

# Contact Sheet X

Contact Sheet X (CSX) is full featured contact sheet generator for Adobe Photoshop CS2, CS3 and CS4. Written in JavaScript, the core of the package has capabilities that include everything found in Contact Sheet II (bundled with Photoshop) and contact sheet facilities found in other applications such as Adobe Lightroom.

CSX builds on the functionality provided in CSII and extends it in many useful ways. For instance, CSII offers a limited number of fonts for captions whereas CSX offers access to the complete set of fonts available to Photoshop.

CSX also has completely new functionality such as being able define a template document into which contact sheets will be inserted as well as the ability to specify captions that contain image metadata. Captions may also have multiple lines.

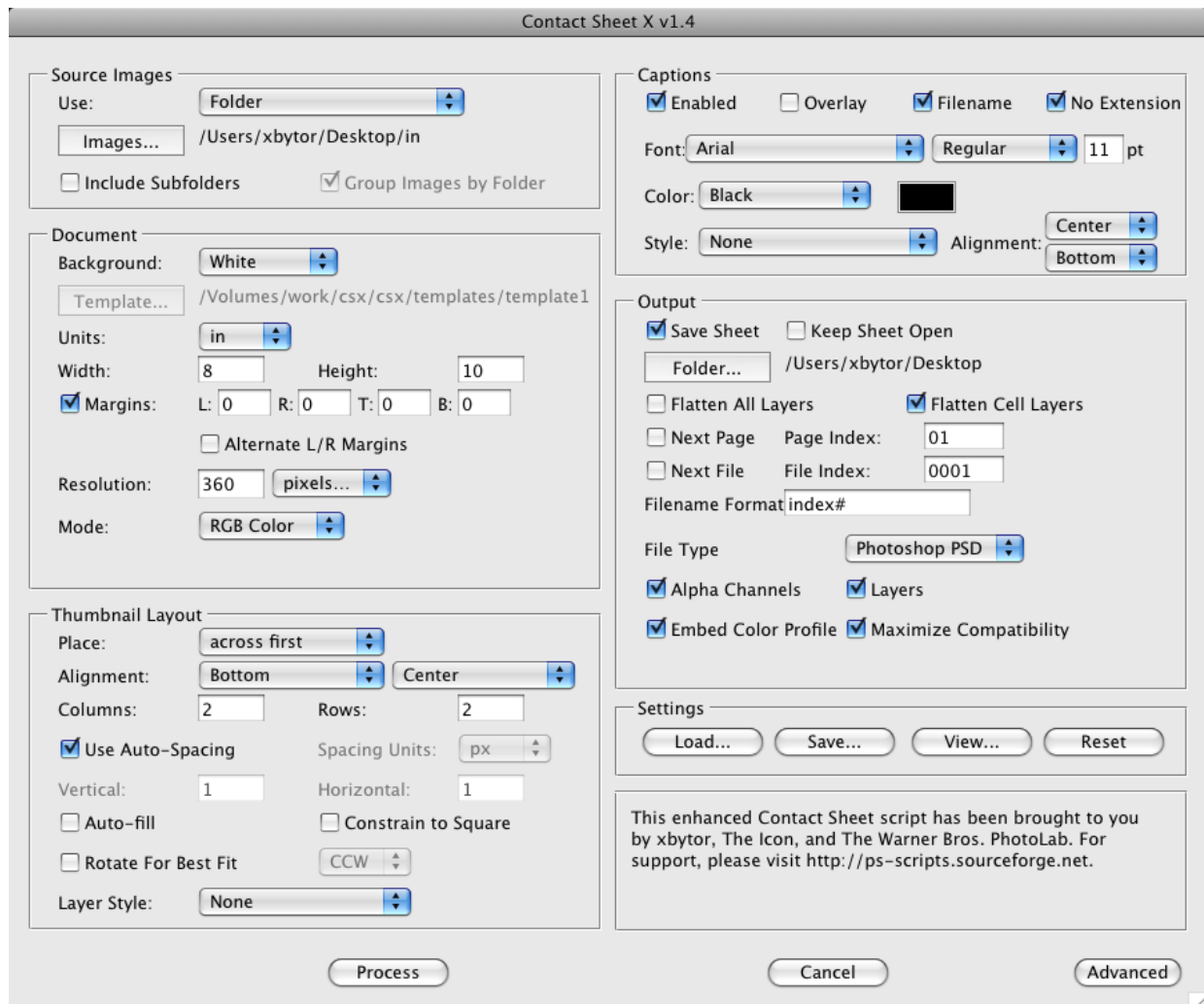
This version also contains new features that allow it to be used as a full fledged Image Caption script, similar to *Dr. Brown's Caption Maker*.

CSX has been tested with CS2, CS3 and CS4 on both the Mac and PC platforms.

From Photoshop CS2, CSX will be in the File Scripts Menu (**File > Scripts**) as 'CSX-1\_4' (or something similar).

From Photoshop CS3 and CS4, CSX will be in the File Automate Menu (**File > Automate**) as 'Contact Sheet X v1.4...' (or something similar).

When CSX is launched, the first dialog you see is this:



The 'Advanced' button in the bottom right of the window will change CSX into a tabbed-panel window. The tabs in the *Advanced* window roughly correspond to the panels in this (the *Standard*) window. Later in this document, the individual panels will be described. The descriptions and screenshots will be of the *Advanced* UI which may contain more or different fields than is contained in the *Standard* UI.

## Installation

This release includes an installation script, *InstallCSX.jsx*. To install CSX using this script, you first need to unzip the package to some temporary location. Where this is doesn't matter since you are going to delete these files after installation. Do not, however, unzip the package in any Adobe folder whatsoever. I typically unzip the package on my desktop.

Now run the script. You can do this by:

1. double clicking it
2. dragging it onto a PS window or application icon  
or
3. running it from inside PS via *File->Scripts->Browse*

The installation process is completely automatic but there are confirmation prompts to let you know what's going on. When the installation is finished, you will have a file called *InstallCSX.log* in your home directory that details what happened during the installation process. If there was a problem with the installation, this log file should contain useful information.

If you are running more than one version of Photoshop, you will need to install CSX on each version.

When you are finished, you can delete the folder that you unzipped the package into and the installation log file now that everything is complete.

### Manual Installation

If you ran the *InstallCSX* script, you do not need to go through these steps.

To install CSX manually, unzip the distribution zip file into your Adobe scripts directory `<Adobe Photoshop>/Presets/Scripts` where `<Adobe Photoshop>` is your Photoshop installation directory. You should end up with a directory structure like this:

```
.../Presets/Scripts/  
    csx/  
    csx/templates  
    csx/Bridge Scripts Only
```

At a minimum, the following files will be present:

```
Presets/Scripts/CSX-1_3.jsx  
Presets/Scripts/csx/README  
Presets/Scripts/csx/CHANGELOG  
Presets/Scripts/csx/RELEASENOTES  
Presets/Scripts/csx/Extensions.txt  
Presets/Scripts/csx/CSX.pdf  
Presets/Scripts/csx/templates/CSXTemplate-Default.psd  
Presets/Scripts/csx/templates/CSXTemplate-LayerText.psd
```

```
Presets/Scripts/csx/templates/*.ini  
Presets/Scripts/csx/Bridge Scripts Only/CSXBridge-1_3.jsx
```

## Bridge Support

CSX provides direct integration with Bridge. This integration includes the ability to launch CSX from within Bridge (from the *Tools* -> *Contact Sheet X...* menu item) and the ability to specify Bridge as a document/image source from within CSX.

### Manual Installation

If you ran the *InstallCSX* script, you do *not* need to go through these steps.

To enable Bridge support, you need to copy the script

```
csx/Bridge Scripts Only/CSXBridge-1_3.jsx
```

to one of the following folders:

#### On WinXP CS2

```
%CommonProgramFiles%\Adobe\StartupScripts
```

#### On WinXP CS3

```
%CommonProgramFiles%\Adobe\Startup Scripts CS3\Adobe Photoshop
```

#### On WinXP CS4

```
%CommonProgramFiles%\Adobe\Startup Scripts CS4\Adobe Photoshop
```

#### On OS X CS2

```
/Library/Application Support/Adobe/StartupScripts
```

#### On OS X CS3

```
/Library/Application Support/Adobe/Startup Scripts CS3/Adobe Photoshop
```

#### On OS X CS4

```
/Library/Application Support/Adobe/Startup Scripts CS4/Adobe Photoshop
```

`%CommonProgramFiles%` is usually equivalent to something like:

```
C:\Program Files\Common Files
```

For CS3 and CS4, the first time you run Bridge after installing this script, Bridge will pop up a window alerting you that a new component (CSX Bridge) is available. Respond 'Yes' when it asks if you want to install the component.

## Runtime Files

CSX automatically creates three files when it executes: an *INI* file (*ContactSheetX.ini*), which is

used to store your settings/preferences, and two *Log* files (*ContactSheetX-UI.log* and *ContactSheetX.log*), which is used to record the actions of the most recent execution. These files are discussed in a later section.

These runtime files are stored in an Application Data/Support folder. The location of this folder on WinXP is something like:

```
C:\Documents and Settings\[user id]\Application Data\xtools
```

On OSX, the location is something like:

```
/Users/[user id]/Library/Application Support/xtools
```

where [user id] is your login ID on your computer.

When reporting a problem, please include the log files in your post or email. They provide information useful in resolving problems.

## CSX Features

### Templates

One of the more innovative features of CSX is the ability to specify a template document into which a sheet will be placed. A CSX template file is typically a multilayer PSD file. Any PSD file will work, but, depending on the names of the layers present in the document, you can have a great deal of control over the final appearance of the contact sheet.

### Special Layer Names

The names of the layers in a template dictate what will be inserted into the layer and how that information will be formatted. Except for *Contact Sheet*, the 'special' layer names that CSX recognizes all begin with the '@' character. All other layers are ignored and will be present in the final contact sheet document without modification. This is useful if you want to have standard background graphic layers for your sheets, for instance.

The layers that CSX does handle are:

- *Contact Sheet* – This names layer into which the sheet will be inserted.
  - There should normally be a mask on the layer to specify where the sheet should be placed. If there is not a mask, the full size of the document will be used.
  - Any layer content (e.g. fill indicating the area that the sheet will occupy) will be replaced by the actual sheet content.  
*Note: I made the decision to do this almost arbitrarily. If you have a need to have this work differently, let me know.*
  - Any style applied to the layer will be retained.  
*Note: I'm not sure how useful this actually is, but I didn't see any reason to not retain the layer style.*
- *@Page Number* – This names the layer used for page numbers. Page numbering starts with the value in the *Output->Page Index* field in the user interface or whatever the next page number is determined to be. The attributes of text layer determines the font, font size, color, alignment, etc... of page number on the contact sheet. The contents of this layer specify how the page information is to be formatted. See the **@Page Number Format Specification** section later in this document for details on formatting this layer.
- *@Date* – This layer name is used for inserting the date and/or time that the contact sheet was generated. The contents of this layer specify how the date/time is to be formatted. If no format is specified, a default format is used. The existing text attributes (font, color, etc...) are used.
- *@Metadata* – This layer name is used for inserting metadata into the template. The metadata is retrieved from the first image of each sheet. The attributes associated with this layer are retained through substitution.
- *@...* - Any other layer that begins with a '@' character. The attributes of text layer

determines the font, font size, color, alignment, etc... of date text on the contact sheet. See the **@Date Format Specification** section later in this document for details on formatting this layer.

## Layer Text Substitution

Layer text substitution is a mechanism whereby the contents of text layers can be substituted by text specified by a user either in an INI file or via the CSX user interface.

There are two predefined substitutions that occur. If you create a text layer in your template with the name *@Date*, the date at the time you ran CSX will be inserted into the text layer based on the format specification in that text layer. *@Page Number* is used for inserting page numbers into a text layer using a similar format specification. These layers are strictly optional. You may also have more than one of each of these layers possibly with different format specifications.

You can also define your own substitutions. An example of one that I defined (and provide a default for) is *@Credits*. To substitute text in for the *@Credits* layer, you can add a line to your INI file that looks like:

```
@Credits : Photos by xbytor
```

CSX also has the ability to edit the template variables directly in the CSX UI. This eliminates the need to edit the INI file to set template variables. See the **Document Panel** section for more information on editing template variables.

The template file *CSXTemplate-LayerText.psd* that is included with script illustrates how these substitution layers can be used.

A more detailed tutorial on creating and using templates will be included in a future version of this document.

## @Page Number Format Specification

*@Page Number* layers have their contents determined by the format specification that resides in that text layer. CSX will replace portions of the original layer contents based on these mappings:

<i>%0p</i>	The current page number, zero padded.
<i>%p</i>	The current page number with no padding.
<i>%P</i>	The total number of pages.

For example, if a *@Page Number* layer in a template contains the format specification of *"Page %p of %P"*, the layer text after substitution will look like *"Page 5 of 25"*. A page format

specification of "*Page %0p*" will result in layer text that looks like "*Page 05*".

### **@Date Format Specification**

@Date layers have their contents determined by the format specification that resides in that text layer. CSX supports two styles of format specification.

A date/time of October 31, 2006 20:43:02 will be used for the example below.

The first style of format specification has these mappings:

<i>YYYY</i>	Four digit year. <i>2006</i>
<i>YY</i>	Two digit year. <i>06</i>
<i>MM</i>	Two digit month. <i>10</i>
<i>DD</i>	Two digit month. <i>31</i>
<i>H</i>	Two digit hour on a 24 hour clock. <i>20</i>
<i>I</i>	Two digit hour on a 12 hour clock. <i>08</i>
<i>M</i>	Two digit minute. <i>43</i>
<i>S</i>	Two digit seconds. <i>02</i>
<i>P</i>	AM or PM. <i>PM</i>

Using these formats, you could create a text layer with the name @Date, set its contents to "*Created on YYYY/MM/DD*", and end up with a layer that looks like "*Created on 2006/10/31*".

The second style of format specification is based on the Unix *strftime(3)* implementation and is explained in detail in **Appendix A: Date Format Specification**.

CSX determines which date format style to use by looking for a '%' character in the format specification. If there's a '%' in the string, the script assumes that the format specification is a *strftime(3)* style format. If not, the *Y/M/D* style is used.

### **@Metadata Format Specification**

@Metadata layers have their contents determined by formatting information in the layer name and/or in the layer text. The formatting options are almost identical to those in the Caption.

If there is more than one image on the sheet, the metadata is selected from the first image on the sheet *only*. If there is only one sheet on the page, CSX is effectively operating as a Caption Maker script.

Examples:

*None Available at this time*



## INI and Log Files

CSX uses an *INI file* to store settings between executions. This text file can be manually edited to change the values of various settings. While most properties are set via the UI, some can only be modified by editing the file manually.

INI files can be saved and reloaded as needed. The "*Presets*" feature of CSX uses saved INI files as the *Presets* files. The major difference between loading a presets file and loading an INI file is that a presets file may only modify a subset of the CSX settings. Additionally, paths to folders and files may have to be dealt with as special cases. This simplifies the sharing of presets and templates even though paths to files and folders are likely to differ from machine to machine.

CSX also keeps two logs of events that happened during an execution. While this may not be very useful for normal work, it aids greatly in tracking down problems when they occur. One log file tracks interactions in the user interface while the other tracks events that occur after the *Process* button is pressed.

The location of the ini and log files on WinXP is something like:

```
C:\Documents and Settings\[user id]\Application Data\xtools
```

On OS X, the location is something like:

```
/Users/[user id]/Library/Application Support/xtools
```

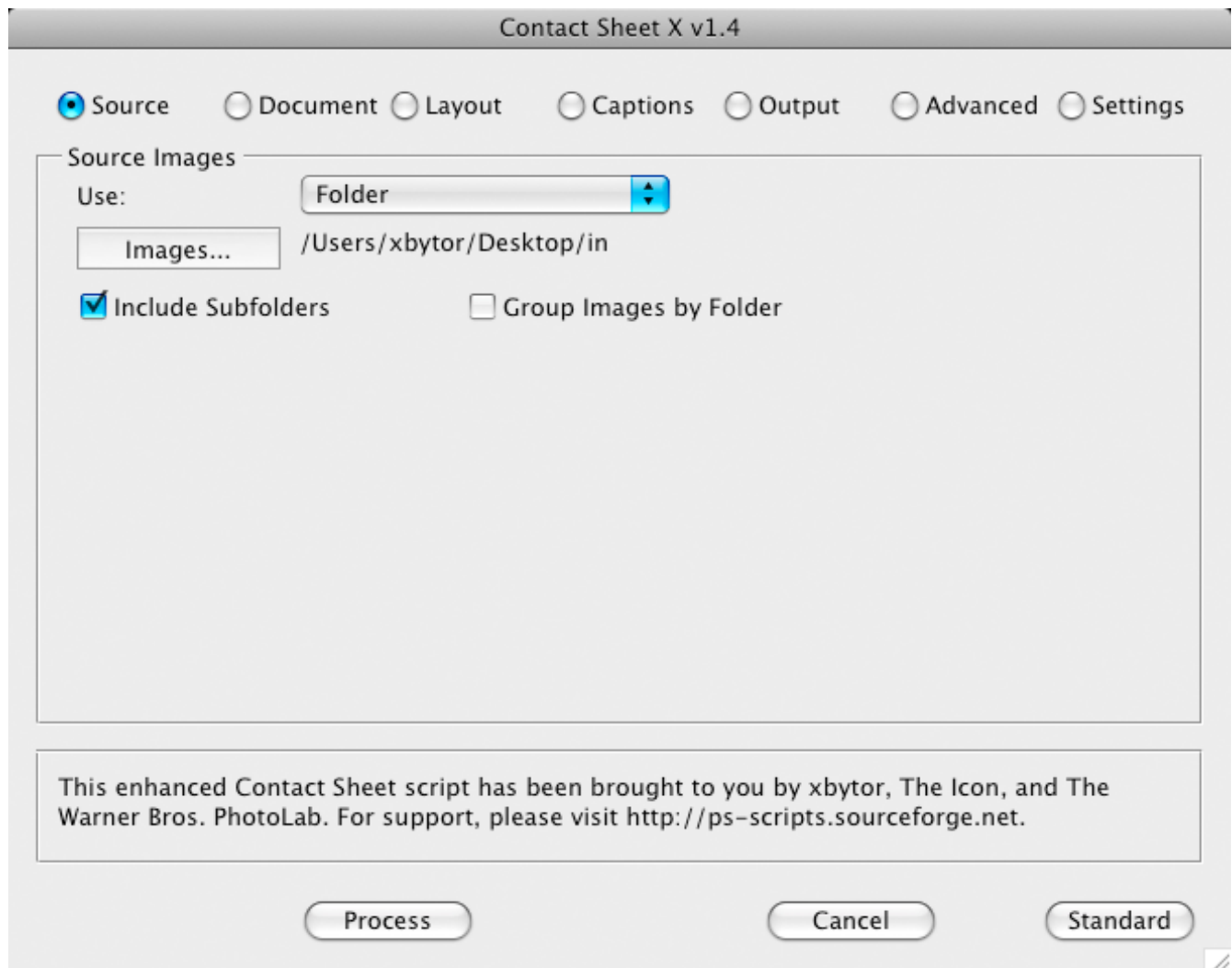
where `[user id]` is your login ID on your computer.

Note that other scripts may also use this folder to store there INI and log files.

## CSX User Interface Panels

### Source Images

The source panel captures all of the information necessary to specify the source of the images for the Contact Sheets that will be generated.



- **Use**

One of *Folder*, *Open Documents*, *Bridge*

*Folder* – use the folder specified by *Images* as the source for sheet images.

*Open Documents* – use the set of currently open documents as the source for sheet images. This option is only present if there are open documents.

*Bridge* – use the current set of selected documents in Bridge. This

option is present only if there are documents selected in Bridge.

- **Images** Opens a folder selection dialog for specification of the folder containing the source images to be used by CSX.
- **Include Subfolders** CSX can work on a hierarchy of folders. This checkbox turns that functionality on and off.
- **Group Images by Folder** By default, when processing a collection of image folders, if the current sheet is not complete when a folder is finished CSX will use images from the next folder to complete the current sheet. Setting this checkbox will force a new sheet to be started at the beginning of each image folder.

**Notes:**

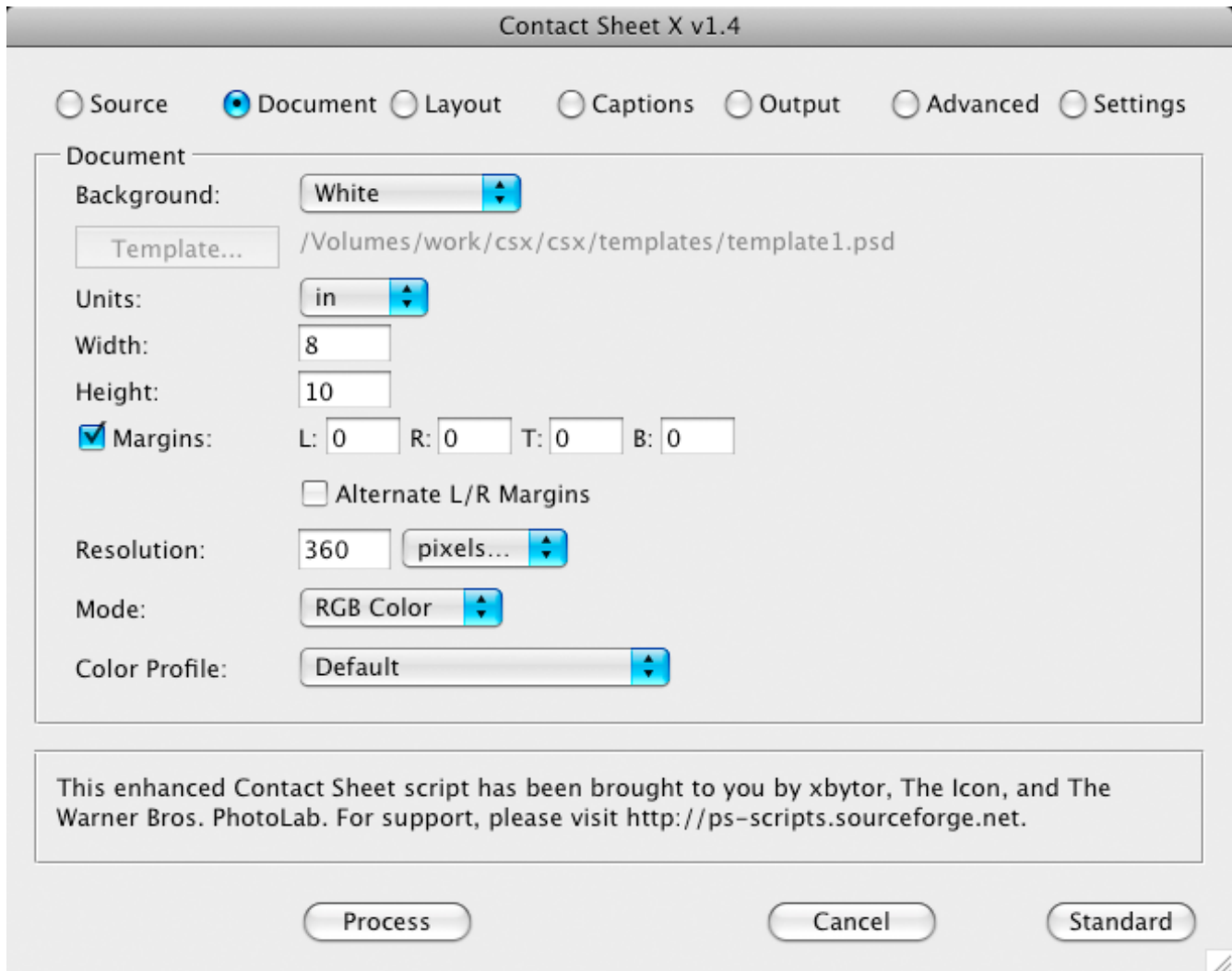
There are no known problems with this panel at this time.

The *Bridge* setting uses the files that were selected in Bridge when CSX was started. If the Bridge selection changes while CSX is running, it will have no effect on the files that are processed.

There was a bug in versions prior to v1.4 where CSX would not descend more than 2 levels of folders when processing a hierarchy. This has been fixed.

## Document Panel

The Document panel lets a user specify either a template file or the document attributes of the sheets being generated.



- **Background**

One of *Template*, *Black*, *White*, *Foreground*, *Background*, *None*

*Template* – Use a psd file as a template for the contact sheets.

*Black* – Use the black as the background color.

*White* – Use the white as the background color.

*Foreground* – Use the current foreground color as the background color.

*Background* – Use the current background color as the background color.

*None* – This either makes the background transparent or removes it altogether.

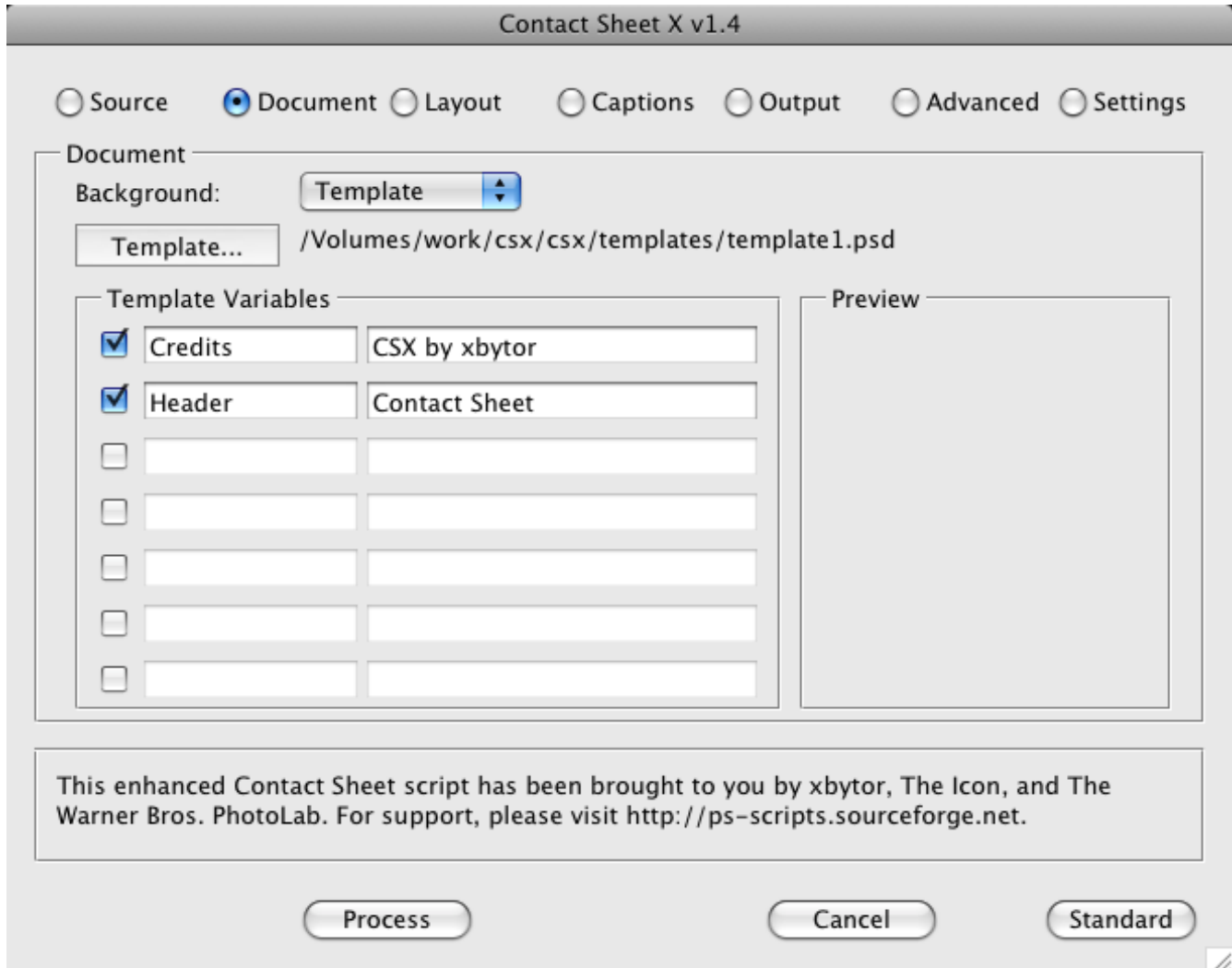
- **Template** A file selector to choose the template for CSX. This is only enabled if **Background** is set to *Template*.

If **Background** is set to *Template*, the following **Document** settings are disabled: CSX determines the appropriate values for these settings by the information in the template file. A different panel is presented for *Template* mode which is discussed later in this section.

- **Units** One of *in,cm,px*. Specifies the unit type for the **Width**, **Height**, and **Margin** settings.
- **Width** The width of the sheet.
- **Height** The height of the sheet.
- **Margins** The left, right, top, and bottom margins.
- **Alternate Margins** If checked, the left and right margins will be swapped for alternate pages. This is intended for use with two sided printing where a binding margin is required.
- **Resolution** The resolution of the sheet.
- **Resolution Units** One of *pixels/in, pixels/cm*. Specifies the unit type for the **Resolution** setting.
- **Mode** One of *RGB Color, CMYK Color, Lab Color*. Specifies the color mode of the contact sheet document.
- **Color Profile** Specifies the color profile to be used for the contact sheet. The *Default* option selects the current working color space. The *Match First Image* option uses the color space of the first image to be processed. Common color spaces are also in the menu. The *Other...* option allows a color profile to be loaded from a file; this particular functionality is not very stable.

## Template Mode

If *Template* is selected as the background, the panel changes to this format:



- Preview**

Some templates may have *Preview* images. If one is found, it is displayed here.  
*Note: A Preview images is an example of the template stored as a .png file in the same folder as the template.*
- Template Variables**

This panel is an editor for template substitution variables. The checkbox on the left turns a variable on. The first field is the name of the variable while the second is the value that is to be substituted.

In the panel above, two variables have been defined, *Credits* and *Header*. When each sheet is generated, CSX looks for text layers named *@Credits* and *@Header* and places the values

entered in this table into the text layers.

The separate **Header** setting that was present in previous version of CSX has been removed.

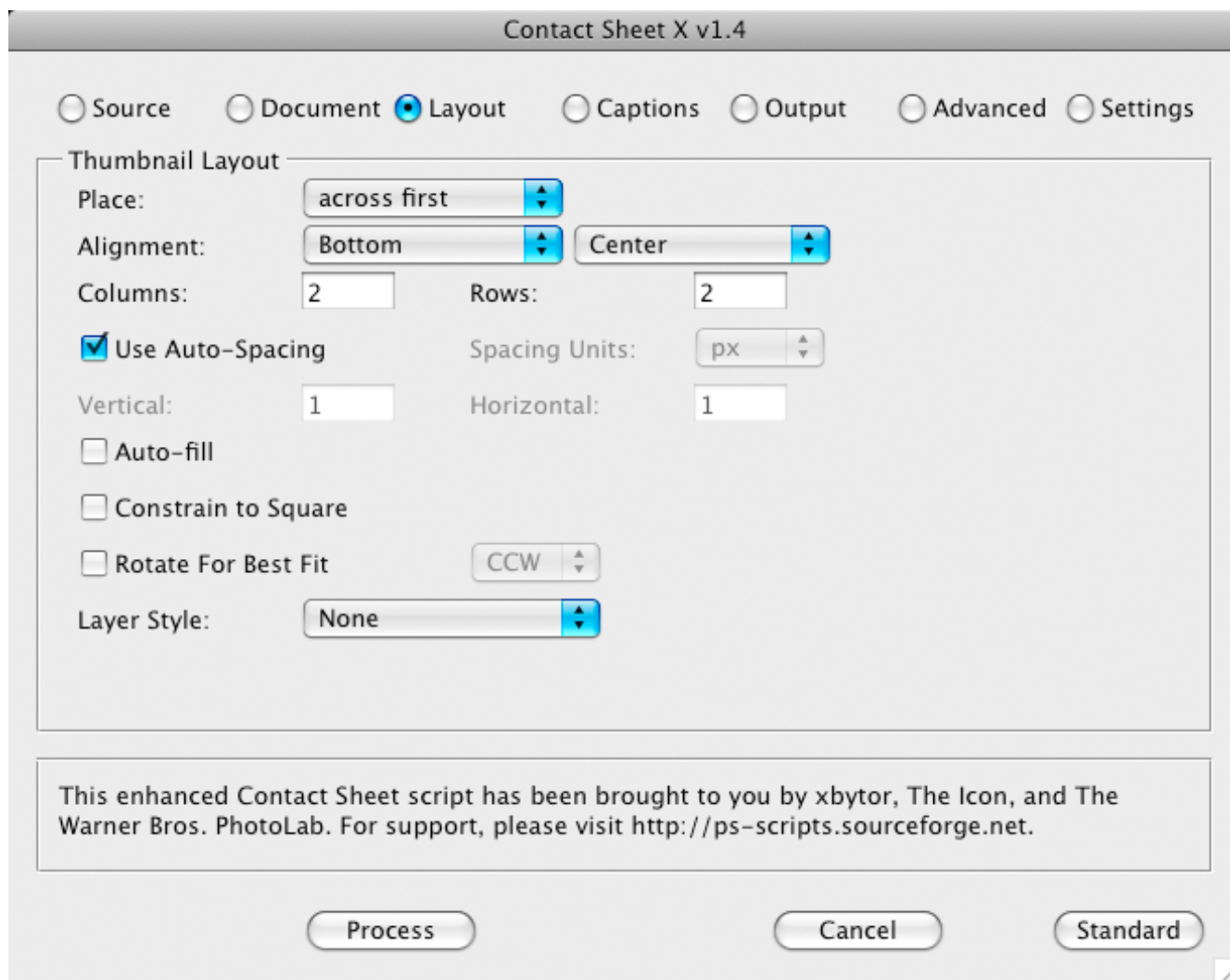
**Notes:**

There are no known problems with this panel at this time.

There is no way to remove template variables once they have been defined except to delete them from the CSX ini file.

## Thumbnail Layout

The Thumbnail Layout panel collects information specifying how thumbnails will be laid out in the sheets.



- **Place** One of *across first*, *down first*. Specifies the sequence in which images are placed in the sheet.
- **Alignment** The first is one of *Top*, *Middle*, *Bottom*. This specifies the vertical alignment of images in a row.  
The second is one of *Left*, *Center*, *Right*. This specifies the horizontal alignment in a column.
- **Columns** Specifies the number of columns of thumbnails in the sheet.
- **Rows** Specifies the number of rows of thumbnails in the sheet.



- **Use Auto-spacing** If this setting is checked, images are automatically spaced apart from each other. The auto-spacing interval works out to be about 1% of the width of the sheet. This value can be changed by modifying the `autospaceFactor` setting in the INI file.

If **Use Auto-spacing** is selected, the **Spacing Units**, **Vertical**, and **Horizontal** settings are disabled.

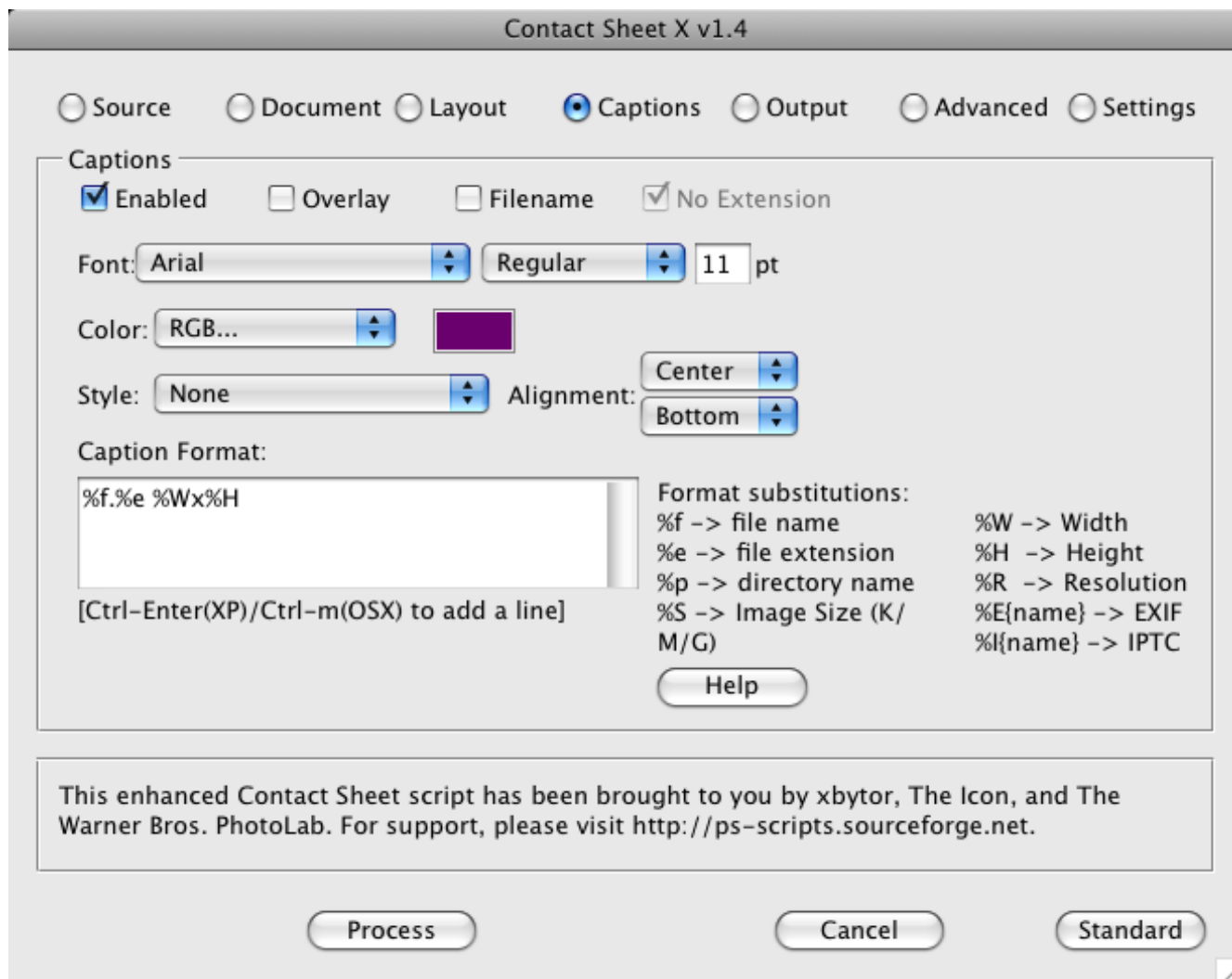
- **Spacing Units** One of *in*, *cm*, *px*. Specifies the unit type for the **Vertical** and **Horizontal** settings.
- **Vertical** The vertical spacing between thumbnails on the sheet.
- **Horizontal** The horizontal spacing between thumbnails on the sheet.
- **Auto-fill** If this setting is checked, the image will *fill* all of the available space even if it requires cropping the image. Normally, an image will be *fit* into the available space.
- **Constrain to Square** This setting forces the area into which a thumbnail is placed to be a square. The effect of this is to keep landscape and portrait images at the same apparent scale.
- **Rotate For Best Fit** If this setting is checked, an image may be rotated 90° to optimally fill the available space.
- **CW/CCW** This determines the direction in which an image will be rotated, if **Rotate For Best Fit** has been selected.
- **Layer Style** The Style to use for the thumbnail. The styles in this list are those that are currently available via the Photoshop Preset Manager.

**Notes:**

- Future releases of CSX will include the ability to specify Layer Styles from a predefined set of styles not currently loaded in Photoshop. There will also be a *Preview* capability to show the effect of layer styles on a thumbnail.
- The presence of captions (and their settings) will also have an impact on thumbnail spacing.

## Captions

The Captions panel is used to specify the form of thumbnail captions.



- **Enabled** Enables captions in contact sheets.
- **Overlay** Determines whether or not the caption will overlay the thumbnail or be placed in a space just outside of the thumbnail.
- **Filenames** Specifies that the image's filename should be used as a caption. If this is not selected, a *Caption Format* should be specified.
- **No Extensions** The filename's extension will not be included in the caption.
- **Font** Specifies the family, style, and size of the font used for the caption.

- **Color** One of *Black, White, Foreground, Background, RGB...*  
  
*Black* – Use the black as the text color.  
*White* – Use the white as the text color.  
*Foreground* – Use the current foreground color as the text color.  
*Background* – Use the current background color as the text color.  
*RGB...* - Use the specified RGB color as the text color.  
  
 The color icon launches a color chooser for selecting the. On CS3 and CS4 the system color picker is used.  
*Note: The CS2 color chooser is not very sophisticated.*
- **Style** The Layer Style to use for the caption text.
- **Alignment** One of *Left, Center, Right*. Determines the horizontal alignment of the caption in the space available.  
 One of *Top, Middle, Bottom*. Determines the vertical alignment of the caption in the space available.
- **Caption Format** This setting specifies what should be placed in the caption. This field can contain multiple lines.

### Caption Format Details

CSX provides a highly flexible facility for specifying image captions that goes far beyond a simple file name. Caption Formats work much like the other substitution specifications used elsewhere in CSX. Patterns like '%f' get substituted with various forms of image metadata. A standard filename caption would be specified like this: %f.%e where %f gets replaced with the filename and %e gets replaced with the file's extension.

Here is a partial list of the basic substitutions available:

%d	Directory (e.g. c:\work\out)
%e	Extension (e.g. .jpg)
%f	Filename (e.g. bigFish001)
%S	Size of File (e.g. 987K)
%B	Bits Per Channel
%C	File Creation Date
%H	Image Height (pixels)
%P	Color Profile(e.g. sRGB IEC61966-2.1)
%R	Resolution (e.g. 300)

%T	File Modification Date
%W	Image Width (pixels)

There is obviously a vast amount of potential metadata available. The three broad categories that CSX can deal with beyond the basics are EXIF, IPTC, and XMP metadata. Those can be specified using a slightly different format specification.

%E{tag}	EXIF field (e.g. %E{GPS Longitude})
%I{tag}	IPTC field (e.g. %I{Author})
%X{ns:tag}	XMP field in namespace <i>ns</i> (e.g. %X{xapMM:DocumentID})
%X{tag}	XMP field (e.g. %X{ICCPProfile})

Because of the wide variety of possible locations in the metadata and name formats, CSX does everything it can to find a requested field. That's why, for instance, the namespace is not strictly required for XMP fields. CSX will search all available namespaces for the desired field if one is not specified.

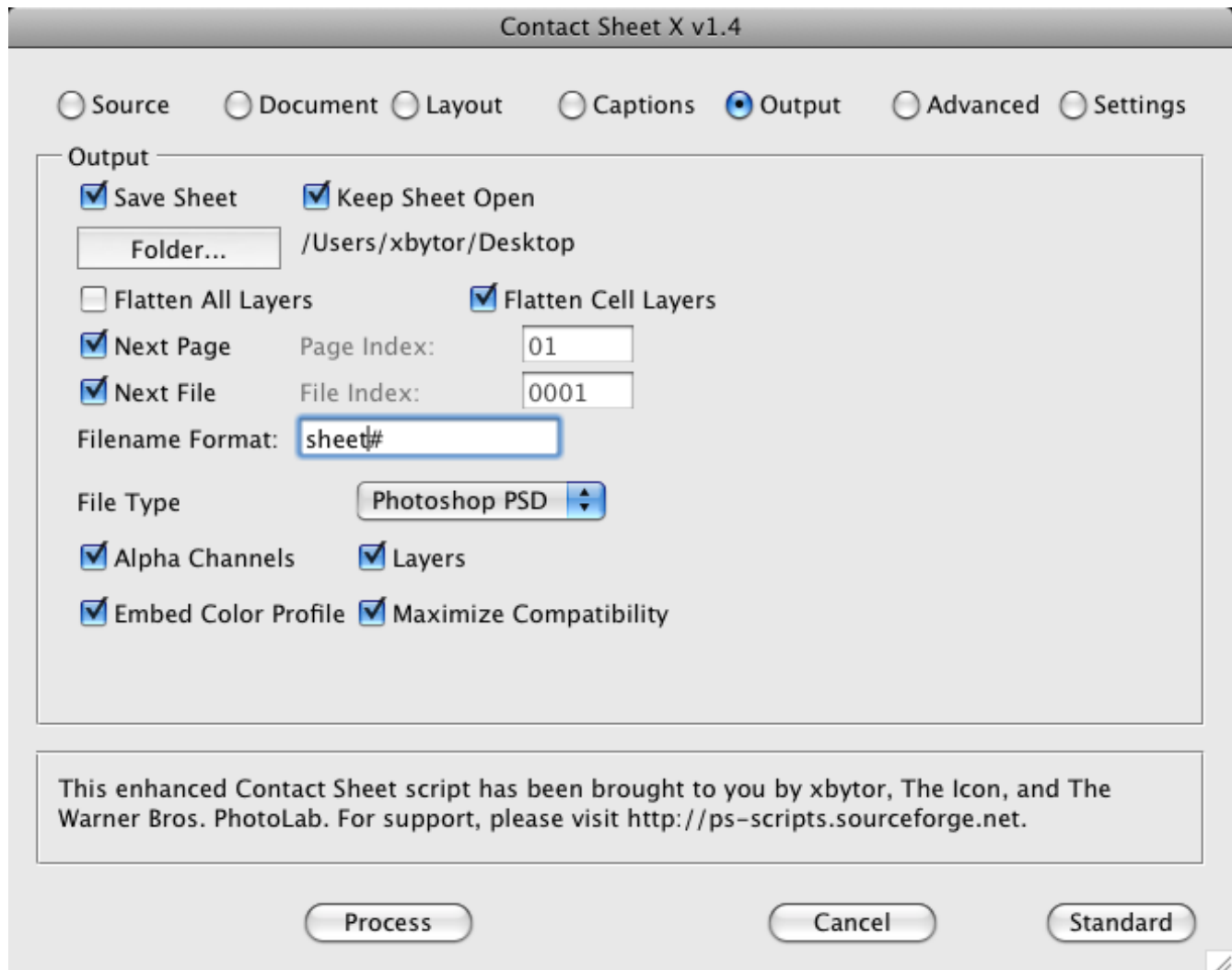
Please refer to ***Appendix B: Caption Format Specification*** for a complete reference on Caption Formats.

### Notes:

- Future releases of CSX may include the ability to specify Layer Styles from a predefined set of styles not currently loaded in Photoshop. There will also be a *Preview* capability to show the effect of layer styles on a caption.

## Output

The Output panel lets a user specify how the final sheets should be handled.



- **Save Sheet** This enables and disables the saving of contact sheets.
- **Keep Open** Setting this will keep all of the contact sheets open after they are created.
- **Folder** Opens a folder selector dialog for specification of the output folder.
- **Flatten All Layers** If this setting is selected, all layers in the document are flattened before saving a sheet. This is not available for all file types.
- **Flatten Cell Layers** If this setting is selected, all of the cell layers (images and captions)

are flattened before saving a sheet. This is useful if the sheets are going to be processed further.

- **Next Page** Makes the starting page number the same as the next file sequence number. If there are no sheets in the folder from which a 'next' page number can be determined, the sequence starts at "01". Zero-padding is determined by the existing sheets in the folder.
- **Next File** Makes the starting file number the next file sequence according to the sheet name and output folder contents. If the *File Prefix* is `fish` and the output folder contains a file "`fish0056.jpg`", the starting File Index will be "0057". If Next Page is also selected, the starting Page Index will be "57". Zero-padding is determined by the existing sheets in the folder.
- **Page Index** Specifies the starting page number to use when adding page numbers to sheets. If the number has leading 0's, this also specifies the number of digits to use when the page number is inserted in a template. For instance, a value of '003' means that the first page number will be 3 and '003' will be inserted into the page number layer in the template. The default starting page index is the same as the starting file index with padding value of two.
- **File Index** Specifies the starting file number to use when creating filenames for sheets. If the number has leading 0's, this also specifies the number of digits to use when the page number is inserted in a template. For instance, '00004' means that the first file number will be 4 and formatted to a width of five characters. The default file index value is determined by looking for numbered files in the output directory. The file index will be the next one in sequence with the same padding.

*Note: The Next Page and Next File settings are essential for Batch execution of CSX where manually setting the Page and File indexes would prove to be problematic.*

- **Filename Format** Specifies the format string to use in the sheet's file name. The '#' character will be replaced with the current file index. If a '#' is not specified, the file index will be appended to the format string.
- **File Type** One of **BMP**, **GIF**, **JPEG**, **EPS**, **PSD**, **PDF**, **PNG**, **TIFF**, or **Targa**. Specifies the file format (and extension) to use when a sheet is saved.
- **File Format Options** When a file type is specified, a set of options specific to that type

becomes available.

## Notes:

### Filenames

With the current release of CSX, filenames will look like:

**<Folder>/<Filename Format>.<File Type>**

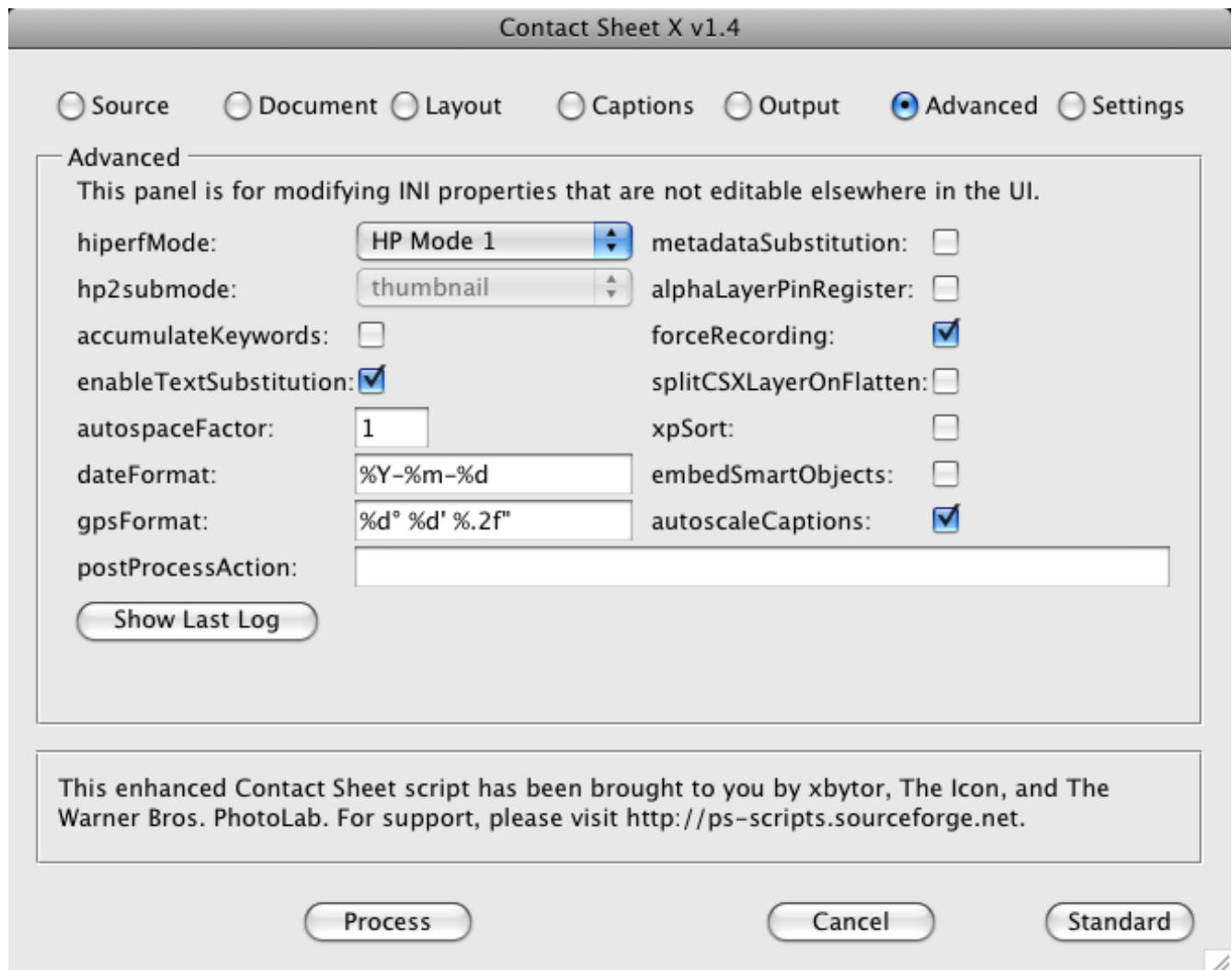
If a # is found in the **Filename Format**, the sequence number for the Contact Sheet is substituted. If one is not found, the sequence number is appended to the **Filename Format**. The length of the sequence number is dependent on the number of digits entered into the **File Index** field (or in the format of existing sheet names, if *Next File* is selected). The value of this field also determines the sequence number used for the first file.

For instance, for the **Output** panel pictured above, the full pathname for the first sheet generated would be C:\work\out\sheet\_0001.jpg while the second would be C:\work\out\sheet\_0002.jpg.

The default **File Index** number is determined when the CSX dialog is first opened. The output folder is examined for files with the same prefix. If one or more is found, the default number is set to one greater than largest sequence value found. The width of that sequence number is also used. This value is also used as the initial **Page Index** value but with a field width of 2. If no file is found with a prefix-sequence number pattern, both of the fields default to a value of 1. Changing the output folder or file prefix will cause the 'next' value to be re-determined.

## Advanced

The Advanced panel contains settings that are rarely needed or are only of limited use.



- **hiperfMode** A selector for which performance mode to use. See below for a detailed discussion. This should normally be set to *HP Mode 1*.
- **hp2submode** A selector for which submode of High Performance Mode 2 to use. See below for a detailed discussion.
- **accumulateKeywords** Setting this will cause all of the keywords of the thumbnails in a sheet to be added to the contact sheet itself.
- **enableTextSubstitution** This enables or disables all layer text substitution for all layers



except *@Date*, *@Page Number*, and *@Metadata* layers.

- **autospaceFactor** Autospacing works by spacing images using a value that is some percentage of the width of the contact sheet. This value should be a percentage value between 0 and 50.
- **dateFormat** This is the default date format to use in *@Date* and *@Metadata* layers and date metadata fields in captions.
- **gpsFormat** This is the format to use in GPS metadata fields in captions and *@Metadata* layers.
- **postProcessAction** The name of an action (*ActionSet@Action*) or the full path to a script to be executed after each contact sheet has been fully populated. This occurs after all thumbnails and captions have been created, after layer text substitution has occurred in templates, but *before* any flattening of the sheet.
- **metadataSubstitution** This enables text substitution for *@Metadata* layers in templates. Setting this to true may cause CSX to open the first image on each sheet so that its metadata can be collected.
- **alphaLayerPinRegister** Performs pin register alignment of images that may be partially transparent.
- **forceRecording** Forces recording of CSX execution as part of an action. Having this set when CSX is not being recorded has no effect.
- **splitCSXLayerOnFlatten** Keeps thumbnail and caption layers separate when layers are flattened. This is particularly useful in situations where a post-processing action or script is run that needs to make changes to the sheet before it is saved.
- **xpSort** Sorts thumbnails based on XP numeric sorting rules instead of simple alphanumeric rules.
- **embedSmartObjects** This setting controls whether or not the thumbnails are kept as Smart Objects or if they are rasterized. This should normally be turned off.
- **autoscaleCaptions** When this is selected, caption text will automatically be scaled

if it does not fit into the space allocated for the thumbnail.

- **Show Last Log....** This button will open the log file from the previous execution of CSX. The information in this log file can be helpful in isolating problems.

## Notes on Performance Modes

CSX currently supports several performance modes, some of which have submodes.

Basic Mode uses typical PS techniques for opening, resizing, rotating and placing images into a contact sheet. While very reliable it is the least efficient technique available.

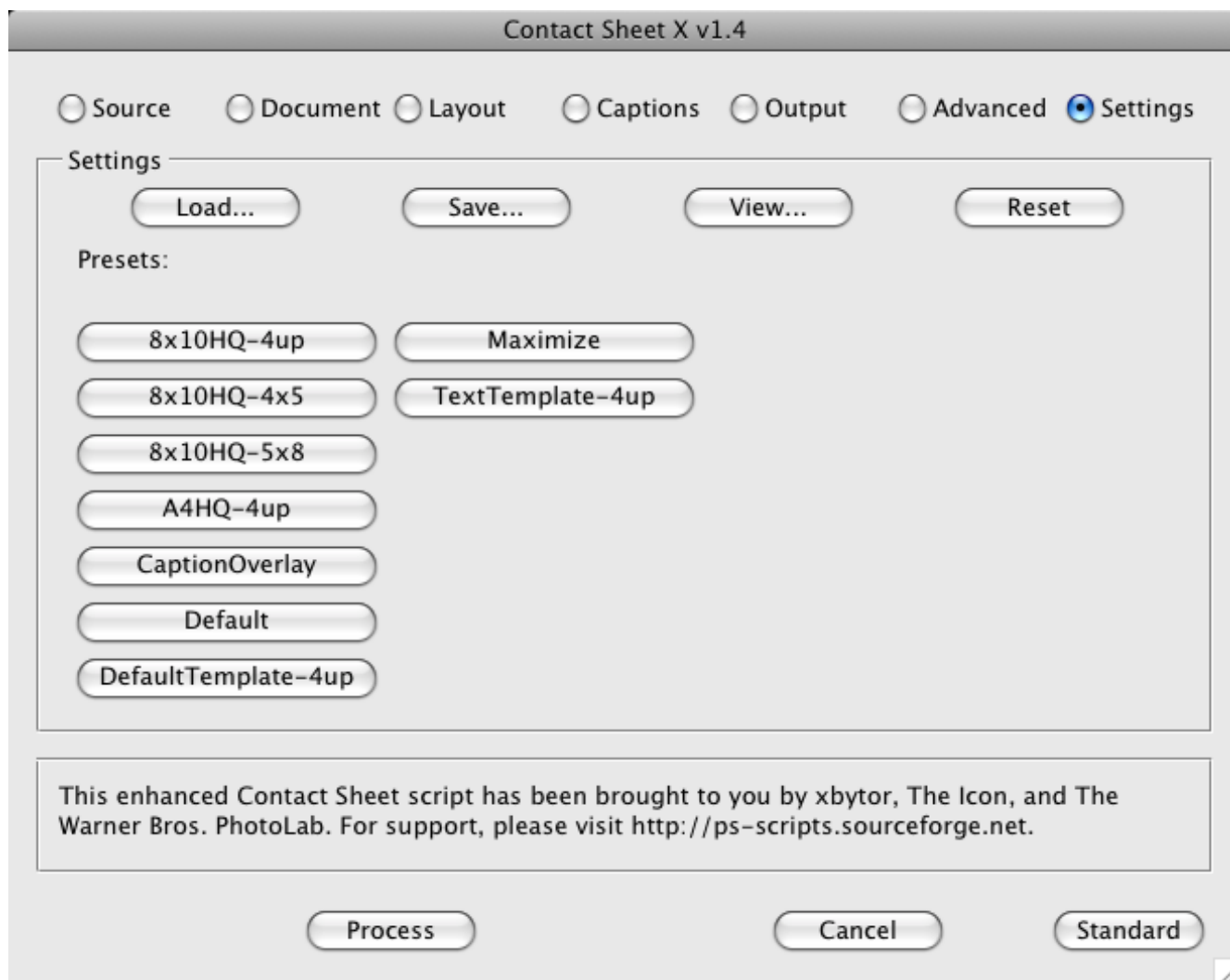
High Performance Mode 1 uses the Photoshop *Place* command to insert an image into a contact sheet. From here, the layer is rotated, resized, and repositioned to the appropriate location on the sheet. It is typically 30-45% faster than Basic Mode. One problem that it has is that metadata from the image is cannot retrieved unless Bridge is also running.

There is another PDF file that is a part of the CSX package called *HighPerformanceModes.pdf*. It goes into more detail on the other HP modes as well as what the installation and configuration requirement modes are.

Unless you clearly understand the implications of using the different HP modes, use *HP Mode 1* or, if absolutely necessary, *Basic* mode.

## Settings

The Settings panel allows a user to load and save settings independent of the application's INI file. There is also a list of built-in alternate settings that can be used.



- **Load** Load CSX settings from an INI file.
- **Save** Save the current CSX settings to an INI file.
- **View** View the current settings in a pop-up window.
- **Restore** Restore the CSX settings to their initial values.
- **Presets** These buttons load their corresponding presets file. The presets are determined when the CSX launches by looking at all INI files in the

CSX/templates folder. Loading a preset is like loading an INI file except that pathnames are modified to point to appropriate places on the current machine and presets typically do not specify values for all available settings. This simplifies the sharing of presets and their associated templates. Presets may also refer to templates.

**Notes:**

- Presets, in their fullest form, are an interesting idea that never quite got developed. Saving and Loading settings/ini files should be enough for almost all situations.

## Appendix A: Date Format Specification

Date formats can be specified using one of two styles.

The first style of format specification has these mappings:

<i>YYYY</i>	Four digit year. <i>2006</i>
<i>YY</i>	Two digit year. <i>06</i>
<i>MM</i>	Two digit month. <i>10</i>
<i>DD</i>	Two digit month. <i>31</i>
<i>H</i>	Two digit hour on a 24 hour clock. <i>20</i>
<i>I</i>	Two digit hour on a 12 hour clock. <i>08</i>
<i>M</i>	Two digit minute. <i>43</i>
<i>S</i>	Two digit seconds. <i>02</i>
<i>P</i>	AM or PM. <i>PM</i>

Using these formats, you could create a text layer with the name *@Date*, set its contents to "*Created on YYYY/MM/DD*", and end up with a layer that looks like "Created on 2006/10/31".

The second style of format specification is based on the Unix *strftime(3)* implementation. The format specification is text that includes characters that will be substituted according to the following rules. Characters in a format specification that are not matched are left unchanged. CSX determines which specification to use by looking for a '%' character in the format specification. If the format does **not** contain a '%', the Y/M/D-style substitution is used. If it does, this style is used.

A date/time of October 31, 2006 20:43:02 will be used for the example below.

<b>Specifier</b>	<b>Substitution</b>
%a	A three-letter abbreviation for the day of the week, one of 'Sun', 'Mon', 'Tue', 'Wed', 'Thu', 'Fri', or 'Sat'. <i>Tue</i>
%A	The full name for the day of the week, one of 'Sunday', 'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', or 'Saturday'. <i>Tuesday</i>
%b	A three-letter abbreviation for the name of the month, one of 'Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', or 'Dec'. <i>Oct</i>
%B	The full name of the month. One of 'January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', or 'December'. <i>October</i>
%c	The complete date and time in this format:

"Tue Oct 31 20:43:02 2006".

%C	The century portion of the year, zero padded. <i>20</i>
%d	The day of the month, zero padded. <i>31</i>
%D	The date in the format "%m/%d/%y". <i>10/31/06</i>
%e	The day of the month, space padded. <i>31</i>
%F	The date in the format "%Y-%m-%d". <i>2006-10-31</i>
%h	A three-letter abbreviation for the name of the month, same as %b. <i>Oct</i>
%H	The hour on a 24 hour clock, zero padded. <i>20</i>
%I	The hour on a 12 hour clock, zero padded. <i>08</i>
%j	The Julian date. <i>304</i>
%k	The hour on a 24 hour clock, space padded. <i>20</i>
%l	The hour on a 12 hour clock, space padded. <i>8</i>
%m	The month number, zero padded. <i>10</i>
%M	The minutes, zero padded. <i>43</i>
%n	A newline character.
%p	Either AM or PM as appropriate. <i>PM</i>
%r	12 hour time to the second with this format: "%I:%M:%S %p". <i>08:43:02 PM</i>
%S	The seconds, zero padded. <i>02</i>
%t	A tab character.
%T	24 hour time to the seconds with this format: "%H:%M:%S". <i>20:43:02</i>
%u	The weekday as a number where Monday is 1 and Sunday is 7. <i>2</i>

%w	The weekday as a number where Sunday is 0 and Saturday is 6. 2
%x	The date in the format "%m/%d/%y". Same as %D.
%X	24 hour time to the seconds with this format: "%H:%M:%S". Same as %T.
%y	The last two digits of the year. 06
%Y	The full four digits of the year. 2006
%%	The '%' character.

Using these formats, you could create a text layer with the name *@Date*, set its contents to "*Created on %Y/%m/%d*", and end up with a layer that looks like "Created on 2006/10/31".

## Appendix B: Caption and @Metadata Layer Format Specification

### Captions

Captions on contact sheets can be specified two different ways. First, there is a **Use Filename** checkbox on the *Captions* panel which will, not surprisingly, create a caption using an image's filename. The **No Extensions** checkbox can be used to remove the filename extensions.

For anything more complicated than a filename caption, a **Caption Format** must be specified.

### @Metadata Layers

Layers in templates that begin with *@Metadata* provide the ability to insert metadata into text layers in a contact sheet, much like the *@Date* and *@Page Number* layers allow the insertion of the current data and page number information. The metadata that is inserted into the document is based on the first document of each individual sheet.

The format specification to be used for the substitution can be specified in one of two ways. The first way is to specify it as a part of the layer name. For example, a layer name of

`@Metadata=%p`

would place the name of the parent folder of the image in the text layer.

The second way is to simply use *@Metadata* as the name of the text layer and to put the format specification as the content of that text layer.

### Format Specifications

A format string has text in it along with 'special' characters (aka *patterns*) that are substituted with additional information about an image, such as EXIF metadata. These substitution patterns always begin with a % character which is followed by one or more characters. For instance, *%W* and *%H* are the pattern specifiers for the width and height of the image, respectively. A caption format of "*%Wx%H*" would result in a caption that looked something like "*640x480*".

This table lists all of the caption format specifiers and what their substitutions are.

Specifier	Substitution
%%	%
%d	Directory (full path)
%e	Extension
%f	Filename without Extension
%p	Parent Directory (name only)
%s	Size of File



%S	Size of File using K/M/G shorthand
%B	Bits Per Channel
%C	File Creation Date
%C{dateFormat}	File Creation Date using specified format
%E{tag}	EXIF field
%F	Filename (same as %f.%e)
%H	Image Height (pixels)
%I{tag}	IPTC file (case insensitive)
%M	Color Mode
%N	Current Date
%N{dateFormat}	Current Date using specified format
%P	Color Profile
%R	Resolution (ppi)
%T	File Modification Date
%T{dateFormat}	File Modification Date using specified format
%W	Image Width (pixels)
%X{space:tag}	XMP Field from specified space
%X{tag}	XMP Field (from first space with tag)

These examples illustrate how basic substitutions occur:

<i>Format</i>	<i>Output</i>
%Wx%H %Rppi	640x480 72ppi
%f.%e	bigFish.jpg
%P %M	sRGB IEC61966-2.1 Lab

### Named Specifiers

In addition to the basic specifiers, there are the *named* specifiers. Because of the very large numbers of possible metadata fields, it is impractical to use simple one letter specifiers for them all. Because of this, the names of EXIF, IPTC, and XMP metadata fields can be used in format specifiers. The general form for a EXIF specifier would be %E{tag} where the %E indicates that this is an EXIF specifier and tag is the name of the EXIF field that we want. For instance %E{Model} specifies that we want the Model EXIF field, which would be substituted by something like *NIKON D200*.

EXIF metadata is image file information typically created by a digital camera (or some other image file creator). The information available varies depending on the creator of the image; one vendor may provide basic fields like *aperture* and *ISO speed* information while another may provide those fields and *GPS* data.

Here is a partial list of EXIF fields that might be found in an image

*Aperture Value, Artist, Color Space, Date Time, Date Time Original, Exposure Bias Value, Exposure Mode, Exposure Program, Exposure Time, F-Stop, Flash, Focal Length, ISO Speed, Ratings, Make, Max Aperture Value, Metering Mode, Model, Orientation, Software, Shutter Speed, White Balance.*

These examples illustrate how EXIF substitutions occur:

<i>Format</i>	<i>Output</i>
%E{F-Stop} %E{Exposure Time}	f/5.0 2.5 sec
ISO%E{ISO Speed Ratings}	ISO320

IPTC metadata is image file information that was originally developed by news organizations and, later, by Adobe. This specifies information like *Title*, *Author*, and *Keywords*.

Here is a partial list of IPTC fields that may be found in an image.

*Author*, *AuthorPosition*, *Caption*, *CaptionWriter*, *Category*, *City*, *Copyrighted*, *CopyrightNotice*, *Country*, *CreationDate*, *Credit*, *Headline*, *Instructions*, *JobName*, *Keywords*, *OwnerURL*, *ProvinceState*, *Source*, *SupplementalCategories*, *Title*, *TransmissionReference*, *Urgency*.

These examples illustrate how IPTC substitutions occur:

<i>Format</i>	<i>Output</i>
%I{Keywords}	Animals, Kangaroo
%I{ProvinceState}, %I{Country}	Queensland, Australia

The remaining category of *named* fields is XMP. XMP is an open-ended XML-based format in which all sorts of information may be stored. XMP metadata is subdivided into namespaces which group fields into collections. There is, for instance, a namespace *xapRights* the specifies rights management (copyright) information for an image. You typically specify the namespace and the field/property within that name space that you want. For instance *%X{xapRights:Copyright}* would be substituted with the *Copyright* property from the *xapRights* namespace and would look something like ©2008 *xbytor*.

These examples illustrate how IPTC substitutions occur:

<i>Format</i>	<i>Output</i>
%X{dc:format}	image/tiff
%X{DocumentID}	uuid:76AC19F1C0EFDB11A4298F44C792E00A

You can, of course, mix the different kinds of format specifiers like this:

<i>Format</i>	<i>Output</i>
%X{format} %S - "%I{Title}"	image/tiff 987K - "Mardi Gras 2007"

### Date Formats

There are three basic specifiers that map to dates (*%C*, *%N*, *%T*). When the substitution occurs for these specifiers, the dates have to be formatted somehow. By default, in CSX, the dates are formatted *YYYY-MM-DD*. *%C* would be substituted by something that looks like *2007-03-21*. The default date format for CSX is specified on the *Advanced* panel in the **dateFormat** setting. Changing this effects the formatting of dates throughout CSX. You can, however, override the default date format by specifying the desired format as part of the specifier. For instance, you could specify a file creation date like this *%C{%m/%d/%Y}* or (equivalently) like this *%C{MM/DD/YYYY}*. See the **Date Format Specification** appendix for details on date formats.

There are also EXIF, IPTC, and XMP date fields that we have to worry about. As a general rule if a *named specifier* contains 'date' anywhere in its name, CSX will attempt to format it using the default date format.

### GPS Formats

CSX provides a mechanism for specifying the format of GPS Coordinates. On the *Advanced* panel there is a **gpsFormat** setting. Much like the **dateFormat** setting, it allows a default format for GPS data to be specified. Instead of inventing some again, I duplicated the format used by *exiftool* (<http://www.sno.phy.queensu.ca/~phil/exiftool/>).

The format for GPS Coordinates uses a syntax similar to other formats in CSX. The specifiers in the format correspond to degrees, minutes, and seconds in that order with the minutes and seconds being optional. In this format, *%d* means the value should be formatted as an integer while a *%f* means the value should be formatted as a floating point number. For precise details on alignment, padding, etc..., see <http://www.opengroup.org/onlinepubs/007908799/xsh/fprintf.html>.

These examples give the output of the same coordinate using various formats:

<i>Format</i>	<i>Output</i>
<i>%d° %d' %.2f"</i>	54° 59' 22.80" (the default)
<i>%d deg %.4f min</i>	54 deg 59.3800 min
<i>%.6f degrees</i>	54.989667 degrees

GPS formatting only occurs for the EXIF fields *GPS Longitude* and *GPS Latitude*.