[**Infinite Scroll jQuery Plugin**](http://www.infinite-scroll.com/infinite-scroll-jquery-plugin/)

June 29th, 2008

**Note**: This documentation is currently out-of-date and will be updated soon. In the mean time, please refer to the GitHub[**README**](https://github.com/paulirish/infinite-scroll/blob/master/Readme.md) and [**Wiki**](https://github.com/paulirish/infinite-scroll/wiki/_pages).

This plugin aims to *progressively enhance* your page. Your navigation/pagination elements should be present in the HTML for non-js users, but the plugin will utilize those links to build out a more rich browsing experience.

**Download the jQuery plugin**

Version 1.5.100504  
[**jquery.infinitescroll.js and minified are now on GitHub**](http://github.com/paulirish/infinite-scroll)

**Usage**

**Minimum required configuration**

This is the minimum amount of configuration you can do, if you want things to work:

*// infinitescroll() is called on the element that surrounds*

*// the items you will be loading more of*

$('#content').infinitescroll({

navSelector : "div.navigation",

*// selector for the paged navigation (it will be hidden)*

nextSelector : "div.navigation a:first",

*// selector for the NEXT link (to page 2)*

itemSelector : "#content div.post"

*// selector for all items you'll retrieve*

});

**All options**

*// usage:*

*// $(elem).infinitescroll(options,[callback]);*

*// infinitescroll() is called on the element that surrounds*

*// the items you will be loading more of*

$('#content').infinitescroll({

navSelector : "div.navigation",

*// selector for the paged navigation (it will be hidden)*

nextSelector : "div.navigation a:first",

*// selector for the NEXT link (to page 2)*

itemSelector : "#content div.post",

*// selector for all items you'll retrieve*

debug : **true**,

*// enable debug messaging ( to console.log )*

loadingImg : "/img/loading.gif",

*// loading image.*

*// default: "http://www.infinite-scroll.com/loading.gif"*

loadingText : "Loading new posts...",

*// text accompanying loading image*

*// default: "<em>Loading the next set of posts...</em>"*

animate : **true**,

*// boolean, if the page will do an animated scroll when new content loads*

*// default: false*

extraScrollPx: 50,

*// number of additonal pixels that the page will scroll*

*// (in addition to the height of the loading div)*

*// animate must be true for this to matter*

*// default: 150*

donetext : "I think we've hit the end, Jim" ,

*// text displayed when all items have been retrieved*

*// default: "<em>Congratulations, you've reached the end of the internet.</em>"*

bufferPx : 40,

*// increase this number if you want infscroll to fire quicker*

*// (a high number means a user will not see the loading message)*

*// new in 1.2*

*// default: 40*

errorCallback: **function**(){},

*// called when a requested page 404's or when there is no more content*

*// new in 1.2*

localMode : **true**

*// enable an overflow:auto box to have the same functionality*

*// demo: http://paulirish.com/demo/infscr*

*// instead of watching the entire window scrolling the element this plugin*

*// was called on will be watched*

*// new in 1.2*

*// default: false*

},**function**(arrayOfNewElems){

*// optional callback when new content is successfully loaded in.*

*// keyword `this` will refer to the new DOM content that was just added.*

*// as of 1.5, `this` matches the element you called the plugin on (e.g. #content)*

*// all the new elements that were found are passed in as an array*

});

**Custom trigger, non-automatic. Twitter-style**

In 1.4 you can trigger the loading of the next page of content at will. You’ll first unbind the default behavior. And then trigger the next pull whenever you like..

*// unbind normal behavior. needs to occur after normal infinite scroll setup.*

$(window).unbind('.infscr');

*// call this whenever you want to retrieve the next page of content*

*// likely this would go in a click handler of some sort*

$(document).trigger('retrieve.infscr');

[**Demo of the triggered infinite scroll functionality.**](http://www.infinite-scroll.com/trigger.html)

**How does it work?**

There is a little known feature in the [.load()](http://docs.jquery.com/Ajax/load#urldatacallback) method that lets you specify the CSS selector of the html you want to include. jQuery will load in any local URL, then parse the html and grab only the elements you’ve defined with your selector. This allows for some pretty fun shit: client-side transclusions (a la [**purple include**](http://ajaxian.com/archives/purple-include-transclusions-you-know-you-want-them)) ; and some really kickass shit when you combo it with a [**local php proxy**](http://developer.yahoo.com/javascript/howto-proxy.html).

This is really the meat of the code:

*// load all post divs from page 2 into an off-DOM div*

$('

').load('/page/2/ #content div.post',function(){

$(this).appendTo('#content'); // once they're loaded, append them to our content area

});

So it basically leverages that load() method at its core. It’s basically scraping your existing page structure, which means**you don’t need to code any custom backend stuff to enable this functionality!** Booyah, right?

[**Plugin compatibility code**](http://www.infinite-scroll.com/lightbox-compatibility-code/)

January 30th, 2008

Many plugins run after the page has loaded and process existing content. Since Infinite Scroll dynamically adds new content to the page, sometimes you have to execute additional code to reinitialize those plugins.

For the plugins below, the following code goes in the **Javascript Callback** option.

[**Lightbox JS 1.0**](http://www.huddletogether.com/projects/lightbox/):  
No compatibility code available.  
[**Lightbox JS 2.0**](http://www.huddletogether.com/projects/lightbox2/):

myLightbox.updateImageList();

[**Shutter Reloaded**](http://www.laptoptips.ca/projects/wp-shutter-reloaded/):

shutterReloaded.Init('sh');

[**Thickbox**](http://jquery.com/demo/thickbox/):

tb\_init( $('a.thickbox, area.thickbox, input.thickbox',**this**) );

[**1 Pixel Out Audio Player**](http://www.1pixelout.net/code/audio-player-wordpress-plugin/):  
Compatible! Version 2.0, too.  
[**Yahoo Media Player**](http://mediaplayer.yahoo.com/):

YAHOO.MediaPlayer.addTracks( **this** );

[**HighslideJS**](http://highslide.com/):

$('a.highslide',**this**).click(**function**() { **return** hs.expand(**this**);})

[**WP-SimpleViewer**](http://wp-simpleviewer-demo.fuggi82.de/):  
Not compatible.  
[**Flare Smith Feed Flare**](http://xentek.net/code/wordpress/plugins/flaresmith/):  
Not compatible. (relies on inline script tags)

ShareThis plugin:  
Not compatible. (relies on inline script tags)

jQuery HoverIntent:

$('a.things',**this**).hoverIntent(......

[**SyntaxHighlighter Evolved**](http://wordpress.org/extend/plugins/syntaxhighlighter/):

SyntaxHighlighter.highlight(undefined,$('pre',**this**).get());

Smooth Scroll Links [SSL]

$('a',**this**).each(**function**(){

ss.addEvent(**this**,'click',ss.smoothScroll);

});

**To refer to the most recently added content:**

**this**; *// as of ver 1.1, the keyword this is the content wrapper (typically a DIV)*

$(**this**); *// the jQuery object including this*

$jQis('#infscr-page-'+INFSCR.currPage) *// old 'n busted ver 1.0 style.*