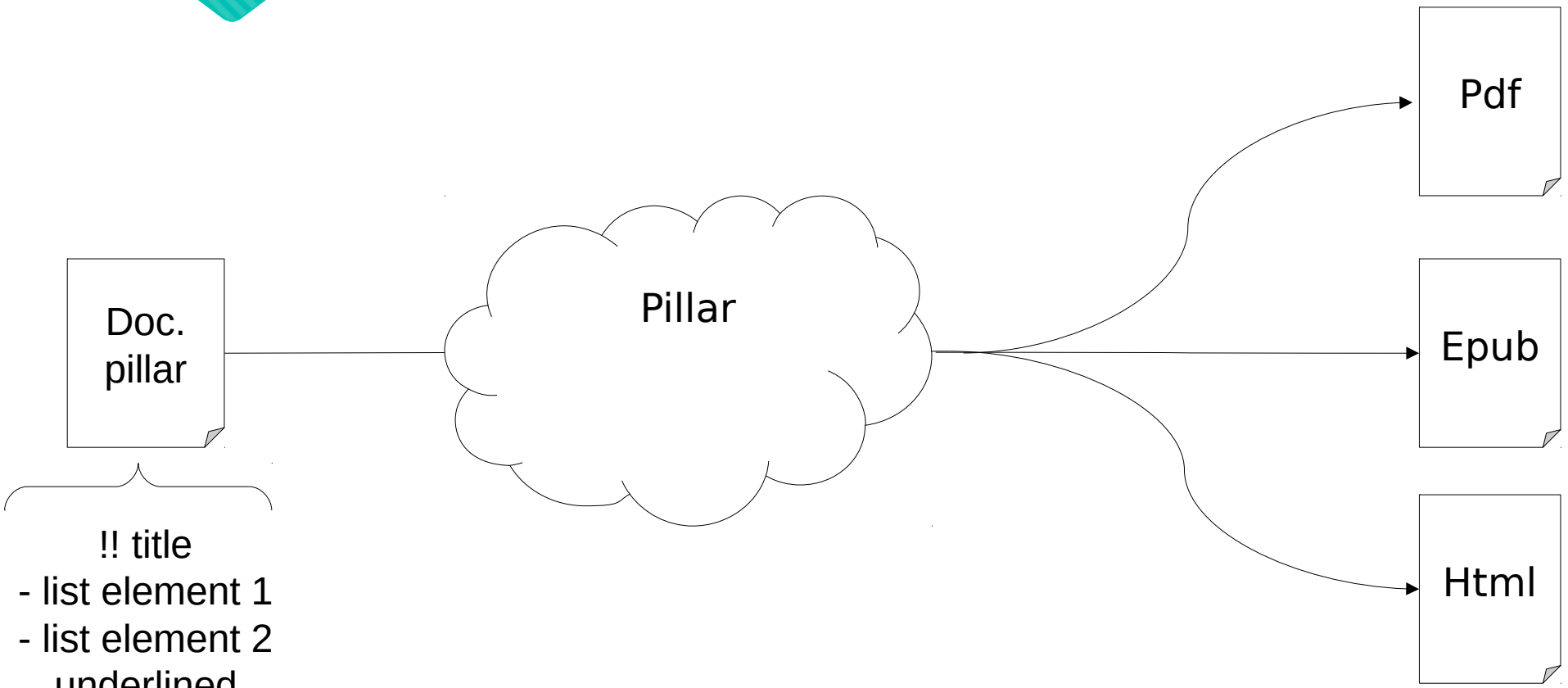
IMPROVING PILLAR GENERATION PROCESSES

Biyalou-Sama Asbathou

Plan

- Context
- Problems
- Solutions
- Next Steps

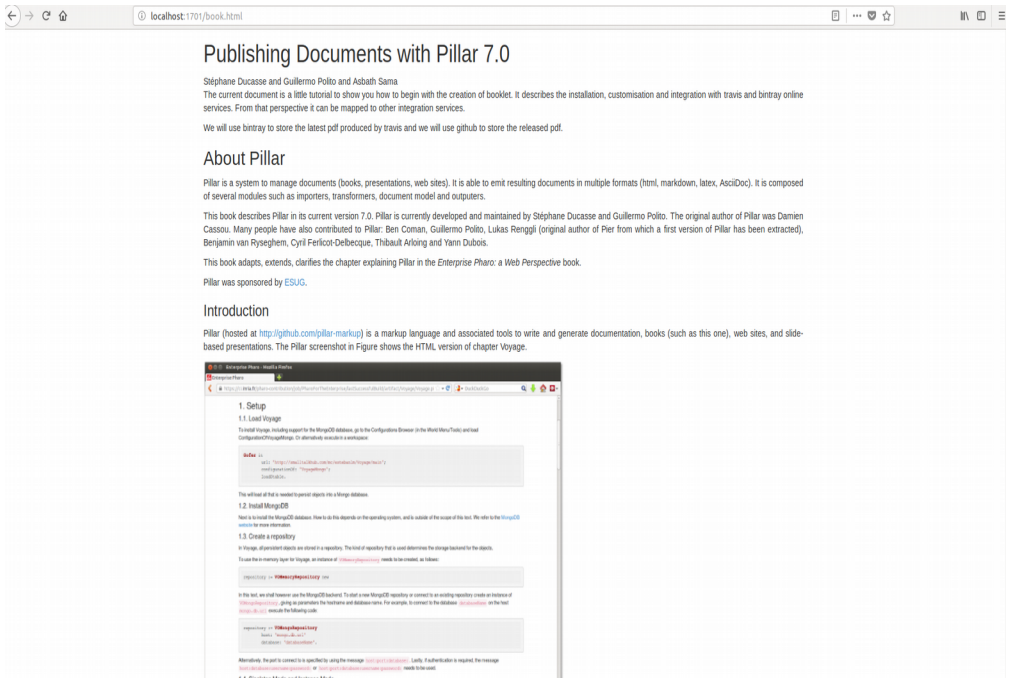
Context



!! title
- list element 1
- list element 2
__underlined__
@Internal Link
${inputFile:path=MonRapport.pillar}$

Problems

- Html templates were not in good state
- No Easy Hosting
- Compilation was not efficient
- Errors Handling and Reporting were not good



My Main Contributions

- Propose new archetypes for books and personal websites
- Implement a pillar module to generate table of contents
- Automatic deploy platform using Continuous Integration
- Blogging / News list components

A new archetype for Html

- Get some existing templates from another template engine
- Adapt templates to the template engine of Pillar
- Manage support files (css, img, js ...) to have an archetype

New Book Archetype Example

Pharo Book

PUBLISHING DOCUMENTS WITH PILLAR 7.0

The current document is a little tutorial to show you how to begin with the creation of booklet. It describes the installation, customisation and integration with travis and bintray online services. From that perspective it can be mapped to other integration services.

We will use bintray to store the latest pdf produced by travis and we will use github to store the released pdf.

About Pillar

Pillar is a system to manage documents (books, presentations, web sites). It is able to emit resulting documents in multiple formats (html, markdown, latex, AsciiDoc). It is composed of several modules such as importers, transformers, document model and outputers.

This book describes Pillar in its current version 7.0. Pillar is currently developed and maintained by Stéphane Ducasse and Guillermo Polito. The original author of Pillar was Damien Cassou. Many people have also contributed to Pillar: Ben Coman, Guillermo Polito, Lukas Renggli (original author of Pier from which a first version of Pillar has been extracted), Benjamin van Ryseghem, Cyril Ferlicot-Delbecque, Thibault Arloing and Yann Dubois.

This book adapts, extends, clarifies the chapter explaining Pillar in the *Enterprise Pharo: a Web Perspective* book.

Pillar was sponsored by ESUG.

Introduction

Pillar (hosted at http://github.com/pillar-markup) is a markup language and associated tools to write and generate documentation, books (such as this one), web sites, and slide-based presentations. The Pillar screenshot in Figure shows the HTML version of chapter Voyage.

Personal Sites Archetype Example

GUILLERMO POLITO

About me | Projects | Research Activities | Publications | Coordinates

Guillermo Polito

Mastering objects

About me

Coding enthusiast. Mad hatter. Software engineer and researcher.

I am currently working as a post-doc researcher at Vrije Universiteit Brussel, on security for programming languages in cloud computing applications.

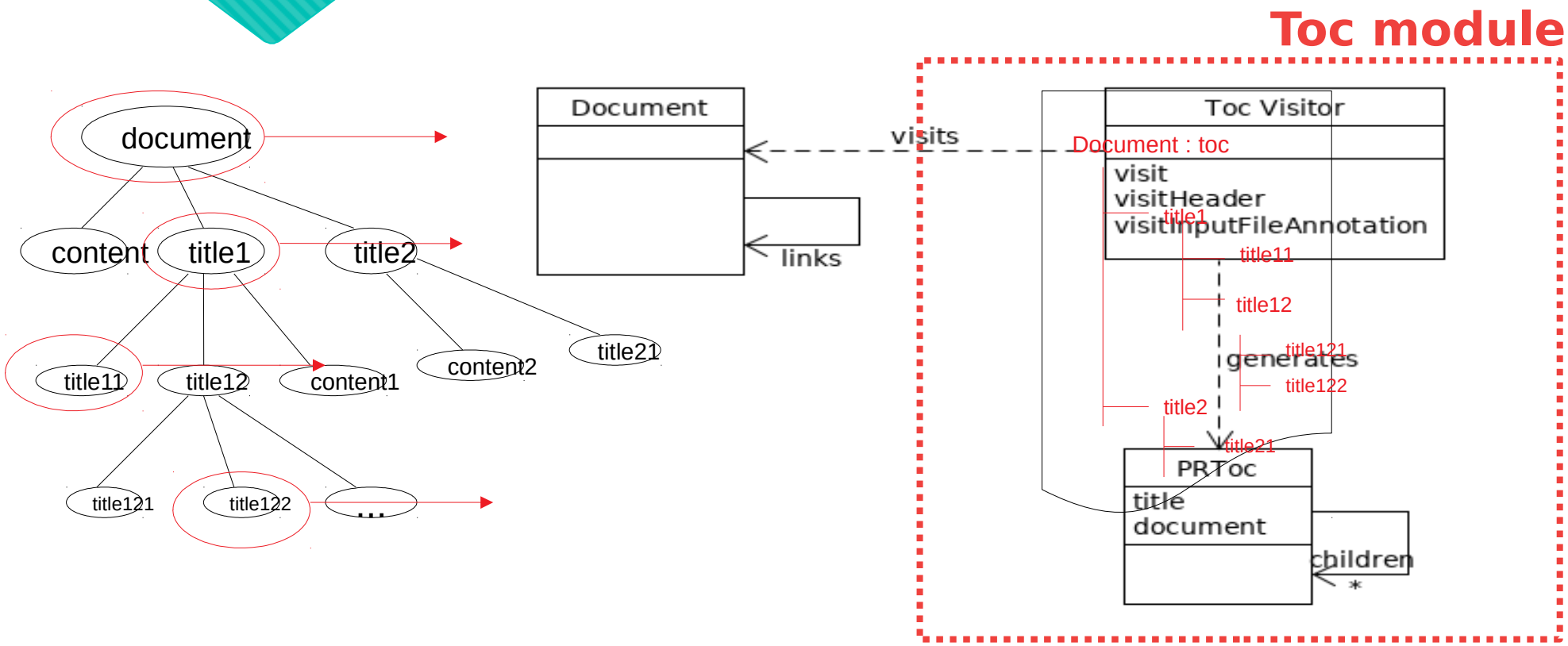
My main topics of interests are the runtime adaptation of applications, Virtual Machine design and development, and programming language tools. For that, I work on reflective applications and programming languages. I participate in the open source community of the Pharo programming language and environment.

I obtained the 13 of april 2015 my Phd entitled "Virtualization support for application runtime specialization and extension" under the direction of Stéphane Ducasse (Inria Rmod team) and the supervision of Noury Bouraqadi and Luc Fabresse(CAR team of Mines Douai).

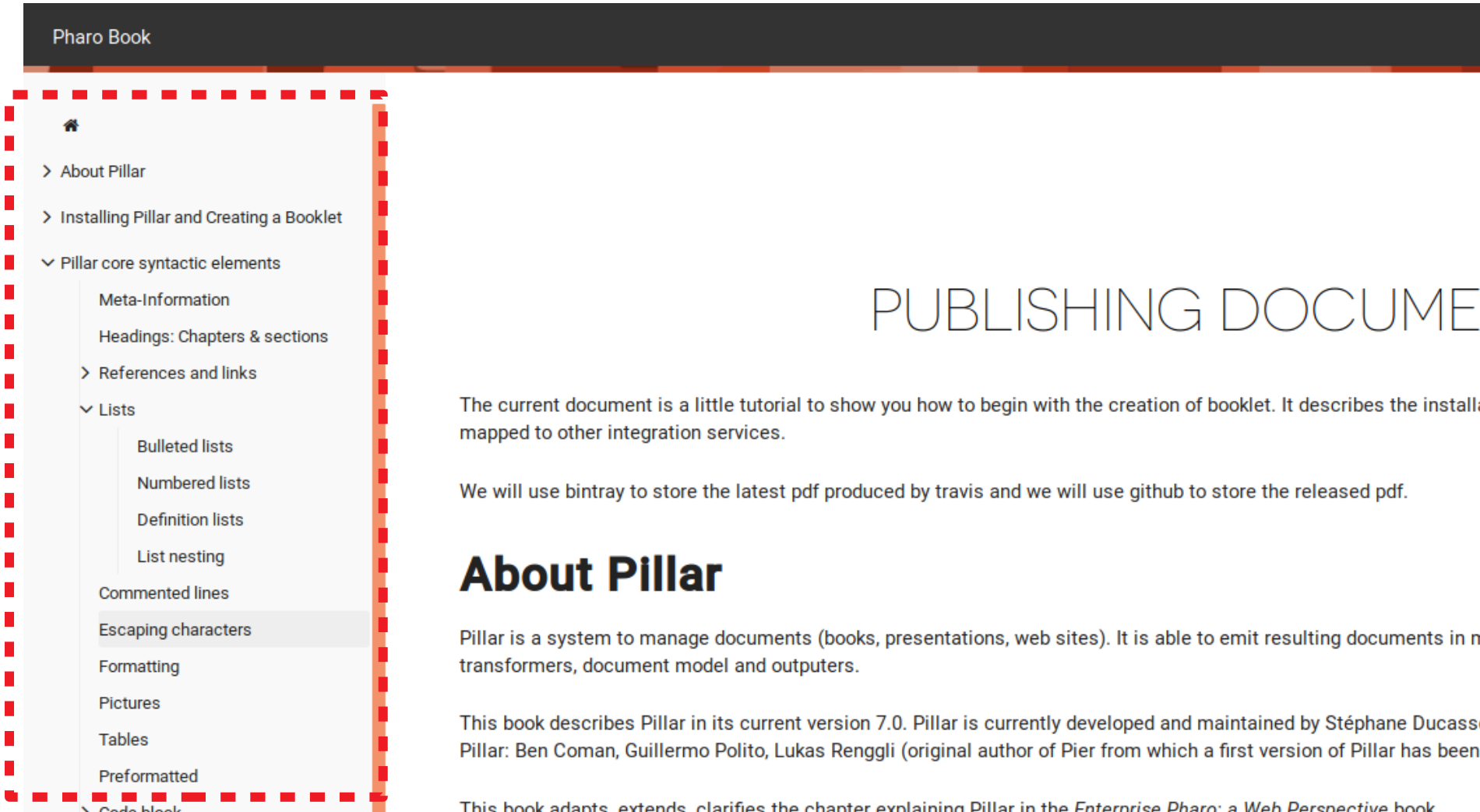
I love coding. I spend a lot of my free time helping the amazing community of Pharo. I also participate in several projects such as the Pharo's database driver suite (DBXTalk), it's shortcut framework, or the static web page generator Ecstatic.

Awards

Pillar Toc module



Rendering the TOC



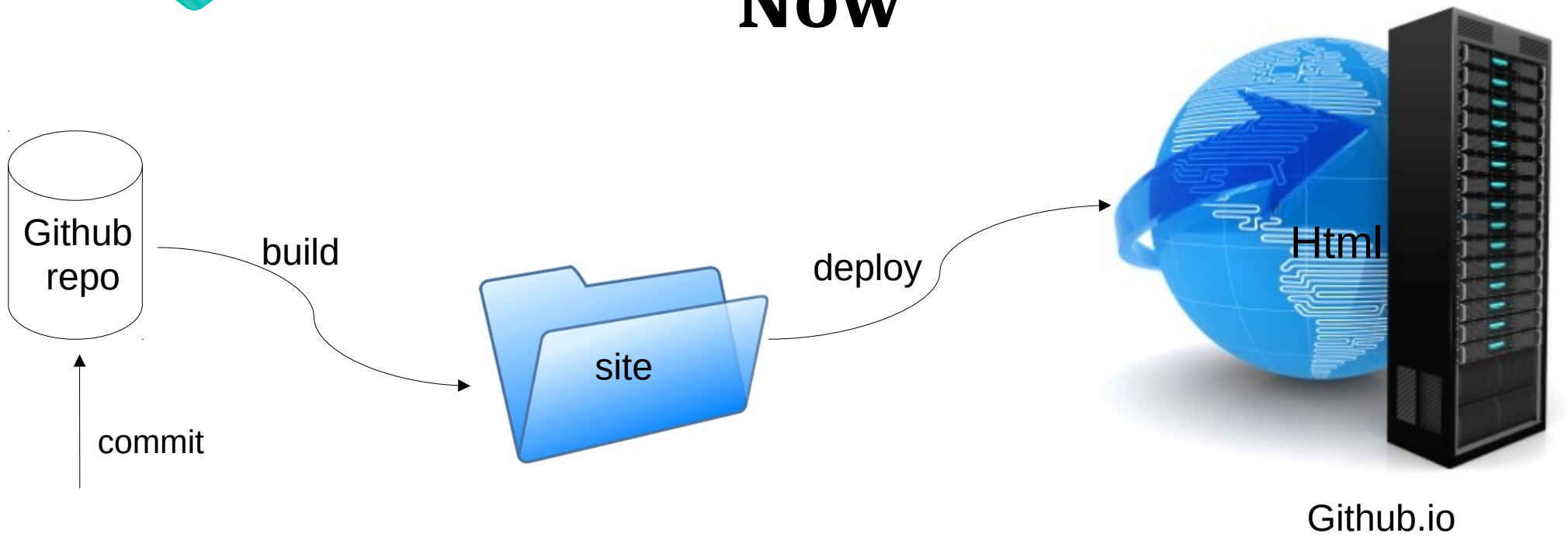
Automatic Deploy : Github + Travis

Before

TRAVIS CONTINUOUS INTEGRATION

Automatic Deploy : Github + Travis

Now



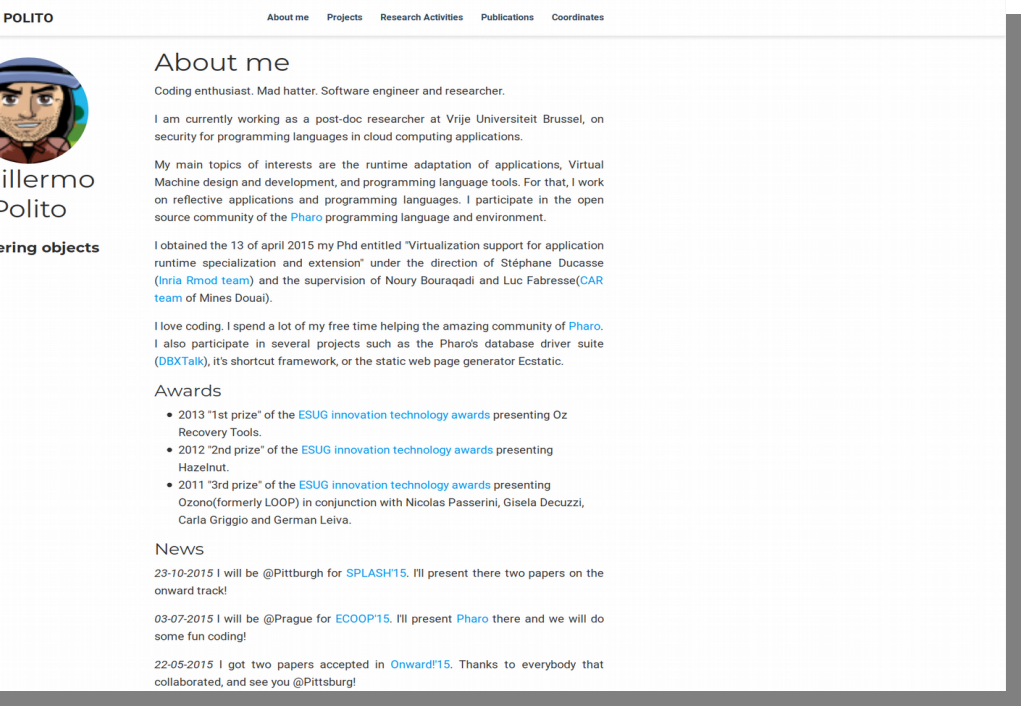
TRAVIS CONTINUOUS INTEGRATION

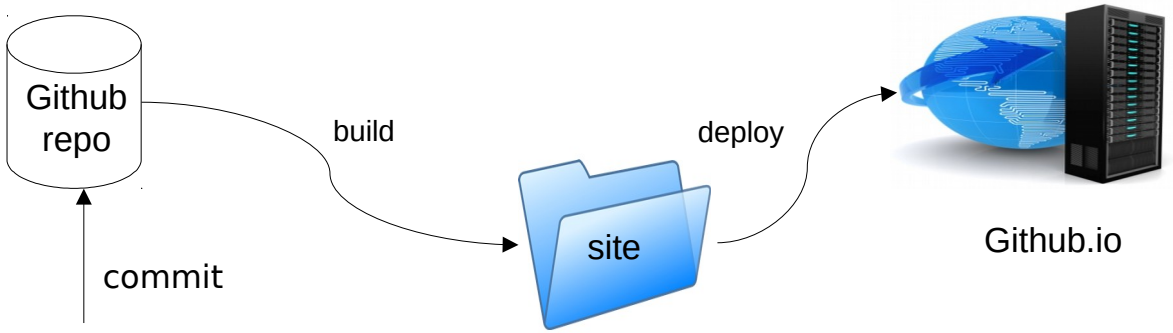
Next Steps

${docList:path=blogs|limit=5|sort=date}$

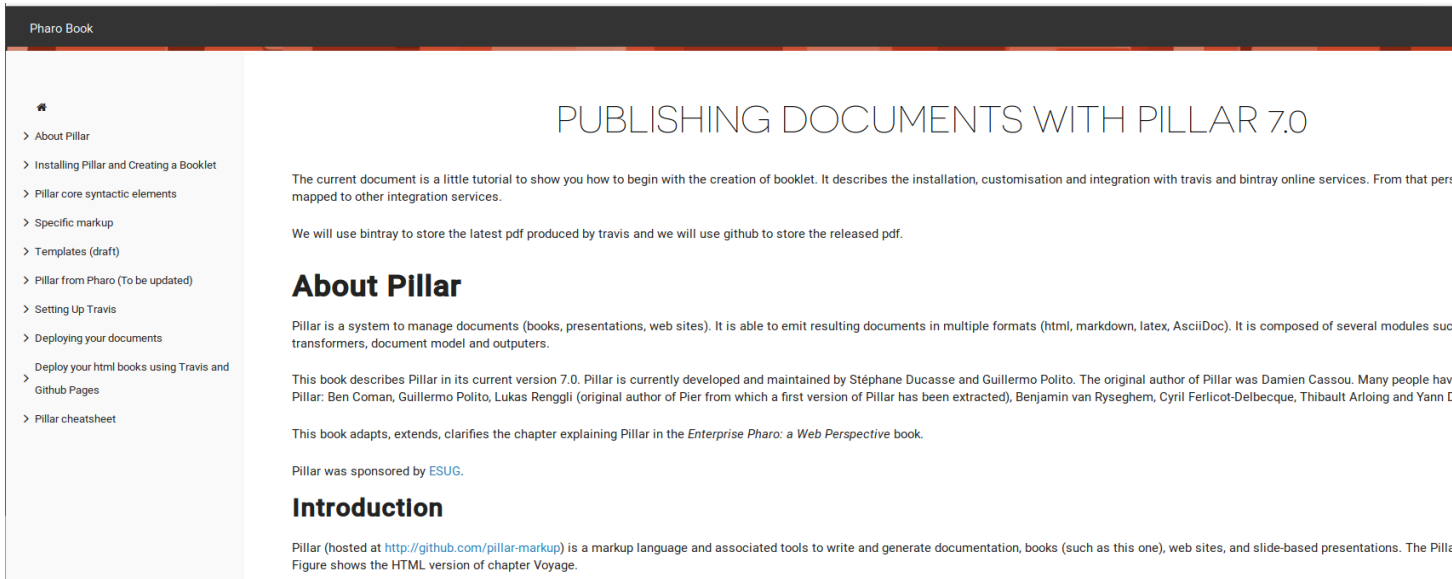
- Blogging Plugin for personal sites
- New system of compilation
- Review the way warnings and errors are reported

Conclusion



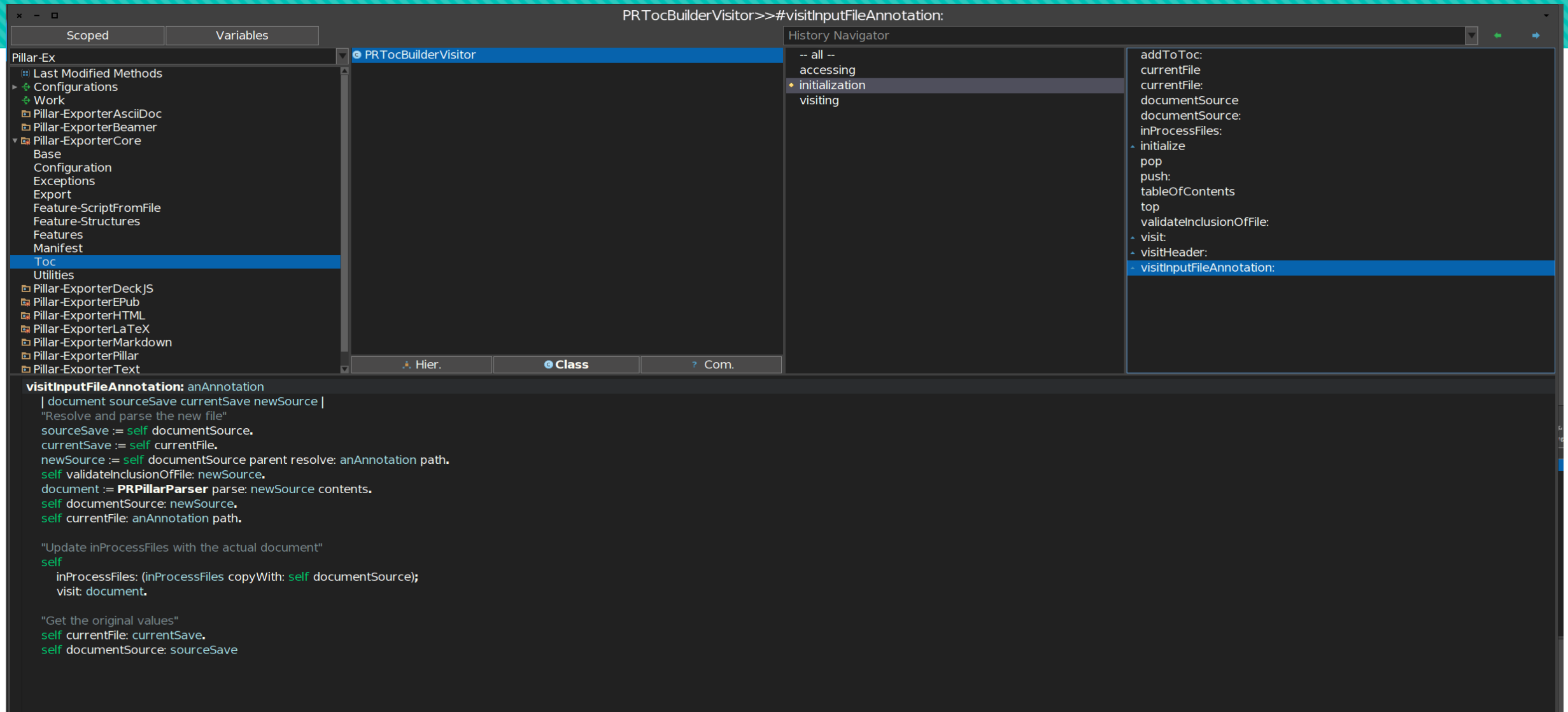


TRAVIS CONTINUOUS INTEGRATION



Demo

Toc Visit


PRTocBuilderVisitor>>#visitInputFileAnnotation:
Scoped
Variables
History Navigator
Pillar-Ex
Last Modified Methods
Configurations
Work
Pillar-ExporterAsciiDoc
Pillar-ExporterBeamer
Pillar-ExporterCore
Base
Configuration
Exceptions
Export
Feature-ScriptFromFile
Feature-Structures
Features
Manifest
Toc
Utilities
Pillar-ExporterDeckJS
Pillar-ExporterEPub
Pillar-ExporterHTML
Pillar-ExporterLaTeX
Pillar-ExporterMarkdown
Pillar-ExporterPillar
Pillar-ExporterText
PRTocBuilderVisitor
Hier.
Class
Com.
-- all --
accessing
initialization
visiting
addToToc:
currentFile
currentFile:
documentSource
documentSource:
inProcessFiles:
initialize
pop
push:
tableOfContents
top
validateInclusionOfFile:
visit:
visitHeader:
visitInputFileAnnotation:
visitInputFileAnnotation: anAnnotation
| document sourceSave currentSave newSource |
"Resolve and parse the new file"
sourceSave := self documentSource.
currentSave := self currentFile.
newSource := self documentSource parent resolve: anAnnotation path.
self validateInclusionOfFile: newSource.
document := PRPillarParser parse: newSource contents.
self documentSource: newSource.
self currentFile: anAnnotation path.
"Update inProcessFiles with the actual document"
self
inProcessFiles: (inProcessFiles copyWith: self documentSource);
visit: document.
"Get the original values"
self currentFile: currentSave.
self documentSource: sourceSave

Toc Tests

- actualClass
- createCyclicPillarFile1
- createPillarFile1
- createPillarFile2
- createPillarFile3
- setUp
- tearDown
- testDocumentWithCyclicInputShouldRaiseError
- testDocumentWithInputAnnotationAndTitle
- testDocumentWithInputAnnotationContainsMultipleTocs
- testDocumentWithInputAnnotationHasTocs
- testDocumentWithOnlyInputAnnotation
- testDocumentWithoutInputAnnotationOnlyContainsTitles
- testTitlesHaveGoodFileRefs
- testTocChildrenHasTheSameLevel
- testTocHasGoodLevelOfChildren
- testTocHasOneChildren
- testTocIsEmpty
- testTocOnlyContainsTocs

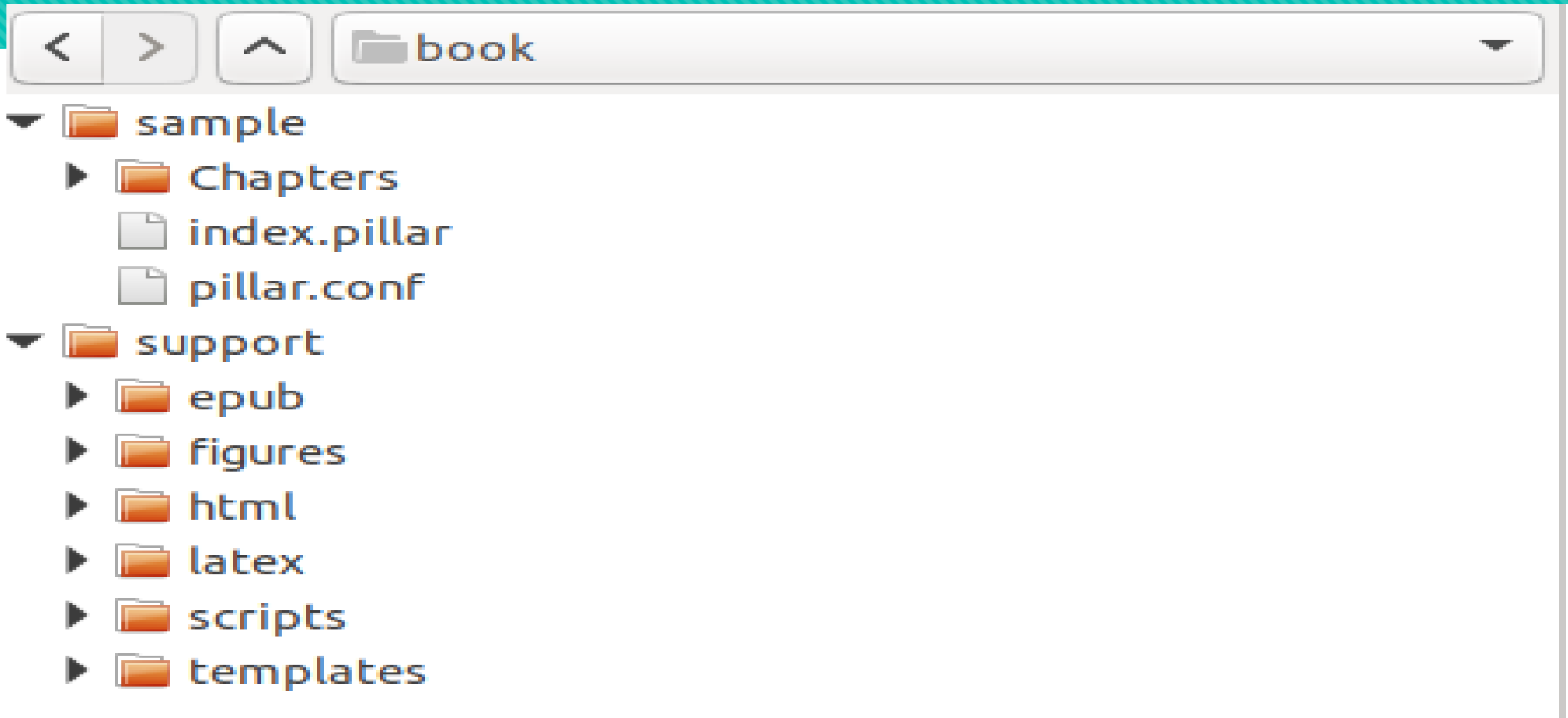
Travis config

```
language: smalltalk


os:
     - linux


smalltalk:
     - Pharo-6.1

env:
  matrix:
     - JOB=test TYPE=System ARCHETYPE=book OUTPUT=html

install:
     # Pillar installation
     - git clone https://github.com/pillar-markup/pillar.git .pillar -b dev-7  # Clone pillar
     - cd .pillar && ./scripts/build.sh && cd ..

script:
     - .pillar/build/pillar build html
     - cd _result/html && touch .nojekyll && cd ..

deploy:
    provider: pages
    skip-cleanup: true
    github-token: $GH_TOKEN
    keep-history: true
    local_dir: _result/html
    on:
```

Mustache template New archetype

```
</header>
<article>
  <aside>
    <ul class="menu">
                <li data-nav-id="123" class="dd-item">
                <a href="{{base_url}}">
                        <i class="fa fa-fw fa-home"></i>
                </a>
        </li>

                {{# toc.children }}
                  <li class="dd-item {{# hasChildren}} haschildren {{/ hasChildren}}">
                        <div>
                          <a href="{{base_url}}{{href}}">
                                        {{name}}
                          </a>
                          {{# hasChildren}} <i class="fa fa-angle-right fa-lg category-icon"></i> {{/ hasChildren}}
                        </div>
                          <ul class="dd-item">
                                {{# children }}
                                <li class="dd-item {{# hasChildren}} haschildren {{/ hasChildren}}">
                                        <div>
                                                <a href="{{base_url}}{{href}}">
                                                        {{name}}
                                                </a>
                                                {{# hasChildren}}<i class="fa fa-angle-right fa-lg category-icon"></i>{{/ hasChildren}}
                                        </div>

                                        <ul class="dd-item">
                                        {{# children }}
                                                <li class="dd-item">
                                                        <a href="{{base_url}}{{href}}">{{name}}</a>
                                                </li>
                                        {{/ children }}
                                        </ul>
                                </li>
                                {{/ children }}
                          </ul>

                  </li>
                {{/ toc.children }}
    </ul>
  </aside>

  <section class="page">

        <h1>{{title}}</h1>

        {{{content}}}
```

Archetype architecture



Pillar syntax

Headers

```
!Header 1
!!Header 2
!!!Header 3
```

Special Paragraphs

Annotation	@@note this is a note
Todo	@@todo this is todo

Lists

- Unordered List	# Ordered List
; Description Term	: Description Definition

Text Formats

```
""bold""
''italic''
--strikethrough--
__underscore__
==monospace==
@@subscript@@
```

Comment

```
% each line starting with % is ignored
```

Tables

|| Centered Cell	|!| Centered Title
|{ Left-Aligned Cell	|} Right-Aligned Cell

Links

Anchor	@anchor
Internal Link	*@anchor*
External Link	*Google>www.google.fr*

Figures

```
+Caption>file://pic.png|width=50|label=fig+
```

Scripts

```
[[[label=hello|language=SmallTalk
Transcript show: 'Hello World'.
]]]
```

Raw

```
{{{latex:
injects raw \LaTeX in your output file
}}}
{{{html:
injects raw <b>html</b> in your output file
}}}
```