community media association

SURREY
COUNTY COUNCIL

Templating Documentation for SOVA:

Simple Online Video Archive

local e-Democracy National Project

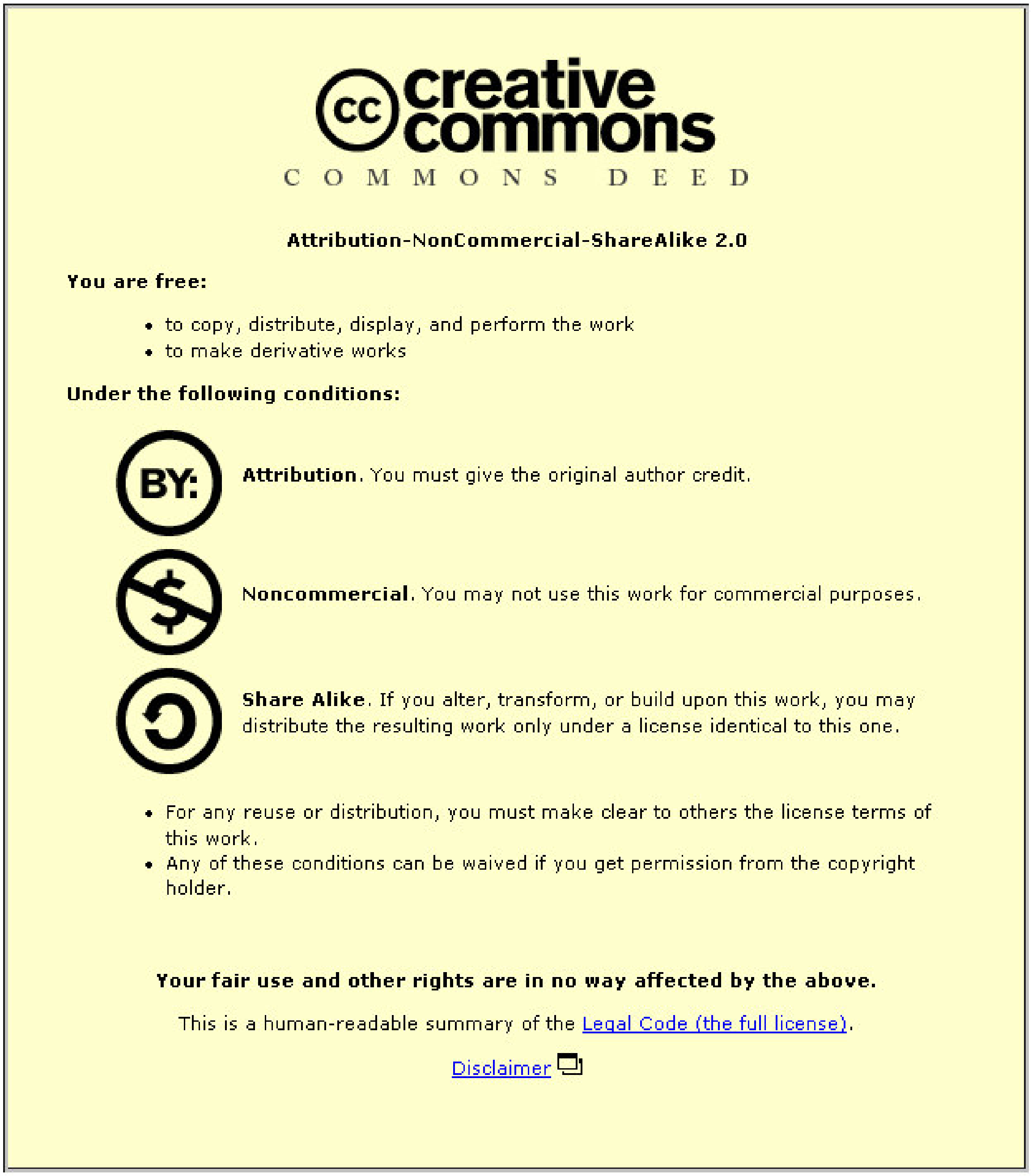
This is a local e-Democracy National Project, part of an £80m ODPM funded programme to deliver e-government solutions.

www.e-democracy.gov.uk

local e-gov National Projects

community media association

SURREY
COUNTY COUNCIL



local e-Democracy National Project

This is a local e-Democracy National Project, part of an £80m ODPM funded programme to deliver e-government solutions.

www.e-democracy.gov.uk

local e-gov National Projects

Contents

1. Introduction

The following documentation has been based on information taken from the website relating to the content management system used for this project: “eZ publish” (http://doc.ez.no). Additional information has been added by the designer/developer at the Community Media Association (http://www.commedia.org.uk) who, along with Surrey County Council (http://www.surreycc.gov.uk/), designed and built the interface for SOVA – Simple Online Video Archive (http://sourceforge.net/projects/sova-dev/). The SOVA platform has first been utilised in Surrey County Council’s Visible Voices project (http://www.urcountyursay.tv/).

The design documentation found at the “eZ publish” website is mostly aimed at developers who have a grasp of PHP (http://www.php.net/) and is either an overview of the terms and concepts used or is almost entirely focused on one section of the product, the ‘eZarticle’ module – the basic building block of “eZ publish”.

Whilst the official “eZ publish” (abbreviated to eZ) documentation can help a designer or developer to construct a site using eZ they obviously do not include instructions for extending its functionality using the special module that has been designed for handling video content. This custom-built module, **ezmediamanage** was originally created for the Community Media Association by developers working on The Showcase – an online archive of community media (http://showcase.commedia.org.uk). You will see later that for The Showcase and SOVA, some adaptation of eZ was required to allow **ezmediamanage** to handle the relation between the uploaded video content, the Dublin Core metadata that it contains and the eZarticle module that holds a basic description of each piece of work.

This documentation starts by explaining the basic concepts involved in creating a web interface with eZ. More explanation is given regarding the video module and the notes finish with an extensive list of all the possible variables that can be used within the ezarticle module. As you can see by visiting the eZ site, no more extensive documentation was written for the other modules but using eZarticle as a starting point it should be possible to apply the logic there to any other module in the site.

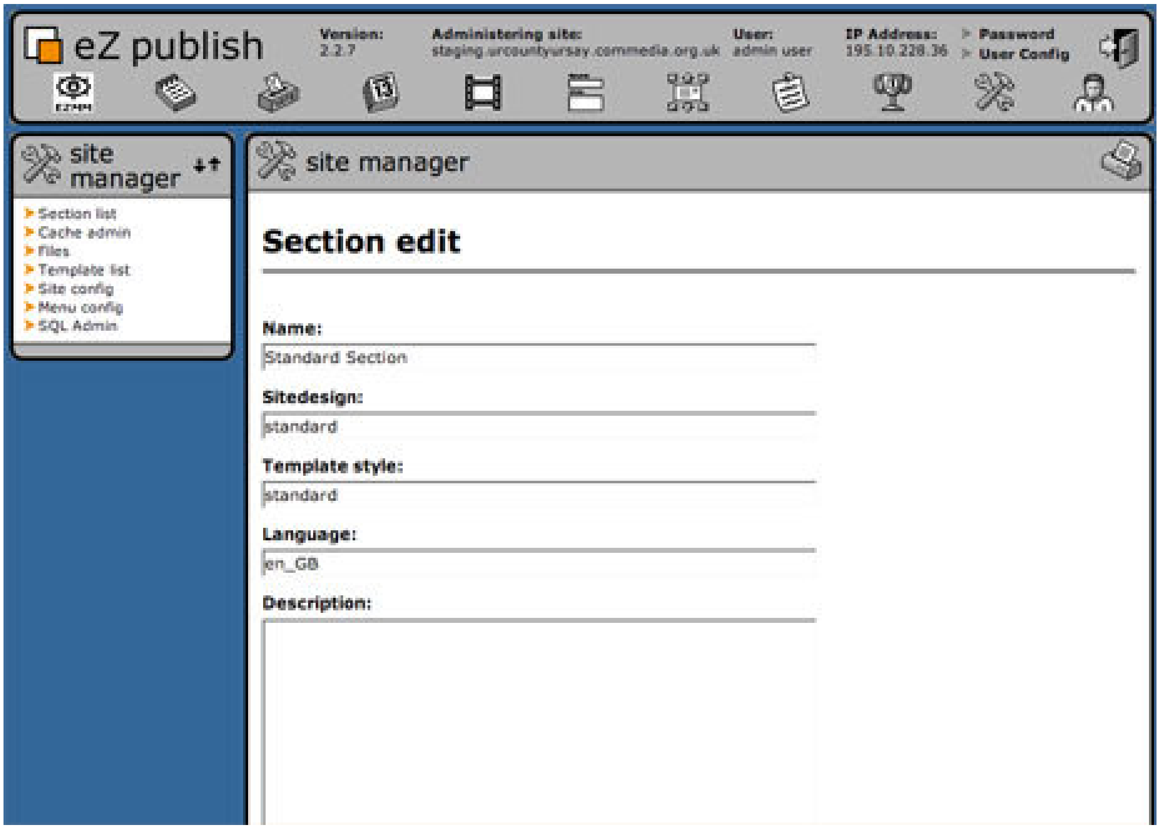
2. Basic Terms and Concepts

In order to understand how “eZ publish” creates pages it is important to have a clear idea of how eZ works and where the relevant files are located in the eZ directory structure.

Content in “eZ publish” is defined by, and contained in a number of separate **modules** that represent different types of site functionality such as news (eZarticle), pictures (eZimagecatalogue) or a calendar (eZcalendar). The output of these modules can be organised through the creation of **sections.** Content and design are separated through the use of a **templating system** comprising a **sitedesign**. Below is a list of definitions of these elements.

Sections

eZ uses a system of **sections** to differentiate potentially separate areas of the site that may have different content and design. For instance a site may have a shop, a news section and a forum, which could be differentiated by having different graphics, background colours or stylesheets in the page. These separate sorts of site content would be given their own sections in the admin backend. But sections don’t necessarily have to be for such disparate sorts of content, you may not have a shop or a forum but just news on a site which is sorted into different subjects which you might to display using slightly different ‘skins’.



Version: 2.2.7
Administering site: staging.urcountyursay.commedia.org.uk
User: admin user
IP Address: 195.10.228.36

Section edit

Name: Standard Section

Sitedesign: standard

Template style: standard

Language: en_GB

Description:

As you can see from the screenshot above whilst editing or creating a Section, information can be entered linking it to a specific sitedesign and template style.

Content

Content is the information that the content providers submit to the site. In other words it is everything written into the HTML forms that is the graphical user interface (GUI) of “eZ publish”.

Everything that is written on an “eZ publish” site (except for a few items sent by e-mail) is saved into a MySQL database.

Module

A module is a collection of functions, objects and pages relating to a service in “eZ publish”. For example, “eZ article” is a module that takes care of news publishing and related areas and “eZ trade” is the e-commerce module.

In the “eZ publish” home directory you will find that modules are subdirectories. Each subdirectory is named after the module except without spaces and all in lower case. “eZ article” is thus found in the directory "ezarticle".

The sitedesign

Sitedesigns are the main page designs. They are created using the **frame.php** documents found in the sub-folders of the **sitedesign** directory at the root of eZ. Looking at the code in a **frame.php** doc you will see that sitedesigns are based on a standard HTML file. They have doctype definitions and metatags in the headers but also have elements of PHP in them that call the eZ functions and pull in the content when the page is built.

To create a new directory of **sitedesign** files, copy one of the existing directories and its contents usually **standard** - and give the directory a new name. The **frame.php** and any graphics can then be amended to the designer’s requirements.

Templates

These are small snippets of HTML that nest within the **sitedesign** format the different module content such as articles or lists of articles. They reside in the module directories in the eZ root. For example, the directory for **ezarticle**, probably the most widely used module, contains a sub-directory called ‘user’. Within ‘user’ is the ‘templates’ folder and within this templates are held in further sub-directories that correspond to the section and **sitedesign** being used. Template files are recognisable as they are appended by .tpl

Although .tpl files cannot contain true PHP code they can contain certain **variables** that allow eZ to find the correct path to objects such as graphics and other files.

A list of these variables is given at the end of this section.

Template Set

A template set is a collection of templates. These are all stored in directories with the name of the set. In the basic “eZ publish” distribution there are only one set: standard. For clarity, ease of use and comprehension, SOVA has also used only the standard template set.

The sets are spread out between all modules in “eZ publish”. Each module has at least a subdirectory called "admin" and most times a directory called "user". In addition there are other directories that necessarily concern the designer.

Everything in the "admin" directory takes place in the administration part of “eZ publish” - the one which editors, content providers and administrators have access to.

Everything in the "user" directory takes place in the user part of “eZ publish” - the public site that visitors to the site see. This is where designers must concentrate their efforts.

Below the "user" directory you will find at least two subdirectories "intl" and "templates". Under templates you will find all the template sets that have defined templates for the current module. Under "intl" you will find one directory for each language that the module has been translated into.

Template Variables

A template variable is a string enclosed by the characters "{" and "}". These are replaced by the PHP-script(s) that use the template in question. Variables that are not replaced remain in plain text in the file.

The variables are divided into two groups: the content variables and the internationalization variables.

Content Variables

The content variables are all the variables that are not precluded by the string "intl-".

Internationalisation Variables

The string “intl-” prefixes the internationalisation variables. These variables are replaced by the appropriate translations of the strings they represent.

{intl-ok} - this is an internationalization variable
{ok} - this is a content variable

URL’s

The URL is the address of a page. When you start up your default “eZ publish” site you will see that the URL’s appear to have a common structure. For example you might write the following into the location bar of your browser "http://www.mysite.com/" to get the front page of “eZ publish”. If you attach the following "article/articleview/1/" you will get an article. If you change the last number to "2" you will see another article (if these articles have already been created in the admin interface). For the purposes of SOVA, article URL’s usually point toward a media file since the description fields of all uploaded media are held in eZ’s article format.

The Module

The first part "article" is the name of the module you are using, minus the "eZ" prefix. So to access the “eZ article” module you use "article", the “eZ trade” module "trade", the “eZ link” module "link", and so on.

The Function

The second part "articleview" refers to a function in the module. Since we are talking about the public site the function will be found in the "user" directory under the module.

The Object ID

The number at last in this example is the object ID for the object of this type. Since we are using "articleview" the object is an article. Note that some of the more complex functions might have subfunctions which will appear before the object ID and you will often find that there are even more numbers that follow the object ID.

Fiddling around with every number you see in your URLs will often give you invaluable insights but much of this is also covered in the design manuals for each module (where available).

3. How it fits together

We can now take a look at how eZ uses the elements described above to create the site pages using the examples created for SOVA.

If you look at the code inside a frame.php document, for instance, from the ‘standard’ sitedesign directory you will see a piece of PHP code:

```
<? print ( $main contents ); ?>
```

This tells eZ that it should deliver the **contents** for that particular page at that point in the document. eZ knows which content to add here based on what is in the URL. For instance if the URL were:

http://mysite.com/frontpage/1

eZ would know that you wanted to display the contents for the front page of section 1. eZ looks for the sitedesign that was assigned to section 1 in the admin backend and collects the frame.php from that sub-directory of the sitedesign . Then, since we are building the front page and the default module is eZarticle, it looks in ezarticle/users/templates/standard folder to collect the frontpage.tpl file. Whatever code is contained in frontpage.tpl will be output.

In the case of SOVA something other than just:

```
<? print ( $main contents ); ?>
```

needs to be added to the frame.php in order to decide whether the content will simply be drawn from the ‘ezarticle’ module or also from the ‘ezmediamanage’ – the module created specifically to allow for the upload and viewing of video content.

If you look at the **frame.php** for the standard sitedesign in SOVA you will see the following conditional statement.

```
<? print( $MainContents ); ?>

<?
    if ( $url_array[2] == "frontpage" )
    {

                    include ( "sitedesign/standard/smallarticles.dat" );

     }

    elseif ( $url_array[2] == "view" || $url_array[2] == "articleview" )

          {

       $ArticleID = "$url_array[3]";
       include( "ezmediamanage/user/showdc.php" );
       unset( $ArticleID );
    };
?>
```

This code checks to see what is in the URL and either adds the **ezmediamanage** code - if the /view or articleview/ elements are present - or omits it if it is not.

For example, if the URL is:

http://mysite.com/article/view/3

then eZ takes article number 3 and using the ‘articleview’ template formats the description field. The code also tells eZ to include the content parsed by **showdc.php**. This is the file that outputs the Dublin Core metadata. **showdc.php** uses the template **showdc.tpl** to format this information into a table. This file resides in /ezmediamanage/user/templates/standard (or whichever user template directory you have created).

If however the URL is of the nature of

http://mysite.com/article/archive/5

or

http://mysite.com/article/topiclist/6

then the code tells eZ to ignore showdc.php and merely display the page using whichever templates are required (in this case ‘articlelist’ and ‘topiclist’).

Appendix A: sitedesign calls and template variables for the module ‘ezarticle’

A list of codes to call templates and how to insert them is given below. These examples are taken from the eZ documentation at http://doc.ez.no/

articleheaderlist.php

This page is activated through the URL: /article/articleheaderlist/1/

This will result in a list of all articles in a category with the number at the end of the URL as the ID. If the ID is zero, all articles in all categories are shown.

Template Variables

The variables provided are:

- **section_id**
  Variable that holds the global section id, i.e. the current section.
- **category_id**
  Variable that holds the listed category's name
- **category_name**
  Variable that holds the listed category's id
- **category_description**
  Variable that holds the listed category's description
- **article_id**
  Variable that holds the article id
- **article_name**
  Variable that holds the article name
- **article_link_text**
  Variable that holds the article link text, if there isn't a text defined for this article, the value is fetched from the variable "DefaultLinkText" in the section "eZArticleMain" of the "site.ini" file.
- **article_published**
  Variable that holds the article publication date
- **category_id**
  Variable that holds the article's main category id
- **article_category_name**
  Variable that holds the article's main category name

articlelinks.php

This file is included like this:

```
<?
   $CategoryID = 2;
   include( "ezarticle/user/articlelinks.php" );
?>
```

This will result in a list of all articles in a category with the supplied ID. An ID to a non-existing category, or to an empty category will not show anything.

Template Variables

The variables provided are:

- **sitedesign**
  Variable that holds the name of the current sitedesign, this is needed for proper inclusion of site design specific information.
- **www_dir**
  Variable that holds the root directory for non-virtualhosts, empty on virtual host setups.
- **index**
  Variable that holds the name of the index.php file for fallback when needed, empty on virtual host setups.
- **category_id**
  Variable that holds the listed category's id
- **current_category_name**
  Variable that holds the listed category's name
- **current_category_description**
  Variable that holds the listed category's description
- **article_id**
  Variable that holds the article id
- **article_name**
  Variable that holds the article name
- **article_link_text**
  Variable that holds the article link text, if there isn't a text defined for this article, the value is fetched from the variable "DefaultLinkText" in the section "eZArticleMain" of the "site.ini" file.
- **article_category_id**
  Variable that holds the article main category name

articlelist.php

This page is activated through the URL: /article/archive/1/2/

- The first number is the category id, $CategoryID.
- The second number is the number of articles from the start to skip, the $Limit, this is used on consecutive pages when the list is longer than one page.

The limit, the number of articles to show, is regulated by the number set for the category, if missing the default set in "site.ini", the "eZArticleMain" section and in the "UserListLimit" variable is used.

Override Template

Before rendering this page, “eZ publish” will check if there is a file in the current template directory which is named "articlelist_override_$CategoryID.tpl" where $CategoryID is replaced by the category id from the URL.

If such a page is found, the template used will be that specific file, in all other cases the section override will be searched for.

The use of an override might be appropriate when you wish to list articles/categories without certain elements, or in cases where you want certain elements to never change. An example use might be for adding extra information, images to the listing.

Section Override Template

Before rendering this page, “eZ publish” will check if there is a file in the current template directory which is named "articlelist_sectionoverride_$GlobabalSectionID.tpl". Where $GlobalSectionID is replaced by the section id fetched from the category.

If such a page is found, the template used will be that specific file, in all other cases the template "articlelist.tpl" will be used.

The use of a section override will do the same as the override, but for all listings in that section.

The Standard Variables

- **section_id**
  Variable that holds the global section id, i.e. the current section.
- **www_dir**
  Variable that holds the root directory for non-virtualhosts, empty on virtual host setups.

- **index**
  Variable that holds the name of the index.php file for fallback when needed, empty on virtual host setups.

The "header_item_tpl" Block Variables

- **current_category_name**
  Variable that holds the listed category's name
- **current_category_description**
  Variable that holds the listed category's description
- **current_image_width**
  Variable that holds the listed category's image width
- **current_image_height**
  Variable that holds the listed category's image height
- **current_image_URL**
  Variable that holds the listed category's image URL
- **current_image_caption**
  Variable that holds the listed category's image caption text
- **current_image_description**
  Variable that holds the listed category's image description text
- **current_image_photographer**
  Variable that holds the listed category's image photographer text

The "path_item_tpl" Block Variables

- **category_id**
  Variable that holds the listed category's id
- **category_name**
  Variable that holds the listed category's name

The "category_list_tpl" Block Variables

- **category_id**
  Variable that holds the current sub-category's id
- **category_name**
  Variable that holds the current sub-category's name
- **category_description**
  Variable that holds the current sub-category's description
- **image_width**
  Variable that holds the current sub-category's image width
- **image_height**
  Variable that holds the current sub-category's image height
- **image_URL**
  Variable that holds the current sub-category's image URL
- **image_caption**
  Variable that holds the current sub-category's image caption text

The "article_list_tpl" Block Variables

- **category_current_id**
  Variable that holds the listed category's id
- **category_id**
  Variable that holds the listed article’s main category's id if "ForceCategoryDefinition" equals "enabled" in "eZArticleMain" in the site.ini, or if the $CategoryID is zero, else it uses the $CategoryID
- **category_def_id**
  Variable that holds the listed article’s main category's id
- **category_def_name**
  Variable that holds the listed article’s main category's name
- **article_id**
  Variable that holds the current article id
- **article_name**
  Variable that holds the current article name
- **article_link_text**
  Variable that holds the current article link text, if there isn't a text defined for this article, the value is fetched from the variable "DefaultLinkText" in the section "eZArticleMain" of the "site.ini" file.
- **article_datevalue**
  Variable that holds the current article's publishing date
- **article_timevalue**
  Variable that holds the current article's publishing time
- **article_published**
  Variable that holds the current article's publishing date and time
- **article_intro**
  Variable that holds the current article's intro text (teaser)
- **author_text**
  Variable that holds the current article's author name
- **topic_id**
  If a topic is defined for this article ,this is the topic id
- **topic_name**
  If a topic is defined for this article ,this is the topic id
- **thumbnail_image_width**
  Variable that holds the current article’s thumbnail image width
- **thumbnail_image_height**
  Variable that holds the current article’s thumbnail image height
- **thumbnail_image_uri**
  Variable that holds the current article’s thumbnail image URL.
- **thumbnail_image_caption**
  Variable that holds the current article’s thumbnail image caption text

articlelistrss.php

This page is activated through the URL: /article/rssheadlines/

The resulting page is an RSS (RDF Site Summary) file that is used for syndication by many different sources. The page is an XML format. This is a set format - no template is available for this function.

articleview.php

This page is activated through the URL:

/article/view/1/2/3/
/article/articleview/1/2/3/
/article/print/1/2/3/
/article/articleprint/1/2/3/
/article/static/1/2/3/
/article/articlestatic/1/2/3/
/article/articleuncached/1/2/3/

- The first digit is the article id, $ArticleID.
- The second digit is the page number, $PageNumber, in the article.
- The third digit is the category id, $CategoryID, which view are viewing this page through. The referred to as the current category.

The design selected in the category will be used when viewing the page. If the third digit is omitted, the main definition category is assumed. If the second digit is omitted, the first page is assumed.

Static Pages

The use of override template and section override template is not possible for static pages (called as articlestatic or static in the URL). Instead static pages use their own template called "articlestatic.tpl". You can use the same method as in the override template on the static template.

The default template for static pages removes author info, publication date and other information that is most appropriate for articles. Static pages are mostly used for pages that contain meta-information about the site, e.g. "terms and conditions", "submission guidelines".

Override Template

Before rendering this page, “eZ publish” will check if there is a file in the current template directory which is named "articleview_override_$CategoryID.tpl" where $CategoryID is replaced by the category id from the URL.

If such a page is found, the template used will be that specific file, in all other cases the section override will be searched for.

This function is useful if you have an article that you want to render very differently from others, or in cases where you do not want to convert an HTML page into eZ article format.

Section Override Template

Before rendering this page, “eZ publish” will check if there is a file in the current template directory which is named "articleview_sectionoverride_$GlobabalSectionID.tpl" where $GlobalSectionID is replaced by the section id fetched from the category.

If such a page is found, the template used will be that specific file, in all other cases the template "articlelist.tpl" will be used.

The usage for this function might be cases where one category of articles should be rendered differently from other categories, and you don't want to create a complete template set for this. Example uses would be extra decoration for articles in a certain category, or perhaps special logos/links added to articles in certain categories.

The Standard Variables

- **section_id**
  Variable that holds the global section id, i.e. the current section.
- **www_dir**
  Variable that holds the root directory for non-virtualhosts, empty on virtual host setups.
- **index**
  Variable that holds the name of the index.php file for fallback when needed, empty on virtual host setups.

The "article_URL_item_tpl" Block Variables

This block is only shown when viewing the printer friendly version of the page.

- **article_URL**
  This is the URL to the page as typed into the browser.
- **article_id**
  Variable that holds the current article id
- **article_name**
  This is the name of the article
- **article_intro**
  Variable that holds the current article's intro text (teaser)
- **article_body**
  Variable that holds the current article's body text
- **link_text**
  Variable that holds the current article link text, if there isn't a text defined for this article, the value is fetched from the variable "DefaultLinkText" in the section "eZArticleMain" of the "site.ini" file.

- **article_datevalue**
  Variable that holds the current article's publishing date
- **article_timevalue**
  Variable that holds the current article's publishing time
- **article_created**
  Variable that holds the current article's publishing date and time
- **author_email**
  This is the e-mail to the writer of the article
- **author_text**
  This is the name of the writer of the article
- **author_id**
  This is the ID of the writer of the article
- **topic_id**
  This is the e-mail to the topic of the article
- **topic_name**
  This is the name of the topic of the article
- **category_definition_name**
  This is the name of the article's main category

The Root Block Variables

- **current_category_image_width**
  Variable that holds the current category's image width
- **current_category_image_height**
  Variable that holds the current category's image height
- **current_category_image_URL**
  Variable that holds the current category's image URL
- **current_category_image_caption**
  Variable that holds the current category's image caption text

The "path_item_tpl" Block Variables

- **category_id**
  Variable that holds the current path item's id
- **category_name**
  Variable that holds the current path item's name

The "image_list_tpl" Block Variables

This block is only shown if there are any images associated with the article that haven't been used in the article text.

- **image_id**
  Variable that holds the current article’s current unused image name

- **image_width**
  Variable that holds the current article’s current unused image width
- **image_height**
  Variable that holds the current article’s current unused image height
- **image_URL**
  Variable that holds the current article’s current unused image URL.
- **image_caption**
  Variable that holds the current article’s current unused image caption text

The "article_header_tpl" Block Variables

The block is not shown if the page is called as "articleuncached", "articlestatic" or "static".

The "article_URL_item_tpl" Block Variables

This block is only shown if the page is called as "print" or "articleprint".

The "article_topic_tpl" Block Variables

The block is only shown if the article has a topic.

The "article_intro_tpl" Block Variables

The block is only shown if the article is on page one, and then only if the intro exists.

The "attribute_list_tpl" Block Variables

This block is only shown if there are any attributes associated with the article.

- **type_id**
  Variable that holds the name of the current attribute type
- **type_name**
  Variable that holds the id of the current attribute type
- **attribute_id**
  Variable that holds the id of the current attribute
- **attribute_name**
  Variable that holds the name of the current attribute
- **attribute_value**
  Variable that holds the value of the current attribute

The "attached_file_list_tpl" Block Variables

This block is only shown if there are any files associated with the article.

- **file_id**
  Variable that holds the id of the current file

- **file_name**
  Variable that holds the value of the current file
- **file_description**
  Variable that holds the description of the current file
- **file_size**
  Variable that holds the file size of the current file
- **file_unit**
  Variable that holds the file size unit of the current file
- **file_URL**
  Variable that holds the URL of the current file
- **original_file_name**
  Variable that holds the original file name of the current file

---

Article Discussions

The article discusses uses of the "ezforum/user/messagesimplelist.php" file.

authorlist.php

This page is activated through the URL: /article/author/list/order/

- The last part of the URL is the sort order, this is optional.

This will list all the article authors.

Sorting Order

Sorting order can be any one of these:

- **name**
  This will order the list based on the author name (a single text string).
- **count**
  This will order the list based on popularity, i.e. how often the article has been viewed.

Default is to list by author name.

Template Variables

The variables provided are:

- **author_id**
  The numerical id of the author. Used in links to /article/author/view/.
- **author_name**
  This is the name of the author.
- **article_count**
  This is the number of times the article has been viewed.

---

authorview.php

This page is activated through the URL: /article/author/list/1/order/2/

- The first digit of the URL is the numerical ID of the author.
- The "order" part of the URL is the sort order.
- The last digit of the URL is the offset to the start of the list; i.e. the number of articles to skip before starting the list.

This will list all the articles by the current author.

Sorting Order

Sorting order can be any one of these:

- **author**
  This will order the list based on the author's name (a single text string).
- **name**
  This will order the list based on the article's name (a single text string).
- **category**
  This will order the list based on the name of the main category of each article.
- **published**
  This will order the list based on the publication date of the article.

Default is to list by publication date.

Template Variables

The variables provided are:

- **author_id**
  The numerical id of the author. Used in links to the author view, "/article/author/view/".
- **author_name**
  This is the name of the author.
- **author_mail**
  This is the e-mail address of the author.

The "article_item_tpl" Block Variables

The variables provided are:

- **article_id**
  This is the numerical id of the article. Used in links to the article "/article/article/view/".
- **article_name**
  This is the name of the article.
- **article_category**
  This is the name of the article's main category.
- **article_published**
  This is the publishing date of the article.

- **author_name**
  This is the name of the author.
- **category_id**
  This is the id of the article's main category.

---

frontpage.php

This page is activated through the URL: /article/frontpage/1/

The last digit is the section id, as found in the section list in the admin interface. This will show the front page of the current section, as it has been configured in the section edit in the admin interface.

The Standard Variables

The variables provided are:

- **section_id**
  variable that holds the global section id, i.e. the current section.
- **www_dir**
  variable that holds the root directory for non-virtualhosts, empty on virtual host setups.
- **index**
  variable that holds the name of the index.php file for fallback when needed, empty on virtual host setups.

The Template Blocks for the Items

The items shown on the front page is configured through the section edit, as mentioned above. Each row in the section edit corresponds to one template block in this template. The different blocks are named the same as the drop down (in English) in the front page, except that numbers are spelled out.

Think of each of these blocks as building blocks for the completed page. If you do not have a one_column_product defined, it will not be shown. If you have four rows of one_short_article in the section edit, then the "one_short_article_tpl" template block will be rendered four times, once for each row, but with different information.

If the selected category for the one_short_article is the same for all four rows, then the four first article from the category (based on the category's sorting order) will be shown. If each row has its own category, then the first article from each category is shown.

The "one_column_article_tpl" Block Variables

This will show one article on this row.

- **article_id**
  This is the id of the article.
- **article_name**
  This is the name of the article.
- **author_text**
  This is the name of the author.
- **category_def_name**
  This is the name of the main category of the article.
- **category_def_id**
  This is the id of the main category of the article.
- **article_datevalue**
  The variable that holds the current article's publishing date
- **article_timevalue**
  The variable that holds the current article's publishing time
- **article_published**
  The variable that holds the current article's publishing date and time
- **article_link_text**
  Variable that holds the article link text, if there isn't a text defined for this article, the value is fetched from the variable "DefaultLinkText" in the section "eZArticleMain" of the "site.ini" file.
- **article_intro**
  Variable that holds the current article's intro text (teaser)
- **thumbnail_image_width**
  Variable that holds the current article's thumbnail image width
- **thumbnail_image_height**
  Variable that holds the current article's thumbnail image height
- **thumbnail_image_uri**
  Variable that holds the current article's thumbnail image URL.
- **thumbnail_image_caption**
  Variable that holds the current article's thumbnail image caption text

The "two_column_article_tpl" Block Variables

This will show two articles on this row. The same variables are available as in the "one_column_article_tpl", but you will have to configure both the left and the right column.

The "one_short_article_tpl" Block Variables

This will show one article link on this row.

- **article_id**
  This is the id of the article.
- **article_name**
  This is the name of the article.
- **author_text**

This is the name of the author.

- **category_def_name**
  This is the name of the main category of the article.
- **category_def_id**
  This is the id of the main category of the article.
- **article_datevalue**
  Variable that holds the current article's publishing date
- **article_timevalue**
  Variable that holds the current article's publishing time
- **article_published**
  Variable that holds the current article's publishing date and time
- **article_link_text**
  Variable that holds the article link text, if there isn't a text defined for this article, the value is fetched from the variable "DefaultLinkText" in the section "eZArticleMain" of the "site.ini" file.

The "one_column_product_tpl" Block Variables

This will show one product on this row.

- **thumbnail_image_width**
  Variable that holds the current product's thumbnail image width
- **thumbnail_image_height**
  Variable that holds the current product's thumbnail image height
- **thumbnail_image_uri**
  Variable that holds the current product's thumbnail image URL.
- **thumbnail_image_caption**
  Variable that holds the current product's thumbnail image caption text
- **product_name**
  The name of the product.
- **product_id**
  The product id, the unique id of the product. This number is used in URLs where the product is to be shown.
- **product_intro_text**
  This is the product's introductory text.
- **product_price**
  This is the price of the product. It will show either one single price, or if the product has options, a minimum and maximum price.

The "two_column_product_tpl" Block Variables

This will show two products on this row. Each product will feature the same variables as for the single product, but each product will be stored in their own blocks: "left_product_tpl" and "right_product_tpl".

- **thumbnail_image_width**
  Variable that holds the current product's thumbnail image width
- **thumbnail_image_height**
  Variable that holds the current product's thumbnail image height
- **thumbnail_image_uri**
  Variable that holds the current product's thumbnail image URL.
- **thumbnail_image_caption**
  Variable that holds the current product's thumbnail image caption text
- **product_name**
  The name of the product.
- **product_id**
  The product id, the unique id of the product. This number is used in URLs where the product is to be shown.
- **product_intro_text**
  This is the product's introductory text.
- **product_price**
  This is the price of the product. It will show either one single price, or if the product has options, a minimum and maximum price.

---

headlines.php

This file is included like this:

```
<?
  $CategoryID = 1;
  $Limit = 1;
  $HeadlineOffset = 0;
  include( "ezarticle/user/headlines.php" );
?>
```

This will result in a list of articles. The list will be of the selected category ($CategoryID), the length of the list will be set by the limit ($Limit; defaults to 10) and starting at the position of the offset ($HeadlineOffset, defaults to 0).

If the $CategoryID is set to 0, all the news in all categories will be collated into one list, sorted by the publication date. All actual categories are sorted by the sorting set for that particular category.

Template Variables

The variables provided are:

- **current_image_width**
  Variable that holds the listed article's image width
- **current_image_height**
  Variable that holds the listed article's image height
- **current_image_URL**
  Variable that holds the listed article's image URL

- **current_image_caption**
  variable that holds the listed article's image caption text

- **article_id**
  This is the id of the article.
- **article_name**
  This is the name of the article.
- **article_published**
  Variable that holds the current article's publishing date and time
- **article_link_text**
  Variable that holds the article link text, if there isn't a text defined for this article, the value is fetched from the variable "DefaultLinkText" in the section "eZArticleMain" of the "site.ini" file.

----

menuarticleview.php

This file is included like this:

```
<?
   $ArticleID = 1;
   include( "ezarticle/user/menuarticleview.php" );
?>
```

This will result in the printing of the body text of the selected article ($ArticleID).

Template Variables

There is only one template variable provided as default:

- **article_body**

  The body of the article.

However, you can use the same variables and sections (which all are empty by default) as in the "articleview.php" page.

----

menubox.php

This file is included like this:

```
<?
   $CategoryID = 1;
   include( "ezarticle/user/menubox.php" );
?>
```

This will result in the menu box for the eZ article module. The menu box will also list the sub categories in the selected category ($CategoryID). If the category is set to zero, all the top categories are listed.

Template Variables

The variables provided are:

1. The Standard Variables

- **section_id**
  variable that holds the global section id, i.e. the current section.
- **www_dir**
  variable that holds the root directory for non-virtualhosts, empty on virtual host setups.
- **index**
  variable that holds the name of the index.php file for fallback when needed, empty on virtual host setups.

2. The "submit_article_tpl" Block Variables

This is shown only if the "UserSubmitArticles" variable in "eZArticleMain" is set to enabled.

3. The "article_category_tpl" Block Variables

- **articlecategory_id**
  The id of the listed category.
- **articlecategory_title**
  The name of the listed category.

----

menumaker.php

This file is included like this:

```
<?
   $CategoryID = 1;
   include( "ezarticle/user/menumaker.php" );
?>
```

This file will result in a menu where all sub-elements of the selected category ($CategoryID) will appear, both articles and categories. If there are no sub-elements nothing will be shown.

Template Variables

The variables provided are:

1. The Standard Variables

- **section_id**
  variable that holds the global section id, i.e. the current section.

- **www_dir**
  variable that holds the root directory for non-virtualhosts, empty on virtual host setups.
- **index**
  variable that holds the name of the index.php file for fallback when needed, empty on virtual host setups.

2. The "menu_box_tpl" Block Variables

This block will only be shown if there is any articles or categories belonging to the category.

3. The "menu_header_tpl" Block Variables

- **current_category_name**
  The name of the selected category.

4. The "article_category_tpl" Block Variables

- **article_id**
  The id of the current article.
- **article_link_text**
  The title of the current article.

5. The "menu_category_tpl" Block Variables

- **category_id**
  The id of the current category.
- **category_link_text**
  The title of the current category.

---

sitemap.php

This page is activated through the URL: /article/sitemap/

This will result in a sitemap that shows all the categories and articles on the site in a tree. This can take a long time to generate on large sites.

Template Variables

The variables provided are:

1. The Standard Variables

- **section_id**
  variable that holds the global section id, i.e. the current section.
- **www_dir**
  variable that holds the root directory for non-virtualhosts, empty on virtual host setups.

- **index**
  variable that holds the name of the index.php file for fallback when needed, empty on virtual host setups.

2. The "category_value_tpl" Block Variables

- **option_value**
  The id of the category.
- **option_name**
  The name of the category.
- **option_level**
  The depth, i.e. level, of the category.

3. The "article_value_tpl" Block Variables

- **option_value**
  The id of the article.
- **option_name**
  The name of the article.
- **option_level**
  The depth, i.e. level, of the article.
- **category_id**
  The ID of the parent category of the article.

---

smallarticlelist.php

This file is included like this:

```
<?
   $CategoryID=1;
   $Offset=0;
   $Limit=3;
   include( "ezarticle/user/smallarticlelist.php" );
?>
```

The result will be a list of articles, including their intros and links to the full article. The articles will come from the selected category ($CategoryID), starting at the offset ($Offset) and with the number of articles selected ($Limit).

If there are no articles in the selected category, nothing will be shown. If there are fewer articles than the selected number, then only those articles available will be shown.

Template Variables
The variables provided are:

1. The Standard Variables

- **section_id**
  variable that holds the global section id, i.e. the current section.
- **www_dir**
  variable that holds the root directory for non-virtualhosts, empty on virtual host setups.
- **index**
  Variable that holds the name of the index.php file for fallback when needed, empty on virtual host setups.

2. The "article_list_tpl" Block Variables

This block will only be shown if there is any content to be shown.

- **current_category_name**
  This is the name of the category from which the articles are shown.
- **current_category_description**
  This is the description of the category from which the articles are shown.

3. The "article_item_tpl" Block Variables

- **article_id**
  This is the id of the article.
- **article_name**
  This is the name of the article.
- **article_published**
  Variable that holds the current article's publishing date and time
- **article_intro**
  This is the intro for the article.
- **category_id**
  This is the id of the category we're looking into.

4. The "read_more_tpl" Block Variables

This block will only be shown if the article has a body.

- **article_link_text**
  variable that holds the article link text, if there isn't a text defined for this article, the value is fetched from the variable "DefaultLinkText" in the section "eZArticleMain" of the "site.ini" file.

topiclist.php

This page is activated through the URL:

/article/topiclist/
/article/topiclist/3/

The first method of calling this page will result in a list of topics that shows all the articles belonging to each topic.

The second method of calling this page will show the articles belonging to a specific topic. The last number in the URL is then the ID of the topic in question.

Template Variables

The variables provided are:

1. The Standard Variables

- **section_id**
  Variable that holds the global section id, i.e. the current section.
- **www_dir**
  Variable that holds the root directory for non-virtualhosts, empty on virtual host setups.
- **index**
  Variable that holds the name of the index.php file for fallback when needed, empty on virtual host setups.

2. The "topic_item_tpl" Block Variables

- **topic_name**
  The name of the topic.
- **topic_description**
  The description of the topic.
- **id**
  The id of the topic.

3. The "article_item_tpl" Block Variables

- **article_id**
  The id of the article.
- **article_name**
  The name of the article.
- **category_id**
  The id of the main category of the article.
- **category_name**
  The name of the main category of the article.

Documentation specific to Visible Voices

This is the master plan of how the Visible Voices site itself has been created listing sections, sitedesign and the templates used with a breakdown of how it was all fitted together.

Since much of what would be included here will also be in the general documents published on Sourceforge it makes sense to merely make this a list of files and functions.

The sitedesign – where it is located – why no CSS2 – explain it's standards compliant XHTML – explain how meta tags can be in site.ini or in the body of frame.php – break down the PHP if/elseif function –
Templates – articleview/menuarticleview/showdc – explain about the article tags and why not to disturb the order of them.
When the site is finished and all logos are in place, Create screenshots of each level and overlay colour to emphasise where the templates are nested and which areas are purely static. This is particularly useful for the frontpage which has a fairly complicated set of interlocking templates.

The above section is not for inclusion in the final docs.

Creating and amending content

Content for Visible Voices is entered via the admin interface and can be found within 3 separate modules, **ezarticle**, **ezimagecatalogue**, and **ezmediamanage**. However, all the content, apart from the text and pictures for the News and Reviews article at the bottom of the front page, is created using the **ezmediamanage** wizard.

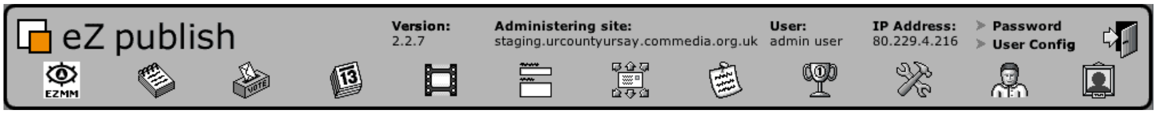
The Dublin Core metadata is held within **ezmediamanage** and can't be edited after entry. The description field is held in **ezarticle** and can be edited through that modules interface. All articles created this way are part of the Area Categories category.

The **News and Reviews** article is held in the **News and Reviews** category.

NOTE: For the article to appear on the front page it must always be article #4. Therefore to create a new review article, replace the title and text of the previous one.

The picture is supplied via the **ezimagecatalogue** module in the Webcast Screencaps category.

NOTE: in order for the picture to appear on the front page it must always be #1. Therefore the previous picture must be deleted and a new picture inserted.



eZ publish

Version: 2.2.7

Administering site: staging.urcountyursay.commedia.org.uk

User: admin user

IP Address: 80.229.4.216

Password

User Config

The screenshot above shows the module bar at the top of the admin section of the site with **ezmediamanage** and **ezarticle** as the first and second icons. **Ezimagecatalogue** is the last icon on the right.

As is illustrated by the 'About this site' article, HTML formatting can be added to articles to include images, tables and a whole range of other data. In order to use HTML in an article certain codes need to surround the area of the article where the tags are placed. In order for eZ to begin parsing the tags **<html></html>**. These are usually best place at the top and bottom of the article. eZ also has a set of easy to use tags that can replace normal HTML tags such as **<link></link>.** However, these do not always produce compliant code and are best avoided.

Design and templating – what goes where?

sitedesign   /home/ucus/sitedesign/standard

The sitedesign directory contains all the files that create the outer skin of the site. Sitedesigns, like most elements of eZ, are based on sections so since VV has only one section all the files are within the 'standard' directory.

These files make up the sitedesign;

frame.php – this file contains the same information as an index.html file would on a normal html site. The doctype definition is set manually as are the links to stylesheets and any javascript functions. Other head elements such as the site title and metatags can be set from within the **site.ini** file that controls the basic configuration of the site.

style.css – contains all the layout and textstyles for both sitedesign and templates.

images/ – only images used within the frame.php structure should be stored here. Images loaded into templates or articles should reside in the root level images folder.

jump.php – this contains the PHP code that drives the menu structure on the front page. It is not a normal element of eZ but was written in specifically to allow pull-down menus on the frontpage to be accessible. It requires that a specific alteration is made to the mod_rewrite section of the eZ servers **httpd.conf** in order to function.

smallarticles.dat – part of the PHP conditional statement within **frame.php** calls this in to the frontpage. It contains the templates **menuarticleview.tpl** and **imageview.tpl**.

templates

These add extra code to areas within the sitedesign and contain variables that format the output from the modules such as ezarticle. Like sitedesign they are sections-based. See below for screenshots showing the positions on the page of each template

ezarticle /home/ucus/ezarticle/user/templates/standard

frontpage.tpl – this template contains all the search engine menus and fields. It is the default page but can also be called through the URL /article/frontpage/1

articlelist.tpl – this is called by the **Video projects by area** menu. It outputs a list of projects based on the category/area selected. It can also be called through the URL /article/archive/1(-13)

topiclist.tpl – this is called by the **Video projects by issue** menu. It outputs a list of projects based on the topic/issue selected. It can also be called through the URL /article/topiclist/1(-10)

search.tpl – this is called by clicking the search button on the front page. It returns a list of search results. It cannot be called by a URL.

articleview.tpl – this template, along with showdc.tpl, is called when a project name is clicked on and displays the body of the article/description that accompanies the metadata.

menuarticleview.tpl – this template formats the text from the small article under **News and Reviews** at the bottom of the front page.

ezimagecatalogue /home/ucus/ezimagecatalogue/user/templates/standard

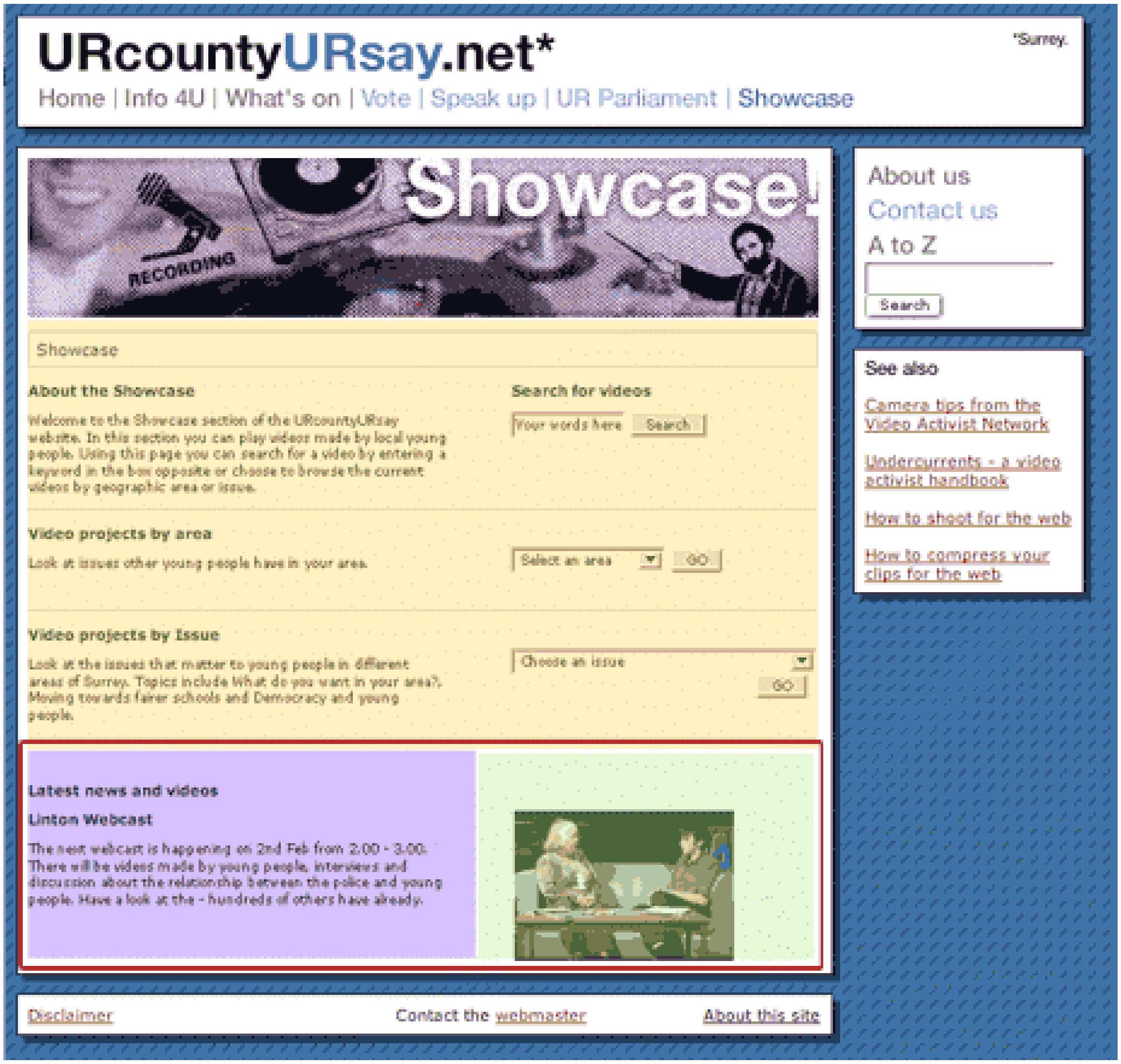
imageview.tpl – this provides the picture that accompanies the **News and Reviews** article.

ezmediamanage /home/ucus/ezmediamanage/user/templates/standard

showdc.tpl – this outputs the Dublin Core metadata that is saved with the media upload. It appears as a formatted table below the media description.

Screenshots

Figure 1. The front page



Showcase

About the Showcase

Welcome to the Showcase section of the URcountyURsay website. In this section you can play videos made by local young people. Using this page you can search for a video by entering a keyword in the box opposite or choose to browse the current videos by geographic area or issue.

Search for videos

Video projects by area

Look at issues other young people have in your area.

Video projects by Issue

Look at the issues that matter to young people in different areas of Surrey. Topics include What do you want in your area?, Moving towards fairer schools and Democracy and young people.

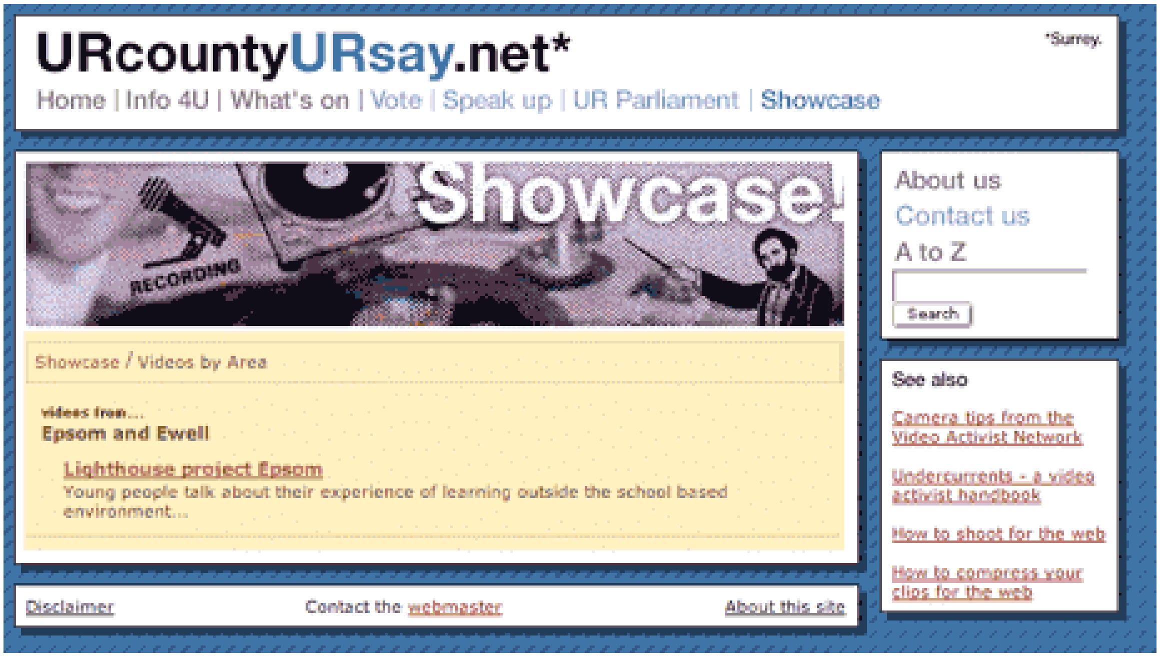
Latest news and videos

Linton Webcast

The next webcast is happening on 2nd Feb from 2:00 - 3:00. There will be videos made by young people, interviews and discussion about the relationship between the police and young people. Have a look at the - hundreds of others have already.

The yellow area covers the content generate through **frontpage.tpl**. The red border surrounds the content contained in **smallarticles.dat**. This file calls the templates **menuarticleview.tpl** (purple area) and **imageview.tpl** (green area).

Figure 2. The Videos by Area page



URcountyURsay.net*

*Surrey.

Home | Info 4U | What's on | Vote | Speak up | UR Parliament | Showcase
Showcase!
RECORDING
About us
Contact us
A to Z
Search

Showcase / Videos by Area

videos from...

Epsom and Ewell

Lighthouse project Epsom

Young people talk about their experience of learning outside the school based environment...

See also

Camera tips from the Video Activist Network

Undercurrents - a video activist handbook

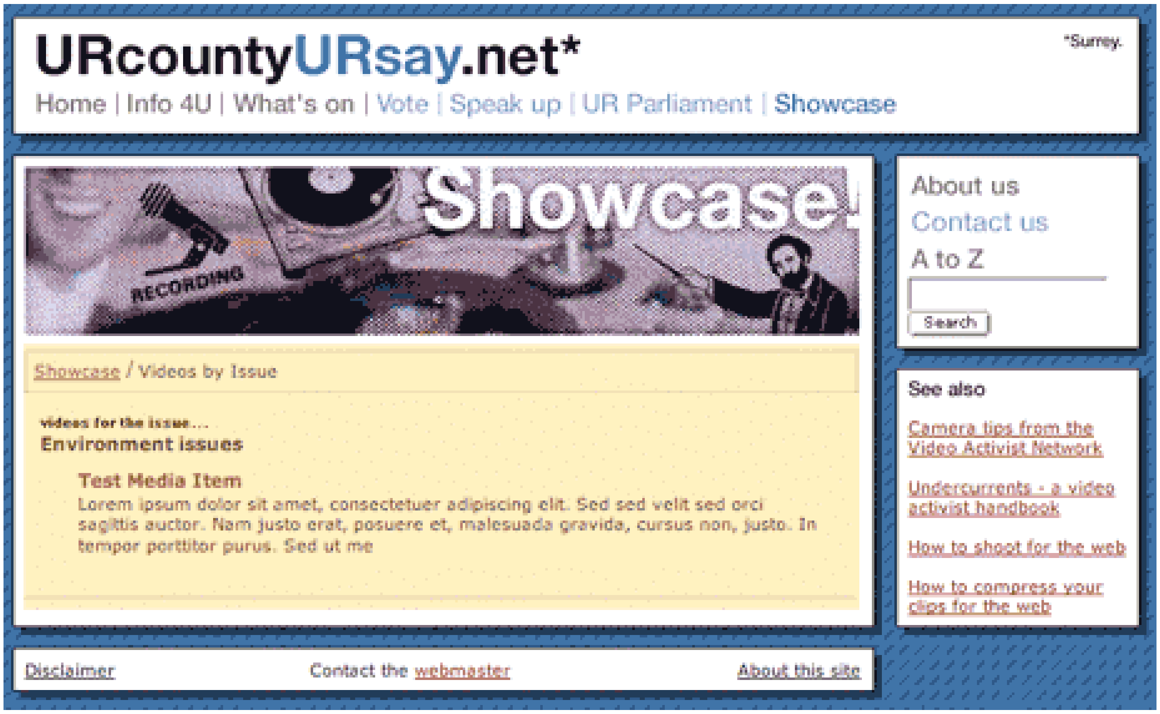
How to shoot for the web

How to compress your clips for the web

Disclaimer Contact the webmaster About this site

The yellow area covers the content generate through **articlelist.tpl**.

Figure 3. The Videos by Issue page



URcountyURsay.net*

*Surrey.

Home | Info 4U | What's on | Vote | Speak up | UR Parliament | Showcase
Showcase!
RECORDING
About us
Contact us
A to Z
Search

Showcase / Videos by Issue

videos for the issue...

Environment issues

Test Media Item

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Sed sed velit sed orci sagittis auctor. Nam justo erat, posuere et, malesuada gravida, cursus non, justo. In tempor porttitor purus. Sed ut me

See also

Camera tips from the Video Activist Network

Undercurrents - a video activist handbook

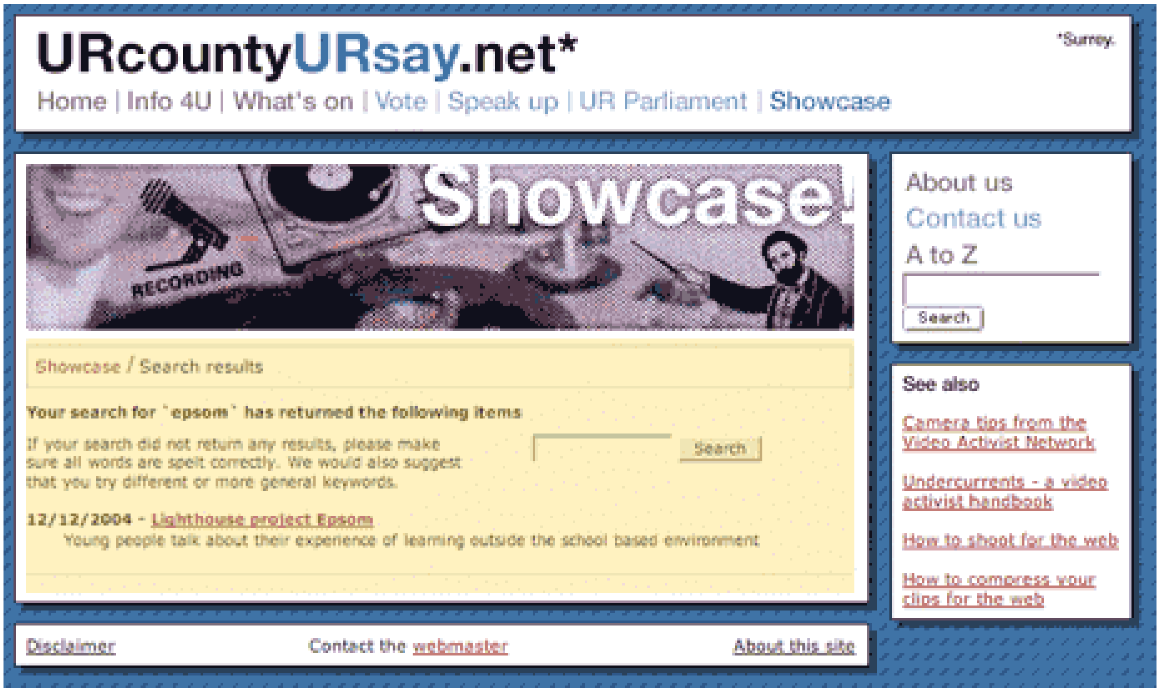
How to shoot for the web

How to compress your clips for the web

Disclaimer Contact the webmaster About this site

The yellow area covers the content generate through **topiclist.tpl**.

Figure 4. The search results page



URcountyURsay.net*
*Surrey.
Home | Info 4U | What's on | Vote | Speak up | UR Parliament | Showcase
Showcase!
RECORDING
About us
Contact us
A to Z
Search
See also
Camera tips from the Video Activist Network
Undercurrents - a video activist handbook
How to shoot for the web
How to compress your clips for the web
Disclaimer
Contact the webmaster
About this site

Showcase / Search results

Your search for `epsom` has returned the following items

If your search did not return any results, please make sure all words are spelt correctly. We would also suggest that you try different or more general keywords.

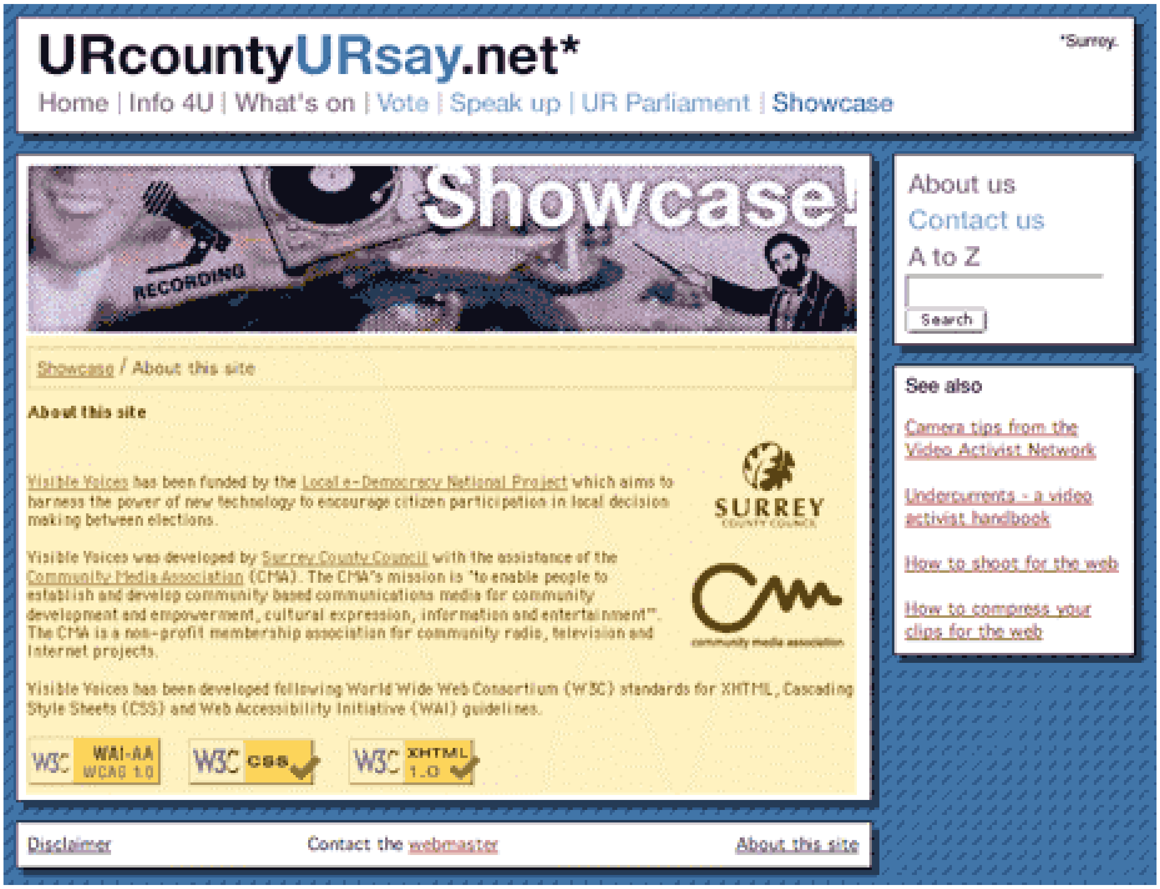
Search

12/12/2004 - Lighthouse project Epsom

Young people talk about their experience of learning outside the school based environment

The yellow area covers the content generated through **search.tpl**.

Figure 5. The ‘About this site’ page



URcountyURsay.net*

*Surrey.

Home | Info 4U | What's on | Vote | Speak up | UR Parliament | Showcase

About us

Contact us

A to Z

Search

Showcase / About this site

About this site

Visible Voices has been funded by the Local e-Democracy National Project which aims to harness the power of new technology to encourage citizen participation in local decision making between elections.

Visible Voices was developed by Surrey County Council with the assistance of the Community Media Association (CMA). The CMA's mission is "to enable people to establish and develop community based communications media for community development and empowerment, cultural expression, information and entertainment". The CMA is a non-profit membership association for community radio, television and Internet projects.

Visible Voices has been developed following World Wide Web Consortium (W3C) standards for XHTML, Cascading Style Sheets (CSS) and Web Accessibility Initiative (WAI) guidelines.

See also

Camera tips from the Video Activist Network

Undercurrents - a video activist handbook

How to shoot for the web

How to compress your clips for the web

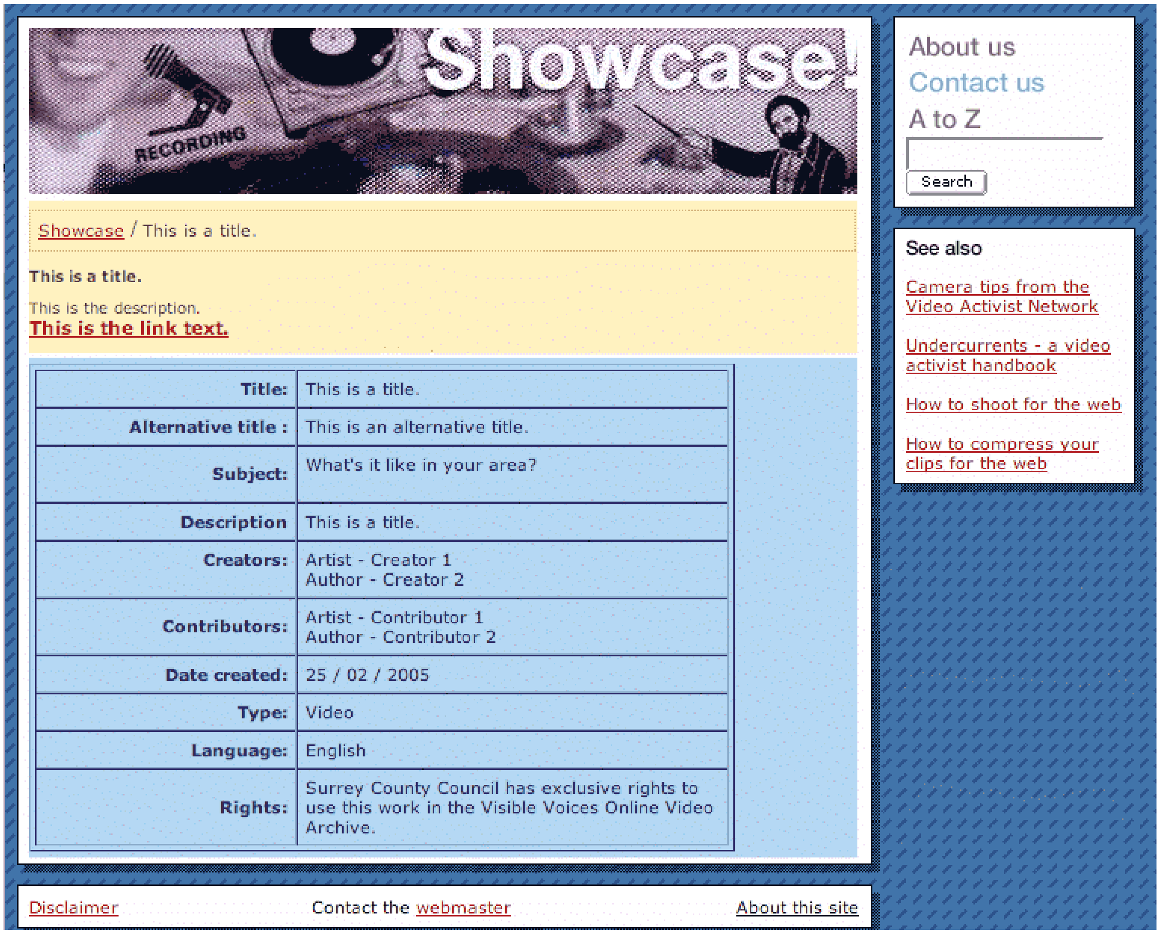
Disclaimer

Contact the webmaster

About this site

The yellow area covers the content generated through **articleview.tpl**. The logos are embedded using HTML in the article.

Figure 6. The Media page



About us
Contact us
A to Z
Search

See also

Camera tips from the Video Activist Network

Undercurrents - a video activist handbook

How to shoot for the web

How to compress your clips for the web

Showcase / This is a title.

This is a title.

This is the description.
This is the link text.

Title:	This is a title.
Alternative title :	This is an alternative title.
Subject:	What's it like in your area?
Description	This is a title.
Creators:	Artist - Creator 1 Author - Creator 2
Contributors:	Artist - Contributor 1 Author - Contributor 2
Date created:	25 / 02 / 2005
Type:	Video
Language:	English
Rights:	Surrey County Council has exclusive rights to use this work in the Visible Voices Online Video Archive.

Disclaimer Contact the webmaster About this site

The yellow area covers the content generated through **articleview.tpl**. The blue area is the metadata table generated through **showdc.tpl**

Appendix B

Creative Commons License

Attribution-NonCommercial-ShareAlike 2.0

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. **"Collective Work"** means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

b. **"Derivative Work"** means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

c. **"Licensor"** means the individual or entity that offers the Work under the terms of this License.

d. **"Original Author"** means the individual or entity who created the Work.

e. **"Work"** means the copyrightable work of authorship offered under the terms of this License.

f. **"You"** means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

g. **"License Elements"** means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, Noncommercial, ShareAlike.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;

b. to create and reproduce Derivative Works;

c. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;

d. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works;

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved, including but not limited to the rights set forth in Sections 4(e) and 4(f).

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any reference to such Licensor or the Original Author, as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any reference to such Licensor or the Original Author, as requested.

b. You may distribute, publicly display, publicly perform, or publicly digitally perform a Derivative Work only under the terms of this License, a later version of this License with the same License Elements as this License, or a Creative Commons iCommons license that contains the same License Elements as this License (e.g. Attribution-NonCommercial-ShareAlike 2.0 Japan). You must include a copy of, or the Uniform

Resource Identifier for, this License or other license specified in the previous sentence with every copy or phonorecord of each Derivative Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Derivative Works that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder, and You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Derivative Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Derivative Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Derivative Work itself to be made subject to the terms of this License.

c. You may not exercise any of the rights granted to You in Section 3 above in any manner that is primarily intended for or directed toward commercial advantage or private monetary compensation. The exchange of the Work for other copyrighted works by means of digital file-sharing or otherwise shall not be considered to be intended for or directed toward commercial advantage or private monetary compensation, provided there is no payment of any monetary compensation in connection with the exchange of copyrighted works.

d. If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and give the Original Author credit reasonable to the medium or means You are utilizing by conveying the name (or pseudonym if applicable) of the Original Author if supplied; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

e. For the avoidance of doubt, where the Work is a musical composition:

   i. **Performance Royalties Under Blanket Licenses**. Licensor reserves the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work if that performance is primarily intended for or directed toward commercial advantage or private monetary compensation.

   ii. **Mechanical Rights and Statutory Royalties**. Licensor reserves the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions), if Your distribution of such cover version is primarily intended for or directed toward commercial advantage or private monetary compensation.

f. **Webcasting Rights and Statutory Royalties**. For the avoidance of doubt, where the Work is a sound recording, Licensor reserves the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions), if Your public digital performance is primarily intended for or directed toward commercial advantage or private monetary compensation.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTIBILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, neither party will use the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time.

Creative Commons may be contacted at http://creativecommons.org/