

Aug 29 11:59:52 PDT 2006

Instructions for editing and installing SBC's website mini-CMS. The website was developed this way so that menus can be reconfigured and new pages can be added more easily.

Basics:

uses perl templateToolkit and the ttree program (built into linux)

Major features:

1. Configurable for installation in multiple directories, i.e., development or production directory.

2. uses "ttree" program, via a shell script. (for instructions on installation of TT, see

<http://template-toolkit.org/docs/tools/ttree.html>

SBC's website source code stored in Subversion. The trunk is web_website/trunk. Use this command:

```
svn+ssh://horcon.msi.ucsb.edu/data01/svn/sbclter/sbc_website/trunk
```

we plan to make subversion available over http.

This is the SHORT set of cmds in a typical edit session. For longer list, see

<https://sbc.lternet.edu> and click on Jenny's cheat-sheet. https://sbc.lternet.edu/website_update_cmds.html

1. svn update (to make sure your version is most recent one)

if you're not mob, you may need to change the destination path in etc/dev_ttree.cfg to point to a valid directory (like your own public_html). The '-a' option on the build command forces a build of all pages, for a visual check of the website. To do one page, add the page name (as in data/index.html, leave off the /src/) instead of the -a option.

2. ./build_dev.sh -a

point your browser to <http://sbc.lternet.edu/~mob/dev/docs>

3. <edit those pages as needed>

4.repeat ad nauseam.

5. svn commit (as needed, send your changes back to subversion)

6. build_production.sh

7. ssh penguin; su www; make sure the permissions are right (password..., you'll have to look it up.) You tried setting the 's' bit to keep the ownership at apache. But if you rip the whole thing out and re-install, it will do it as mob again. (can you install from mob as apache?)

Some notes:

A. if you replace the entire website, you will want to add the three readmes

```
build_production.sh affiliates/readme.txt
```

```
build_production.sh news/items/readme.txt
```

```
build_production.sh data/readme.txt
```

and the robots.txt

```
build_production.sh robots.txt
```

these need to be manual because all text files (extension .txt) are protected by default from the copy process to keep the news items from getting a new time stamp. Kind of lame I know, but have not taken the time to make the script accept some text file, but not all. 3 more commands is a small price to pay.

However, there is a rule to copy these text files

Here's how to archive a news item (also see the news dir for info on the bloxom script):

1. go into the news/items/DIRECTORY/ and

2. cp existingstory.txt newstory.txt

This ensures you have the META tag, which keeps the file from getting the wrapper added. Title is line 1, and story on

line 2. no newlines within a paragraph.

3. ./build_production.sh --trim news/items/DIRECTORY/newstory.txt

4. The --trim option cleans out extra blank lines. But you should still go into www/docs/news/items to check the story and take off the write permission.

Here is a short list of the THE FILES in svn: /website/

A longer list follows.

bin/: scripts

build_dev.sh -- sends output to ~mob/public_html/dev/docs

bulid_production.sh -- sends output to /var/www/sbc.ltnet.edu

bulid_jenny.sh -- sends output to ~j_dugan/public_html/dev/docs

docs/: generated pages from build script.

deprecated. we use another dir as output, either ~mob/public_html/dev/docs, or (future)

/var/www/sbc-dev.ltnet.edu/docs

etc/: misc control files.

*_ttree.cfg: sets dirs for src, lib, output. 'pre_process config' and 'wrapper' directives. a different one used by each build script, above.

lib/: reusable components, Template Toolkit (TT) files

config/: these files with variables set for css, js, image directories

1. config/main:

2. config/build_pages.tmpl: sets conditions for what page types are to have the wrapper added, or simply copied.

3. config/site_cfg.tmpl: contains vars definitions for sbclter urls

4. config/offline_url.tmpl: contains vars definitions for off-site urls

(these are deprecated, not using target="offline" anymore.

these template files (in lib) need to be accessed by cgi scripts at run time, and should be copied to an output dir (either a dedicated script, or by build).

1. wrapper.html: banner, top menu, and footer for all pages

src/: source templates for pages, eg,

help/index.html, help/faq.html, help/server_connection.html

These have "html" extensions, but a better style would be to call them html_src, and set the switch to change that extension to html in output (see TT docs)

these are parts of various sandboxes:

2. displayTaxonomyList.tmpl: creates a table of taxonomy csv and posts it (could input from database view.

3. displaySites.tmpl: creates a table of siteDescription.csv info for display

Notes:

bin and docs aware already at /var/www/sbc.ltnet.edu/

Other to be considered (also at var/www):

cgi-bin:

need to activate the development location, either ~mob/cgi-bin, or set cgi templates to look at /var/www/sbc.ltnet.edu/lib?

private:
done, 2007.

php:

Accompanying docs:
TT_notes.txt
directory_contents.txt

file: directory_contents.txt

/home/sbclter/mob/website

bin:

build-website.sh* configure_example.pl* tmpl-copy.sh*
cgi-build.sh* fancy_build.pl*

cgi:

ldapweb.cgi*
ldapweb2006.cgi*
ldapweb2009.cgi
test_config.cgi*
test_sites_display.cgi*
test_sites_display2.cgi*
test_taxon_display.cgi*

dev:

navigation_additional.js

docs:

empty, should no longer be set as destination dir.

documentation:

editing_notes.txt README.html* TT_notes*

etc:

cgi_ttree.cfg -- for building the cgi scripts
test-ttree.cfg -- for messing around
tmpl_ttree.cfg
ttree.cfg -- a default, not used
dev_ttree.cfg -- used heavily, destination = ~mob/public_html/dev/docs
production_ttree.cfg -- destination = /var/www/sbc.lternet.edu/docs
jenny_ttree.cfg -- destination = /home/sbclter/j_dugan/public_html/dev/docs

lib: (you've got a lot, and should consider subdirs now)

build_pages.tpl
data_cruises20012006_year.tpl
data_formSearchByCoreResearchArea.tpl
data_formSearchByHabitat.tpl
data_formSearchByMeasurementtype.tpl
displayDataCollections.tpl
display_Projects.tpl
displaySites.tpl
displayTaxonomyList.tpl
emptyTemplate.tpl
menu_box.tpl
people_nsfPIs2009.tpl
people_searchResults2009.tpl
people_searchResults.tpl
people_showAllByTitle2009.tpl

people_showAllByTitle.tpl
people_showIndividual2009.tpl
sbclterDataUseAgreement.tpl
wrapper.html

lib/config/
main
offline_urls.tpl
page_cfg.tpl
page_description.tpl
README.html
sitemap.tpl
site_cfg.tpl

lib/deprecated/ -- old stuff

skeleton:
bin/ etc/

src:
admin_index.html
affiliates/
brown_arrow7x12.gif
collaborations/
data/
education/
futurePages/
help/
images/
index.html*
info_management/
ITRS_2006/
location/
navigation.js*
navigation_saveme.js*
news/
outreach/
people/
project/
publications/
research/
robots.txt*
sbclter_content.css
sbclter_wrapper.css
sbclter_wrapper_testnav.css
sites/
underConstruction.html
w3_recommended.css

TT_notes.txt

Notes on template toolkit:

nob 14. 2007:

had been using v2.14, but having problems accessing the plugins for xml. These were supposed to be included with this version, but maybe not. So updated to version 2.19. But in versions 2.15 and later, the xml plugins are not included, so these need to be installed, too. Some of these have dependencies, which are not all installed yet.

XML.XPath works. XML.Simple works

XML.LibXML - not yet,

pkgs installed:

```
1001 cd ~mob/downloads
```

```
1002 tar zxf Template-Toolkit-2.19.tar.gz
```

```
1003 cd Template-Toolkit-2.19
```

```
1004 ls
```

```
1005 perl Makefile.PL
```

```
1006 make
```

```
1007 make test
```

```
1008 sudo make install
```

```
1037 tar zxf XML-XPath-1.13.tar.gz
```

```
1038 cd XML-XPath-1.13
```

```
1039 perl Makefile.PL
```

```
1040 make
```

```
1041 make test
```

```
1042 sudo make install
```

```
1018 tar zxf XML-Simple-2.18.tar.gz
```

```
1019 cd XML-Simple-2.18
```

```
1020 perl Makefile.PL
```

```
1021 make
```

```
1022 make test
```

```
1023 sudo make install
```

packages not installed yet (lots of dependencies):

```
1001 cd ~mob/downloads
```

```
1009 cd ../Template-XML-2.17
```

```
1010 ls
```

```
1011 perl Makefile.PL
```

```
1012 cd ..
```

```
1013 ls
```

```
1014 tar zxf XML-RSS-1.31.tar.gz
```

```
1015* cd XML-RSS-1.3
```

```
1016 perl Makefile.PL
```

```
1017 cd ..
```

```
1027 tar zxf XML-Template-3.20.tar.gz
```

```
1028 cd XML-Template-3.20
```

```
1029 ls
```

```
1030 perl Makefile.PL
```

```
<div id="panel-left">
  <div id="menu-box">
    [% PROCESS menu_box.tmpl %]
  </div>
</div>
```

```
<div id="panel-right">
```

```
</div>
```