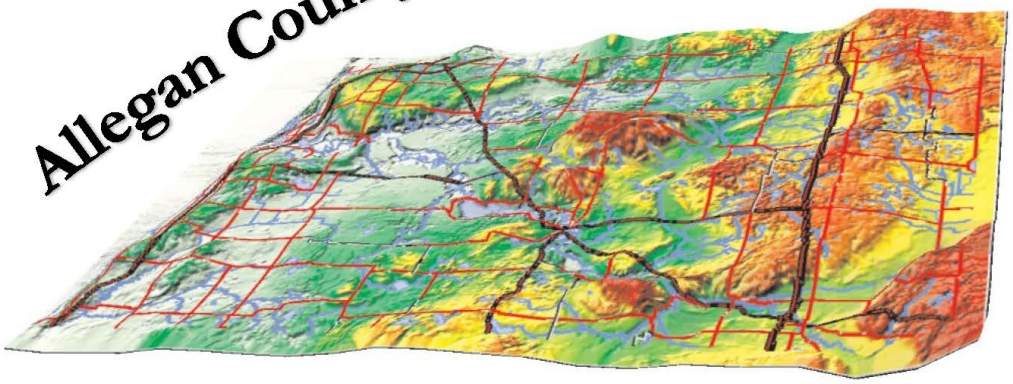
Allegan County GIS Services



How The Jalapeño Works

WWW.ALLEGANCOUNTY.ORG/GIS

JULY 24, 2019

Contents

0.0.1 How Jalapeño Works

Problem and Analysis

Background

GIS Services has complicated and evolving workflows and uses everchanging technologies

Statement of Problem

GIS documentation has traditionally been done in different formats and stored in many different files and folders in the county network. This has resulted in problems with:

- version control
- finding the documentation
- disseminating the documentation

Analysis

The Jalapeño folder along with some open-source software provides a robust documentaion tool for GIS documentation.

Default sizes in Jalapeño

Element	Default Size
Paragraph Heading	Large
Paragraph text	normalsize
Subparagraph Heading	large
Subparagraph Text	normalsize

Table 0.1: Default Sizes

Examples:

Schema Change Procedure large size

large size type

Schema Change Procedure Default size

default size type

Schema Change Procedure Large size

Large size type space neg pouint 3in here

Schema Change Procedure Large size

LARGE size type

Schema Change Procedure Default size

default size type

Schema Change Procedure large size

large size type

Schema Change Procedure Large size

Large size type

Schema Change Procedure LARGE size

LARGE size type

Colors

Blues

HeaderBlueA
HeaderBlueB
HeaderBlueC
HeaderBlueD
HeaderBlueE

Golds

HeaderGoldA
HeaderGoldB
HeaderGoldC
HeaderGoldD
HeaderGoldE

Oranges

HeaderOrangeA
HeaderOrangeB
HeaderOrangeC
HeaderOrangeD
HeaderOrangeE

Greens

HeaderGreenA
HeaderGreenB
HeaderGreenC
HeaderGreenD
HeaderGreenE

Others

HyperlinkBlue1
graphicOrange

Project Notes:

- jalapeno folder is a git package https://github.com/nbesteman/jalapeno
- Project is coded with relative paths and jalapeno can be located anywhere.

Project File Structure:

...\jalapeno\..

folder	description
.git	versioning repository for Jalapeño
documentation	resources used in Jalapeño
processing	.tex douments and build folders
source	common image files
temp	untracked folder for temp storage

...\jalapeno\documentation\..

folder or file	description
classDocs	LaTeX class documentation
DevNotes	Notes and Mind Maps for Jalapeño
latexamples	LaTeX example code
moduleTemplates	.tex templates
packageDocs	LaTeX package documentation
readingRoom	Resources linked in Jalapeño
unsorted	Unsorted documentation
gitnotes.txt	git commands notes

...\jalapeno\processing\..

folder or file	description
archive	Processing backup folder
...Part	Folders of book *parts*
build	LaTeXfolder for .pdf output and temp files
build\referenceEntries.bib	Entries that appear in references
preamble.tex	preamble code for all documents
titlePages	Assortment of .tex title pages
compileFull.sh	pdflatex, bibtex, makeglossaries, makeindex, pdflatex, pdflatex
compileMainX2.sh	pdflatex, pdflatex
GISDocumentation.tex	Master document code
glossaryEntries.tex	Entries that appear in glossary
indexEntries.tex	Entries that appear in the index

...\jalapeno\processing\preamble..

folder or file	description
chapterStyles.tex	Sets chapter title page attributes with Memoir Class
colorDefs.tex	Defines custom colors
graphicsPath.tex	Defines graphics variable
pageLayoutCommands.tex	Sets spacing and typeface for headings of Sections down to Subparagraphs in mainmatter
pageLayoutCommandsAlt.tex	Sets spacing and typeface for headings of Sections down to Subparagraphs in backmatter
pageStyles.tex	Sets header and footer properties
preamble.tex	Preamble used to compile main document
subSectionPreamble.tex	Preamble used to compile any subsection document

USING THE GLOSSARY

Glossary Requirements

Glossary commands require a Perl interpreter. Activeperl is a free Perl interpreter and can be downloaded from:
https://www.activestate.com/activeperl/downloads (A typical installation adds Perl to your path). Compiling the glossary requires running the makeglossaries command either in a LaTeX IDE or in command line as described here. PDFLatex must be run first to create a .aux file that is used by makeglossaries to create an .gls file. After the .gls file is created, PDFLatex must be run again to insert the glossary at the \printglossaries location.

Creating a new glossary entry

To **create a new glossary entry:** Add an entry to glossaryEntries.tex. Save it there and then use the makeglossaries command to recompile the .gls file.

Rebuilding the glossary

To Recompile the .gls. In the (main document)build folder:

- Launch command prompt
- enter command: **makeglossaries GISDocumentation***

Note that this command reads the .aux file and creates the .gls file. The .aux file is created by compiling with PDFLatex. If there is no .aux file the command will fail

Using glossary terms in a subdocument:

In the subdocument you must add code to input the glossaryEntries file.
ie. After the line:

```
\input{../../../preamble}
```

Add the line:

```
\input{../../../glossaryEntries}
```

To use a glossary term in the subdocument:

In place of the term, use code referencing the key (in the glossaryEntries file):

- \gls{key}

To add the glossary to the subdocument:

- Add the line \makeglossaries to the preamble of the subdocument.
- Add the line \printglossaries to the subdocument.
- Run makeglossaries in command line on the subdocument similar to how is described above.

USING THE BIBLIOGRAPHY (REFERENCES)

Bibliography requirements

Compiling the bibliography requires running bibtex either in a LaTeX IDE or in command line as described here. PDFLatex must be run first to create a .aux file that is used by bibtex to create a .bbl file. After the .bbl file is created, PDFLatex must be run again to insert the bibliography at the \bibliography location.

Inserting the bibliography

In the LaTeX code:

```
\bibliography\{referenceEntries}
```

Inserts a bibliography called referenceEntries.bib from the same folder as the project .aux file

Creating a new bibliography entry

To **create a new bibliography entry:** Add an entry to referenceEntries.bib. Save it there and then use bibtex to recompile the .bbl file.

Rebuilding the bibliography

To Recompile the .bbl. In the (main document)build folder:

- Launch command prompt
- Enter command: **bibtex GISDocumentation**

Note that this command reads the .aux file and creates the .bbl file. The .aux file is created by compiling with PDFLatex. If there is no .aux file the command will fail

To cite a bibliography source in a subdocument

In the place that you want the citation In the LaTeX code:

```
~\cite[pg.#]{key}
```

USING THE INDEX

Index requirements:

Compiling the index requires running the makeindex command either in a LaTeX IDE or in command line as described here. PDFLatex must be run first to create a .aux file that is used by makeindex to create an .idx file. After the .idx file is created, PDFLatex must be run again to insert the index at the \printindex location.

Creating a new index entry

To **create a new index entry:** Add an entry to indexEntries.tex. Save it there and then use the makeindex command to recompile the .idx file.

Rebuilding the index

To Recompile the .idx In the (main document)build folder:

- Launch command prompt
- enter command: **makeindex GISDocumentation***

Note that this command reads the .aux file and creates the .idx file. The .aux file is created by compiling with PDFLatex. If there is no .aux file the command will fail. Run PDFLatex first

Access the index from a subdocument

In the subdocument you must add code to input the indexEntries file. For example: After the line:

```
\input{../../../preamble}
```

Add the line:

```
\input{../../../indexEntries}
```

To use a index term in the subdocument:

In place of the term, use code referencing the key (in the indexEntries file):

- \index {key}

To add the index to the subdocument:

- Add the line \makeindex to the preamble of the subdocument.
- Add the line \printindex to the subdocument.
- Run makeindex in command line on the subdocument similar to how is described above.

USING THE APPENDICES