version 4.

Example (print multiple reports to memory)

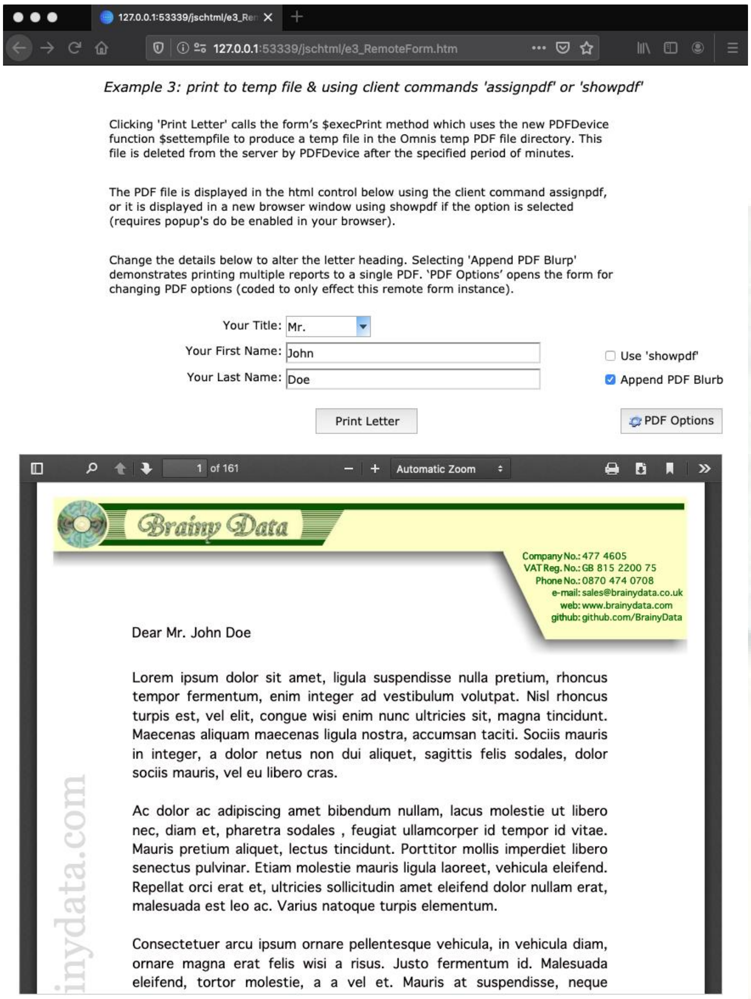
```
# ultra-thin call to print report and return PDF

# 1. prepare our output device (tell PDFDevice to send output to our local variable)
Do $cdevice.$assign(kDevPdf)
Do PDF Device.$setmemoryoutput(nam(lvPdfData),pParams.FileName)

# 2. print multiple reports to same PDF document
Do $cdevice.$open()
Set report name eX_Report
Print report * (pParams)  ## pass form data to report instance
Set report name eX_ReportPDFKeyActions
Print report
Set report name eX_ReportPDFBlurb
Print report
Do $cdevice.$close()

# 3. return result for display on client
Quit method lvPdfData
```

The above code snippet demonstrates aptly how little code is required to prepare PDFDevice, print multiple reports and return the resulting PDF data to the client.



127.0.0.1:53339/jschtml/e3_RemoteForm.htm

Example 3: print to temp file & using client commands 'assignpdf' or 'showpdf'

Clicking 'Print Letter' calls the form’s $execPrint method which uses the new PDFDevice function $settempfile to produce a temp file in the Omnis temp PDF file directory. This file is deleted from the server by PDFDevice after the specified period of minutes.

The PDF file is displayed in the html control below using the client command assignpdf, or it is displayed in a new browser window using showpdf if the option is selected (requires popup's do be enabled in your browser).

Change the details below to alter the letter heading. Selecting 'Append PDF Blurp' demonstrates printing multiple reports to a single PDF. ‘PDF Options’ opens the form for changing PDF options (coded to only effect this remote form instance).

Your Title: Mr.
Your First Name: John
Your Last Name: Doe

□ Use 'showpdf'
☑ Append PDF Blurb

Company No.: 477 4605
VAT Reg. No.: GB 815 2200 75
Phone No.: 0870 474 0708
e-mail: sales@brainydata.co.uk
web: www.brainydata.com
github: github.com/BrainyData

Dear Mr. John Doe

Lorem ipsum dolor sit amet, ligula suspendisse nulla pretium, rhoncus tempor fermentum, enim integer ad vestibulum volutpat. Nisl rhoncus turpis est, vel elit, congue wisi enim nunc ultricies sit, magna tincidunt. Maecenas aliquam maecenas ligula nostra, accumsan taciti. Sociis mauris in integer, a dolor netus non dui aliquet, sagittis felis sodales, dolor sociis mauris, vel eu libero cras.

Ac dolor ac adipiscing amet bibendum nullam, lacus molestie ut libero nec, diam et, pharetra sodales , feugiat ullamcorper id tempor id vitae. Mauris pretium aliquet, lectus tincidunt. Porttitor mollis imperdiet libero senectus pulvinar. Etiam molestie mauris ligula laoreet, vehicula eleifend. Repellat orci erat et, ultricies sollicitudin amet eleifend dolor nullam erat, malesuada est leo ac. Varius natoque turpis elementum.

Consectetuer arcu ipsum ornare pellentesque vehicula, in vehicula diam, ornare magna erat felis wisi a risus. Justo fermentum id. Malesuada eleifend, tortor molestie, a a vel et. Mauris at suspendisse, neque

Setting PDF options

The new server features make it now much easier to implement a JS-Client interface to allow users to manipulate device settings safely, i.e. changing device settings that only affects the client's server connection. The new PDFDevice examples at https://supportpublic.brainydata.com/examples/PDFDeviceAndJSClient.zip demonstrate how you may implement such an interface for JS-Client. Allowing the client to manipulate PDF settings requires nothing more than our provided interface, a remote form instance variable and an extra line of code when printing.

Example (print using client device settings)

```
# print for embedding in our ctrlEmbedPDF control

# 1. prepare our output device
Do $cdevice.$assign(kDevPdf)
#    1.a  apply device params to external device associated with this server stack
Do $cdevice.$setparam(kDevPdfAll,ivDeviceSettings)
#    1.b  tell PDFDevice to send output to our local variable
#          providing a mime file name prefixes data with a PDF mime header
Do PDF Device.$setmemoryoutput(nam(lvPdfData),"report.pdf")

# 2. print our reports
Do $cdevice.$open()
Set report name eX_Report
Print report * (ivFormData)  ## pass form data to report instance
Set report name eX_ReportPDFKeyActions
Print report
Set report name eX_ReportPDFBlurb
Print report
Do $cdevice.$close()

# 3. return result for display on client
Quit method lvPdfData
```

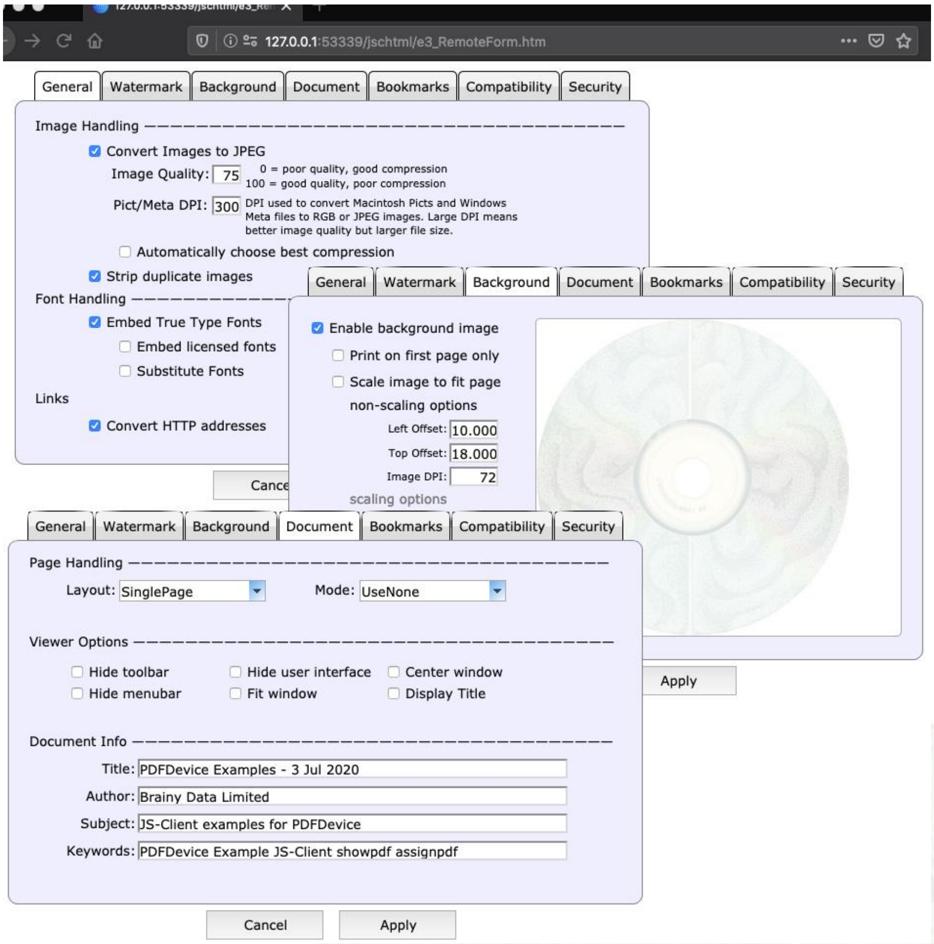


Image Handling

☑ Convert Images to JPEG

Image Quality: 75 — 0 = poor quality, good compression; 100 = good quality, poor compression

Pict/Meta DPI: 300 — DPI used to convert Macintosh Picts and Windows Meta files to RGB or JPEG images. Large DPI means better image quality but larger file size.

☐ Automatically choose best compression

☑ Strip duplicate images

Font Handling

☑ Embed True Type Fonts

☐ Embed licensed fonts

☐ Substitute Fonts

Links

☑ Convert HTTP addresses

☑ Enable background image

☐ Print on first page only

☐ Scale image to fit page

non-scaling options

Left Offset: 10.000

Top Offset: 18.000

Image DPI: 72

scaling options

Page Handling

Layout: SinglePage

Mode: UseNone

Viewer Options

☐ Hide toolbar ☐ Hide user interface ☐ Center window

☐ Hide menubar ☐ Fit window ☐ Display Title

Document Info

Title: PDFDevice Examples - 3 Jul 2020

Author: Brainy Data Limited

Subject: JS-Client examples for PDFDevice

Keywords: PDFDevice Example JS-Client showpdf assignpdf

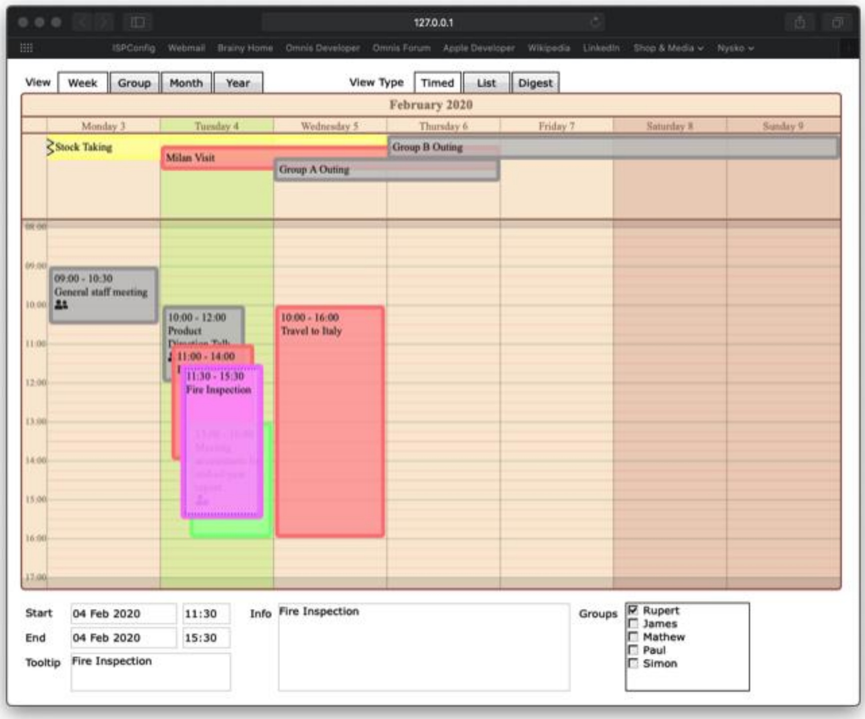
jsoCal development

Back in July of this year we released alpha 2 of our jsoCal port. However, we are saddened to say that we have had no feedback over the past five months, nor has there been any further interest in this component. Of course we understand that these are difficult times and we will continue our efforts to port our software to the Javascript client. In contrast, demand for our other desktop products has steadily increased during 2020, which will help to finance these important ongoing JS Client ports. In regards to the jsoCal development, we have decided we will close the current development cycle as the existing feature set is sufficiently comprehensive. We will produce full release and demo versions in the New Year in the hope that this will generate renewed interest.

In the meantime, if you are interested in jsoCal, please send us an email. Our contact details can be found on our contacts page at https://home.brainydata.com/contact.htm. For your reading pleasure, we have included below descriptions of the current main features of jsoCal.

The Week/Day View

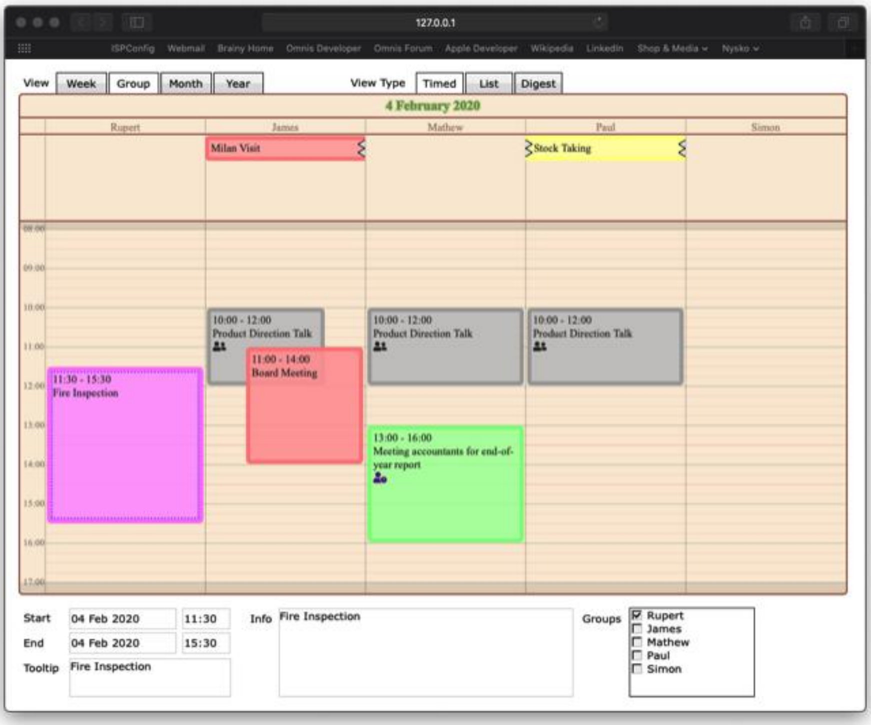
The week/day view can display events in two different modes. The timed mode displays events across the configurable vertical time scale on each day. The list view will list the event without overlapping them, displaying the earliest event of the day at the top of the column. Multi-day events are displayed in a separate multi-day pane above the standard day view columns. The week view can also be configured to show fewer or more days, i.e. just a single day, or only Monday to Friday, or 14 days, etc.



Start 04 Feb 2020 11:30
End 04 Feb 2020 15:30
Tooltip Fire Inspection
Info Fire Inspection
Groups ☑ Rupert ☐ James ☐ Mathew ☐ Paul ☐ Simon

Group View

The group view allows the display of the events for a single day within configurable group columns. A scheduled event may belong to one or more groups. Up to 255 individual groups could theoretically be displayed, although practically there may not be enough space on screen to show them all at once. However, OCal makes it easy to show or hide ranges of groups by assigning a sequence of YN states. As with the week view, events can be displayed in timed or list mode. Events that are shared amongst several groups will all select when one is clicked.

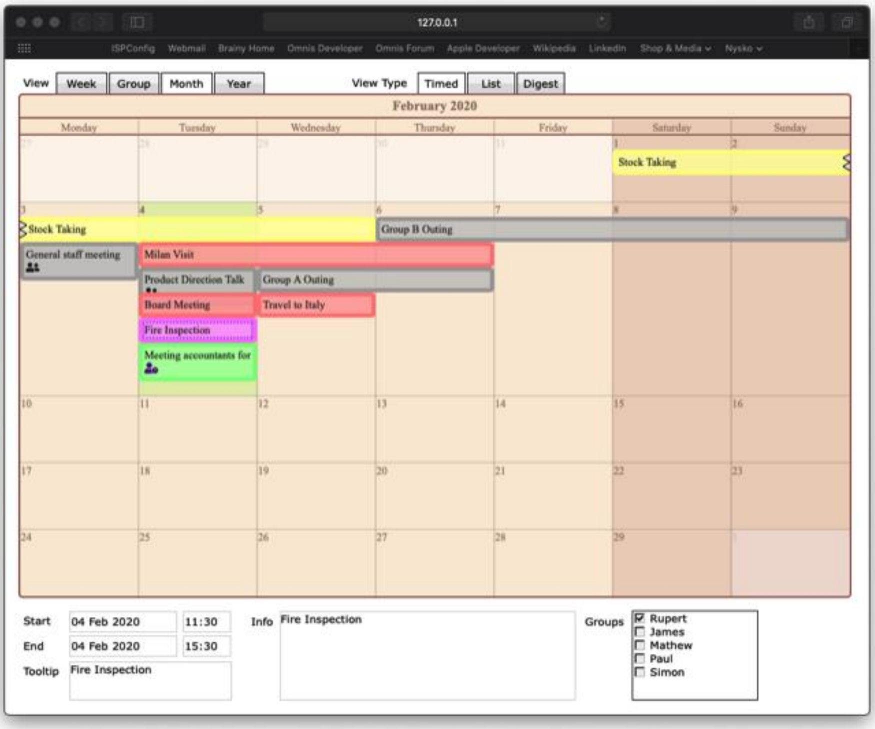


Start 04 Feb 2020 11:30
End 04 Feb 2020 15:30
Tooltip Fire Inspection
Info Fire Inspection
Groups ☑ Rupert ☐ James ☐ Mathew ☐ Paul ☐ Simon

Both these views allow the dragging and resizing of events using the mouse. Of course, events can also be repositioned using date and time fields provided by your interface.

Month View

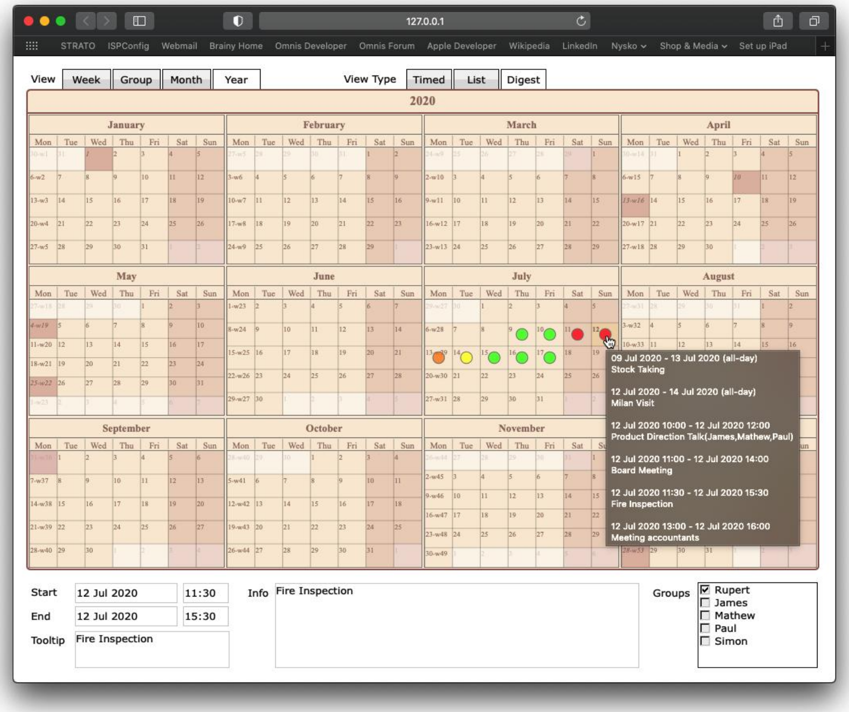
The month view displays the calendar using a traditional month view layout where the month is divided into rows of weeks. In this mode events are displayed in list mode only. A calendar row will expand to make room so all events in a day are visible. If need be, OCal will add a scroll bar should there not be enough room to display all rows of the month. The month view is also capable of displaying the ISO week numbers, a feature that was added in alpha 2.



Start 04 Feb 2020 11:30
End 04 Feb 2020 15:30
Tooltip Fire Inspection
Info Fire Inspection
Groups ☑ Rupert ☐ James ☐ Mathew ☐ Paul ☐ Simon

Annual Digest View

The annual digest view displays an entire year of events using colours to highlight busy or less busy days. The calculations and colours used are highly configurable. The jsoCal control uses HSV colour manipulation techniques to produce the smooth shades that morph the idle colour with the busy colour. Hovering over a digest bullet will pop-up a tool-tip displaying a configurable summary of all the events of that day.



09 Jul 2020 - 13 Jul 2020 (all-day)
Stock Taking

12 Jul 2020 - 14 Jul 2020 (all-day)
Milan Visit

12 Jul 2020 10:00 - 12 Jul 2020 12:00
Product Direction Talk(James,Mathew,Paul)

12 Jul 2020 11:00 - 12 Jul 2020 14:00
Board Meeting

12 Jul 2020 11:30 - 12 Jul 2020 15:30
Fire Inspection

12 Jul 2020 13:00 - 12 Jul 2020 16:00
Meeting accountants

Start 12 Jul 2020 11:30
End 12 Jul 2020 15:30
Tooltip Fire Inspection
Info Fire Inspection
Groups ☑ Rupert ☐ James ☐ Mathew ☐ Paul ☐ Simon

```
.jsocal .horzDividerLineMajor {
    /* seperates hours */
    stroke:black;
    stroke-width:1;
    stroke-linecap:butt;
    stroke-opacity: 0.2;
}
.jsocal .horzDividerLineMinor {
    /* seperates $dayviewtimescale minutes */
    stroke:red;
    stroke-width:1;
    stroke-linecap:butt;
    stroke-opacity: 0.1;
}
```

Appearance

The calendar appearance, with the exception of the main colours and border style which are controlled by Omnis properties, is entirely controlled by configurable CSS. To manipulate individual elements within the control, the provided CSS file “ctl_ocal.css” can be edited. For example, to change the appearance for the time lines, simply edit the CSS for “.jsocal .horzDividerLineMajor” and “.jsocal .horzDividerLineMinor”.

jsoGantt development

jsoGantt

Having completed the initial design stages for our port of oGantt to the JS-Client, we have begun scheduling the development of this component. As with jsoCall, we will implement the design-interface using the Omnis Studio JSON control editor. Development is still in its earliest stages, but we are confident that we should be able to produce a beta in late Q2 or early Q3 of next year. If you are interested in receiving prior alpha releases of this component, please talk to us in the New Year so we can discuss the commitments on both sides.

The plan so far is to utilize the existing non-visual layer object of the oGantt desktop version. This object will be used to construct the GANTT data for the client control, just as it is used now to construct the GANTT data for the desktop control. We will add a new feature to convert this data to JSON which the jsoOGantt control will be able to load and display on the client in a similar way that the oGantt desktop control implements the display of this data.

We are not yet sure if there will be some differences or limitations in behaviour and feature set, but we will endeavour to implement jsoGantt as close to the current desktop behaviour as possible.

If you wish to help with this development, either by ensuring its financial success by pre-ordering the component, or by providing other resources, please contact us via email and we will be in touch. Contact details are available on our contacts page at https://home.brainydata.com/contact.htm.

oWrite/jsoWrite version 5 alpha 4

We have been working on version 5 for a while and are nearing the end of its development. We expect to produce a beta release early in the New Year. The following are the new features that are available in the alpha 4 release.

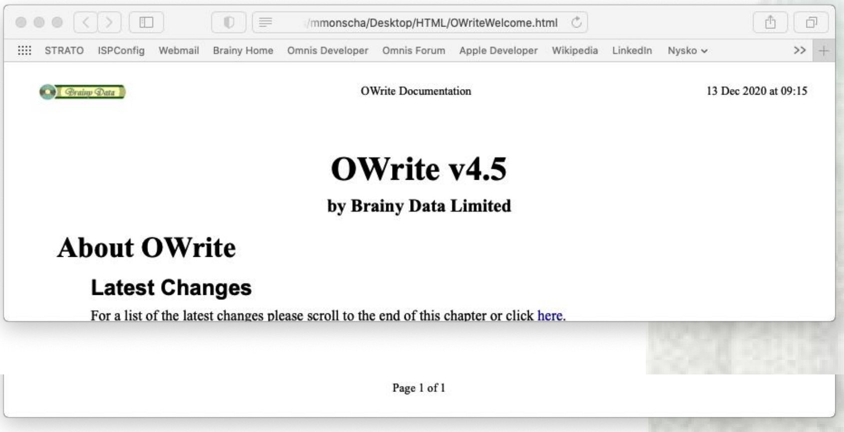
Sub-lists

1. Main list *1*
2. Main list *2*
    a. sub-list *1.a*
        i. sub-sub-list *2.a.i*
        ii. sub-sub-list *2.a.ii*
    b. sub-list *1.b*
3. Main list *3*

oWrite now supports sub-lists. There is a new property called $curlistlevel which can be manipulated programmatically. The list level can also be manipulated by the user using the usual ‘tab’ and ‘return’ keys.

For example, when in a main list item, pressing return (creates a new main list item) followed by tab will increase the list level. Pressing shift-tab when the input caret is at the start of the list item will decrease the list level.

Headers & Footers



OWrite Documentation

13 Dec 2020 at 09:15

OWrite v4.5

by Brainy Data Limited

About OWrite

Latest Changes

For a list of the latest changes please scroll to the end of this chapter or click here.

Page 1 of 1

Traditionally, non-paginated export formats such as plain text and HTML did not export oWrite headers and footers. In version 5 oWrite will now output the “default” header and footer (being single page formats no other headers can be exported) if the custom parameter kWriOutputHF is specified when calling $savedata.

$papercontinuous improvements

Some of the other major enhancements are less visual, albeit more functional in nature. Amongst these are important improvements that we made to the existing $papercontinuous feature.

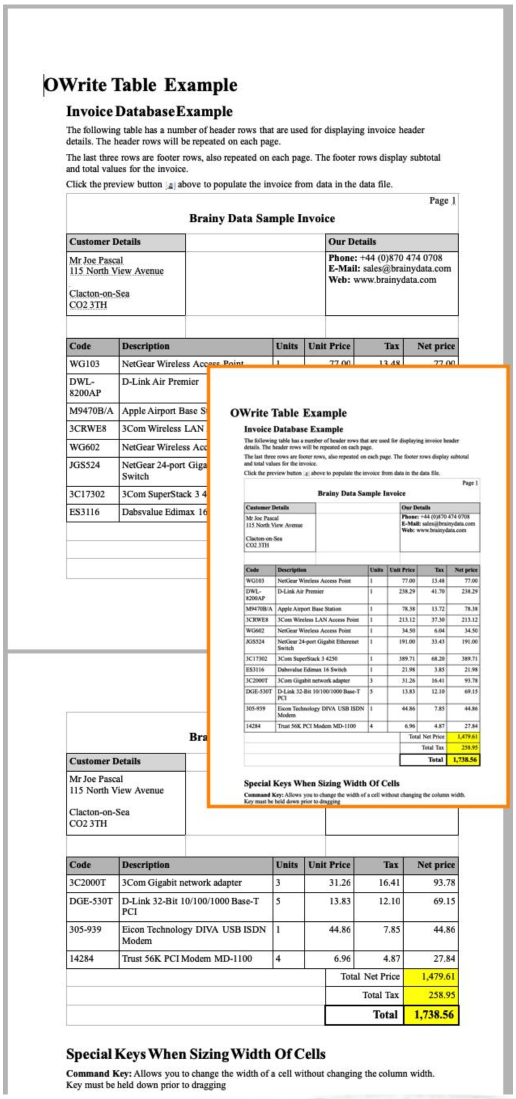
Background

Traditionally, this feature has been used to present documents in a single page mode for editing after the document has been evaluated. This feature is especially important when the document used table fields that pull in list data during evaluation and generate table headers and footers around document page breaks. If the user was to edit the document in this state it would be very difficult for oWrite to maintain the headers and footers around these breaks. By turning on $papercontinuous mode, oWrite temporarily removes all generated headers and footers, allowing the user to edit the document and remaining table cells within a single page view without adversely affecting the document’s integrity. When the user is done, the $papercontinuous mode can be turned off and oWrite will reproduce the headers and footers around the natural page breaks of the document, re-evaluating any total and subtotal calculations within these headers and footers.

The improvements

Firstly, static table fields (tables that are not linked to an Omnis data list) were not handled by the $papercontinuous feature. This has now been rectified. In other words, static tables will now generate headers and footers around page breaks when using this feature to edit documents.

Secondly, when this mode is turned on, all the document headers and footers were hidden. In version 5, the default header and footer will be shown when in the single page mode so that their content can also be edited.



OWrite Table Example

Invoice Database Example

The following table has a number of header rows that are used for displaying invoice header details. The header rows will be repeated on each page.

The last three rows are footer rows, also repeated on each page. The footer rows display subtotal and total values for the invoice.

Click the preview button above to populate the invoice from data in the data file.

Page 1

Brainy Data Sample Invoice

Customer Details		Our Details
Mr Joe Pascal 115 North View Avenue Clacton-on-Sea CO2 3TH		**Phone:** +44 (0)870 474 0708 **E-Mail:** sales@brainydata.com **Web:** www.brainydata.com

Code	Description	Units	Unit Price	Tax	Net price
WG103	NetGear Wireless Access Point	1	77.00	13.48	77.00
DWL-8200AP	D-Link Air Premier	1	238.29	41.70	238.29
M9470B/A	Apple Airport Base Station	1	78.38	13.72	78.38
3CRWE8	3Com Wireless LAN Access Point	1	213.12	37.30	213.12
WG602	NetGear Wireless Access Point	1	34.50	6.04	34.50
JGS524	NetGear 24-port Gigabit Etherenet Switch	1	191.00	33.43	191.00
3C17302	3Com SuperStack 3 4250	1	389.71	68.20	389.71
ES3116	Dabsvalue Edimax 16 Switch	1	21.98	3.85	21.98
3C2000T	3Com Gigabit network adapter	3	31.26	16.41	93.78
DGE-530T	D-Link 32-Bit 10/100/1000 Base-T PCI	5	13.83	12.10	69.15
305-939	Eicon Technology DIVA USB ISDN Modem	1	44.86	7.85	44.86
14284	Trust 56K PCI Modem MD-1100	4	6.96	4.87	27.84
				Total Net Price	1,479.61
				Total Tax	258.95
				Total	**1,738.56**

Special Keys When Sizing Width Of Cells

Command Key: Allows you to change the width of a cell without changing the column width. Key must be held down prior to dragging

Thirdly, static page breaks (page-breaks inserted manually) were still producing structural breaks in this mode. This has been changed so that static page breaks are now mere visual place holders and do not produce additional structural pages in a mode that is supposed to be presenting a single page. Static page breaks will generate the correct structural breaks when $papercontinuous is turned off.

Other changes

There have been numerous other smaller changes (enhancements and gremlins that have been expelled). These include improvements to oWrite-desktop border handling, the ability to disable editing of headers and footers, support for different bullet characters, the ability to limit row heights during evaluation, as well as some improvements to RTF parsing.

For a full list of changes see the alpha 4 release notes at

https://supportpublic.brainydata.com/rn/owrite_jsowrite_4904.pdf.

Downloads:

macOS:	https://support.brainydata.com/owrite_sm/patches/owrite_4904_major_release5a4_mac.zip https://support.brainydata.com/jsowrite/patches/jsowrite_4904_major_release5a4_mac.zip
winOS:	https://support.brainydata.com/owrite_sw/patches/owrite_4904_major_release5a4_win.zip https://support.brainydata.com/jsowrite/patches/jsowrite_4904_major_release5a4_win.zip
additional files:	https://demos.brainydata.com/software/owrite_jsowrite_4904_additional_files.zip

We still have a number of outstanding support cases that we need to address in version 5. However, if you have reported an issue previously and feel it is important, please get in touch to ensure we look into it before we finalise version 5.

⚠ We need your help!

We have had requests to update our oLDAP component to support LDAP S with secure socket support. These requests have come on the back of the announcement by Microsoft that their Active Directory servers will discontinue support for non-secure connections. Our first investigation suggests that the actual changes required to our software may not be difficult, but we cannot evaluate this impression further without having access to an Active Directory server and to a client machine that has in place all that is required for the new signing protocols to test our changes. If you do have access to an Active Directory server and clients that can successfully connect to the server, perhaps you could assist us in the implementation and testing of our component? See Microsoft Article at

https://support.microsoft.com/en-us/help/4520412/2020-ldap-channel-binding-and-ldap-signing-requirements-for-windows

Technical Notes

We have updated the following technical note.

TN0032: Document List Structure

Technical notes can be viewed at

https://supportpublic.brainydata.com/technotes.htm

Technical Hints

The 'Technical Hint' is a regular article containing one or more interesting things that may have come up in technical support queries or that we encountered during development.

Undocumented oWrite constants

It has come to our attention that there are undocumented features that were implemented in the recent past. We will add these to our documentation in due course. For now, please find some details below.

We implemented the $watermark property in oWrite version 3.8.5. This property is well documented in the current documentation. However, there is also a set of watermark constants for use in the first column of the watermark list that have largely remained undocumented. These are:

- **kWriWMallPages** - print specified watermark on all pages
- **kWriWMfirstPage** - print specified watermark on first page
- **kWriWMnotFirstPage** - print specified watermark on all except the first page

Headers and footers in oWrite can contain special info objects. Traditionally these displayed basic document information such as the date, time, page count and current page number. Such info objects can be inserted using the notation

```
$::insert(kWriObjTypeInfo,kWriObjTypeInfoDate)
```

However, we added additional types to oWrite to allow the display of page count and numbers using roman numerals, and to exclude the first page in the count. The constants that were added are as follows:

- **kWriObjTypeInfoPgCntFP** - display the page count excluding the first page
- **kWriObjTypeInfoPgCntLR** - display the page count using lower case roman numerals
- **kWriObjTypeInfoPgCntUR** - display the page count using upper case roman numerals
- **kWriObjTypeInfoPgNumFP** - display the page number excluding the first page
- **kWriObjTypeInfoPgNumLR** - display the page number using lower case roman numerals
- **kWriObjTypeInfoPgNumUR** - display the page number using upper case roman numerals

When we implemented the kWriSaveAsList feature in version 4, we also added a large number of constants to aid with the deciphering of the document list structure, specifically data related to document style settings. There are a huge number of constants and we will not document them here. We will merely draw your attention to their location in the Studio catalog. The additional constants are organised into these groups:

- **OWrite-Style Codes (Paragraph)** - these are constants that identify paragraph styles
- **OWrite-Style Codes (Special)** - these are constants that identify special style runs that mark object place holders, object result data and spell-checker highlights.
- **OWrite-Style Codes (Text)** - these are constants that identify text styles used both in paragraph style settings and text style runs.
- **OWrite-Style Values (Paragraph)** - these constants are used as values for some of the paragraph styles.
- **OWrite-Style Values (Text)** - these constants are used as values for some text styles.

We want your Feedback!

We would like to add your feedback to our customer page at https://home.brainydata.com/customers.htm. Please take a look and if you feel you would like to contribute to this page please send us an email with your feedback about our services and products. You are also very welcome to add details about your own products! **Even if you have already provided feedback previously**, perhaps it is time to update your feedback or add some additional comments?

There is no limit to the size of your feedback, but in addition to your comments we would like to list your company name with a link to your website, your name (although required for us to identify you, you can request for your name not to be shown on our feedback page) and your position within the company (i.e. job title).

Please send all feedback to feedback@brainydata.co.uk (do not forget to include your name, job title and website address). We appreciate your support.

Important Links

News: https://home.brainydata.com/news.htm

Products & pricing: https://products.brainydata.com

Demo/Examples Downloads: https://demos.brainydata.com/download.htm

Github: https://github.com/BrainyData

Sponsors: https://home.brainydata.com/sponsors.htm

Feedback: https://home.brainydata.com/customers.htm

Online Documentation: https://supportpublic.brainydata.com/documentation.htm

Technical notes: https://supportpublic.brainydata.com/technotes.htm

Support Request Form: visit https://supportpublic.brainydata.com and click “Software Downloads”

Software Downloads: visit https://supportpublic.brainydata.com and click “Contact Support”

This newsletter is for informational purposes only. Brainy Data assumes no responsibility for its accuracy, and the information is subject to change without notice. Any use of, or actions taken based upon, any of the information contained in this newsletter is done entirely at your own risk.

This document was produced by OWrite and PDFDevice.