Projects / Hua

Print version of this document (Hua.pdf, 36K).

Table of Contents

Hua Basics

Hua (simplified Chinese for flower) is a simple, feature−rich, template−based static content generator used for maintaining web sites and blogs.

Written in PowerShell, Hua was originally written in the Ruby scripting language. It was inspired, in part, by the venerable Perl−based blogging tool Blosxom and similar static content generators. Simplicity is a core principle: The database containing site entries, site content, includes, and template files are maintained in plain text. Comments are provided through a third−party engine like Disqus or IntenseDebate. Hua is open source software (MIT License).

Features

- Supports as many sites as there are configuration files.
- Template−based: Hua gives the designer complete control over a site's visual and interaction design.
- No need to code the content in HTML—Hua supports Markdown.
- Configurable paging.
- Supports tagging at the article and site−level.
- Supports article−level and site meta information.
- Runs on Windows and Linux.
- 100% text−based configuration and content.
- Command line−based: it's easy to integrate Hua into an existing toolchain.
- PDF output: Hua supports PDF output through Pandoc and the GNU roff (groff) typesetting system.

Requirements & Download

Hua has been tested on Windows 10, Ubuntu 22.04 and Cygwin (see note regarding the latter in Using Hua). It should run on any reasonably up−to−date PowerShell installation. An example site (configuration file, entries file, meta info file, and content) is included in the distribution so you can try it out immediately.

Hua supports Markdown using the MarkdownToHTML module. For PDF output, Pandoc and GNU roff (groff) are required.

Hua Revision 287 − Fri Feb 14 2025 Hua.zip, 81K | Hua−dist.tar.gz, 50K

Installation

Hua.zip can be used on either Windows or Linux systems. Simply unzip and place it wherever you want.

Hua.tar.gz is intended for use on Linux systems. To install, decompress the file and extract the tar archive. It is designed to be installed in `/usr/local`. Don't forget to place `/usr/local/bin` in the executable path and `/usr/local/share/man` in the manual path. What's installed and where.

Configuring Hua

The Hua configuration file specifies the locations of support files and variables used by Hua (articles/entries and meta info, output/content/include directories, URLs).

```
Hua files and config variables:

    entries_file      Delimited text file of blog entries
    meta_file         Delimited text file containing blog meta info
    content_dir       Directory of the blog content
    markdown_dir      Directory containing the markdown template
    output_dir        Directory of the blog output HTML
    index_file        File path of the primary blog page
    archive_file      File path of page listing all article titles
    header_file       File path of header include
    footer_file       File path of footer include
    read_more_file    File name of 'Read More...' include
    comments_file     File name of comments include
    blog_root         The web accessible root of the blog
    web_root          The web accessible root of the web site
    title_sep         Character(s) used to separate elements in the title
    entries_pp        Number of entries per page (for paging)
    next_label        Label of next page link (replace spaces with  )
    prev_label        Label of previous page link (replace spaces with  )
    log_dir           Directory in which log file is output
    log_file          File name of log file

entries_file   =  .\entries.csv
meta_file      =  .\meta.csv
content_dir    =  .\content
markdown_dir   =  .\markdown
output_dir     =  .\blog
index_file     =  index.html
archive_file   =  archive.html
header_file    =  .\content\inc\header.html
footer_file    =  .\content\inc\footer.html
read_more_file =  .\content\inc\read_more.html
comments_file  =  .\content\inc\comments.html
blog_root      =  /blog/
web_root       =  http://example.local
title_sep      =  /
entries_pp     =  2
next_label     =  Next Page
prev_label     =  Previous Page
log_dir        =  .
log_file       =  hua.log
```

The Entries File

The entries file is a comma−delimited plain−text database of articles. It can be edited with a text editor or a spreadsheet application (provided the plain−text format is preserved). Each line consists of the following fields:

1. ID: A numeric field. Articles are sorted and presented in descending order using this field.
2. Title: Appears at the top of the article and is linked to the article's permalink.
3. File: The filename of the article's template, stored in the `content_dir`.
4. Tags: Article−specific tags ("|" delimited). In addition to index and permalink pages, articles appear in the appropriate tag pages.
5. Date: The date presented with the article. *Note:* This doesn't control the position of the article on the index and tag pages, ID does that.
6. Author: Individual or organization name associated with the article.
7. Contact: Link to individual or organization web page (precede with `http://`) or email (precede with `mailto:`) associated with the article.

The Meta File

Article meta information is stored in a separate, pipe−delimited file. Like the entries file, it can be edited with a text editor or a spreadsheet application (provided the plain−text format is preserved). Each line consists of the following fields:

1. ID: A numeric field. Must correspond to the appropriate article in the entries file.
2. Keywords: A comma−delimited list of subject keywords appropriate to the article.
3. Description: A description of the article.

Command Line Arguments

- Pass `-A` or `--article` followed by an article ID to generate *only* that article. This is useful when an article is in development and requires review. It is also significantly faster. Only the article matching the ID is is generated. Index, archive, and tagged−with files are *not* updated in this mode. Note Article ID *must* be contained within quotes.
- Pass `-G` or `--groff` in article mode to output GNU roff (groff) code (in addition to HTML) for the specified article. This mode is reliant upon Pandoc being installed and in the system path. On Linux systems Hua automatically produces a PDF provided pdfroff (included with groff) is in the system path.
- Pass `-O` or `--orphans` as the second argument to list all files (irrespective of type) in the `output_dir` that are not referenced in the `entries_file` (typically *.html and *.md). If there are no orphans, nothing is output. Content is *not* processed in this mode.
- Pass `-V` or `--version` as the first argument to show Hua's version information.

Using Hua

Basic Usage

On Windows Hua is run by by specifying the Hua script and configuration file. In the example below it is assumed the user has unzipped and changed to the Hua directory:

```
>pwsh hua.ps1 example.cfg
```

The same in a PowerShell console:

```
$ .\hua.ps1 example.cfg
```

Hua can be run in a Cygwin console as well. Note that PowerShell 7.4 (or above) should be used.

```
$ pwsh hua.ps1 example.cfg
```

On Linux, with the archive expanded to `/usr/local`, the sample content can be used to generate a blog with the following commands:

```
$ mkdir temp
$ cd temp
$ cp /usr/local/share/hua/example.cfg .
$ cp -r /usr/local/share/hua/content .
$ mkdir blog
$ hua example.cfg
```

Additional Examples

Article mode (only article ID 00041 is processed):

```
$ .\hua.ps1 example.cfg -A '00041'
```

Article and groff mode (article ID 00041 HTML and groff is output):

```
$ .\hua.ps1 example.cfg -A '00041' -G
```

Converting groff to PDF:

```
$ pdfroff -ms -mpdfmark -mspdf article.ms > article.pdf
```

Orphans mode:

```
$ .\hua.ps1 example.cfg -O
another-orphan.html
orphan-1.html
```

Version information:

```
$ .\hua.ps1 -V
Hua version: $Revision: 287 $
```

Errors

Hua is normally silent, so if everything goes well, nothing is output on the command line and Hua exits with a status of `0`. Errors are output in the following situations:

- No configuration file is specified. Hua stops. Exit status of `1`.
- The configuration file can't be found. Hua stops. Exit status of `2`.
- A configuration variable doesn't exist in the configuration file. Hua stops. Exit status of `3`.
- The directory specified for the log file doesn't exist. Hua stops. Exit status of `4`.
- The entries file doesn't exist. Hua stops. Exit status of `5`.
- The entries file doesn't contain any articles. Hua stops. Exit status of `6`.
- Duplicate IDs in the entries file. Hua stops. Exit status of `7`.
- Filename conflicts in the entries file (e.g., duplicates or article.html and article.md). Hua stops. Exit status of `8`.
- Filename conflicts in the content directory (e.g., article.html and article.md). Hua stops. Exit status of `8`.
- The meta info file doesn't exist. Hua stops. Exit status of `9`.
- The header include doesn't exist or is empty. Hua stops. Exit status of `10`.
- The footer include doesn't exist or is empty. Hua stops. Exit status of `11`.
- The output directory doesn't exist or isn't a directory. Hua stops. Exit status of `12`.
- Unknown mode passed as second argument. Hua stops. Exit status of `13`.
- The ID specified in article mode does not exist in the entries file. Hua stops. Exit status of `14`.
- The content directory doesn't exist or isn't a directory. Hua stops. Exit status of `15`.
- Orphans mode and the output directory doesn't exist or isn't a directory. Hua stops. Exit status of `16`.
- Content directory and output directory specified as the same in configuration file. Hua stops. Exit status of `17`.
- The read more include doesn't exist or is empty. Hua continues.

- The comments include doesn't exist or is empty. Hua continues.
- An article content file doesn't exist. Hua continues to the next article.
- The ID field of an entry in the entries file is empty. Hua continues to the next article.
- The file field of an entry in the entries file is empty. Hua continues to the next article.
- An article file exists but contains no content. Hua continues to the next article.
- A markdown file is referenced in the entries file, but MarkdownToHTML is not installed. Hua continues to the next article.
- A markdown file is referenced in the entries file, but the markdown template cannot by found. Hua continues to the next article.
- Groff output mode is specified, but Pandoc is not installed. Hua produces specified article HTML but groff is not output.

Known Issues & Future Enhancements

There are no known issues at this time.

Related Articles

All articles tagged with Hua on the blog.