

Validating Books: a Rule-based Approach with Microdown

**S. Ducasse, T. Ramarozaka
and F. Randriamihajarivo**



IWST 2026



Outline

- Challenges of producing books
- Looking at an author guide
- Valiboky
- Rule illustration
- Tooling

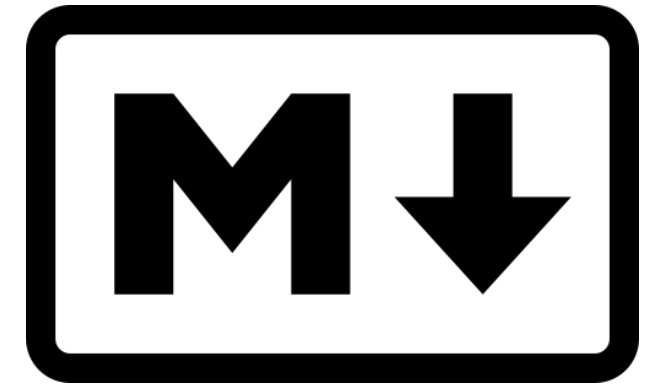
Challenges of book production

- Why this figure is not referenced?
- “oh this example does not work?”
- “oh this example was working on the previous version and now it does not anymore :(“

Author guide

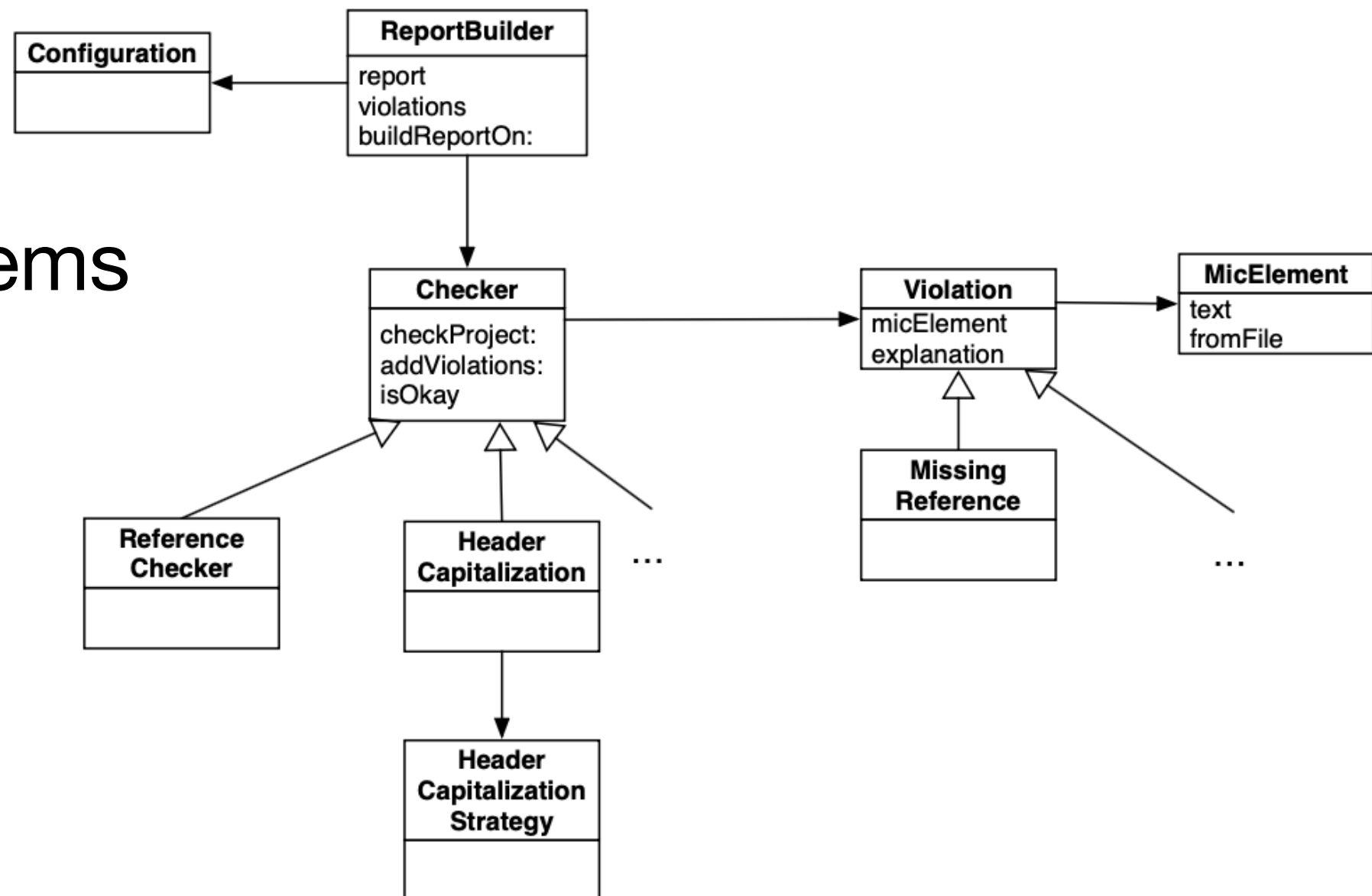
- * Use American English, e.g. "colors", not "colours". Pharo's class names and method selectors are written in American English too.
- * Write "Spec 2.0" only when needed, e.g. when comparing to Spec 1.0. If there is no ambiguity, write "Spec".
- * Use the Pharo Light theme to make screenshots.
- * Write "backend", not "back-end" or "back end".
- * Write "subpresenter", not "sub-presenter".
- * Methods without comment have an empty line after the method selector.
- * Methods do not have a period on the last line.
- * Do not put a space before a semicolon in a cascaded message.
- * Write "Pharo image", not "Pharo Image".
- * Do not use protocol references because they are not useful and may change.
- * Write "Section 6.1", not "section 6.1".
- * Write "Figure 6-2", not "figure 6-2".
- * Do not use a prefix "Sp" for example class names because the prefix is reserved for Spec classes.
- * Use sentence case for headers, e.g. "Editable tables", not "Editable Tables".
- * Use sentence case for chapter titles, e.g. "Testing Spec applications", not "Testing Spec Applications".
- * Start a note with "**Note**".
- * I love musicians such as Metallica (it means that I do), I love musicians like Metallica (it does not mean I like Metallica).
- * For method 'SpPresenter>>methodName' and not 'SpPresenter>>#methodName'
- * Do not write a space before a colon, e.g. "the following code:", not "the following code :". In French one would write the space, but not in English.
- * Use tabs to indent code.

Valiboky

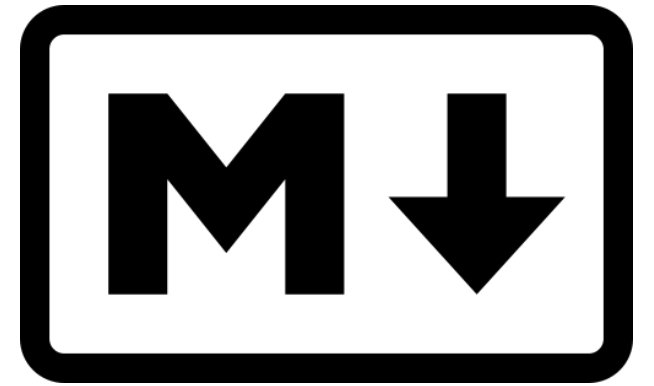


Valiboky

- input books in Microdown
- applies rules
- reports problems



Valiboky basics



- Only works on reachable files
- Authors can have broken draft files
- Report broken files
- Configuration of rules
- Start to handle false positives

6 Rule kinds

- Typography
- Vocabulary
- Code display
- Infrastructure
- Semantics
- Evolution

Overview of rules

Rule name	Kind	Description
ENGLISH TYPOGRAPHY	typography	Check that punctuation follows English typography: in English, there is no space before diacritics (;?!), and there is space after diacritics and (.,). I pay attention that a diacritic can be the last character of a line: in this case, I do not report errors.
FRENCH TYPOGRAPHY	typography	Check that punctuation follows French typography: in French, there is a space before diacritics (;?!), and there is space after diacritics and (.,).
NOTTHEFIGUREX	typography	Check that references to figures, sections, and tables are correctly capitalized before an anchor and that there is no 'the' before.
HEADER CAPITALIZATION	typography	Check that headers follow a consistent capitalization strategy (uppercase or lowercase, including stopword handling).
LAST PERIOD IN CAPTION	typography	Check that caption in figure, math equation or code block either ends with a terminating period or not, depending on my configuration.
LAST PERIOD IN LABELS	typography	Check that paragraph labels consistently ends with a period.
EMPTY CAPTION	typography	I verify that captions are not left empty.
VOCABULARY	vocabulary	I check that certain vocabulary is not used. I'm configured with a list of pairs showing the vocabulary that should not be used and the correct one.
METHOD REFERENCE	code	Check that method references are consistent within a document.
METHOD STRUCTURE	code	Check that methods in code blocks have an empty line after the selector when there is no comment.
CODE INDENTATION	code	Check that code blocks use tabs for indentation instead of spaces.
CODE LAST PERIOD	code	Check that a code block either ends with a terminating period or not.
REFERENCE CHECKS	infrastructure	Check duplicated anchor definitions, missing references
NO SPACE IN FILENAME	infrastructure	Identify files with space in their names.
CODE EXPRESSION TEST	semantics	Check that code block 'evaluation' is correct.
DESYNCHRONIZED CODE	evolution	Verify that synchronizable code blocks are in sync with the system version.

Typographic validation

English/French Typography

Space and diacritics

Validation d'ouvrages :

vs.

Book validation:

Typographic validation

Reference to anchors in English

The Figure 3.1

->

Figure 3.1

Typographic validation

Header capitalisation

Who should read this book?

Finding information in Pharo

vs.

Who Should Read this Book?

Finding Information in Pharo

vs.

Who Should Read This Book?

Finding Infirmination In Pharo

Typographic validation

Consistent last period in captions

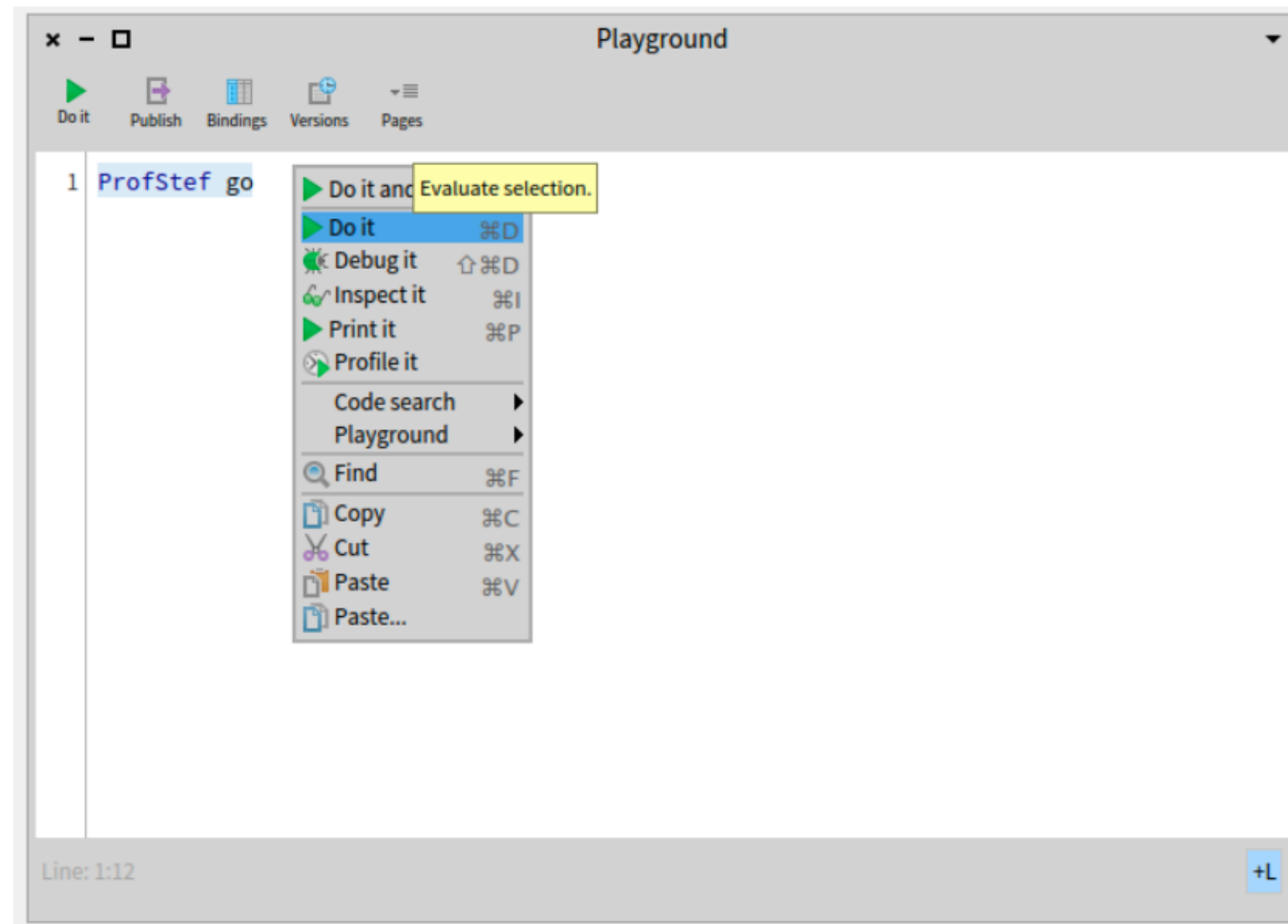


Figure 2-3 Executing an expression is simple with the **Do it** menu item.

Typographic validation

Consistent last period in captions

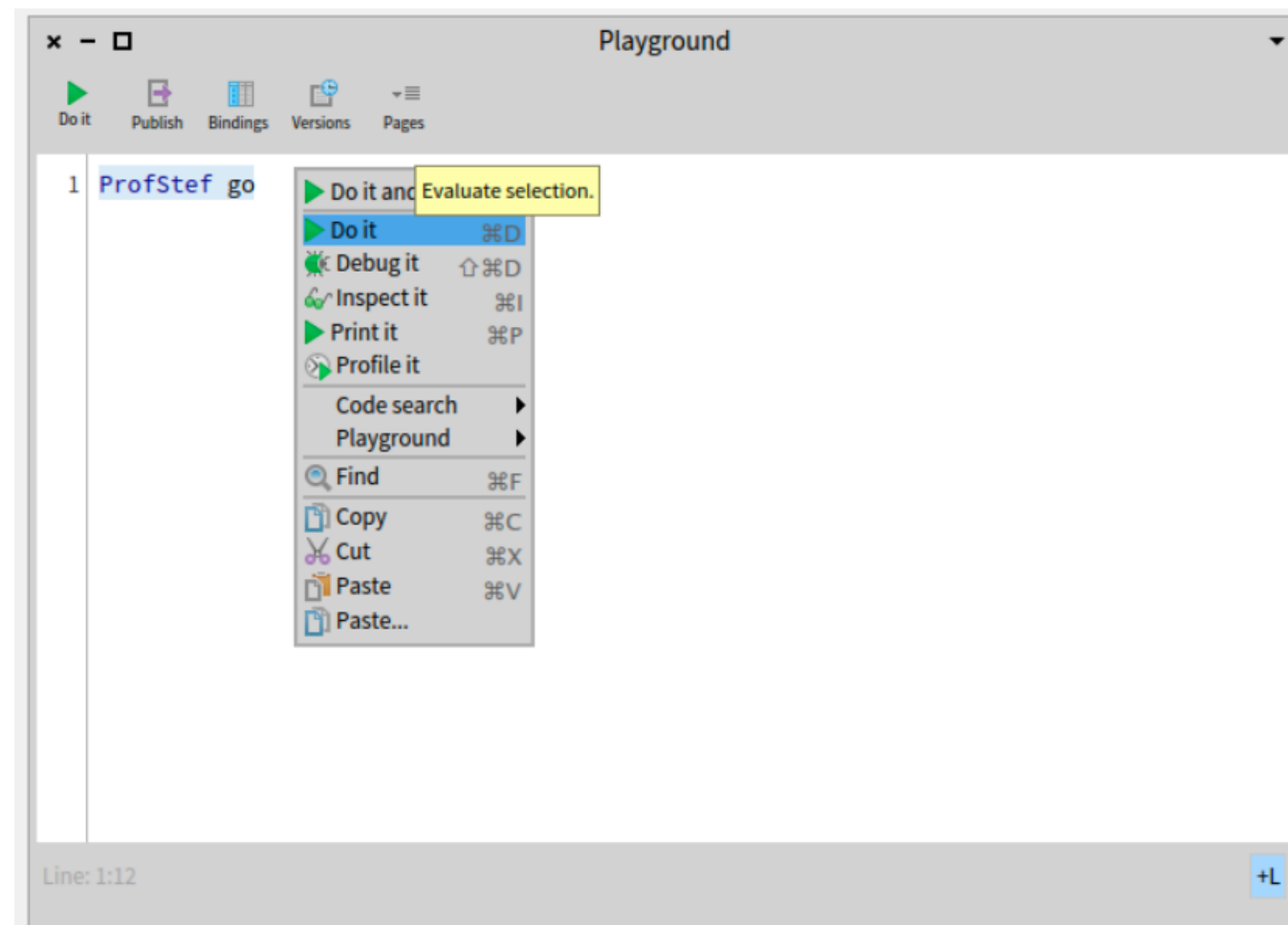


Figure 2-3

Vocabulary validation config

```
"checkerConfigurations" :  
  [  
    "LastPeriodInCaption" : true,  
    "Vocabulary" :  
      [  
        ["sub-presenter", "subpresenter" ],  
        ["sub presenter","subpresenter"],  
        ["meta object","metaobject"]  
      ],  
    "LastPeriodInParagraphLabel" : {  
      "isEndingWithPeriod" : true,  
      "levelForParagraph" : 5  
    }  
  ]
```

Code validation

Method reference

Point >> #degrees vs. Point >> degrees

Code validation

Empty line vs. comments

```
[ ImdbFilmListPresenter >> defaultLayout  
  ^ SpBoxLayout newTopToBottom  
    add: filmList;  
    yourself
```

Code validation

Last period in method

```
[ ImdbFilmListPresenter >> defaultLayout  
  ^ SpBoxLayout newTopToBottom  
    add: filmList;  
    yourself
```

Code validation

Tabs in method

```
[ ImdbFilmListPresenter >> defaultLayout  
  ^ SpBoxLayout newTopToBottom  
    add: filmList;  
    yourself
```

Infrastructure validation

Duplicated reference definitions

![A layout.

%width=60&anchor=LayoutPresenters\figures/
FilmList-01-LayoutInititalizePresenters2.png)

![Another layout.

%width=60&anchor=LayoutPresenters\figures/
FilmList-01-LayoutInititalizePresenters.png)

Infrastructure validation

Undefined references

![[A layout. %width=60&anchor=**FooBAAR**
(figures/FilmList-01-
LayoutInitilalizePresenters.png)

Infrastructure validation

Filename with spaces

Spaces in filename leads to more complex VCS management

Semantics validation

```
```example=true
```

```
3 + 4
```

```
>>> 7
```

```
```
```

```
```example=true&expectedFailure=true
```

```
1/0
```

```
```
```

Semantics validation by example

```
```example=true
```

```
3 + 4
```

```
>>> 7
```

```
```
```

```
```example=true&expectedFailure=true
```

```
1/0
```

```
```
```

Reporting an error

```
```example=true
```

```
3 + 4
```

```
>>> 12
```

```
```
```

Evolution rule

How to detect that a method definition is not in sync with the current version in Pharo/Library?

- Pillar offers method source inlining
 - The code is directly taken from the Pillar image version
 - Still difficult to know that something may have changed

Evolution rule

A better approach

- Tag a definition with its origin
- Compare this definition with a current one

Evolution rule

```
""sync=true&origin=MicMethodBodySyncTest >>  
#simpleCode
```

simpleCode

"This is not the definition of simpleCode"

^ 100 slowFactorial + 100

```
""
```

Tool support

Microdown Checker Dashboard

☒ NotTheFigureX

☒ CodeBlockValidator

☒ CodeIndentation

☐ EnglishTypography

☒ EmptyCaption

☒ NoSpaceInFilename

☐ FrenchTypography

☐ HeaderCapitalization

☐ UnnamedChecker

☐ MethodReferenceUsingHash

I check that references to figures, sections and tables are correctly capitalized before an anchor and that there is no 'the' before.
When there is a number (given by an anchor) the section/figure... must be uppercased and not preceeded by the or an.

In Microdown, references should be written as and the following texts are correct

...

- in Figure `*@figureExample@*`
in Section `*@sectionExample@*`

/Users/ducasse/Workspace/FirstCircle/MyBooks/Bk-Writing/PharoBooks2/NewPharoByExample9/index.md

Browse

Run Checkers

Export HTML Report

Results

File Content

Unreferenced Figures

- test status: Passed

isExpectedFailure? false

Description of the test result : Test passed

the concerned code Block : 3 + 4

>>> 7

from the source file:

/Users/ducasse/Workspace/FirstCircle/MyBooks/Bk-Writing/PharoBooks2/NewPharoByExample9/Chapters/PharoObjectModel/PharoObjectModel.md

- test status: Passed

Configuration example

```
{  
    "attribution" : "Stéphane Ducasse, Guillermo Polito and  
Juan Pablo Sandoval",  
    "keywords" : "SUnit, DrTests, Testing, Tests, Agile  
development, Pharo, Smalltalk",  
    "title" : "Testing in Pharo",  
    "onlyCheckers" : [ "EnglishTypography" ],  
    "latexWriter" : "miclatex:sbabook",  
}
```

Rule configuration & results

- Configuring the rules
- Emits microdown-based report to close the loop

pillar check index.md [17:13:31]

Checker: MicEnglishTypographyChecker is checking.

File index.md has reference problems.

English Typography Violations

- Text: " :" in file/Users/ducasse/Workspace/FirstCircle/MyBooks/Bk-Writing/PharoBooks2/Booklet-TestingInPharo/Chapters/Web/Web.md contains a mistake: There is a space before :

Left challenges

- Better support for code formatting violations
- Put in production evolution rules
- Validate all the rule configurations
- Automated analysis in book CI/CD
- How to say that a given piece of text should not follow a rule?

Conclusion

- ValiBoki is a book validation testing framework
- Helps you to produce better books :)
- Ready to write more books and get help