

zoowee bubbles

DEVELOPMENT DOCUMENTATION

Documentation for the Zoowee Library v2.0

The Zoowee javascript library may be installed on any given webpage to enhance that pages functionality and provide access to numerous information resources from anywhere upon the network/internet. Bubbles can be built using the Zoowee Bubbles Content Management system or manually using the XML feed.

Embedding the Library:

Bubbles can live on any webpage you can embed javascript on. The first step to placing bubbles on the webpage is to place call the zoowee Library. The Zoowee Library calls all of the necessary elements and styles for displaying your bubbles.

There are two compressed files that can be sourced:

1. Calls the library when a bubble link is clicked:

```
<script src='http://my.zoowee.com/workspace/loader.js' type='text/javascript' language='javascript'>
</script>
```

2. Calls the library when the page loads if a bubble is embedded in a div:

```
<script src='http://my.zoowee.com/workspace/library.js' type='text/javascript' language='javascript'>
</script>
```

Place this script anywhere before the `</body>` tag. Similar to how you would embed Google Analytics onto your webpage. The first script call, with `loader.js`, calls the javascript library when a link is clicked on your webpage that calls a bubble. This is the most common use of Zoowee Bubbles. The second script call, the one with `library.js`, calls the whole library when the page loads. This is used when a bubble is embedded within a div (or table) within your webpage. This allows you to use bubbles as content management containers.

Calling Your Zoowee Bubbles:

If you have created a Bubble using the `my.zoowee.com` content management system, and you want to call the bubble when the user clicks on a link, you will call the bubble using the bubble ID:

```
<a href='javascript:Zoo.createBubble({zid:"01_photo.cmsfeed"})'>Bubble Name</a>
```

If you have created a Bubble using the `my.zoowee.com` content management system, and you want to embed the bubble in your webpage, you will call the bubble using the bubble ID:

```
<div class='zoo'><div zid='60_photo.cmsfeed' style='width:520px;height:400px'></div></div>
```

If you have created a Bubble using the Zoowee XML feed structure, you will call your XML file instead of the bubble ID:

```
<a href='javascript:Zoo.createBubble({zid:"@http://mysite.com/bubblefile.xml})'>Bubble Name</a>
```

Building a bubble using the feed structure is a bit more advanced, but flexible, way of building and maintaining bubbles.

Skinning Your Bubble:

All bubbles can have a default zoowee skin applied to them, which you can find by using the my.zoowee.com workspace in the bubble properties menu, or a custom skin applied to it which you build yourself. All of the default skin images are hosted on zoowees server. However, if you build your own custom skin you will need to host the images on your own server and refer to them using absolute references when you embed your bubbles.

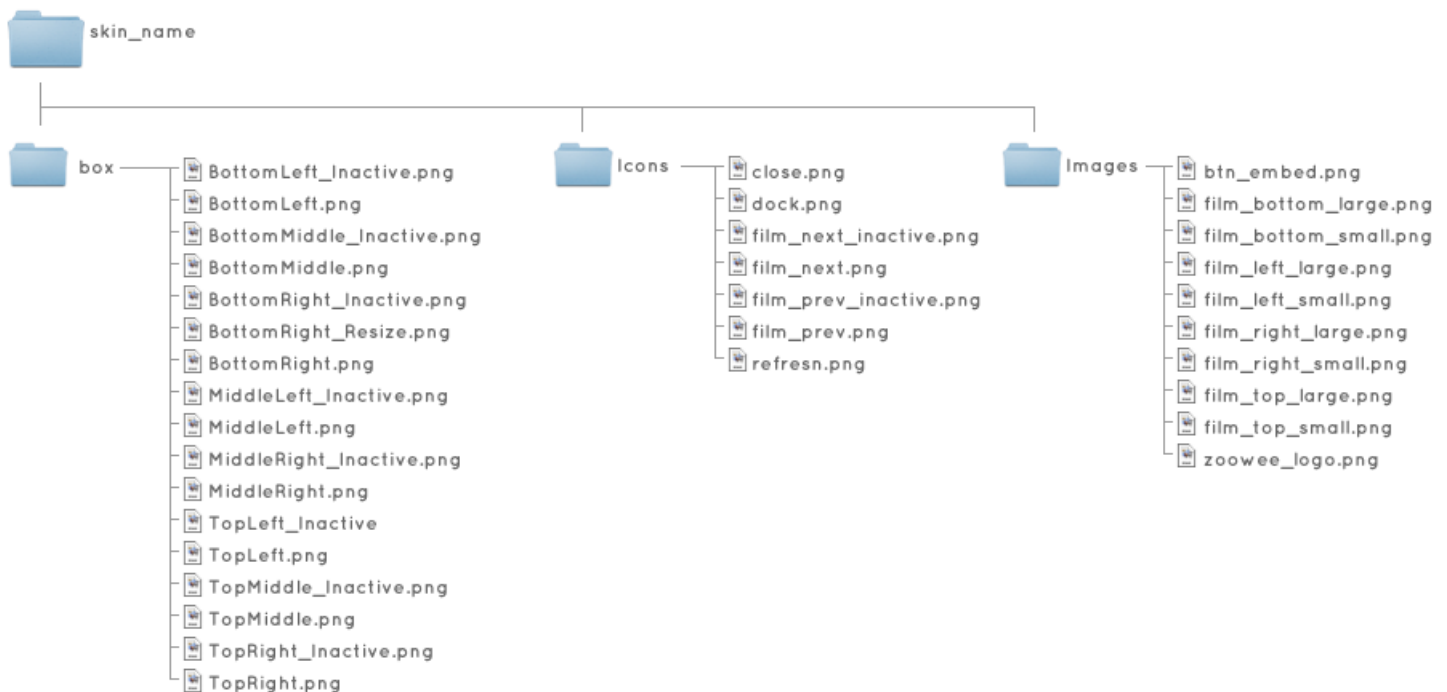
The skin is composed of **34 total images**.

17 box images: make up the active and inactive state of the bubble, including one image to indicate resizing of the bubble in the bottom right.

7 icon images: will generally be the buttons, or icons, in the header and footer of the bubble.

10 general images: make up the layout of various skin dependent elements for the various bubble types.

Bubble Skin File Structure:



The images must be in this structure for the bubble call the skin files appropriately. We would recommend downloading this file structure zip file from the zoowee website and replacing the images with your own skin images. You can download this directory in your my.zoowee.com workspace under the properties tab. By clicking on the custom skin button.

Applying Your New Skin:

Your skin will be applied in the embed code (or in the feed structure if you are building or maintaining your bubble with XML). You will call a variety of parameters that apply the images along with their width and height to the bubble. This is also where you will call the background, font, and border colors. The properties you apply to the bubble will be included in the embed code if someone clicks the embed button on your skinned bubble.

Original Embed Code:

```
javascript:Zoo.createBubble({zid:"01_video.cmsfeed"})
```

Embed Code With New Skin:

```
javascript:Zoo.createBubble({zid:'01_video.cmsfeed', width:445, height:400, showTitle:false, skin:{path:'http://www.yourWebsite.com/bubbleSkinDirectory/', ext:'png', heightBottom:54, heightTop:56, widthLeft:11, widthRight:11}, colors:{background:'#e1d4a1', primary:'#140e03'}})
```

Properties:

Width: number - Width of your bubbles content area (not including the skin)

Height: number - Height of your bubbles content area (not including the skin)

showTitle: true/false - Show title at the top of the bubble

skin: This effects the external edges of the bubble. Properties applied to the bubble skin.

path: string - URL of your bubble skin (the folder containing box, icons, and images directories)

ext: png/gif - The file extension of your skin images (typically png)

heightTop: number - height of top_left, top_middle, and top_right.png/gif images in for the skin

heightBottom: number - height of bottom_left, bottom_middle, and bottom_right.png/gif images

widthLeft: number - width of top_left, middle_left, and bottom_left.png/gif images

widthRight: number - width of top_right, middle_right, and bottom_right.png/gif images

colors: This applies to the content within the bubble. Colors applied to multiple parts of the bubble.

background: Bubble Content background color

primary: primary colors for all of the bubbles

alert: a high contrast alert color for your bubble

inactive: value for inactive text color

title: color of title bar text

highlight: link text color

menu: color for the unselected menu text color

menubg: color for the unselected menu background color

menuhighlight: color for the selected menu text color

menuhighlightbg: color for the selected menu text color

Building Bubbles using XML:

All bubbles use this basic feed structure in building and maintaining information and media types. The my.zoowee.com workspace provides a user interface to assist in the construction of this feed. The XML for each bubble can be found in the zoowee workspace under the embed tab. You can view this XML feed information by clicking on the “xml data” button next to the “Feed Link:”.

Basic Bubble Feed Structure:

```
<zoowee type="feed">
  <branch title="Bubble Name" owner="Zoowee Username" zid="01_html.cmsnode" version="2.0">
    <style options="layout:view_left|padding:0px" width="480" height="365" show_embed="false">
      <view_type>view</view_type>
      <skin path="https://my.zoowee.com/workspace/skins/default/default/" ext="png" top="32" left="8"
        right="8" bottom="32"/>
      <colors primary="#333333" highlight="#996600" background="#eaeaea" tablebg="#939393"
        menu="#333333" menubg="#cdcdcd" menuhighlight="#333333" menuhighlightbg="#eaeaea"
        title="#333333"/>
    </style>
    <items>
      <node zid="01_html.cmsnode" has_children="true">
        <atoms>
          <text name="title"/>
          <html name="content">
            HTML goes here
          </html>
        </atoms>
      </node>
    </items>
  </branch>
</zoowee>
```

The complete feed is contained in the <zoowee> block. The branch of content is contained in the <branch> block. The branch block may give a title attribute that will become the title of the bubble. The <style> block contains rendering details – in this case it gives the <view_type> as “view” which is a top down vertical display of all items in the feed. The <items> block will contain all of the data to be rendered. Each data point is given as a <node> with atomic elements of content to be displayed. These <atoms> can be of five different types. <text>, <html>, <media>, <image>, and <link>. Text atoms are rendered without accepting any HTML markup. HTML atoms are rendered directly in the bubble as they give HTML. Media atoms contain a reference to an audio or video file type. Image atoms will render the source given. Link atoms can be href links or bubble links. All atoms must have a name attribute and may have a tooltip.

Bubble Template Structure:

```
<zoowee type="feed">
<branch icon="/icons/icon_album.png" zid="@sample/simple.xml">
<style>
  <view_type>template</view_type>
  <template>
    <div [[zoo:loop:items]]>
      <div [[zoo:loop:node]]> <img style='margin:10px;float:left;border:5px solid #fff'
        src='[[zoo:mobile:[[zoo:property:pic:thumb_source]]][[zoo:bubble:[[zoo:property:pic:original_
source]]]]'/><div style='margin:15px;'><span style='font-weight:bold;' [[zoo:atom:title]]> </span>
      <span style='font-family:arial,Helvetica;margin:10px;height:100%' [[zoo:atom:description]]></span>
    </div>
  </div>
</div>
</template>
<skin path="http://music2.zoowee.com/main/skins/rust_2/" ext="png" top="35" left="25"
right="25" bottom="35" tab_options="background_image_both"/><colors primary="#312614" high-
light="#945529" background="#E4CDAB" tablebg="#ece2d2" menu="#E4CDAB" menubg="#945529"
menuhighlight="#945529" menuhighlightbg="#E4CDAB" title="#E4CDAB"/>
</style>
<items>
<node zid="SOME-UNIQUE-ID">
<atoms>
<text name="title">Bionic</text>
<html name="description">
Subtlety not being part of Christina Aguilera's vocabulary, she trades the retro-swing of Back to Basics
for the future-pop of Bionic.
</html><image name="pic" original_source="http://music2.zoowee.com/main/pop/cov200/drn900/n919/
n91930bwid4.jpg" thumb_source="http://music2.zoowee.com/main/pop/cov200/drn900/n919/n91930b-
wid4_thumb.jpg"/>
</atoms>
</node>
</items>
</branch>
</zoowee>
```

Templates are Zoowee's way of providing a functional mechanism to control the layout and navigation of the content of a bubble feed. Zoowee bubbles are generated from two buckets. A rendering bucket, and a content bucket. The content bucket is a list of nodes that contain things like text, images, links, media, and html. The rendering bucket is the template piece and it provides the necessary components to draw the content. In this bucket includes outer skin images and dimensions, default colors for content within, and an HTML framework for the layout.

The Style Block:

The style block can have a variety of attributes applied to it. The attributes that are applied to the style block will effect the layout and behaviors of your bubble. Some of these attributes will effect how the content is displayed, while some will effect how the bubble itself acts. The available attributes are listed below.

Style Block Attributes:

show_embed: Boolean - Turn the embed button off/on in the bubble footer

show_title: Boolean - Show the title in the header

show_tooltips: Boolean - Show rollover tooltips for photos or for the rss feed

resizable: Boolean - Allow users to resize your bubble (does not work well with all bubble types)

scrolling: Boolean - Turn scroll bars off/on

width: Number - Width of the content area of the bubble

height: Number - Width of the content area of the bubble

options: String - properties specific to the content built through the my.zoowee.com workspace

The the options property (<style options =""...>) can be removed from the tabbed bubble type because it does not require any specific properties applied to the containing bubble (you will notice an additional options property within the style block on the Basic Bubble Feed Structure section on the previous page). This options property will apply a set of properties for the various bubble types. These options are specific for each bubble type.

The style block will have 2 child nodes (skin and color) which will apply the skin and color to your bubble. These two nodes will have attributes applied to them in the same way the attributes were applied to the embed code.

Skin Block Attributes:

path: string - URL of your bubble skin (the folder containing box, icons, and images directories)

ext: png/gif - The file extension of your skin images (typically png)

top: number - height of top_left, top_middle, and top_right.png/gif images in for the skin

bottom: number - height of bottom_left, bottom_middle, and bottom_right.png/gif images

left: number - width of top_left, middle_left, and bottom_left.png/gif images

right: number - width of top_right, middle_right, and bottom_right.png/gif images

Color Block Attributes:

background: Bubble Content background color

primary: primary colors for all of the bubbles

alert: a high contrast alert color for your bubble

inactive: value for inactive text color

title: color of title bar text

highlight: link text color

menu: color for the unselected menu text color

menubg: color for the unselected menu background color

menuhighlight: color for the selected menu text color

menuhighlightbg: color for the selected menu text color

Combining Bubbles using the feed:

One advantage of using the feed structure is combining multiple bubble types in one bubble. This allows you to build bubbles in the my.zoowee.com workspace and combine those bubble types into one single bubble. Currently most but not all bubbles built in the workspace work in this feed structure. The Photo and Video grid, Text, HTML, and RSS Bubbles work in this feed structure. The Contact, Photo and Video Filmstrip, Podcast and Hover Bubbles still have bugs that will be resolved in the future.

Calling Bubbles in the Feed:

```
<zoowee type="feed">
  <branch title="Bubble Name" owner="Zoowee Username" zid="">
    <style options="null" width="480" height="365" embedable="false">
      <view_type>tab</view_type>
      <skin path="https://my.zoowee.com/workspace/skins/default/default/" ext="png" top="32" left="8"
right="8" bottom="32"/>
      <colors primary="#333333" highlight="#996600" background="#eaeaea" tablebg="#939393"
        menu="#333333" menubg="#cdcdcd" menuhighlight="#333333" menuhighlightbg="#eaeaea"
        title="#333333"/>
    </style>
    <items>
      <node zid="01_html.cmsfeed" has_children="true">
        <atoms>
          <text name="title">HTML Bubble Tab</text>
        </atoms>
      </node>

      <node zid="02_photo.cmsfeed" has_children="true">
        <atoms>
          <text name="title">Photo Grid Tab</text>
        </atoms>
      </node>

      <node zid="03_info.cmsfeed" has_children="true">
        <atoms>
          <text name="title">Text Bubble Tab</text>
        </atoms>
      </node>
    </items>
  </branch>
</zoowee>
```