Student Conference 2015

Academic Writing – Part III

Fabian Benduhn, Veit Köppen, Gunter Saake

(based on slides by Christian Kästner)
Keep it short and simple

• Remove clutter
• Use short words
• Keep sentences short with simple structure
• One topic per sentence/paragraph/section
Active Voice

• Write in active voice, avoid passive
• Make clear who is acting

The following the theorem can now be proved.

The data were collected.

Several of the data sets were lost.

To get advice, an expert must be consulted.
Who is Acting?

• Common knowledge in the field? Or contribution by the author?

   It is shown that stable graphs are closed.
   It is hypothesized that …

• Do not hide your contribution

   We have shown that stable graphs are closed.
   Hermann has shown that stable graphs are closed [12].
   We hypothesize that …
Missing subjects

Always provide a subject

Termination occurs after 23 iterations.

The program terminates after 23 iterations.
Strong Verbs

• Use strong verbs
• Avoid “be/is/are”, “has”, “do”, “use”
• Also weak: achieve, carry out, conduct, occur, effect

  the algorithm supports effective garbage collection in distributed systems

  the algorithm collects garbage effectively in distributed systems
Nominalization

• Verbs are stronger than nouns
  – discovery $\rightarrow$ discover
  – movement $\rightarrow$ move
  – collaboration $\rightarrow$ collaborate

• Verbs are stronger than nouns
  – difficulty $\rightarrow$ difficult
  – applicability $\rightarrow$ applicable
  – difference $\rightarrow$ different
Replace Nominalization

Use strong verbs

The police conducted an investigation of the matter.

Term structures can be utilized for dynamic storage of terms.

Local packet transmission was performed to test error rates.
Connectives

• Adding
  – and, also, as well as, moreover, too, furthermore, besides, in addition

• Sequencing
  – next, then, first, second, third…, finally, meanwhile, eventually, after, before

• Emphasizing
  – above all, in particular, especially, significantly, indeed, notably, most of all

• Comparing
  – equally, in the same way, similarly, likewise, as with, like, compared with

• Cause and Effect
  – because, so, therefore, thus, hence, consequently

• Qualifying
  – however, although, unless, except, if, as long as, apart from, yet, despite

• Contrasting
  – whereas, instead of, alternatively, otherwise, unlike, on the other hand, in contrast
Ambiguity / Undefined This

• be more specific than “this”
• “this” is often ambiguous

Lyn asked Max to use the p-test. This proved to be ill-advised.

The test was a poor suggestion.
Disturbing Max proved to be ill-advised.
Making this request proved to be ill-advised.
Max misunderstood the point of the test completely.
Formatting

• Never use **bold** font or *underlined* font in text
• Use *italics* for emphasis
• (On slides use **bold** font instead)

• Emphasize keywords and definitions of abbreviations

• You may use a different font for code
In Figure 3, we show the distribution using a p-metric.

With Table 1, we summarize these results.
Numbers

• Spell out numbers < 10, numerals otherwise
• Spell out numbers at the beginning of a sentence
  – Twenty-five people were...
• Use numerals for units and when referring to sections, pages or millions
  – 1 cm
  – Section 4
  – 3 million
• First, second, ... finally
Figure, Section, Chapter

• When used with a number, capitalize words such as Figure, Section, Chapter, Equation, ..., otherwise don’t.
  – In this chapter, we combine Equations 3 and 4 with Line 12 and the previous figure... as shown in Figure 4, see also Section 4.

• No line break before a number or references
  – Latex: Figure\textsuperscript{4}, p.\textsuperscript{3—4},
Commas

• Use commas for series
  There can be different methods, bodies, or parameters.
• Use comma after introductory word/phrase
  In summary, we described… However, it did… First, it…
  For all typing judgments for terms, we need an environment
• Use commas for logical pauses, or when beginning new thought
• Comma splice: Do not use commas to join sentences, use semicolon or period instead
  It is nearly half past five; we cannot reach town before dark.
  It has been developed based on CIDE; however, it can…
Composite Words

• Usually no hyphen
  – front end, software product line, decision making

• Hyphen when two words modify a third
  – front-end code, decision-making process, rule-based parsing, internet-specific registration code

• Avoid connecting more than three/four words
  – operating-system--command error -> error of an operating-system command
    high-school student vs high school student
Examples

• aspect oriented programming
• annotation based approaches
• software product lines
• software product line implementation
• software product line implementation techniques
• commercial product line tools
• commercial quality Eclipse plug in
• Boolean satisfiability problem solver
• off the shelf SAT solver
Repetition

• Technical writing != prose
• Don’t worry too much about repetition
• It’s okay to start sentences with the same subject
• For important concepts, prefer repetition to synonyms -> less confusion to reader
• Define terms and use them

• Avoid monotony by writing clearly (avoid nominalization, passive, ...) instead
Cap/lc

• Titles and headings: often most words are capitalized
• Capitalize every word >= 4 letters, capitalize first and last word
• Do not capitalize propositions (in, on), articles (a, the), connectives

An Analysis of the Variability in Forty Preprocessor-Based Software Product Lines

A Model of Refactoring Physically and Virtually Separated Features

On the Impact of the Optional Feature Problem
Tables

- No vertical lines, only three horizontal lines (latex: booktabs)
- Provide units in header

<table>
<thead>
<tr>
<th>SPL</th>
<th>( t_{\text{var}} ) (sec)</th>
<th>( t_{\text{SPL}} ) (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MobileMedia (rel. 6)</td>
<td>0.2</td>
<td>1.3</td>
</tr>
<tr>
<td>MobileMedia (rel. 8)</td>
<td>0.3</td>
<td>1.8</td>
</tr>
<tr>
<td>Mobile RSS Reader</td>
<td>0.6</td>
<td>8.3</td>
</tr>
<tr>
<td>Lampaio</td>
<td>2.0</td>
<td>19.0</td>
</tr>
<tr>
<td>Berkeley DB</td>
<td>2.6</td>
<td>21.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SPL</th>
<th>FE</th>
<th>CD/MD</th>
<th>AN</th>
<th>D</th>
<th>FM</th>
<th>CR/MR</th>
<th>GH</th>
</tr>
</thead>
<tbody>
<tr>
<td>GraphPL</td>
<td>20</td>
<td>16/163</td>
<td>167</td>
<td>←</td>
<td>29</td>
<td>41/29</td>
<td>-</td>
</tr>
<tr>
<td>Bali</td>
<td>18</td>
<td>40/503</td>
<td>122</td>
<td>←</td>
<td>18</td>
<td>26/9</td>
<td>-</td>
</tr>
<tr>
<td>Berkeley</td>
<td>38</td>
<td>283/6515</td>
<td>2297</td>
<td>→</td>
<td>99</td>
<td>338/954</td>
<td>858</td>
</tr>
<tr>
<td>Prevayler</td>
<td>5</td>
<td>140/994</td>
<td>175</td>
<td>→</td>
<td>8</td>
<td>13/19</td>
<td>28</td>
</tr>
</tbody>
</table>

FE: number of features; CD/MD: class/member declarations; AN: annotated code fragments; D: direction of initial refactoring; FM: feature modules; CR/MR: class refinements/method refinements; GH: generated ‘hook’ methods

**Table 1.** Statistics before and after refactoring
I vs. We

• Never use “I” in academic writing
• Use “we” for things the authors did (even for a single author, don’t ask)
• Prefer “we show” to “the authors show”

• Figures or papers do not do anything, authors do
  – Figure 3 shows the results. [12] confirms...
  – -> In Figure 3, we show the results. Apel et al. confirm ...
    ...[12].
Singular vs. Plural

• Write definitions in singular where possible
• -> Less ambiguity

Product lines generate variants.
Feature models describe a set of features.
Features are distinguishing characteristics.

A product line generates a variant?
A product line generates variants?
Multiple product lines generate a variant?
Which vs. That

• Which provides additional information
  – the meaning does not change when you remove the subsentence

• That makes things more specific
  – picks a subset, modifies the meaning

• Comma before which, no comma before that.

In embedded systems, which have scarce resources, we…
In embedded systems that have scarce resources, we…

Feature selections that fulfill all constraints are valid.
Dependencies that cannot be expressed with X are called …
Proof reading

• Let others have a look at your text

• Use a good spellchecker and grammar checker (MS Word finds many common problems)

• Especially for Latex users:
  – Convert your text to a word-file (e.g., latex2rtf)
Be precise

• is likely
• may

• can lead to large savings -> led to savings of 13-14% in our experiments.

• Avoid unnecessary modifiers
  – very slow
  – much more energy
Footnotes & Abbreviations

• Avoid footnotes

• Avoid abbreviations
  – Unless really common (SOA, UML, HTML)
  – Or used very often (>10) in the paper; define on first use
    • A Software Product Line (SPL) is a set of ...
  – For plural add an “s”, consider pronunciation ->
    • an SPL; several SPLs
  – do not use “e.g.”, ”i.e.”, “etc”, “Fig.”, “Sec.” in main text (in brackets is ok)
Quotation Marks

• Use for quoting and irony only
• Do not use for emphasis, for using a word in an odd way, or when you are unsure about a word

He said “come on”. (quote)
His “honesty” was appreciated. (irony?)
Consider this “simple” solution… (irony?)
As a “gold standard”, we choose… (irony?)
The computer “decides” which course to take (irony?)
“Fresh” fish here! (irony?)
Articles

For Germans: Try to drop *the* and listen how it sounds. Plural usually does not require *the*.

- A and an are indefinite articles. They refer to something not specifically known to the person you are communicating with.
- You use the when you know that the listener knows or can work out what particular person/thing you are talking about.
- You should also use the when you have already mentioned the thing you are talking about.
- We also use the before certain nouns when we know there is only one of a particular thing.
- We usually use no article to talk about things in general

I ate a banana for lunch.
She's got two children; a girl and a boy. The girl's eight...
Books are expensive.
The conditional compilation with the preprocessors like cpp is simple but effective means to implement variability. By annotating the code fragments with #ifdef and #endif directives, different program variants with or without these annotated fragments can be created. Although, annotation-based approaches are frequently used in a practice, the researchers often criticize them for a negative effect on the code quality and maintainability. In contrast to modularized implementations such as components or aspects, the annotation-based implementations typically neglect a separation of concerns, can entirely obfuscate the source code, and are prone to introduce subtle errors.
Names are Important

• Check spelling of all names
• Most people are sensitive to misspelled names
Misc

• Use either British or US-English; do not mix
• e.g., and i.e., are always followed by a comma
• it is “Related work” and not “Related works”
• it is “Conclusion”, and not “Conclusions”
• don't use blurry figures (prefer vector graphics to bitmaps)