

## Making web page modifications

### Introduction

This document is a brief discussion of how to make changes to and add additional pages to the SJC website. It contains five sections. The first of these is a step-by-step list of instructions. The second is a list of common problems and how to resolve them. A third section is a brief discussion of HTML (the language used to format web pages) and CSS, the means used to style and prettify them. See also the website policy document for further discussions of what the website is **for** – this document is about changing it and assumes the goals are being met by the change<sup>1</sup>. Next, there is a brief discussion of “accessibility” and why it matters. Finally, I discuss using other tools to build site pages and other content.

Please contact me if there is any problem using this document or if you feel it should include something that it currently does not. It is only available in English at this time, which should not be a problem, as grasp of English at this level is necessary to use the software necessary to make the changes in question anyway<sup>2</sup>.

The SJC at present uses Dreamweaver to manage and maintain its website. This Adobe (formerly) Macromedia product can be intimidating to the new user, but have courage – the SJC site is quite simple and does not require mastery of anywhere near all of Dreamweaver’s features.

### Instructions

The first task is to determine what sort of update is needed. If a page merely needs modifications (e.g. text added or deleted), follow the steps in part A. If a page needs to be added, follow part B. If the common area to a bunch of pages (usually grouped by language) must be changed, use part C. If a new common area has to be created (e.g. if the site is to gain a new language), see me, as this requires a new template. This latter category of updates is considerably involved.

### **Part A – Modifying an existing page**

Start by finding the page you wish to modify in the local collection. Then locate the section you wish to modify. There are several possible such areas:

- 1) Title
- 2) Heading

---

<sup>1</sup> There are many potential topics that could be regarded as within our topic. For example, I have not addressed how to move the entire site somewhere else, how to add certain features (such as mailto: links, forms, etc. which might be useful in future versions of the site). If there is sufficient demand, I will revise this document to include such matters.

<sup>2</sup> That said, Keith is open to discussing the matters herein in French, particularly if there are one or two minor items that are unfamiliar to the reader.

- 3) Primary content
- 4) Location (i.e. what is in Action Alerts belongs in Publications, or the like)
- 5) Scripts

(1) Title is the item that usually appears on the tops of windows. (See the HTML discussion, below, for more.) Simply type the title you wish into this section.

The title cannot be marked up in any fashion, so this is it for this sort of modification.

(2) Headings come several styles and sizes. Follow a hierarchy within your document (smallest within largest). The SJC has decided to only use two levels of headings, 1 (main heading) and 2 (subheading). Dreamweaver's "paragraph format" submenu in the Text menu will include these as choices. Otherwise one can use (in code or split view) the HTML tags `<h1>` and `<h2>`. (See the HTML section below for more on this.)

(3) Primary content is the hardest portion to discuss. Some guidelines are as follows:

- (a) In general it is best to type up the web page directly in Dreamweaver's supplied editor. Do not import directly or copy from a Word document. You may copy and paste, but only from a text editor like TextPad or Notepad.
- (b) If you wish to emphasize or otherwise mark up text, use the CSS methods, not the old-timey methods. In other words, no `<font>`, `<i>` `<b>`, etc. You may have to create such methods; see me for further details.
- (c) Images can be "placed" in a document (they are actually still external to the web page itself; the user agent is responsible for placing them on the page if necessary) with the Insert menu item. Be sure to enter an "alt" attribute. This is so that users who cannot see images can understand what the image is for. These images can be created in any program desired (so long as they are saved in an appropriate format – see below), and should all be placed in the images directory at the top level of the site. (This way they are equidistant from both languages, amongst other virtues.) If you are creating an image that has textual function (like as sub banner), be sure to create an appropriate one in all languages.
- (d) If you wish to add a link to a page, do this by selecting the appropriate item from Dreamweaver's Modify menu. Here, one should be careful for another reason: too many different means of navigation looks ugly and also confuses users. Remember also that each page (via the template) has the navigation bar and the banner, both of which contain some links. Finally, the style sheet attached to the pages will style this link like the others; no formatting is required.
- (e) If you wish to add a "block" to a page, matters are a bit more complicated. First off, resist the temptation to use a table, since that spoils the markup policy. This policy is designed to have maximum flexibility when designing and using the site: a table is disastrous for programs like screen readers, etc. so we should avoid them. Instead a new or existing `<div>` block should be used. It should have its position specified either in the style sheet or in the HTML page in which it occurs.

Dreamweaver will automatically create some (style-sheet format) position code if you drag around a div block to position it.

- (f) To the extent that you can, it is best to enter your modifications in split or code view, rather than design. This way you have the highest possible control over positioning and will minimize the number of clashes with the template you encounter. (Dreamweaver will use grey to indicate where the template portions of the page are.)

(4) If you wish to move a page, move it using the Dreamweaver files panel. You can drag and drop files to move them around here and Dreamweaver will do the work of updating relative links for you. However, since we use subbanners to indicate relative location within the site, please update the relevant subbanner if the category of file changed as a result of your move. This involves both selecting both a new images file and rewriting the `alt` text. For example, if you moved a file from News to Publications, update the line that says something like:

```

```

to one that says:

```

```

(5) Sitemeter and other javascript-based items are available from a great many web pages. One can add these to our pages relatively easily – follow the supplied instructions as necessary, remembering that the template prevents you from adding to a large portion of the page. However, do not add such items willy-nilly, as they can do all sorts of horrendous things to the user experience, including crashing browsers, creating incompatibilities, etc. Certainly do not add any scripts that you do not understand. This document does not discuss javascript at all as it is well beyond its scope.

Once all of (1)-(5) is done, upload any changed files (complete folders, in the case of moved files) or do a synchronize. I have set up this to function with the selection of a particular menu item (i.e., Dreamweaver can do the login, transfer, etc. for you; it still has to be initiated. I recommend strongly against having automatic uploads, etc. as the remote site can function as something of a backup if you accident mess up files locally) from some computers. Presently “karen” and “corner” are set up that way, though at present only to the staging area, not the production area. (Use Dreamweaver to change to folder and account information by using the Sites menu.)

Please note that some pages or folders when opened in Dreamweaver will produce a “design notes” window. This will inform the modifier about certain important matters about changing that specific page. If you find that you have rendered a design note irrelevant, update its content appropriately. Similarly, feel free to comment your pages

and use plenty of design notes to document your work. This will make it easier for you to collaborate with others on the site.

Also, although this has nothing to do with web site design and implementation *per se*, it is important to note that the SJC has a bilingualism policy where possible on new pages. So, it is recommended to provide a translation of whatever page you have changed if you have changed the meaning of what was said. If you cannot do this on your own due to insufficient grasp of the other language, ask for assistance from volunteers or others.

### **Part B – Creating a new page**

To add a page, use a template. Open the appropriate one up (based on the language of the page you are creating – mockup2.dwt is the correct one in English; FR\_mockup2.dwt is the one in French) and save as the appropriate name in the appropriate location. The latter ensures the links will point to the correct relative locations. Then add content to the editable portions of the page per part A. If you find you cannot add in the relevant location, use the code view in Dreamweaver to make sure you are not trying to modify the read-only parts of the page. As it stands you may only change the subcategory banner, the title of the page, and in a restricted box of content in the lower right hand portion of the page.

As with the case of modifying a page, it is important to note that the SJC has a bilingualism policy where possible on new pages. So, it is recommended to provide a translation of whatever page you have added.

When you save your new page, please abide by the following guidelines:

1. Do not use any characters other than the letters a-z (in either case), 0-9 and the dash, period and underscore. The period should only be used to indicate a file extension in Windowsish style. If you feel the urge to use accented characters, don't use them. Instead use the relevant unaccented character. É should be E, for example.
2. The extension in question should be .html for all web pages on the site. Do not use .htm; life is easier if all files are named uniformly.
3. Be sure to save it in the right location; this allows Dreamweaver to calculate the correct relative links used in the template.
4. If you are providing (as you should) a document in multiple languages, use the same name in all places in which it appears. The existing site is more or less uniformly named in English. This practice makes parallelism easier and hence design faster and more productive.

### **Part C – Changing the look and feel of many pages**

First, think carefully about your decision. Do you really want to change all the pages that use a particular template? (This will usually be all the pages on the site in a given

language.) If you want to change just one, then stop and think again<sup>3</sup>. What you are proposing is that the templates were not designed correctly to begin with: by construction the templates are to be common at least for all pages of a given language. If you still wish to go ahead, see me, since this suggests a new version of the site at least to some degree. It is possible to have any number of templates, in principle. But they cannot “cascade” – i.e. you cannot base a template on another template and expect the changes to propagate through in the way you might want. (This would be a very nifty feature, but it would be very difficult to do properly.)

Next, locate the template from its local location and open it. Make the changes and save the template. Dreamweaver will ask if the files which use it should be updated or left alone. Update them: this is the whole point of using the template. Then, upload all the changed files. Note that there is not some “magic area” that stays as you transverse parts of the website. Rather there is some area that looks the same on all the pages of a given sort. The templating mechanism is all performed by Dreamweaver by automatically changing files for the user and is yet another good reason to use Dreamweaver as one’s primary web design and implementation tool.

Please note also that it is somewhat important to resist the temptation to add many features to all the pages. Pages with many graphics, large numbers of choices for the user, etc. are confusing and hard to use.

Finally, if you are replacing the main banner for the site, be sure to edit the <map> tags that allow users to click on parts of the image and get linked to some of the pages. You can use Dreamweaver to help with this, as it allows you to draw “hotspots” visually rather than having to manually work out coordinates.

### Problems

- (1) The part of the page I want to modify in part A won’t let me change it. (Or, I find I cannot add my content in part B.)

This is likely because you are trying to modify the templated portion of the page. Use the instructions in part C instead to make your change. Pay close attention to the first step of the instructions. Note that using the graphical editor sometimes presents problems with this. Use the code or split view to at least get started adding or changing content. Look for the <!-- InstanceBeginEditable ... --> <!-- InstanceEndEditable --> comment blocks for the beginning and end of areas one can change.

- (2) New content does not appear in a browser when checked.

---

<sup>3</sup> There is one page that is not templated like the rest of the site. This is the splash screen (the screen that appears at the top level of the site and has nothing but an image and a language choice), and is not discussed in this document.

First, always refresh/reload the page. Sometimes user agents fail to notice that pages are new. Also, make sure you uploaded the changes. (When in doubt, upload again.)

If necessary, quit and relaunch the relevant web browser. Some browsers too aggressively cache items such as `<iframe>` blocks and need to be restarted in order to have these items refresh.

View the source of the page and make sure all tags are typed correctly and have been matched with their ending in the correct place. In particular, do not stray out of the appropriate `<div> ... </div>` block.

(3) Content appears but has the wrong style (font, size, colour, etc.)

Again, make sure that the content is in the correct location. Also make sure that you enter the correct `<div>` to place the content if you are creating a new page. Check the spelling of its class or id.

(4) A link fails to work.

Check it carefully. Chances are it contains a typo. Make sure you have the right relative location, using `..` if necessary. (The dots mean “move up a directory”.) If it is a link in the template that fails to work, see me. A common error that can produce hard to track broken links is to not match case exactly. UNIX file systems are generally case sensitive. Windows ones are case preserving only. This entails that a link may work when tested locally but not on the uploaded version. The Dreamweaver broken link checker can help find some sorts of broken links.

(5) An update to a template failed to show up on the pages that use it.

Make sure that you have, indeed, allowed Dreamweaver to update the files that use the template you modified. If it turns out you forgot to, go to part C above and try again.

(6) An image fails to show up.

Make sure the file name is exactly correct and that you are pointing to the right location; also make sure that you have uploaded it correctly. (Upload again if necessary.) If you are linking to an offsite image, make sure the image is still at the location you pointed to. Also, make sure that you are allowed to make use of it in that fashion. (It is possible to configure a web server so that referrals to images only can be obtained from specific sites. In practice, this is hardly ever done, but in principle it is possible.)

(7) The page looks funny in Dreamweaver’s “Design” view.

Don't panic. This is normal; Dreamweaver is not a web browser, and it is certainly not a graphic design tool. Always use the "Preview in browser" submenu in DW's file menu to initially test your pages. (And, of course, test them on the real site once uploaded.)

- (8) The page I created looks different on different browsers or computers.

Yes, it probably does. Some of this is due to genuine differences in user settings and so forth. Many of our colour selections, for example, can be overridden by user preferences. Some of this difference, most notably when using Internet Explorer, is due to bugs in the browsers. This can be worked around with various "hacks" and the like. However, it is my opinion that such is not worth the effort. Careful use of the Target Browser Check in Dreamweaver can prevent egregious incompatibilities. None of these should occur if you don't "get fancy".

- (9) The content I wrote looks more or less correct except for certain specific characters (quotes, accented letters, etc. are the usual trouble).

Chances are you typed these characters in directly and Dreamweaver didn't catch them to turn them into their proper marked up equivalents. These characters can be variable across platforms or have special meaning to HTML and so have to be created in an unusual way. One must use what are called "entities" to create them. These are short words surrounded by the ampersand and semi colon characters. One such character that should be so entered is the ampersand itself; it is written: `&amp;`

Accented characters are generally written as the letter followed by the first few letters of the accent sort in English. Note that this is, of course, case sensitive. So `&eacute;` is é and `&Eacute;` is É. Enter these in Dreamweaver's code view to make sure they are entered correctly; it will help you to autocomplete the entity name. For a complete list of such names, go to:

[http://www.w3schools.com/tags/ref\\_entities.asp](http://www.w3schools.com/tags/ref_entities.asp)

### HTML and CSS

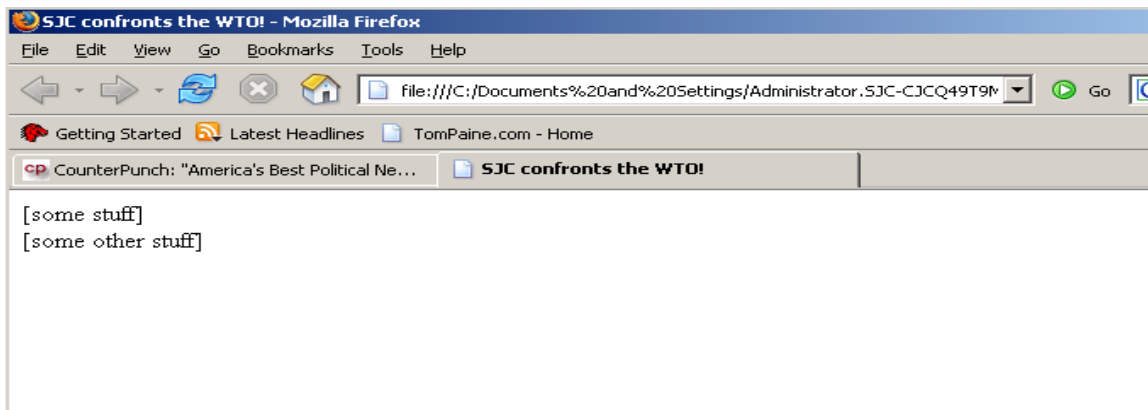
HTML (Hypertext Markup Language) and CSS (Cascading Style Sheets) are two "languages" used to create web pages. No tool (Dreamweaver in our case) is perfect, so some knowledge of what goes on is useful to understand errors that result, etc.

A web page first and foremost is a text file (i.e. a file with no extra control characters, etc.) placed on a server so that it can be accessed by a "user agent" (usually a web browser, though other sorts of user agents exist) across a network (or viewed locally, though that's of little concern to us here). A web page is downloaded and parsed by a user agent, so it can be displayed or turned into speech (etc.) by such software. This parsing

first distinguishes between the content of the page and how is to be arranged, formatted, etc. For example, a page may look like this as it downloaded (greatly simplified, of course):

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" http://www.w3.org/TR/html4/loose.dtd>
<html>
  <head>
    <title>SJC confronts the WTO!</title>
  </head>
  <body>
    <div class="navigation">
      [some stuff]
    </div>
    <div class="content">
      [some other stuff]
    </div>
  </body>
</html>
```

It will be displayed by a visual user agent perhaps something like this (predicting exactly how is difficult. This is important: HTML by itself is not a layout language, much less a print layout one):



The items in `<>` in the page are called tags. Most of these are written in lower case, in part because of ease of typing, partially because other web standards we may wish to comply with some day require it. They get parsed out by the web browser (notice they aren't visible in the version of the page as it appears in a web browser) and inform it how to arrange and structure the content. From our perspective, all pages will have that weird `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">` as the first item. This tells the web browser what sort of web page it is. There are various web standards, and the SJC's website is designed to comply with one of them. Next is the `<html>` tag, which tells the browser where the HTML for the page actually begins. Inside this section is the `<head>`, which includes features for search engines, a link to the SJC's style sheet (of which more in a moment), and a `<title>` area, which determines the title of the page. This is often displayed at the top of a window in graphical user agents, here:

In this case, 'CounterPunch: "America's Best Political Newsletter"' is the title.

The `<head>` area ends with a `</head>` tag. Generally tags come in pairs: one indicates the beginning of a section; the other, with the `/`, indicates the end. Some tags in the current HTML standard are also "self-ending", so for example the `<img>` tag actually looks like ``

So, you can see that the `<html>` tag also includes a `<body> ... </body>` section. This is where most of the content of the page goes. Other tags structure pieces of the layout, indicate different sorts of headings, link to other pages, reference images, etc. Pages created with our templating mechanism will also have a lot of "comment" tags, which look like `<!-- [stuff] -->`. A user agent ignores comments. However, since they are ignored, they can be used by programs like Dreamweaver to hide mechanisms for their templating process and other purposes. This is vital to remember if one is ever making modifications to a page using any other tool. One can, of course, use comments to indicate to humans reading one's HTML various important things one may wish them to know. Note also that tags include a name (the first characters after the `<`) and often one or more attributes, such as the `src="robin.jpg"` in the above `<img>` tag.

CSS is the modern way used to select fonts, colours, positioning, etc. (There is an old way to do these things, which for many reasons is now regarded as obsolete, not the least of which is the level of flexibility CSS offers is much higher. Those who know something about HTML but not CSS may remember the `<font>` tag. This tag is amongst the most obsolete.) A CSS document is associated with a web page in various fashions. I'll ignore the details; all one needs to understand at this point is that most of our style sheets are in external files. (I.e., in different files from the web pages that use them.) This is one of their strengths: one can have many pages use the same style sheet and thus make it easier to change stylings (and even structure, as positioning can also be controlled by CSS). For example, using old timey markup, we would have to change hundreds of pages on the site manually if we wanted to turn a brown background in one common place to a green one. With CSS, all one does is change the colour "set aside" for that section in the style sheet document. CSS syntax is quite complicated (at least relative to HTML) so I will not address it here. Both it and HTML can be studied by means of the tutorials at <http://www.w3schools.com/>. I encourage interested readers to look at this good reference and tutorial site.

If you are interested in modifying colour selection on the site and other such factors, it is also important to realize that the factors selected on the site have been studied and came somewhat out of a process of user testing. Any substantial change to the site should follow similar procedures. As running such a test is outside the scope of this document (and relies on skills I do not feel qualified to teach, e.g., elementary statistics), I will not discuss this matter further. See the accompanying document, "SJC Internal Beta Test.doc" for an example questionnaire. A general piece of advice about web site design

and implementation applies here: a web page is not in general a work of art, which is not to say it is free from artistic qualities. (If anyone wishes to learn more on this subject, see me, as I have some papers in this area.)

### Accessibility

Software and web pages are said to be accessible to the degree that users of varying abilities, platforms, etc. can make use of them. There are thus many aspects of this concern forming a huge topic. I will only touch upon these issues here. (Interested readers may consult the massive literature available at <http://www.w3c.org/> on this topic to learn more.) There are three areas I would like to mention briefly: handicaps, bandwidth and external file formats.

Handicapped users sometimes have a great deal of trouble using a computer to do various tasks that non-impaired users take for granted. There are many assistive technologies that can alleviate some of these problems, but these work best when content providers and designers such as the SJC keep them in mind.

First, the blind often use screen readers to make use of computers. For this reason I noted above that `<table>` tags used for layout are a recipe for disaster. Similarly, it is vital that all `<img>` tags get flagged with an `alt` attribute. Any substantial amount of written words within an image should appear in this attribute, and large images that take the place of text should be avoided. (Note: there is such a page on the existing site that should be phased out eventually.)

The colourblind remind us that colour should never be the only cue to important navigation items. This is true in the existing site design, and only needs to be remembered if the CSS file is changed for whatever reason.

It is usually (for bandwidth reasons) bad practice to require sound to navigate the site, but if it is ever so used on the SJC site a written version of any sound files so needed is good practice to permit deaf users.

The existing pages are created also in such a way as that they do not become unmanageable for those using screen magnification tools. Similarly, our text sizes are selected to be reasonably clear on common resolutions of displays.

As for bandwidth, if the SJC were a commercial organization I would not worry about users complaining about bandwidth on the actual web pages. But since the SJC does deal with world wide issues pertaining to poverty, it stands to reason that some of our design policy and the like should take it into account. Large, graphics rich pages are obnoxious even for users with “first world” bandwidth; users with dialup connections on noisy lines will find such pages unusable. For the same reason large files linked to on the site should be labeled with a warning as to their size. (There is no sense in providing a download time, as too many users do not know their own connection speed and are not likely willing to perform the calculation necessary to work out the time from the provided one

either.) At present, I have only flagged two large video files in this way. I recommend any files over about 10 megabytes should be so indicated. Warning: the SJC quota (1 gigabyte) at its hosting service will severely limit the number of such large files as well. Be judicious in deciding what needs to be made available on the site as enlarging our quota will cost the SJC more money.

This brings me to my last subtopic within accessibility, namely file formats. Most pages on the site are to be brief and to the point. Substantial policy papers and other documents do not work out too well in HTML format without a great deal of work. For this reason I stress using another format for such documents, particularly ones with graphics, charts, tables, equations, etc. It might seem like the world uses Microsoft Office, but this is a horrible format to use for any number of reasons. Instead, such documents should be made available in Acrobat Reader (PDF) format. Such a format is a bit better supported and is slightly less obnoxious for various reasons. Note, however, that it and any file format other than GIF, JPEG and PNG (three graphics formats) should be labeled as to their format. For example consider the following possibility for a link on a page:

[“A Global Village” 164 page analysis of small communities and their trading partners.](#)

Instead this should say something like:

[“A Global Village” 164 page analysis of small communities and their trading partners.](#) [Acrobat Reader format; large file due to many colour photographs]

Audio format should be MPEG layer three (.MP3) with a sensible bit rate (128-256 for music, 32 or 64 for voice) unless the file is extremely short in duration in which case Windows Wave (.WAV) or Sun audio (.au) are suitable. In all of these cases, the format must be mentioned to the user. Video should be (and isn't in the existing site, alas) in QuickTime format with a default codec or a wrapperless MPEG-1 format file. These ensure maximal playbackability across platforms and locations. Note that the SJC has no capacity for streaming media, whence all files have to be downloaded prior to being viewed/listened to. Also, it is strongly suggested that translations of audio items be made available if possible; in any case indicate the language used in media files. If you do not know what these are or you find this part confusing, consult with Keith on such formats before using audio and video formats.

There are several file formats which new content should avoid. Real Media (.rm, etc.) and Windows Media Player (.wmv, etc.) are two. These are horrible formats for any number of reasons. Similarly, although Photoshop is a great program, one cannot rely on the user having a copy, hence no Photoshop files should be distributed casually. (Photoshop format files also might contain private information like an undo history, etc. that one might not want to distribute.) Similarly, make sure that if you used a digital camera to create images that they are in a format that can be displayed by web browsers. (Sometimes cameras use their own JPEG format.) As noted above, Word and Excel are also bad choices; WordPerfect and other word processors are even worse choices, as they are substantially less popular. This brings me to the next and final section of this

document, namely other tools. (On the off-chance that the site has to include files that the user is supposed to be able to edit, see me.)

### Other Tools

This document has been written with using Dreamweaver in mind. This is not essential; if the user wishes to make use of other tools she is welcome to do so. However, many of these are really unsuitable for the job. The easiest to integrate with Dreamweaver are a good text editor (TextPad is a good Windows one; TextWrangler is good on the off chance someone wants to work on the site from their own MacOS X machine) and an FTP client such as the one built into Windows or various others. These are also the hardest to use from the perspective of learning the ins and outs of web design and implementation. If you are needing to read this document, chances are this approach is not for you. Moreover, such an approach does not lend itself to using the Dreamweaver templates in use. This latter concern also makes using FrontPage, GoLive and other graphical web design tools other than DW awkward or pointless. So, be careful if you wish to use these. At best they will complement DW, not replace it.

What is even worse is using Word (and Excel, but there is less danger of that) to create pages. Word does have a “web page” format listed in its Save As ... dialogue, but this should **NEVER** be used. The absolute worst of the worst pages will result. This is true even if you copy out the page that results and paste into a templated page created by Dreamweaver. Similarly, this advice applies to virtually every word processor on the planet.

There is also a danger of using content created for other purposes and trying to cut and paste “just the text” into new web pages from Word. Web pages, in order that they are viewable on a wide variety of operating systems, user settings and so forth, use their own means of doing things like accents, font choices and so on. There is much to be discussed in this area – too much for this document to detail. Writing a plain text version of what content you want to be made available on the site is for the best, but I have found that many people do not know (through no fault of their own) what plain text involves. To this and other ends, I have proposed a small workshop on “writing for the web”. You might have gotten this document through such an event; otherwise tell Keith you are interested in having such a workshop.