This document explains how to install Point on a website.

Contents of the Point package

The Point package contains two directories: *docs*, which contains this installation manual plus a template Point help page for your site, and *point*, which contains the following installation files:

- *point_config.js*
- *point_waiticon.gif*
- *point.css*
- *buttons* (directory)

How to install Point

Upload all the files within the *point* directory to a location on your web server. The example used throughout this document is *https://www.example.com/point/*.

Now you will need to configure Point to work on your website. Do this by performing the following basic steps, specified in detail below:

1. Configure Point JavaScript that runs in the user's browser
2. Customise CSS, if required
3. Add Point code to the HTML of your site's pages
4. Add a button to your site's pages to allow the user to turn Point on or off
5. Add / update your cookie site information
6. Add a Point help page to your site, if required
7. Test Point on your site.
   When testing that Point has been updated, you will need to empty your cache and hard reload your browsers to refresh for the new version (SHIFT refresh or SHIFT F5)

point_config.js

This file contains all your site-specific settings for Point.

REQUIRED: Minimal Configuration

- **Set your account name on line 7: (same as licence name)**
  ```
  Point.accountName = "ExampleAccount";
  ```

- **Selecting content to symbolise**

Point can be enabled on particular areas of the page using either element IDs or CSS classes (the default option).

Either:

Set the name of the CSS class or classes for which Point will be enabled on line 11 and 12:

```
Point.findContentByClass = true;

Point.pointEnabledDivClass = "point_sym_hover";
```

You can optionally instead specify multiple CSS classes using the following syntax:

```
Point.pointEnabledDivClass = ["point1", "point2", "point3"];
```

And/Or:

Set the ID or IDs of the elements for which Point will be enabled on lines 16 and 17:

```
Point.findContentById = true;

Point.pointEnabledDivId = "point_content_div";
```

You can instead optionally specify multiple IDs using the following syntax:

```
Point.pointEnabledDivId = ["point_content_div", "point_other_div"];
```

OPTIONAL: Extra Configuration

- **Changing the symbol size**

  The default symbol size (measured as the height of each symbol in pixels) is 80. You can change this on line 25:

  ```
  Point.symbolSize = 100;
  ```

- **Changing the maximum number of symbols per tooltip**

  By default, Point will display a maximum of four symbols in each tooltip. You can change this limit on line 28:

  ```
  Point.maxSymbols = 4;
  ```

- **Using secure HTTP for symbol images**

  You can specify that Point symbol images should be served via secure HTTP (HTTPS) on line 32. You may wish to do this if some or all of your site is served via HTTPS to avoid browser warnings.

  ```
  Point.httpsImageUrls = true;
  ```

- **Enabling Point when the page loads**

  By default, Point loads once the document loads. If you would prefer to load it manually instead (for example, by using a button on the page) then you can do so by changing line 36:

  ```
  Point.enableImmediately = false;
  ```

- **Caching tooltips**

  Point caches tooltips that have already been displayed on the current page. If you want to disable this feature, you can do so on line 39:

```
Point.cacheTooltips = false;
```

You can also change the length of time tooltips may be cached on a single page by changing the value of `Point.cacheExpiry` on line 42. The value is measured in milliseconds and defaults to 24 hours.

```
Point.cacheExpiry = 3600000; // One hour
```

- **Point Enable/Disable images**

  Set the full addresses for the images for the button to enable and disable Point (see below) on lines 44 and 45. See the section "Enabling and disabling Point" below for more information:

  ```
  Point.enabledPointImage = "/point/buttons/light_background/on.gif";
  Point.disabledPointImage = "/point/buttons/light_background/off.gif";
  ```

- **Point waiting icon**

  Point by default displays a spinning animated icon whilst symbols are loading. To use the default icon provided with Point, make sure you specify a URL for the icon on line 46:

  ```
  Point.waitIconImage = "/point/point_waiticon.gif";
  ```

  If you wish to specify your own icon then enter a URL for your icon instead. If you do not wish to have any icon then remove this line.

- **Setting blacklisted elements**

  Point has a list of element types for which symbols will not appear. A default list of HTML elements is provided. You should only edit this list if you have a specific reason to do so.

  To edit the list, add (or remove) lower case tag names surrounded by speech marks, followed by a comma. The list starts on line 49 with

  ```
  Point.blacklistedElements = [
  ...
  ```

- **Setting excluded words**

  Words included in the excluded words list will not show symbols when the mouse is hovered over them. You may wish to hide symbols that are abstract in nature. A default list of common abstract symbols is included.

  To edit the list, add (or remove) words surrounded by speech marks, followed by a comma. The list starts on line 73 of the file with

  ```
  Point.blacklistedWords = [
  ...
  ```

- **Setting excluded CSS classes**

  An element in the page having any of the blacklisted classes and all descendants of such an element will not be Point-enabled. By default, the list of blacklisted classes contains classes for symbolised HTML content generated by Widgit's Insite product.

  To edit the list, add (or remove) classes surrounded by speech marks, followed by a comma. The list starts on line 91 of the file with

  ```
  Point.blacklistedClasses = [
  ...
  ```

- **Language support**

  Point currently supports content written in English UK:

  Point.defaultLanguage = "English_UK"

- **Setting the hover delay**

  This controls the delay before a symbol tooltip is shown, in milliseconds. The default value is 200 milliseconds. You can change this on line 105:

  ```
  Point.hoverTimeout = 500;
  ```

- **Point spans inheriting styles from parent elements**

  If enabled, this option forces the element that Point creates surrounding every word to inherit all of its style properties from its parent element. Enabling this option incurs a performance penalty on the user's browser because Point has to do more work when it is enabled on each page visit, so you should only set this to true if you are having layout issues with Point-enabled content, which is very rare.

  Set this option on line 109:

  ```
  Point.spansInheritParentStyles = true;
  ```

- **Smart symbolisation**

  If enabled, this option symbolises each word in context. In other words, Point will look at the words around the word the user is hovering over and deduce which symbol best fits the word on that context. For example, in the phrase "I drink a drink", when the user hovers over the first "drink", they will see the symbol for the verb "to drink" at the top of the Point tooltip, and when they hover over the second "drink", they will see the symbol for the noun "drink" at the top.

  This option is enabled by default. You can change it on line 113:

  ```
  Point.symboliseContext = false;
  ```

- **Symbolisation of idioms**

  If this option is enabled and the user hovers over a word that is part of a phrase or idiom that Point recognises as a single concept, the Point highlight will extend to the whole phrase and show a tooltip containing possible symbols for the whole phrase.

  This option is disabled by default and has no effect if `Point.symboliseContext` is `false`. You can change it on line 117:

  ```
  Point.symboliseIdioms = false;
  ```

- **Creating Point elements**

  Whether or not Point always creates elements around each word within Point-enabled content when the page loads. If `Point.spanifyOnLoad` is `false` and `Point.enableImmediately` is `false`, and the user has not previously enabled Point, Point waits until the user enables Point before performing "spanification".

  `Point.despanifyWhenDisabled` controls whether Point should remove its <span>s when Point is switched off. Setting this to true will have a slight impact on performance when the user switches Point off.

  By default, Point creates elements when the page loads and does not remove them when Point is switched off. You can change this on lines 123 and 127:

  ```
  Point.spanifyOnLoad = false;
  ```

  ```
  Point.despanifyWhenDisabled = true;
  ```

- **Access key**

  Point can be switched on and off using an access key. By default, this is set to "P". Access keys are built into browsers, and the actual keypresses required vary:

  Microsoft Edge: Alt + P
  Google Chrome / Safari on Windows: Alt + P
  Google Chrome / Safari on Mac: Ctrl + Opt + P
  Firefox and Opera no longer in use

  The access key used can be specified on line 130:

  ```
  Point.accessKey = "P";
  ```

- **Using cookies**

  By default, Point uses a cookie to track whether the user has previously enabled or disabled Point for your site. The implications of using cookies are discussed elsewhere in this document. You can opt not to use a cookie by changing the value of `Point.useCookie` on line 134:

  ```
  Point.useCookie = false;
  ```

- **Skin tone**

  Point allows you to specify a preferred skin tone for symbols that feature people. This value may be 0, 1, 2, 3, 4 or 5. The default value is 1.

  You can change this by changing the value of `Point.skinTone` on line 140:

  ```
  Point.skinTone = 4;
  ```

- **Vary group skin tones**

  In symbols that contain more than one person, the skin tones of the people in the symbol are different. You can disable this so that all people in a symbol have the same skin tone by changing the value of `Point.varyGroupSkinTones` on line 143:

  ```
  Point.varyGroupSkinTones = false;
  ```

point.css

This file contains the styling for the elements that Point adds to the page. You can use it to customize the appearance of Point's tooltip and word highlighting.

OPTIONAL: Tooltip and word highlighting customization

- **Tooltip style**

  The Point tooltip has the class *point_sym_tooltip*. Edit the CSS rule within the point_site.css file on lines 1-8 to modify the tooltip's appearance.

- **Word highlighting**

  In Point-enabled areas of the page, Point highlights each word as the user hovers over it. Where a word has or may have symbols, it is highlighted with an orange background. You can customize this by amending the rule on lines 15-17 within the point_site.css file. Where a word does not have

symbols, it is highlighted with a grey background. You can customize this by amending the rule on lines 19-21, also within the point_site.css file.

Enabling Point on your web pages

Now the Point code has been uploaded to your website, you can enable Point on any page within the site.

Web page header

To activate a page with Point, add the following lines of code in inside your page *<head>* element. The first two lines refer to JavaScript and CSS files hosted on Widgit's servers. You should copy these lines exactly as they appear below. The remaining two lines refer to files on your server, so you will need to update these paths to point to where you uploaded the files.

```
<script type="text/javascript" src="https://static.widgit.com/point/pointv2.js"></script>
<link rel="stylesheet" href="https://static.widgit.com/point/point.css" type="text/css" />

<script type="text/javascript" src="/point/point_config.js"></script>
<link rel="stylesheet" href="/point/point.css" type="text/css" />
```

Web page content

You need to now set the area of the page you wish to be Point-enabled. This can be the whole page or just few lines. It's recommended to use Point around the main story area of your web page and not over the menus.

You can enclose the area with a *<div>* (or other element) with a specific CSS class (defined in *point_config.js*). If you already have a container with this class, you do not need to modify your page. If you need to add a container, you can do so:

```
<div class="point_sym_hover">
  <p>Content that will have Point enabled.</p>
</div>
```

You can repeat this if you would like more than one area on the page to work with Point.

Enabling and disabling Point

If you wish to allow the user to enable and disable point on your site, you can add an on/off button to your web pages. The button indicates whether Point is switched on or off, so you need an image for each state (on and off). Included with Point are images for two different buttons: one set for dark backgrounds and one set for light backgrounds. Each button has images called *on.gif* and *off.gif*, which can be found within the *light_background/* and *dark_background/* subfolders within in the *buttons/* folder.

There is also an alternative set of buttons with different wording.

To allow users to enable and disable Point by clicking on a button, add the following code somewhere on your page:

```
<img src="/point/buttons/light_background/off.gif" id="point_togglebutton"
onclick="Point.togglePoint();" width="113" height="32" alt="Toggle Point
Symbols" title="Toggle Point Symbols" />
```

Ensure you update the *src* attribute to refer to the correct path to the disabled image you want to use.

Adding a Point help page to your website

Users of your website may find it helpful to have a page on your site explaining what Point is and how to use it. Included in the docs directory is a template HTML help page for Point which you can use as a template for your own Point help page. You should provide a link to this help page on every page that uses Point.

Note that the section entitled "How do I turn Point on and off" will require some tailoring depending on how you have set up the button for enabling and disabling Point.

Point and cookies

Point uses a cookie to remember whether a visitor to your website has enabled or disabled Point. It only creates this cookie if the user's choice differs from the default Point settings for your website (e.g. if you have Point symbols disabled by default and a user turns them on then a cookie will be created). The cookie is called 'point_enabled' and is a single true/false value. By default, it lasts for 90 days.

If you have a cookie policy on your website or have added cookies to a section of your privacy policy, then you may want to mention Point. Included in the docs directory is some sample HTML that provides a brief description of cookies and a description of the Point cookie specifically. If you use the sample, you will need to replace all occurrences of '[YOUR WEBSITE]' with the name of your website

1st Floor, Bishops House
Artemis Drive
Tachbrook Park
Warwick
CV34 6UD
United Kingdom

Phone: 01926 333680
support@widgit.com
www.widgit.com