



XtPrinter 1.0.1

Reference Manual ver. 0.0.1

Homepage: <http://e2cs.mm-mendez.com/>

Manual Translations credits:

English version: Carlos Mendez – cmendez21@gmail.com

Any person who would like to translate please take this document as the starting point. (Word 07 version for modifications or translations available for download on the site, meanwhile the site is up please contact us to send you a copy for the word doc only.) Also Author's credits will be set in the *E2CS home page* (manual page)

Notes:

This is the first revision and initial documentation on the **XtPrinter** widget so many features or documentation may be still missing so we apologize for this and we appreciate all your reports about this issue thanks..!

CONTENTS

- **Class Ext.grid.Xprinter**
- **Properties**
- **Methods**
- **Usage**
- **History Log**

File: extjs_printergrid_0.0.1.js

Class: Ext.grid.XPrinter

Custom object to get the grid structure (renderers) or custom store (with additional configuration use Pagination toolbar in case of need) and generates a HTML table with custom format for printing, also capabilities to generate a PDF file for exporting.

For export PDF it uses **dompdf 0.5.1** <http://www.digitaljunkies.ca/dompdf/>

TESTED

- On Firefox 2.0 and IE6, IE7

TO DO

- Create Grouping rows
- Use a custom template provided for multiple records or one (a lot of fields)
- Parameters to search and to filter in the output generated (custom store)

Properties

Grid: grid panel or gridEditor panel object

Used if the report will generate data according to the grid if using custom settings with a custom store this property it's not necessary

styles: string

Name of the stylesheet used for the new page (output that's gonna be printed), in fact there's 2 style sheets `default.css` and `default_printer_out.css`

pathPrinter: string

The path to the *printer* folder where every resource for output/printing resides

pdfEnable: boolean

Enable PDF output based on HTML generated

paperOrientation: string

Values are 'p' for portrait or 'l' for landscape any other will be set as 'p'

logoURL:string

Path for the filename (image) that will be displayed as the logo for the report

wWidth: integer

Width for the window that will popup with the output HTML file

wHeight:integer

Height for the window that will popup with the output HTML file

localeData: config object

Object containing the string used for labels in the output page

```
localeData:{
    Title:'Title',
    subTitle:'Subtitle',
    footerText:'Page footer',
    pageNo:'Page #',
    printText:'Print document',
    closeText:'Close document',
    pdfText:'Create PDF file'
}
```

useCustom: config object

Object containing the parameters needed for the use of custom store and some basic uses

custom: boolean

Specifies if the printer object will use a custom store

customStore: store object

columns: array

Array of text for the title of the columns

alignCols: array

Array of text specifying the alignment of each column

pageToolbar: PaginationToolbar object

This is in case there are a lot of records and could need pagination for displaying records also its necessary to set a number of records for each page according the style or contents so each page will not be over written or exceeds the space for a presentation report.

uselt: boolean

If the printer is bound to a grid then the printer uses the grid's paging toolbar to set the page number otherwise the page always be 1 (for custom store not applies)

location: string

Tells the printer the position of the paging toolbar on the grid (binding grid mode) so it can perform the generation of the page number

```
useCustom:{
    custom:false,
    customStore: null,
    columns:[],
    alignCols:[],
    pageToolbar:null,
    uselt: false,
    location: 'bbar'
}
```

showdate: boolean

Used for displaying the date of generation on the top right of the report.

showdateFormat: string

Any format string based on DATE object / util in Extjs

showFooter: boolean

Used for displaying the footer on the report.

footerText: string

String to be displayed on the footer

METHODS

prepare()

prepares the HTML file for output and waits it has too be loaded (see USAGE)

printGrid()

Creates the HTML for the popup HTML file and renders in this document.

newPage()

the report needs pagination or refreshing the data (custom store mode) its necessary this function so it updates the HTL in the output file.

USAGE (GRID binding)

```
Ext.onReady(function() {
    Ext.QuickTips.init();

    [Create your store and GRID here]
    myprinter=null;
    function printmygridPDF(){
        myprinter = new Ext.grid.XPrinter({
            grid: myrecords_grid,
            pathPrinter: './printer',
            logoURL: 'companylogo.jpg',
            pdfEnable: true,
            localeData: {
                Title: 'myrecords',
                subTitle: 'subtitle here...',
                footerText: 'myrecords report footer',
                pageNo: 'Page.#',
                printText: ' print document',
                closeText: ' close document',
                pdfText: ' download PDF file'
            },
            useCustom: {
                custom: false,
                customStore: null,
                columns: [],
                alignCols: [],
                pageToolbar: null,
                useIt: true,
                showIt: false,
                location: 'bbar'
            },
            showdate: true,
            showdateFormat: 'Y-F-d H:i:s',
            showFooter: true,
            styles: 'default'
        });
        myprinter.prepare();
    }
});

// this function will be called always by the new HTML page when it finishes loading and has the
// document DOM ready to be used.
function printmygridGO(obj){
    myprinter.printGrid(obj);
}
```

(CUSTOM store)

The custom store it's planned for use it with more data (fields) than the grid does cause sometimes one uses the grid for displaying few fields and the record has more

```
Ext.onReady(function() {
    Ext.QuickTips.init();

    [Create your store and GRID here]

});

function createreportgrid(){
    storex = new Ext.data.Store({
        url: 'yourserverfile.php',
        baseParams: {
            task: 'getrpt'
        },
        reader: new Ext.data.JsonReader({
            root: 'records',
            id: 'cve',

```

```

                                totalProperty: 'totalCount'
                                },
                                [{name:'id', mapping: 'id', type: 'int'},
                                {name:'name', mapping:'name', type: 'string'},
                                {name:'address',mapping:'address', type: 'string'}]
                                ),
                                remoteSort: false
                                ))
                                report_pbar = new Ext.PagingToolbar({
                                pageSize: 3,
                                store: storex,
                                displayInfo: true,
                                displayMsg: 'Showing Records {0} - {1} of {2}',
                                emptyMsg: "Theres no data o display"
                                })
                                global_printer = new Ext.grid.XPrinter({
                                grid: yourgrid,
                                pathPrinter: './printer',
                                logoURL: 'companylogo.jpg',
                                pdfEnable:true,
                                paperOrientation:'p',
                                localeData:{
                                Title:'Employees catalog with Adress',
                                subTitle:'your subtitle here',
                                footerText:' footer text here',
                                pageNo:'Page # ',
                                printText:' Print ',
                                closeText:' Close ',
                                pdfText:' download PDF '
                                },
                                useCustom:{
                                custom:true,
                                customStore:storex,
                                columns:['ID','Employee Name','Description'],
                                alignCols:['center','left','left'],
                                pageToolbar:report_pbar,
                                useIt: false,
                                showIt: true,
                                location: 'bbar'
                                },
                                showdate:true,
                                showdateFormat:'Y-F-d H:i:s',
                                showFooter:true,
                                styles:'default'
                                });
                                storex.on('load', function(){
                                if (global_printer.flagdatachange){
                                global_printer.newPage();
                                }
                                });
                                storex.load();
                                global_printer.prepare();
                                }

                                function printmygridGO(obj){
                                global_printer.printGrid(obj);
                                }

```

FAQ's

I want to change the design and Xtemplate which uses the XtPrinter how can I do it?

A: Replace the HTML code used on the JS file on Line 99 and Line 292 but be aware of some {} fields which the XtPrinter uses to create the output

Is there Skins for the Printer Output, because I need it for my application?

A: Use the EXTjs Skins normally and Change also in the base_print_001.php and base_print_002.php files, also you can change the CSS (default and printer on the Printer folder to Make quick changes.

I need it FOR ASP, JAVA, COLDFUSION Print output what can I do?

A: You will need to adapt code and server side code to make it work, also DOMPDF I think it's only for PHP so you could use something to generate PDF files.

I have a deadline and I can't make the XtPrinter work can you help me giving me support or develop more code or modules for my application?

A: Sure you have the following options

- Donate (for quick and urgent support responses according donation)
- \$100 Us. dollars per Hour for consulting, coding modules or custom code (JavaScript, php, mysql, Extjs)
 - Payment in advance for Foreign Clients (outside México).
 - For Mexican Clients 50% in Advance and 50% on Release on my Test site and then I deliver the code (also I give invoices tax deductible, also tax it's not included on the price).
- Per project .- Depending on the project to make and the time that will take to do it (please request for pricing and time availability for development)

History Log

17-11-2008 First release