Adding a Tooltip Type MouseOver Effect to Content in Desire2Learn
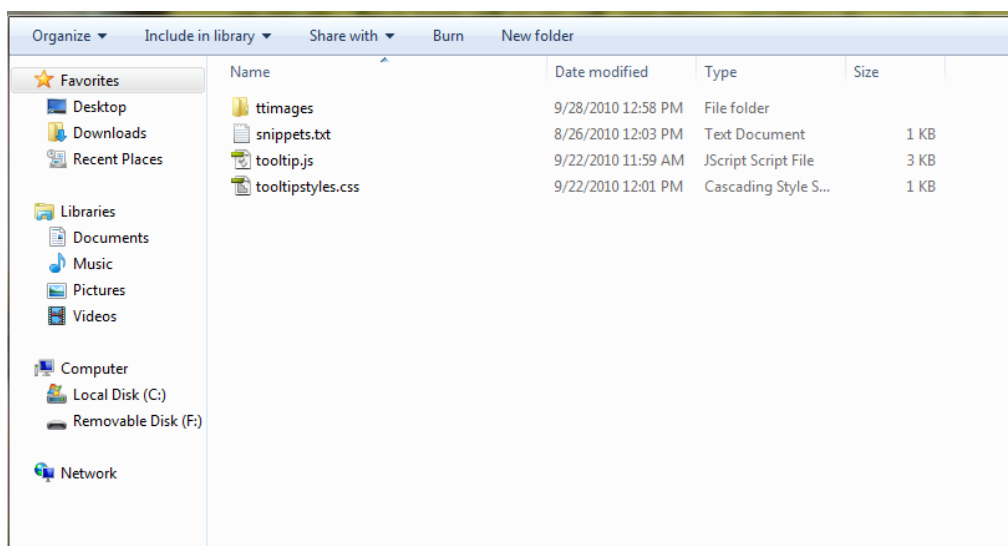
This document shows how to add a tooltip type mouse over effect, so that definitions or pictures could be displayed in context within a LMS page coded in HTML. I will be screen capturing stills from within Desire2Learn's LMS, from within Firefox browser on the Windows Operating System. The techniques shown should be applicable within Blackboard, Moodle, Sakai or any other web based LMS that uses HTML to display pages. The procedure shown will be slightly different based on which web browser one uses to access the LMS.

1. Import the tooltip.zip package by uploading or importing the entire package so that it maintains the structure of the backend files. The structure will look something like this.

The javascript file, tooltip.js contains the javascript that creates the mouse over effect and handles the fade in as well as the creation of the layering effect that occurs.

The CSS file, tooltipstyles.css contains minimal style information that shouldn't conflict with Desire2Learn's styles.

Snippets.txt contains the code snippets that you will have to copy and paste into the page. The folder ttimages contains several images that round corners that surround the text information.

2. Go to the page you wish to add the tooltip. While many LMS's allow you to upload and view Word or PDF files, this procedure will only work with HTML files. You can also use this technique to add tooltips to any webpage. Edit this topic.

Most modern LMS's have a built-in editor to create topics or pages that students can access. In Desire2Learn, this window similar to the screen capture on the next page.

3. You will want to edit the HTML of this page by clicking the Code Editor button.

4. Open snippets.txt (by double clicking) to have the code snippets available to you. Review the contents of the text document, which gives you the code that you will need to add and where in the HTML you need to add the snippets. Adding this functionality to a page has two parts. The first only has to occur once, where you link this document to the javascript and CSS files. The second occurs whenever you want to show a definition.

5. Looking at the HTML code of the page we've created in the editor, there is not a lot of content there.

Here's the code in text format if you wish to copy the steps exactly on your own LMS or webpage.

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
<head>
      <title>Untitled document</title>
      <meta http-equiv="Content-Type" content="text/html; charset=utf-
8" />
</head>
<body style="font-family: verdana, sans-serif;">
We've got some words, that need definition. You'll have to add a whole
bunch of text to what you have displayed on the screen.
</body>
</html>
```
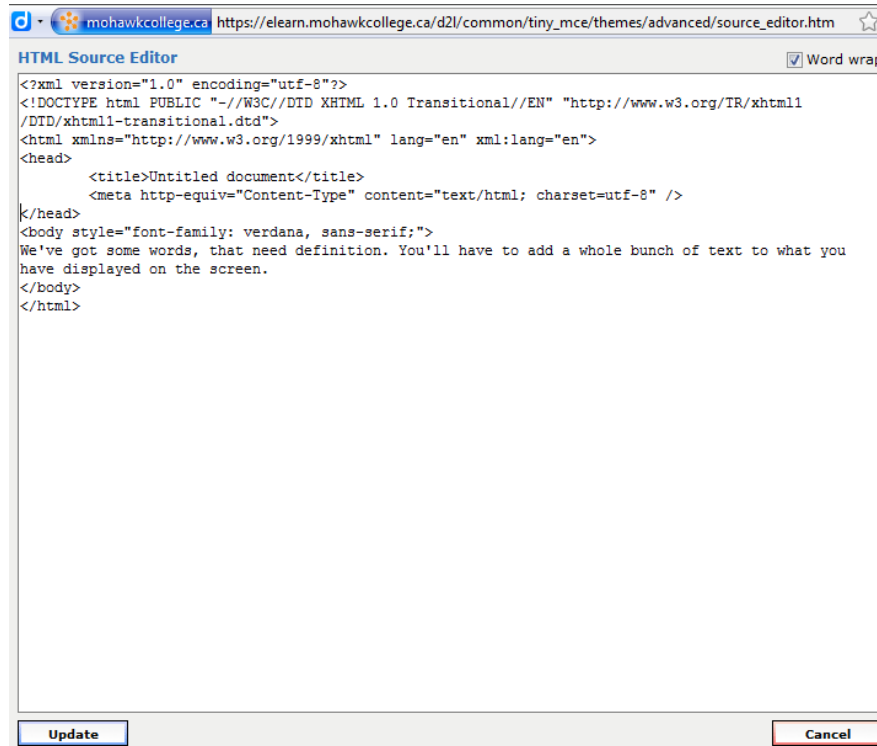
6. **We will now need to insert the first snippet after the line** `<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />`. **After this we will need to add two lines:**

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

The finished piece will look like:

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
<head>
      <title>Untitled document</title>
      <meta http-equiv="Content-Type" content="text/html; charset=utf-
8" />
      <script type="text/javascript" src="tooltip.js"></script>
      <link rel="stylesheet" type="text/css" href="tooltipstyles.css"/>
</head>

<body style="font-family: verdana, sans-serif;">
We've got some words, that need definition. You'll have to add a whole
bunch of text to what you have displayed on the screen.
</body>
</html>
```

You have now linked the javascript and CSS files to this document. Next we need to apply the tooltip to a section of text.

7. To apply the tooltip, find the word that requires a definition. Add this code snippet in front of the word:

```
 <span class="hotspot" onmouseout="tooltip.hide();"
onmouseover="tooltip.show('definition goes here',250);">
```

And close the tag pairing with a following the word. If you were going to define the word definition, your complete HTML would look like this:

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
<head>
      <title>Untitled document</title>
      <meta http-equiv="Content-Type" content="text/html; charset=utf-
8" />
      <script type="text/javascript" src="tooltip.js"></script>
      <link rel="stylesheet" type="text/css" href="tooltipstyles.css"/>
</head>
```

```
<body style="font-family: verdana, sans-serif;">
We've got some words, that need <span class="hotspot"
onmouseout="tooltip.hide();" onmouseover="tooltip.show('The definition
of definition is a statement of the meaning of the word or
phrase.',250);">definition</span>. You'll have to add a whole bunch of
text to what you have displayed on the screen.
</body>
</html>
```
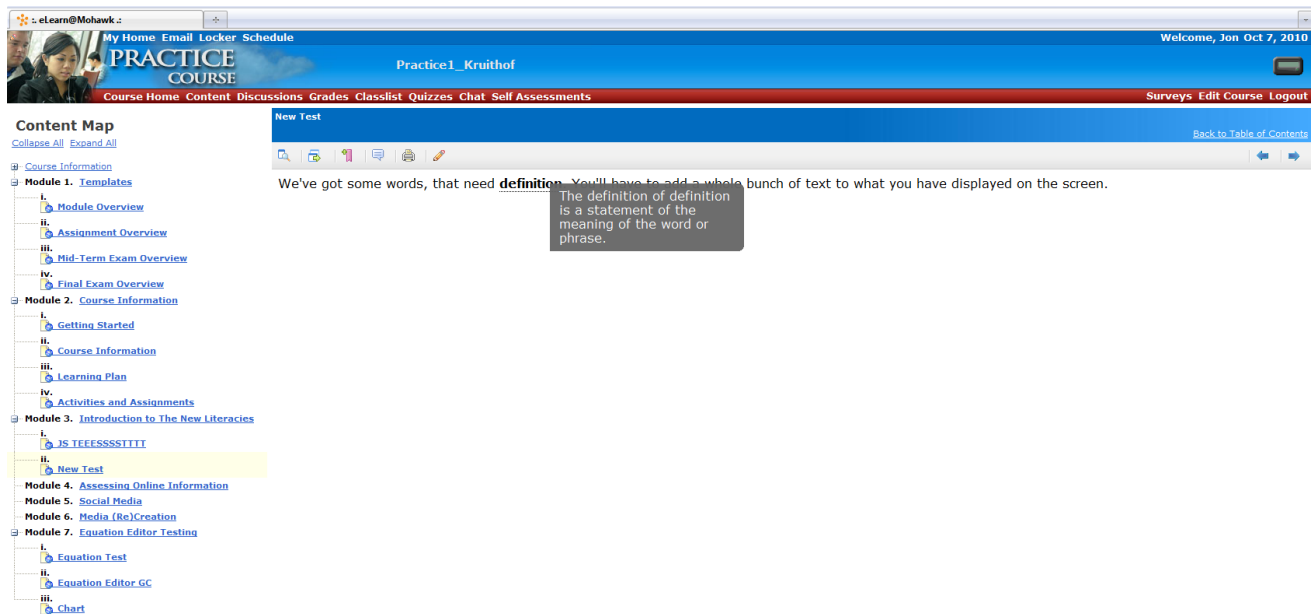
Notice that the definition itself is passed in the code, so one can reuse the same functions (which live in tooltip.js) but change the definition based on the context. It is best practice to actually edit all your tooltips in one attempt. We have seen the HTML editor rewrite code when it is re-opened at a later date, so you have to load all your definitions at once. A second edit of the page will break any previous javascript work. Save your work when finished by clicking the Update button. Save the edited topic.

8. Here's what the effect looks like:

Notice that there is a fade-in and fade-out effect added to the mouse over. This way the effect is not harsh and jarring to the user and is much more elegant. If you are comfortable with javascript and coding in general, there are variables you can change within the javascript file. These variables include the speed of the fade-in and fade-out, offset of the tooltip from the mouse pointer.

NOTE: When the tooltip is engaged close to the right hand edge of the window, sometimes the definition appears off screen. If you move your cursor further left then back to the edge of the window, this will usually correct the behavior.