

photo

/**

* This software is intended for use with Oxwall Free Community Software
<http://www.oxwall.org/> and is
* licensed under The BSD license.

* ---

* Copyright (c) 2011, Oxwall Foundation
* All rights reserved.

* Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the
* following conditions are met:

*

* - Redistributions of source code must retain the above copyright notice, this
list of conditions and
* the following disclaimer.

*

* - Redistributions in binary form must reproduce the above copyright notice, this
list of conditions and
* the following disclaimer in the documentation and/or other materials provided
with the distribution.

*

* - Neither the name of the Oxwall Foundation nor the names of its contributors
may be used to endorse or promote products
* derived from this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES,
* INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED
* AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

*/

/**

* @author Kairat Bakitow <kainisoft@gmail.com>
* @package ow_plugins.photo
* @since 1.6.1

*/

(function(window, \$, params) {'use strict';

```

                                photo
var utils = window.photoUtils;
var _vars = $.extend({}, (params || {}),
    {cache: {}, minSize: {width: 400, height: 400}, spacing: 30, infoBar: 400,
cmpListForDeleting: [], curPos: 0, fullScreen: (function()
{
    if ( !params.isEnableFullscreen )
    {
        return false;
    }

    var val, valLength;
    var fnMap = [

['requestFullscreen','exitFullscreen','fullscreenElement','fullscreenEnabled','fulls
creenchange','fullscreenerror'],

['webkitRequestFullscreen','webkitExitFullscreen','webkitFullscreenElement','webkitF
ullscreenEnabled','webkitfullscreenchange','webkitfullscreenerror'],

['webkitRequestFullScreen','webkitCancelFullScreen','webkitCurrentFullScreenElement'
,'webkitCancelFullScreen','webkitfullscreenchange','webkitfullscreenerror'],

['mozRequestFullScreen','mozCancelFullScreen','mozFullScreenElement','mozFullScreenE
nabled','mozfullscreenchange','mozfullscreenerror'],

['msRequestFullscreen','msExitFullscreen','msFullscreenElement','msFullscreenEnabled
','MSFullscreenChange','MSFullscreenError']
    ];
    var i = 0;
    var ret = {};

    for ( ; i < fnMap.length; i++ )
    {
        val = fnMap[i];

        if ( val && val[1] in document )
        {
            for ( i = 0, valLength = val.length; i < valLength; i++ )
            {
                ret[fnMap[0][i]] = val[i];
            }

            return ret;
        }
    }

    return false;
}))(),

```

```

                                photo
{globalList: [], lists: [], ID_LIST_LIMIT: 2, DEFAULT_LIST: 'latest'},
{toString: Object.prototype.toString},
{
  commentEvents: [
    'base.comments_list_init',
    'base.comment_add',
    'base.comment_delete',
    'base.attachment_added',
    'base.attachment_deleted',
    'base.comment_added',
    'base.comment_textarea_resize',
    'base.comment_attachment_added',
    'base.comment_button_show',
    'base.add_photo_attachment_submit',
    'base.comment_attachment_deleted',
    'base.comment_video_play',
    'base.comment_attach_media'
  ],
  commentUnknownEntity: [
    'base.add_photo_attachment_submit',
    'base.comment_video_play',
    'base.comment_attach_media'
  ]
},
),
_elements = {
  preloader: $('<div>').addClass('ow_floatbox_preloader ow_photo_preload')
},
_methods = {
  isLoading: function()
  {
    return _vars.fetchRequest && _vars.fetchRequest.readyState !== 4;
  },
  setId: function( photoId, listType, extend, data, photo )
  {
    if ( _methods.isLoading() )
    {
      return false;
    }

    _methods.history.replaceState(photoId, listType, extend, data, photo);
  },
  isCached: function( photoId )
  {
    return !isNaN(+photoId) && _vars.cache.hasOwnProperty(+photoId);
  },
  cachePhotoComponents: function( photos )
  {

```

```

                                photo
    var keys, idList = [];

    if ( !photos || utils.getObjectType(photos) !== 'Object' || (keys =
Object.keys(photos)).length === 0 )
    {
        return false;
    }

    keys.forEach(function( item )
    {
        if ( !_methods.isCached(item) )
        {
            var cmp = _vars.cache[item] = photos[item];
            idList.push(item);

            if ( cmp.photo.dimension )
            {
                try
                {
                    cmp.photo.dimension = JSON.parse(cmp.photo.dimension);
                }
                catch ( ignore ) { }
            }

            OW.trigger('photo.afterPhotoCached', [cmp.photo.id, cmp],
_methods);
        }
    });

    _methods.backgroundLoadPhotos(idList, _methods.fullscreen.isFullscreen()
? 'fullscreen' : 'main');
},
backgroundLoadPhotos: function( photoIdList, type )
{
    if ( !Array.isArray(photoIdList) || photoIdList.length === 0 ) return;

    photoIdList.forEach(function( item )
    {
        if ( _methods.isCached(item) )
        {
            var cmp = _methods.getPhotoCmp(item);
            var url = (type === 'fullscreen' && +cmp.photo.hasFullsize) ?
cmp.photo.urlFullscreen : cmp.photo.url;

            setTimeout(function( url )
            {
                new Image().src = url;
            }, 1, url);

```

```

                                photo
                                }
                            });
    },
    setState: function( loading )
    {
        if ( utils.getObjectType(loading) !== 'Boolean' )
        {
            loading = true;
        }

        switch ( loading )
        {
            case false:
                _methods.utils.hidePreloader();
                break;
            case true:
            default:
                _methods.utils.showPreloader();
                break;
        }
    },
    getPhotoCmp: function( photoId )
    {
        if ( !_methods.isCached(photoId) )
        {
            return null;
        }

        return _vars.cache[photoId];
    },
    deletePhotoCmp: function( photoId )
    {
        delete _vars.cache[photoId];
    },
    getIsLoadPrevList: function( index )
    {
        return !_vars.isPrevCompleted && index < _vars.ID_LIST_LIMIT;
    },
    getIsLoadNextList: function( index, list )
    {
        return !_vars.isNextCompleted && list.length - index <
_vars.ID_LIST_LIMIT;
    },
    getPrevId: function( index, list )
    {
        if ( ['most_discussed', 'toprated'].indexOf(_vars.listType) !== -1 )
        {
            return index === 0 ? list[list.length - 1] : list[index - 1];
        }
    }

```

```

        photo
    }

    return index === 0 ?
        (_vars.lists.length === 0 ?
            _vars.globalList[_vars.globalList.length - 1] :
            _methods.arrayUtils.getMinInList(_vars.lists)) :
        list[index - 1];
},
getNextId: function( index, list )
{
    if ( ['most_discussed', 'toprated'].indexOf(_vars.listType) !== -1 )
    {
        return index === list.length - 1 ? list[0] : list[index + 1];
    }

    return index === list.length - 1 ?
        (_vars.lists.length === 0 ?
            _vars.globalList[0] :
            _methods.arrayUtils.getMaxInList(_vars.lists)) :
        list[index + 1];
},
clearHistory: function()
{
    _vars.globalList.length = 0;
    _vars.lists.length = 0;
},
showFloatBox: function()
{
    if ( _elements.hasOwnProperty('photoFB') || _vars.layout === 'page' )
    {
        return;
    }

    var options = {};

    if ( _vars.isClassic )
    {
        options.addClass = 'ow_photoview_overlay_pint';
        options.$title = '';
    }
    else
    {
        options.addClass = 'ow_photoview_overlay floatbox_overlayBG';
        options.layout = 'empty';
    }

    _elements.photoFB = new OW_FloatBox(options);
    _elements.photoFB.bind('show', function()

```

```

                                photo
{
    _vars.fbHasContent = true;
});
_elements.photoFB.bind('close', function()
{
    _methods.history.goInitState();
    _methods.beforeLoadCmp(_vars.photoId);

    delete _elements.photoFB;
    delete _elements.content;
    delete _vars.direction;
    delete _vars.photoId;

    _vars.minSize.width = 400;
    _vars.minSize.height = 400;
    _vars.fbHasContent = false;

    _vars.cmpListForDeleting.forEach(function( item )
    {
        _methods.deletePhotoCmp(item);
    });
    _vars.cmpListForDeleting.length = 0;

    if ( _methods.fullscreen.isFullscreen() )
    {
        _methods.fullscreen.exit();
    }

    OW.trigger('photo.onFloatboxClose', [], _methods);
});
},
getDataToSend: function( photoId, listType )
{
    var dataToSend = {photos: [], photoId: photoId, layout: _vars.layout},
params = {};
    var index = _vars.globalList.indexOf(photoId);

    if ( index === -1 )
    {
        var listIndex = -1, keyInListIndex = -1;

        _vars.lists.some(function( index, i )
        {
            listIndex = i;

            return (keyInListIndex = index.indexOf(photoId)) !== -1;
        });
    }

```

```

        photo
    if ( listIndex === -1 || keyInListIndex === -1 )
    {
        params.photoId = photoId;

        dataToSend.photoId = photoId;
        dataToSend.loadPrevList = 1;
        dataToSend.loadPrevPhoto = 1;
        dataToSend.loadNextList = 1;
        dataToSend.loadNextPhoto = 1;
        dataToSend.photos.push(photoId);
    }
    else
    {
        var list = _vars.lists.slice(0);
        list.push(_vars.globalList);

        params.prevId = keyInListIndex === 0 ?
            _methods.arrayUtils.getMinInList(list) :
            _vars.lists[listIndex][keyInListIndex - 1];
        params.nextId = keyInListIndex === _vars.lists[listIndex].length
- 1 ?
            _methods.arrayUtils.getMaxInList(list) :
            _vars.lists[listIndex][keyInListIndex + 1];
        params.listIndex = listIndex;

        if ( _vars.hasOwnProperty('direction') )
        {
            switch ( _vars.direction )
            {
                case 'left':
                    dataToSend.loadPrevList = keyInListIndex <
_vars.ID_LIST_LIMIT ? 1 : 0;
                    break;
                case 'right':
                default:
                    dataToSend.loadNextList =
_vars.lists[listIndex].length - keyInListIndex < _vars.ID_LIST_LIMIT ? 1 : 0;
                    break;
            }
        }
        else
        {
            dataToSend.loadPrevList = keyInListIndex <
_vars.ID_LIST_LIMIT ? 1 : 0;
            dataToSend.loadNextList = _vars.lists[listIndex].length -
keyInListIndex < _vars.ID_LIST_LIMIT ? 1 : 0;
        }
    }
}

```

```

                                photo
        }
        else
        {
            params.prevId = _methods.getPrevId(index, _vars.globalList);
            params.nextId = _methods.getNextId(index, _vars.globalList);

            if ( _vars.hasOwnProperty('direction') )
            {
                switch ( _vars.direction )
                {
                    case 'left':
                        dataToSend.loadPrevList =
                        _methods.getIsLoadPrevList(index) ? 1 : 0;
                        break;
                    case 'right':
                    default:
                        dataToSend.loadNextList =
                        _methods.getIsLoadNextList(index, _vars.globalList) ? 1 : 0;
                        break;
                }
            }
            else
            {
                dataToSend.loadPrevList = _methods.getIsLoadPrevList(index) ? 1
: 0;
                dataToSend.loadNextList = _methods.getIsLoadNextList(index,
_vars.globalList) ? 1 : 0;
            }
        }

        if ( !_methods.isCached(photoId) )
        {
            params.photoId = photoId;
            params.loadCurrent = true;

            dataToSend.photos.push(photoId);
        }

        if ( !_methods.isCached(params.prevId) )
        {
            dataToSend.photos.push(params.prevId);
        }

        if ( !_methods.isCached(params.nextId) )
        {
            dataToSend.photos.push(params.nextId);
        }
    }

```

```

                                photo
if ( ['hash', 'user', 'desc', 'all'].indexOf(listType) !== -1 )
{
    var listData = {};

    if ( window.hasOwnProperty('browsePhoto') )
    {
        listData = window.browsePhoto.getListData();
    }
    else
    {
        var searchParse =
_methods.locationUtils.parseSearch(location.search.substr(1));

        if ( searchParse.hasOwnProperty('searchVal') )
        {
            listData = searchParse;
        }
    }

    $.extend(dataToSend, listData);
}

return {
    dataToSend: dataToSend,
    params: params
};
},
showPhotoCmp: function( photoId, listType, data, photo )
{
    OW.trigger('photo.onBeforeShow', [photoId, listType], _methods);

    if ( photoId !== _vars.photoId )
    {
        OW.trigger('photo.onBeforePhotoChange', [_vars.photoId, photoId],
_methods);
    }

    _methods.showFloatBox();

    if ( _vars.isDisabled && _vars.layout !== 'page' )
    {
        _elements.photoFB.setContent($(document.getElementById('ow-photo-view-error')));
        _elements.photoFB.fitWindow({width: 500, height: 300});

        return;
    }
}

```

```

                                photo
    _methods.beforeLoadCmp(_vars.photoId);
    _vars.photoId = +photoId;
    _vars.listType = listType || _vars.DEFAULT_LIST;

    if ( data && data.main && data.main.length === 2 && data.mainUrl )
    {
        _methods.createFloatboxContent();
        _elements.photoFB.setContent(_elements.content);

        var dimension = _methods.getScreenDimension(data.main[0],
data.main[1]),
            imgCss = {}, floatboxCss = {width: dimension.width +
_vars.infoBar, height: dimension.height}, stageCss = {height: dimension.height};
        var proportion = _methods.utils.getScreenProportion(dimension.width,
dimension.height, data.main[0], data.main[1]);

        if ( proportion[0] > dimension.imageWeight && proportion[1] >
dimension.imageHeight )
        {
            imgCss.width = dimension.imageWeight;
            imgCss.height = dimension.imageHeight;
        }
        else
        {
            imgCss.width = proportion[0];
            imgCss.height = proportion[1];
        }

        if ( _vars.isClassic )
        {
            stageCss.float = 'none';
            floatboxCss.height = 0;
        }
        else
        {
            floatboxCss.top = dimension.screenHeight / 2 - dimension.height
/ 2;
            stageCss.width = dimension.width;

            _elements.content.find('.ow_photoview_info_wrap').height(floatboxCss.height);
        }

        OW.trigger('photo.setStage', [photo, data, stageCss, imgCss],
_elements.content);

        _elements.photoFB.fitWindow(floatboxCss);
        _elements.content.find('.ow_photoview_stage_wrap').css(stageCss);
        _elements.content.find('img.ow_photo_view').css(imgCss).attr('src',

```

```

                                photo
data.mainUrl);
    }

    _methods.setState(true);

    var send = _methods.getDataToSend(_vars.photoId, _vars.listType);

    if ( _methods.isCached(send.dataToSend.photoId) )
    {
        _methods.loadCachedComponent(send.dataToSend.photoId);
    }

    _methods.backgroundLoadPhotos([send.params.nextId, send.params.prevId],
    _methods.fullscreen.isFullscreen() ? 'fullscreen' : 'main');

    if ( _methods.utils.isEmptyObject(send.dataToSend) )
    {
        $.extend(send.dataToSend, {ajaxFunc: 'getFloatbox', listType:
    _vars.listType});
        _methods.fetchCmp(send.dataToSend, send.params);
    }
    },
    fetchCmp: function( dataToSend, params )
    {
        _vars.fetchRequest = $.ajax(
        {
            url: _vars.ajaxResponder,
            type: 'POST',
            cache: false,
            data: dataToSend,
            dataType: 'json',
            beforeSend: function( jqXHR, settings )
            {
                OW.trigger('photo.onBeforeLoadPhoto', [dataToSend.photoId],
    _methods);
            },
            success : function( response )
            {
                if ( !response || !response.result )
                {
                    return;
                }

                OW.trigger('photo.onRequestComplete', [dataToSend, params,
response], _methods);
                utils.includeScriptAndStyle(response.scripts);
                _methods.cachePhotoComponents(response.photos);
            }
        }
    );
}

```

```

        photo
    if ( params.loadCurrent )
    {
        _methods.loadCachedComponent(params.photoId);
    }

    if ( response.prevCompleted && !_vars.isPrevCompleted )
    {
        _vars.isPrevCompleted = true;
        _methods.arrayUtils.push(_vars.lists,
_methods.arrayUtils.unionToList(response.firstList));
    }

    if ( response.nextCompleted && !_vars.isNextCompleted )
    {
        _vars.isNextCompleted = true;
        _methods.arrayUtils.push(_vars.lists,
_methods.arrayUtils.unionToList(response.lastList));
    }

    var list = _methods.arrayUtils.unionToList(response.prevList,
params.photoId, response.nextList);

    if ( _vars.globalList.length === 0 )
    {
        _vars.globalList = list;
    }
    else
    {
        if ( _methods.arrayUtils.isMatch(_vars.globalList, list) )
        {
            _vars.globalList =
_methods.arrayUtils.unionToList(_vars.globalList, list);
        }
        else
        {
            if ( params.listIndex )
            {
                _vars.lists[params.listIndex] =
_methods.arrayUtils.unionToList(_vars.lists[params.listIndex], list);
                _vars.lists.sort(_methods.arrayUtils.sortBySum);
            }
            else
            {
                _methods.arrayUtils.push(_vars.lists, list);
            }

            while ( true )
            {

```

```

        photo
var listIndex;
var match = _vars.lists.some(function( item, i )
{
    listIndex = i;

    return
_methods.arrayUtils.isMatch(_vars.globallist, item);
});

if ( match )
{
    _vars.globallist =
_methods.arrayUtils.unionToList(_vars.globallist, _vars.lists[listIndex]);
    _vars.lists.splice(listIndex, 1);
}
else
{
    var index, innerIndex;
    var some = _vars.lists.some(function( item, i )
    {
        index = i;

        for ( var j = i + 1; j < _vars.lists.length;
j++ )
        {
            innerIndex = j;

            if ( _vars.lists[j][0] + 1 >=
item[item.length -1] )
            {
                return true;
            }
        }
    });

    if ( some )
    {
        _vars.lists[index] =
_methods.arrayUtils.unionToList(_vars.lists[index], _vars.lists[innerIndex]);
        _vars.lists.splice(innerIndex);
    }
    else
    {
        break;
    }
}
}
}

```

```

        photo
    }
    },
    error: function( jqXHR, textStatus, errorThrown )
    {
        throw textStatus;
    },
    complete: function( jqXHR, textStatus )
    {
    }
    });
},
loadCachedComponent: function( photoId )
{
    _methods.createFloatboxContent();
    OW.trigger('photo.onBeforeLoadFromCache', [photoId], _methods);

    if ( _vars.layout === 'page' )
    {
        var contant = $(document.getElementById('ow-photo-view-page'));

        if ( contant.is(':empty') )
        {
            contant.html(_elements.content);
        }
    }
    else if ( _elements.photoFB.$body[0] &&
!_elements.photoFB.$body[0].children.length )
    {
        _elements.photoFB.setContent(_elements.content);

        if ( _vars.isClassic )
        {
            _elements.photoFB.fitWindow({width: _vars.minSize.width});
        }
        else
        {
            _elements.photoFB.fitWindow({width: _vars.minSize.width, height:
_vars.minSize.height});
        }
    }

    _methods.setContent(photoId);
    _methods.setComment(photoId);
    _methods.addCmpMarkup(photoId);

    OW.trigger('photo.markupReady', [_methods.getPhotoCmp(photoId)],
_elements.content);
}

```

```

        photo

        _methods.fitSize(photoId, function( offset )
        {
            OW.trigger('photo.beforeResize', [photoId], _methods);

            if ( !_methods.fullscreen.isFullscreen() )
            {
                if ( _vars.layout !== 'page' )
                {
                    var options = {};

                    if ( _vars.isClassic )
                    {

                        _elements.content.find('.ow_photoview_stage_wrap').css({float: 'none', height:
                        _vars.minSize.height});

                        options.width = offset.width;
                    }
                    else
                    {

                        _elements.content.find('.ow_photoview_stage_wrap').css({width: _vars.minSize.width,
                        height: _vars.minSize.height});

                        _elements.content.css('height', _vars.minSize.height);
                        options = offset;
                    }

                    _elements.photoFB.fitWindow(options);
                    _methods.updateComment({entityType: 'photo_comments',
                    entityId: _vars.photoId});
                }
                else
                {

                    _elements.content.find('.ow_photoview_stage_wrap').css({float: 'none', height:
                    _vars.minSize.height});
                }
            }

            OW.bindAutoClicks(_elements.content);
            OW.bindTips(_elements.content);

            OW.trigger('photo.afterResize', [photoId], _methods);
            OW.trigger('photo.photo_show', {photoId: photoId}, _methods);

            _methods.setState(false);
        });
    },

```

```

                                photo
fitSize: function( photoId, callback, resize )
{
    if ( !_methods.isCached(photoId) )
    {
        return;
    }

    var cmp = _methods.getPhotoCmp(photoId);
    var complete = function()
    {
        if ( !_vars.isRunSlideshow )
        {
            var img =
_elements.content.find('img.ow_photo_view').css({width: 'auto', height: 'auto'})[0];

            if ( _methods.fullscreen.isFullscreen() &&
+cmp.photo.hasFullsize )
            {
                img.src = cmp.photo.urlFullscreen;
            }
            else
            {
                img.src = cmp.photo.url;
                img.alt = escape(cmp.photo.description);
            }
        }

        var imageWeight = this.naturalWidth;
        var imageHeight = this.naturalHeight;

        var screnWeight = _vars.layout === 'page' ?
$(document.getElementById('ow-photo-view-page')).width() : $(window).width();
        var screnHeight = $(window).height();

        var minWidth = screnWeight < 400 ? (screnWeight - 2 * _vars.spacing)
: 400;
        var minHeight = screnHeight < 400 ? (screnHeight - 2 *
_vars.spacing) : 400;

        var maxWidth = _vars.layout === 'page' ? screnWeight : (screnWeight
- 2 * _vars.spacing - 400);
        var maxHeight = screnHeight - 2 * _vars.spacing;

        if ( _vars.layout === 'page' )
        {
            callback({width: maxWidth, height: maxHeight});

            return;

```

```

        photo
    }

    var width = (imageWeight > minWidth && imageWeight < maxWidth) ?
imageWeight : (imageWeight < minWidth) ? minWidth : maxWidth;
    var height = (imageHeight > minHeight && imageHeight < maxHeight) ?
imageHeight : (imageHeight < minHeight) ? minHeight : maxHeight;

    if ( resize )
    {
        _vars.minSize.width = (_vars.minSize.width + _vars.infoBar >
screnWeight) ? maxWidth : (width <= _vars.minSize.width) ? _vars.minSize.width :
(width >= maxWidth) ? maxWidth: width;
        _vars.minSize.height = (_vars.minSize.height > screnHeight) ?
maxHeight : (height <= _vars.minSize.height) ? _vars.minSize.height : (height >=
maxHeight) ? maxHeight : height;
    }
    else
    {
        _vars.minSize.width = (_vars.minSize.width + _vars.infoBar >
screnWeight) ? maxWidth : (width > _vars.minSize.width) ? width :
_vars.minSize.width;
        _vars.minSize.height = (_vars.minSize.height > screnHeight) ?
maxHeight : (height > _vars.minSize.height) ? height : _vars.minSize.height;
    }

    OW.trigger('photo.fitSize', [_methods.getPhotoCmp(photoId),
_vars.minSize], _elements.content);

    width = _vars.minSize.width + _vars.infoBar;
    height = _vars.minSize.height;

    callback({width: width, height: height, top: screnHeight / 2 -
(_vars.minSize.height / 2)});
};

var img = new Image();

if ( _methods.fullscreen.isFullscreen() && +cmp.photo.hasFullsize )
{
    img.src = cmp.photo.urlFullscreen;
}
else
{
    img.src = cmp.photo.url;
}

img.complete ? complete.call(img) : img.onload = complete;
},

```

```

                                photo
    getScreenDimension: function( imgWidth, imgHeight, screenMinWidth,
screenMinHeight )
    {
        var screenWidth = _vars.layout === 'page' ?
$(document.getElementById('ow-photo-view-page')).width() : $(window).width(),
        screenHeight = $(window).height(),

        minWidth = screenMinWidth || 400,
        minHeight = screenMinHeight || 400,

        maxWidth = _vars.layout === 'page' ? screenWidth : (screenWidth - 2
* _vars.spacing - _vars.infoBar),
        maxHeight = screenHeight - 2 * _vars.spacing,

        width = imgWidth > maxWidth ? maxWidth : imgWidth < minWidth ?
minWidth : imgWidth,
        height = imgHeight > maxHeight ? maxHeight : imgHeight < minHeight
? minHeight : imgHeight;

        return {
            imageWidth: imgWidth,
            imageHeight: imgHeight,
            maxWidth: maxWidth,
            maxHeight: maxHeight,
            width: width,
            height: height,
            minWidth: minWidth,
            minHeight: minHeight,
            screenWidth: screenWidth,
            screenHeight: screenHeight
        };
    },
    setContent: function( photoId )
    {
        var content = _elements.content;
        var cmp = _methods.getPhotoCmp(photoId);

        $('.ow_user_list_picture a', content).attr('href', cmp.avatar.url);
        $('.ow_user_list_picture img', content).attr({alt: cmp.avatar.title,
src: cmp.avatar.src}).show();
        $('.ow_photoview_albumlink', content).attr('href', cmp.albumUrl).html(
            "'" + utils.truncate(cmp.album.name) + "': " + cmp.photoIndex + ' '
+ OW.getLanguageText('photo', 'of') + ' ' + cmp.photoCount
        );
        $('.ow_user_list_data a.ow_photo_avatar_url', content).attr('href',
cmp.avatar.url).html(cmp.avatar.title);
        $('.ow_user_list_data .ow_timestamp',
content).html(cmp.photo.addDatetime);
    }

```

```

                                photo
                                $( '.ow_user_list_data .ow_photo_album_url', content ).attr( 'href',
cmp.albumUrl )
                                .find( '.ow_photo_album_name' ).html(
                                OW.getLanguageText( 'photo', 'album' ) + ': ' +
                                utils.truncate( cmp.album.name, 60 )
                                );

                                if ( cmp.contextAction.length )
                                {

$(cmp.contextAction).addClass('ow_photo_context_action').insertAfter($('ow_photovie
w_bottom_menu_wrap', content)).show();
                                }

                                $( '.ow_photo_share', content ).empty().html( function()
                                {
                                    return cmp.share || '';
                                } );

                                if ( !_methods.fullscreen.enabled() )
                                {
                                    $( '.ow_photoview_fullscreen', content ).attr( { target: '_blank', href:
(cmp.photo.urlFullscreen ? cmp.photo.urlFullscreen : cmp.photo.url) } );
                                }

                                $( '#btn-photo-edit' ).off().on( 'click', function()
                                {
                                    var photoId = $( this ).attr( 'rel' );

                                    _elements.editFB = OW.ajaxFloatBox( 'PHOTO_CMP_EditPhoto', { photoId:
photoId }, { width: 580, iconClass: 'ow_ic_edit', title: OW.getLanguageText( 'photo',
'tb_edit_photo' ),
                                    onLoad: function()
                                    {
                                        owForms[ 'photo-edit-form' ].bind( 'success', function( data )
                                        {
                                            _elements.editFB.close();

                                            if ( data && data.result )
                                            {
                                                if ( data.photo.status !== 'approved' )
                                                {
                                                    OW.info( data.msgApproval );

                                                    if ( _elements.photoFB )
                                                    {
                                                        _elements.photoFB.close();
                                                    }
                                                }
                                            }
                                        }
                                    }
                                }

```

photo

```
$('#photo-description').html(OW.getLanguageText('photo', 'pending_approval'));

        if ( window.hasOwnProperty('browsePhoto') )
        {
            browsePhoto.removePhotoItems(['photo-item-'
+ photoId]);
        }
    }
    else
    {
        OW.info(data.msg);

        cmp.albumUrl = data.albumUrl;
        cmp.album.name = data.albumName;
        cmp.photo.description = data.description;

        $('#ow_user_list_data .ow_photo_album_url',
content).attr('href', data.albumUrl)
            .find('.ow_photo_album_name').html(
                OW.getLanguageText('photo', 'album') +
                ': ' + utils.truncate(data.albumName, 60)
            );

        var event = {text:
utils.descToHashtag(cmp.photo.description, utils.getHashtags(cmp.photo.description),
params.tagUrl)};

        OW.trigger('photo.onSetDescription', event);
        $('#ow_photoview_description span',
content).html(event.text);

        if ( window.hasOwnProperty('browsePhoto') )
        {
            browsePhoto.updateSlot('photo-item-' +
data.id, data);

            browsePhoto.reorder();
        }
    }
}
else if ( data.msg )
{
    OW.error(data.msg);
}

delete _elements.editFB;
});
}}
```

```

                                photo
        });
    });

    $('#photo-mark-featured').off().on('click', function()
    {
        var status = $(this).attr('rel');
        var photoId = $(this).attr('photo-id');
        var $this = $(this);

        $.ajax(
        {
            url: _vars.ajaxResponder,
            type: 'POST',
            data: {ajaxFunc: 'ajaxSetFeaturedStatus', entityId: photoId,
status: status},
            dataType : 'json',
            success : function(data)
            {
                if ( data.result === true )
                {
                    var newStatus = status === 'remove_from_featured' ?
'mark_featured' : 'remove_from_featured';
                    var newLabel = status === 'remove_from_featured' ?
OW.getLanguageText('photo', 'mark_featured') : OW.getLanguageText('photo',
'remove_from_featured');
                    $this.html(newLabel);
                    $this.attr('rel', newStatus);
                    _vars.cmpListForDeleting.push(+photoId);
                    OW.info(data.msg);
                }
                else if ( data.error != undefined )
                {
                    OW.warning(data.error);
                }
            }
        });
    });

    $("#photo-delete").off().on('click', function( event )
    {
        var photoId = $(this).attr("rel");

        if ( confirm(OW.getLanguageText('photo', 'confirm_delete')) )
        {
            $.ajax(
            {
                url: _vars.ajaxResponder,
                type: 'POST',

```

```

        photo
        data: {ajaxFunc: 'ajaxDeletePhoto', entityId: photoId},
        dataType: 'json',
        success: function( data )
        {
            if ( data.result === true )
            {
                OW.info(data.msg);
                if ( data.url )
                {
                    document.location = data.url;
                }
            }
            else if ( data.error != undefined )
            {
                OW.warning(data.error);
            }
        }
    });
});

$('#btn-photo-flag').off().on('click', function()
{
    OW.flagContent('photo_comments', $(this).attr("rel"));
});

$('#photo-approve').off().on('click', function()
{
    $.ajax({
        url: $(this).attr('url'),
        cache: false,
        type: 'POST',
        dataType: 'json',
        success: function( data )
        {
            // TODO show message
            OW.info(OW.getLanguageText('photo',
'approve_success_message'));
            $('#photo-approve').parent().remove();
        }
    })
});

if ( !_methods.fullscreen.enabled() )
{
    _elements.content.find('.ow_photoview_fullscreen').attr({target:
'_blank', href: cmp.photo.url});
}

```

photo

```
    if ( cmp.photo.status !== 'approved' )
    {
        $($('.ow_photoview_description span',
content).html(OW.getLanguageText('photo', 'pending_approval')));

        return;
    }

    var event = {text: utils.descToHashtag(cmp.photo.description,
utils.getHashtags(cmp.photo.description), params.tagUrl)};
    OW.trigger('photo.onSetDescription', event);
    $($('.ow_photoview_description span', content).html(event.text);

    var userScore = cmp.userScore;
    var rateInfo = cmp.rateInfo;
    var rateItems = $($('.ow_rates_wrap',
content).show().find('.rate_item').off());

    if ( +userScore > 0 )
    {
        content.find('.rate_title').html(
            OW.getLanguageText('photo', 'rating_your', {count:
rateInfo.rates_count, score: userScore})
        );
    }
    else
    {
        content.find('.rate_title').html(
            OW.getLanguageText('photo', 'rating_total', {count:
rateInfo.rates_count})
        );
    }

    content.find('.active_rate_list').css('width', (rateInfo.avg_score * 20)
+ '%');

    var eventRate = {
        canRate: true,
        photo: cmp.photo
    };
    OW.trigger('photo.canRate', eventRate, this.node);

    if ( !eventRate.canRate )
    {
        rateItems.on('click', function( e )
        {
            e.stopPropagation();
```

photo

```
var event = {
    msg: ''
};
OW.trigger('photo.canRateMessage', event, self.node);
OW.error(event.msg);
});
}
else
{
    rateItems.on('click', function()
    {
        var ownerError;

        if ( +_vars.rateUserId === 0 )
        {
            OW.error(OW.getLanguageText('base',
'rate_cmp_auth_error_message'));

            return;
        }
        else if ( +cmp.ownerId === +_vars.rateUserId )
        {
            OW.error(OW.getLanguageText('base',
'rate_cmp_owner_cant_rate_error_message'));

            return;
        }

        var event = {
            canRate: true,
            photo: cmp.photo
        };
        OW.trigger('photo.canRate', event, this.node);

        if ( !event.canRate ) return;

        var rate = $(this).index() + 1;

        $.ajax({
            url: _vars.ajaxResponder,
            dataType: 'json',
            data: {
                ajaxFunc: 'ajaxRate',
                entityId: cmp.photo.id,
                rate: rate,
                ownerId: cmp.ownerId
            },
```

```

        photo
        cache: false,
        type: 'POST',
        success: function( result, textStatus, jqXHR )
        {
            if ( result )
            {
                switch ( result.result )
                {
                    case true:
                        OW.info(result.msg);

content.find('.active_rate_list').css('width', (result.rateInfo.avg_score * 20) +
'%');

                        content.find('.rate_title').html(
                            OW.getLanguageText('photo',
'rating_your', {count: result.rateInfo.rates_count, score: rate})
                        );

                        cmp.userScore = rate;
                        rateInfo.avg_score =

result.rateInfo.avg_score;

                        rateInfo.rates_count =

result.rateInfo.rates_count;

                        OW.trigger('photo.onSetRate', {
                            entityId: cmp.photo.id,
                            userScore: cmp.userScore,
                            avgScore: rateInfo.avg_score,
                            ratesCount: rateInfo.rates_count
                        }, _methods);
                        break;
                    case false:
                    default:
                        OW.error(result.error);
                        break;
                }
            }
            else
            {
                OW.error('Server error');
            }
        },
        error: function( jqXHR, textStatus, errorThrown )
        {
            throw textStatus;
        }
    });

```

```

                                photo
        })
        .hover(function()
        {
            $(this).prevAll().add(this).addClass('active');
            content.find('.active_rate_list').css('width', '0px');
        }, function()
        {
            $(this).prevAll().add(this).removeClass('active');
            content.find('.active_rate_list').css('width',
(rateInfo.avg_score * 20) + '%');
        });
    }
},
setComment: function( photoId )
{
    var cmp = _methods.getPhotoCmp(photoId);

    $('ow_feed_comments', _elements.content).append(cmp.comment);

    _vars.commentEvents.forEach(function( item )
    {
        OW.bind(item, function( data )
        {
            _methods.updateComment(data, item);
        });
    });
},
updateComment: function( data, eventName )
{
    if ( !_elements.hasOwnProperty('photoFB') ||
(_vars.commentUnknownEntity.indexOf(eventName) === -1 && (data.entityType !==
'photo_comments' || data.entityId !== _vars.photoId)) )
    {
        return;
    }

    var height, cmp = _methods.getPhotoCmp(_vars.photoId),
        scrollCont = _elements.content.find('ow_photo_scroll_cont'),
        padding =
parseInt(_elements.content.find('ow_photoview_info').css('padding-top')),
        appendTo = _elements.content.find('ow_feed_comments_input_sticky');

    if ( eventName == 'base.comment_textarea_resize' )
    {
        var textarea;

        if ( !cmp.isMoved )
        {

```

```

        photo
        textarea = $('textarea', scrollCont)[0];
    }
    else
    {
        textarea = $('textarea', appendTo)[0];
    }

    if ( textarea.selectionStart || textarea.selectionStart == '0' )
    {
        _vars.curPos = textarea.selectionStart;
    }
    else if ( document.selection )
    {
        textarea.focus();

        var sel = document.selection.createRange();

        sel.moveStart('character', -textarea.value.length);
        _vars.curPos = sel.text.length;
    }
}

if ( !_vars.isClassic && _vars.layout !== 'page' )
{
    OW.addScroll(scrollCont, {autoReinitialise: true}).scrollToBottom();

    scrollCont.css('height', '');

    var jsp = scrollCont.data('jsp'), scrollContHeight =
jsp.getContentHeight();

    if ( _methods.fullscreen.isFullscreen() )
    {
        height =
_elements.content.find('.ow_photoview_info_wrap').height();
    }
    else
    {
        height =
_elements.content.find('.ow_photoview_info_wrap').height() - appendTo.height() -
padding;
    }

    if ( scrollContHeight >= height )
    {
        if ( !cmp.isMoved )
        {
            OW.trigger('base.move_comments_form',

```

```

        photo
    {
        entityType: 'photo_comments',
        entityId: data.entityId || _vars.photoId,
        customId: cmp.customId,
        appendTo: appendTo
    });
    cmp.isMoved = true;

_elements.content.find('.ow_photoview_info_wrap').addClass('sticked');
}

scrollCont.height(_elements.content.find('.ow_photoview_info_wrap').height() -
appendTo.height() - padding);
jsp.reinitialise();
jsp.scrollToBottom();

    if ( eventName == 'base.comment_textarea_resize' )
    {
        $('textarea', appendTo).photoFocus();
    }
}
else
{
    if ( jsp != null && jsp.getContentHeight() >= height )
    {

_elements.content.find('.ow_photo_scroll_cont').height(height);
    }
    else
    {
        if ( cmp.isMoved )
        {
            OW.trigger('base.move_comments_form',
            {
                entityType: 'photo_comments',
                entityId: data.entityId,
                customId: cmp.customId
            });
            cmp.isMoved = false;
        }

        OW.removeScroll(scrollCont);

_elements.content.find('.ow_photo_scroll_cont').css('height', 'auto');
    }

    if ( eventName == 'base.comment_textarea_resize' )

```

```

                                photo
                                {
                                    $('textarea',
_elements.content.find('.ow_photo_scroll_cont')).photoFocus();
                                }
                            }

                            if ( jsp && ['base.comment_added',
'base.comment_textarea_resize'].indexOf(eventName) !== -1 )
                            {
                                setTimeout(function()
                                {
                                    jsp.scrollToBottom(true);
                                }, 400);

                                _elements.content.find('.ow_feed_comments_input_sticky
textarea').photoFocus();
                            }

                            if ( ['base.comment_add', 'base.comment_delete'].indexOf(eventName) !==
-1 )
                            {
                                _vars.cmpListForDeleting.push(+data.entityId);
                            }
                        },
loadPhoto: function( direction, listType )
{
    if ( _methods.isLoading() )
    {
        return;
    }

    _vars.direction = (['left', 'right'].indexOf(direction) === -1) ?
'right' : direction;

    var data = _methods.getDataToSend(_vars.photoId, _vars.listType),
photoId, extend;

    switch ( direction )
    {
        case 'left':
            photoId = +data.params.prevId;
            break;
        case 'right':
            photoId = +data.params.nextId;
            break;
    }
}

```

```

                                photo
    if ( isNaN(photoId) )
    {
        return false;
    }

    OW.trigger('photo.onPhotoSwitch', [photoId, _vars.direction], _methods);

    if ( window.hasOwnProperty('browsePhoto') )
    {
        extend = browsePhoto.getMoreData();
    }

    _methods.setId(photoId, listType, extend);
    _methods.updateSlideshow();
},
loadPrevPhoto: function()
{
    _methods.loadPhoto('left', _vars.listType);
},
loadNextPhoto: function()
{
    _methods.loadPhoto('right', _vars.listType);
},
addCmpMarkup: function( photoId )
{
    if ( !_methods.isCached(photoId) )
    {
        return;
    }

    var cmp = _methods.getPhotoCmp(photoId);

    if ( cmp.css )
    {
        OW.addCss(cmp.css);
    }

    if ( cmp.cssFiles )
    {
        $.each(cmp.cssFiles, function( key, value )
        {
            OW.addCssFile(value);
        });
    }

    if ( cmp.scriptFiles )
    {
        OW.addScriptFiles(cmp.scriptFiles, function()

```

```

                                photo
        {
            if ( cmp.onloadScript )
            {
                OW.addScript(cmp.onloadScript);
            }
        });
    }
    else
    {
        if ( cmp.onloadScript )
        {
            OW.addScript(cmp.onloadScript);
        }
    }

    if ( cmp.meta )
    {
        Object.keys(cmp.meta).forEach(function( item )
        {
            var _meta = cmp.meta[item];

            Object.keys(_meta).forEach(function( attr )
            {
                $('meta[' + item + '="' + attr + '"').remove();
                $(document.head).append('<meta ' + item + '="' + attr + '"
content="" + _meta[attr] + '>');
            });
        });
    }
},
beforeLoadCmp: function( photoId )
{
    if ( _methods.isCached(photoId) )
    {
        var cmp = _methods.getPhotoCmp(photoId);

        if ( cmp.isMoved )
        {
            if ( _vars.fbHasContent )
            {
                OW.removeScroll(_elements.content.find('.ow_photo_scroll_cont'));

                if ( cmp.isMoved )
                {
                    OW.trigger('base.move_comments_form',
                    {

```

```

        photo
        entityType: 'photo_comments',
        customId: cmp.customId,
        entityId: photoId
    });
    cmp.isMoved = false;

    _elements.content.find('.ow_photoview_info_wrap').removeClass('sticked');
    }
    }

    OW.trigger('base.comments_destroy',
    {
        entityType: 'photo_comments',
        customId: cmp.customId,
        entityId: photoId
    });

    $('.ow_feed_comments', _elements.content).empty();
    _elements.content.find('.ow_photoview_info_wrap').height('');
    $('.ow_rates_wrap', _elements.content).hide();

    var index;

    if ( (index = _vars.cmpListForDeleting.indexOf(photoId)) !== -1 )
    {
        _methods.deletePhotoCmp(photoId);
        _vars.cmpListForDeleting.splice(index, 1);
    }

    $('.ow_photo_context_action', _elements.content).remove();
    }

    ['base.move_comments_form',
'base.comments_destroy'].concat(_vars.commentEvents).forEach(function( item )
    {
        OW.unbind(item);
    });

    if ( _elements.hasOwnProperty('editFB') )
    {
        _elements.editFB.close();
    }
    },
    fullscreen: {
        request: function( elem )
        {
            var keyboardAllowed = typeof Element !== 'undefined' &&
'ALLOW_KEYBOARD_INPUT' in Element;

```

```

                                photo
elem = elem || document.documentElement;

if ( /5\.1[\.\\d]* Safari/.test(navigator.userAgent))
{
    elem[_vars.fullScreen.requestFullscreen]();
}
else
{
    elem[_vars.fullScreen.requestFullscreen](keyboardAllowed &&
Element.ALLOW_KEYBOARD_INPUT);
}
},
exit: function()
{
    document[_vars.fullScreen.exitFullscreen]();
},
isFullscreen: function()
{
    return !!document[_vars.fullScreen.fullscreenElement];
},
enabled: function()
{
    return _vars.fullScreen &&
!!document[_vars.fullScreen.fullscreenEnabled];
},
getFullscreenElement: function()
{
    return document[_vars.fullScreen.fullscreenElement];
},
isPhotoInit : function()
{
    return _methods.fullscreen.getFullscreenElement() ===
document.documentElement || _vars.isPhotoFullscreenInit;
},
onchange: function( event )
{
    if ( !_methods.fullscreen.isPhotoInit() ) return;

    var content = _elements.content, cmp =
_methods.getPhotoCmp(_vars.photoId);

    if ( _methods.fullscreen.isFullscreen() )
    {
        _vars.isPhotoFullscreenInit = true;
        $(window).off('resize.photo');

        content.height('');
    }
}

```

```

                                photo
content.find('.ow_photoview_stage_wrap').addClass('ow_photoview_stage_wrap_fullscreen');
content.find('.ow_photoview_info_wrap').css({height:
screen.height});
content.find('.ow_photoview_play_btn').removeClass('stop');

if ( _vars.layout === 'page' || _vars.isClassic )
{
    $(document.body).addClass('ow_photo_page_fullscreen');
    content.find('.ow_photoview_stage_wrap').css({float: 'left',
width: screen.width, height: screen.height});
content.find('.ow_photoview_info').removeClass('ow_photoview_info_onpage');
}
else
{
    _elements.photoFB.fitWindow({top: '0', width: screen.width,
height: screen.height});
    content.find('.ow_photoview_stage_wrap').css({width:
screen.width, height: screen.height});
}

if ( cmp.photo.hasFullsize && cmp.photo.dimension )
{
    var dimension = cmp.photo.dimension;

    if ( dimension.fullscreen && dimension.fullscreen.length ===
2 )
    {
        var css = {};

        if ( dimension.fullscreen[0] > screen.width &&
dimension.fullscreen[1] > screen.height )
        {
            var proportion =
_methods.utils.getScreenProportion(dimension.fullscreen[0],
dimension.fullscreen[1]);

            css.width = proportion[0];
            css.height = proportion[1];
        }
        else
        {
            css.width = dimension.fullscreen[0];
            css.height = dimension.fullscreen[1];
        }
    }
}

```

```

                                photo
_elements.content.find('img.ow_photo_view').css(css).attr('src', cmp.photo.url);
    _methods.fitSize(_vars.photoId, function(){});
    }
}

if ( _vars.fbHasContent )
{
OW.removeScroll(_elements.content.find('.ow_photo_scroll_cont'));
    }
else
{
    _vars.isPhotoFullscreenInit = false;
    _vars.isRunSlideshow = false;
    _methods.stopSlideshow();

content.find('.ow_photoview_info_btn').removeClass('close').addClass('open');

content.find('.ow_photoview_stage_wrap').removeClass('ow_photoview_stage_wrap_fullscreen');

    content.find('.ow_photoview_info_wrap').css({height: ''});
    content.find('.ow_photoview_play_btn').addClass('stop');

    if ( _vars.layout === 'page' || _vars.isClassic )
    {

_elements.content.find('.ow_photoview_stage_wrap').css({float: 'none'});

_elements.content.find('.ow_photoview_info').addClass('ow_photoview_info_onpage');
    $(document.body).removeClass('ow_photo_page_fullscreen');
    }

    if ( _vars.isClassic )
    {

_elements.content.find('.ow_photoview_info').addClass('ow_photoview_pint_mode');
    }

    setTimeout(function()
    {
        var dimension = _methods.getScreenDimension(), css = {};

        if ( _vars.isClassic )
        {
            css.width = 'auto';
            css.height = dimension.maxHeight;

```

```

                                photo
                                }
                                else
                                {
                                    css.width = dimension.maxWidth;
                                    css.height = dimension.maxHeight;
                                    _vars.minSize.width = dimension.maxWidth;
                                    _vars.minSize.height = dimension.maxHeight;
                                }

                                content.find('.ow_photoview_stage_wrap').css(css);
                                _methods.updateComment({entityType: 'photo_comments',
entityId: _vars.photoId});

                                if ( _vars.layout != 'page' )
                                {
                                    content.height(css.height);
                                    _elements.photoFB.fitWindow({width: dimension.maxWidth +
_vars.infoBar, height: dimension.maxHeight, top: dimension.screenHeight / 2 -
(dimension.maxHeight / 2)});
                                }

                                $(window).on('resize.photo', _methods.resizeWindow);
                                }, 100);
                            }

                            content.find('.jspPane').css('left', 0);
                            content.find('.ow_photoview_info_wrap').height('');

                            var dataToSend = _methods.getDataToSend(_vars.photoId,
_vars.listType);

                            _methods.backgroundLoadPhotos([dataToSend.params.nextId,
dataToSend.params.prevId], _methods.fullscreen.isFullscreen() ? 'fullscreen' :
'main');

                            OW.trigger('photo.onSizeChange', [_vars.photoId,
_methods.fullscreen.isFullscreen()], _methods);
                        }
                    },
                    utils:
                    {
                        showPreloader: function()
                        {
                            if ( _vars.layout != 'page' || _elements.photoFB )
                            {
                                if ( !_elements.photoFB.$canvas.hasClass('ow_floatbox_loading')
)

```

```

                                photo
_elements.photoFB.$canvas.prepend(_elements.preloader.css('left', $(window).width()
/ 2 - 160));
    }
},
hidePreloader: function()
{
    _elements.preloader.detach();
},
isEmptyObject: function( object )
{
    var keys;

    if ( !object || object !== Object(object) || (keys =
Object.keys(object)).length === 0 )
    {
        return false;
    }

    return keys.some(function( item )
    {
        if ( ['photoId', 'layout', 'searchVal', 'id'].indexOf(item) !==
-1 )
        {
            return false;
        }

        switch ( _vars.toString.call(object[item]) )
        {
            case '[object Array]':
                return object[item].length !== 0;
            case '[object Number]':
                return !isNaN(object[item]) && object[item] > 0;
            default:
                return object[item] !== null;
        }
    }
));
},
getScreenProportion: function( width, height, screenWidth, screenHeight
)
{
    screenWidth = screenWidth || window.screen.width;
    screenHeight = screenHeight || window.screen.height;

    var rw = screenWidth / width, rh = screenHeight / height, ratio = rw
> rh ? rw : rh;

    return [Math.round(screenWidth / ratio), Math.round(screenHeight /
ratio)];
}

```

```

        photo
    }
},
arrayUtils:
{
    unionToList: function()
    {
        var arr = [].splice.call(arguments, 0)
            .reduce(function( previousValue, currentItem )
            {
                return previousValue.concat(currentItem);
            }, [])
            .map(function( item )
            {
                return +item;
            })
            .filter(function( item, i, arr )
            {
                return !isNaN(item) && item > 0 && arr.indexOf(item) === i;
            });

        var event = {listType: []};
        OW.trigger('photo.collectListType', event, _methods);

        if ( ['most_discussed',
'toprated'].concat(event.listType).indexOf(_vars.listType) !== -1 )
        {
            return arr;
        }

        return arr.sort(function( a, b )
        {
            return b - a;
        });
    },
    isMatch: function( array1, array2 )
    {
        if ( array1.length === 0 || array2.length === 0 )
        {
            return false;
        }

        return (array1[array1.length - 1] <= array2[0] + 1 || array1[0] + 1
<= array2[array2.length - 1]);
    },
    push: function( dst, src )
    {
        if ( src.length )
        {

```

```

                                photo
                                dst.push(src);
                                dst.sort(_methods.arrayUtils.sortBySum);
                            }
    },
    getMinInList: function( list )
    {
        var min;

        list.forEach(function( item )
        {
            var _min = Math.min.apply(0, item);

            if ( min === undefined )
            {
                min = _min;
            }
            else
            {
                if ( _min < min )
                {
                    min = _min;
                }
            }
        });

        return min;
    },
    getMaxInList: function( list)
    {
        var max;

        list.forEach(function( item )
        {
            var _max = Math.max.apply(0, item);

            if ( max === undefined )
            {
                max = _max;
            }
            else
            {
                if ( _max > max )
                {
                    max = _max;
                }
            }
        });
    });

```

```

                                photo
        return max;
    },
    sortBySum: function( a, b )
    {
        return b.reduce(function( p, c ){return p + c}) - a.reduce(function(
p, c){return p + c});
    }
},
locationUtils: {
    splitPath: function( pathname )
    {
        pathname = pathname || location.pathname.substr(1);

        return pathname.split('/');
    },
    parseSearch: function( search )
    {
        search = search || location.search.substr(1);

        var chunks = search.split('&'), result = {}, chunk;

        chunks.forEach(function( item )
        {
            chunk = item.split('=');

            if ( chunk.length === 2 )
            {
                result[decodeURIComponent(chunk[0])] =
decodeURIComponent(chunk[1]);
            }
        });

        return result;
    },
    compile: function ( uriComponents )
    {
        var keys;

        if ( !uriComponents || uriComponents !== Object(uriComponents) ||
(keys = Object.keys(uriComponents)).length === 0 )
        {
            return '';
        }

        var result = [];

        keys.forEach(function( item )
        {

```

```

                                photo
                                result.push(encodeURIComponent(item) + '=' +
encodeURIComponent(uriComponents[item]));
                                });

                                return result.join('&');
                                },
                                parseUrl: function( url, component )
                                {
                                    var a = document.createElement('a'), result = {};
                                    a.href = url;

                                    component.forEach(function( item )
                                    {
                                        result[item] = a[item];
                                    });

                                    return result;
                                }
                                },
                                history: {
                                    replaceState: function( photoId, listType, extend, data, photo )
                                    {
                                        if ( !photoId || isNaN(+photoId) )
                                        {
                                            return false;
                                        }

                                        listType = listType || _vars.DEFAULT_LIST;

                                        var state;

                                        if ( (state = window.history.state) == null )
                                        {
                                            state = {
                                                initPath: location.pathname,
                                                initSearch: location.search
                                            };
                                        }

                                        var url = _methods.locationUtils.parseUrl(_vars.urlHome,
['pathname'])(['pathname'] + 'photo/view/' + photoId + '/' + listType, search;

                                        if ( url.charAt(0) != '/' )
                                        {
                                            url = '/' + url;
                                        }

                                        if ( extend && (search =

```

```

                                photo
_methods.locationUtils.compile(extend)).length )
    {
        url += '?' + search;
    }

    if ( !_methods.fullscreen.isFullscreen() )
    {
        window.history.replaceState(state, null, url);
    }

    $(window).triggerHandler('popstate.photo', {
        photoId: photoId,
        listType: listType,
        data: data,
        photo: photo
    });
},
popstate: function( event, data )
{
    if ( data && data.photoId && !isNaN(+data.photoId) )
    {
        _methods.showPhotoCmp(+data.photoId, data.listType, data.data,
data.photo);
    }
},
goInitState: function()
{
    var state = window.history.state;
    var url = (state.initPath ? state.initPath :
_methods.locationUtils.parseUrl(_vars.urlHome, ['pathname'])['pathname'] +
'photo/viewlist/' + _vars.DEFAULT_LIST) + state.initSearch;

    window.history.replaceState(null, null, url);
}
},
createFloatboxContent: function()
{
    if ( _elements.hasOwnProperty('content') )
    {
        return _elements.content;
    }

    _elements.content =
$(document.getElementById('ow-photo-view')).clone(true).removeAttr('id');
    _elements.content.find('.ow_photoview_slide_time').slider({min: 3,
max:10, range: 'min',
        slide: function( event, ui )
        {

```

```

                                photo
var tip = $(this).data('owTip');

tip.find('.ow_tip_title').html(OW.getLanguageText('photo',
'slideshow_interval') + ui.value);
_vars.interval = ui.value * 1000;
_methods.updateSlideshow();
    }
});
OW.trigger('photo.onAfterContentRender', [_elements.content], _methods);

return _elements.content;
},
resizeWindow: function(event)
{
    if ( _methods.fullscreen.isFullscreen() ||
!_methods.fullscreen.isPhotoInit() ) return;

    if ( _vars.timerId )
    {
        clearTimeout(_vars.timerId);
    }

    _vars.timerId = setTimeout(function()
    {
        _methods.fitSize(_vars.photoId, function( offset )
        {
            OW.trigger('photo.beforeResize', [_vars.photoId], _methods);

            if ( _vars.layout !== 'page' )
            {
                var options = {};

                if ( _vars.isClassic )
                {
                    _elements.content.find('.ow_photoview_stage_wrap').css({float: 'none', height:
_vars.minSize.height, width: ''});
                    options.width = offset.width;
                }
                else
                {
                    _elements.content.find('.ow_photoview_stage_wrap').css({width: _vars.minSize.width,
height: _vars.minSize.height});
                    _elements.content.css('height', _vars.minSize.height);
                    options = offset;
                }
            }
        });
    }, _vars.interval);
}

```

```

        photo
        _elements.photoFB.fitWindow(options);
    }
    else
    {

_elements.content.find('.ow_photoview_stage_wrap').css({height: offset.height,
width: offset.width});
    }

    _methods.updateComment({entityType : 'photo_comments', entityId:
_vars.photoId});

    OW.trigger('photo.afterResize', [_vars.photoId], _methods);
    }, true);
    }, 100);
},
runSlideshow: function()
{
    _vars.isRunSlideshow = true;
    _vars.slideshowTimeId = setInterval(function()
    {
        if ( _vars.animate )
        {
            return;
        }

        _vars.animate = true;
        var dataToSend = _methods.getDataToSend(_vars.photoId,
_vars.listType);

        if ( !_methods.isCached(dataToSend.params.nextId) )
        {
            _vars.animate = false;
            var dataForNext =
_methods.getDataToSend(dataToSend.params.nextId, _vars.listType);

            if ( _methods.utils.isEmptyObject(dataForNext.dataToSend) )
            {
                $.extend(dataForNext.dataToSend, {ajaxFunc: 'getFloatbox',
listType: _vars.listType});
                dataForNext.params.loadCurrent = false;
                _methods.fetchCmp(dataForNext.dataToSend,
dataForNext.params);
            }
        }
        else
        {
            _methods.setId(dataToSend.params.nextId);

```

photo

```
var nextCmp = _methods.getPhotoCmp(dataToSend.params.nextId),  
image = new Image(), url;
```

```
if ( +nextCmp.photo.hasFullsize )  
{  
    url = nextCmp.photo.urlFullscreen;  
}  
else  
{  
    url = nextCmp.photo.url;  
}
```

```
image.src = url;
```

```
if ( !image.complete )  
{  
    _vars.animate = false;  
    _methods.stopSlideshow();  
    _methods.runSlideshow();  
  
    return;  
}
```

```
var css = {display: 'inline-block'}, properties = {};
```

```
if ( _vars.effect === 'slide' )  
{  
    css.top = 0;  
    css.right = screen.width;  
    properties = {left: 0};  
}  
else  
{  
    css.opacity = 0;  
    properties = {opacity: 1};  
}
```

```
var img = _elements.content.find('img.ow_photo_view');  
var slideImg = $('img.slide', _elements.content).css(css);
```

```
slideImg[0].src = url;  
img.animate({opacity: 0}, {duration: 300});  
slideImg.animate(properties, {duration: 300, complete:
```

```
function()
```

```
{  
    var complete = function()  
    {
```

```

        photo
        img.css({opacity: 1});
        slideImg.hide();
        _vars.animate = false;
    };

    img[0].src = url;
    img[0].complete ? complete.call(img) : img[0].onload =
complete;

        }));
    }
    }, _vars.interval || 3000);
},
stopSlideshow: function()
{
    if ( _vars.slideshowTimeId )
    {
        clearInterval(_vars.slideshowTimeId);
    }
},
updateSlideshow: function()
{
    _methods.stopSlideshow();

    if ( _vars.isRunSlideshow )
    {
        _methods.runSlideshow();
    }
},
init: function()
{
    if ( _vars.isInitialized )
    {
        return;
    }

    _vars.isInitialized = true;

    $(window).on({'resize.photo': _methods.resizeWindow, 'popstate.photo':
_methods.history.popstate});
    $(document).on(_vars.fullScreen.fullscreenchange + '.photo',
_methods.fullscreen.onchange);

    var content = $(document.getElementById('ow-photo-view'));

    if ( _methods.fullscreen.enabled() )
    {
        $('.ow_photoview_fullscreen', content).on('click', function()
        {

```

```

                                photo
                                _methods.fullscreen.isFullscreen() ? _methods.fullscreen.exit()
: _methods.fullscreen.request();
    });

    $('._ow_photoview_info_btn', content).on('click', function( event )
    {
        if ( !_methods.fullscreen.isFullscreen() )
        {
            event.stopImmediatePropagation();
        }

        OW.trigger('photo.beforeResize', [_vars.photoId], _methods);

        if ( $(this).hasClass('open') )
        {
            $(this).removeClass('open').addClass('close');
            var wrap =
_Elements.content.find('._ow_photoview_stage_wrap');

            wrap.width(screen.width - 400);

            if ( _vars.isClassic )
            {
                wrap.css('float', 'left');
            }

            _Elements.content.find('._ow_photoview_info').removeClass('ow_photoview_pint_mode');
        }
        else
        {
            _Elements.content.find('._jspPane').css('left', 0);
        }

        _methods.updateComment({entityType: 'photo_comments',
entityId: _vars.photoId});

        OW.trigger('photo.afterInfoShow', [_vars.photoId],
_methods);
    }
    else
    {
        $(this).removeClass('close').addClass('open');

        _Elements.content.find('._ow_photoview_stage_wrap').width(screen.width);

        if ( _vars.fbHasContent )
        {

OW.removeScroll(_Elements.content.find('._ow_photo_scroll_cont'));

```

```

        photo
    }

    OW.trigger('photo.afterInfoHide', [_vars.photoId],
_methods);
}

OW.trigger('photo.afterResize', [_vars.photoId], _methods);
});

var timerId;

$('.ow_photoview_slide_settings', content).on({
    mouseenter: function()
    {
        if ( timerId )
        {
            clearTimeout(timerId);
        }

        $('.ow_photoview_slide_settings_controls',
_elements.content).stop(true).css({display: 'inline-block', opacity: 1});
    }
});
$('.ow_photoview_bottom_menu_wrap', content).on({
    mouseleave: function()
    {
        if ( timerId )
        {
            clearTimeout(timerId);
        }

        timerId = setTimeout(function()
        {
            $('.ow_photoview_slide_settings_controls',
_elements.content).fadeOut(1000);
        }, 5000);
    }
});

$('.ow_photoview_slide_settings_effect', content).on('click',
function()
{
    $('.ow_photoview_slide_settings_effect',
_elements.content).removeClass('active');
    _vars.effect = $(this).addClass('active').attr('effect');
    _methods.updateSlideshow();
});

```

```

                                photo
event )    $('ow_photoview_play_btn', _elements.content).on('click', function(
{
    if ( !_methods.fullscreen.isFullscreen() )
    {
        event.stopImmediatePropagation();
    }

    if ( $(this).hasClass('stop') )
    {
        _vars.isRunSlideshow = false;
        _methods.stopSlideshow();
        $(this).removeClass('stop');
    }
    else
    {
        _methods.runSlideshow();
        $(this).addClass('stop');
    }
});
}

OW.bind('photo.onAfterEditPhoto', function( photoId )
{
    _methods.deletePhotoCmp(photoId);
});
OW.bind('photo.onChangeListType', function( listType )
{
    if ( listType != _vars.listType )
    {
        _methods.clearHistory();
    }
});
OW.bind('photo.deleteCache', function( photos )
{
    if ( !Array.isArray(photos) || photos.length === 0 ) return;

    photos.forEach(function( photoId )
    {
        _methods.deletePhotoCmp(photoId);
    });
});

var event = {
    selectors: [
        'ow_photoview_stage_wrap',
        'img.ow_photo_img'
    ]
}

```

```

        photo
    };
    OW.trigger('photo.collectChangePhotoSelectors', event);
    content.on('click', event.selectors.join(','), function( event )
    {
        if ( !$(event.target).is(event.currentTarget) )
        {
            return;
        }

        if ( (event.pageX - $(this).offset().left) > ($(this).width() / 2) )
        {
            _methods.loadNextPhoto();
        }
        else
        {
            _methods.loadPrevPhoto();
        }

        event.stopImmediatePropagation();
    });

    $('.ow_photoview_arrow_left', content).on('click',
    _methods.loadPrevPhoto);
    $('.ow_photoview_arrow_right', content).on('click',
    _methods.loadNextPhoto);

    $.fn.extend({
        photoFocus: function () {
            this.each(function () {
                try
                {
                    if(this.setSelectionRange)
                    {
                        this.focus();
                        this.setSelectionRange(_vars.curPos, _vars.curPos);
                    }
                    else if (this.createTextRange)
                    {
                        var range = this.createTextRange();

                        range.collapse(true);
                        range.moveEnd('character', _vars.curPos);
                        range.moveStart('character', _vars.curPos);
                        range.select();
                    }
                }
                catch ( e ){ }
            }
        }
    });

```

```

        photo
    });
    return this;
  }
});
};

window.photoView = Object.defineProperties({}, {
  init: {value: _methods.init},
  setId: {value: _methods.setId}
});

})(window, window.jQuery, window.photoViewParams);

```