Lecture 3
Style

Prof. Dr. Jasmin Blanchette

Chair of Theoretical Computer Science and Theorem Proving

Version of October 21, 2025

Basic Decisions

Varieties of English

Usually you will write in either **American** or **British English**.
Differences concern spelling, vocabulary, grammar, punctuation, and idioms.

The choice is often yours, but stick to one style.
If multiple authors are involved, the "editor" (or, rarely, an actual copy editor) should enforce a **consistent style**.

The two varieties may be mixed when, e.g., a Briton writes about the *behaviour of the* `color` *attribute*.

Selected Differences between Varieties of English

	Mainly American	**Mainly British**
-dg(*e*)*ment* ending	*judgment*	*judgement*
-er/*-re* ending	*center*	*centre*
-ize/*-ise* ending	*realize*	*realise*
-o(*u*)*r* ending	*behavior*	*behaviour*
-ed/*-t* ending	*learned*	*learnt*
-ward(*s*) ending (adverbs)	*toward*	*towards*
Intraword hyphens	*nonempty*	*non-empty*
Interword hyphens	*first-class citizen*	*first class citizen*
Vocabulary	*chips*	*crisps*
Subject–verb agreement	*the committee agrees*	*the committee agree*
Quotes	*"sweet spot."*	*'sweet spot'.*
Dashes	*Ada—the language*	*Ada – the language*
Enumerations	*1, 2, and 3*	*1, 2 and 3*
Dates	*January 1, 1990,*	*1 January 1990*
Idioms	*skeleton in the closet*	*skeleton in the cupboard*

Registers

Scientific and technical English belong to the **formal register**.
"Formal" is situated between "informal" and "solemn"—more formal than chat messages, but less formal than legal documents.

Avoid colloquialisms as well as stilted language.

Selected Differences between Registers

Informal	Formal
boils down to	*amounts to*
by the way	*incidentally*
can't	*cannot*
code up	*program*
comes with	*includes*
distro	*distribution*
from scratch	*from the ground up*
have to	*need to* or *must*
on the fly	*dynamically*
pays off	*is useful*
prints (to the console)	*outputs*
ships with	*includes*
snippet	*extract* or *fragment*
the sixties	*the 1960s*
write code	*develop code*

Choice of Pronouns

Theses and papers might refer to the authors as *we* or *I*.
Technical documentation tends to avoid these pronouns.

For single-author texts, *I* is usually preferable to the **royal 'we'**,
but only where you are relevant to the story.
There is also the **inclusive 'we'**, which encompasses the reader—e.g.:

> *We can now prove the main theorem.*

In technical documentation, there is often a choice between *you* and *we*.
We is more formal. *One* is even more formal and usually best avoided. Compare:

> *You can also provide the string data as an array of* `chars`*.*
> *We can also provide the string data as an array of* `chars`*.*
> *One can also provide the string data as an array of* `chars`*.*

Gender neutrality can be achieved using **singular** or **plural 'they'**.

Principles

Consistency

Your text should read **as if** it had been written by a **single author**, even if it was not. It should be difficult to tell who wrote what.

To quote Halmos:

> Consistency, by the way, is a major virtue and its opposite is a cardinal sin in exposition. Consistency is important in language, in notation, in references, in typography—it is important everywhere, and its absence can cause anything from mild irritation to severe misinformation.

Neutrality

You have a point of view informed by your beliefs, knowledge, and life experience. In scientific and technical texts, it is usually best to adopt a **neutral perspective**. Even unassuming phrases such as *surprisingly* and *of course* can reveal your beliefs.

When writing technical documentation, a certain level of **objectivity** is expected.

- It is perfectly acceptable to cover all the best aspects and to gloss over the weaknesses or deficiencies.
- But it is not acceptable to deceive regarding failings or to exaggerate capabilities.

Purchasing decisions can be influenced by technical documentation, and misrepresentation can lead to court cases.

Clarity

Some texts seem impossible to understand, because they appear to have been written to show off the author's erudition (or to obfuscate their ignorance) without any concern for making the material **understandable**, let alone interesting.

Even the most difficult concepts can be explained in **clear, accessible English**.
To do this, you must begin by thinking about the reader.

- What is their educational level?
- What specialist knowledge of the field are they assumed to have?
- What is their motivation for reading the document?

Beware of the **curse of knowledge**.

Clarity also requires that you use words **precisely**, with their **literal meanings**.
Developing a graphical user interface in Java is not *painful* (unless your hands actually hurt) but perhaps *tedious* or *unpleasant*. And you certainly did not choose Java because you are *allergic* to C++.

Conciseness

Strunk urges you to **omit needless words**:[1]

> Vigorous writing is concise. A sentence should contain no unnecessary words, a paragraph no unnecessary sentences, for the same reason that a drawing should have no unnecessary lines and a machine no unnecessary parts. This requires not that the writer make all his sentences short, or that he avoid all detail and treat his subjects only in outline, but that every word tell.

Conciseness saves the reader's time. It also helps when working to a page limit.

[1] In other words, you should refrain from using words and phrases that are not strictly necessary to convey your messages.

Examples of Verbosity and Remedies

Verbose	**Concise**
in order to	*to*
is applicable	*applies*
provide assistance to	*help*
This is a subject that	*This subject*
owing to the fact that	*since* or *because*
because of the fact that	*because*
as far as she is concerned	*in her opinion*
in an approximate manner	*approximately*
for all intents and purposes	*essentially*
It is important to note that	*Importantly,*
For example, let us consider	*Consider*
The next point that I want to emphasize is that	*Next,*
The usefulness of this technique resides in the fact that	*This technique is useful because*

Naturalness

Van Leunen writes:

> Scholarly writing is formal, precise, and allusive. It has to be. It does not have to be wooden, finicking, and cabalistic. The idea of this book is to help you achieve the first set of characteristics without sinking into the second.

To find out more, **buy her book**.

Strategies

Fresh Language

Clichés generally do not belong in formal writing—e.g.:

last but not least
at the end of the day
take with a grain of salt
in any way, shape, or form
does not exist in a vacuum
42

Some clichés are tempting in titles—e.g.:

To Be or Not to Be Parallel
A Tale of Two Rust Compilers
Database Performance: The Good, the Bad, and the Ugly

Resist the temptation. Also be sparing with **jokes**.

Gender Neutrality

Because English nouns have no grammatical gender,
the language lends itself well to gender-neutral writing.

He or she, *s/he*, etc., can be replaced by **singular 'they'**.
Often, **plural 'they'** is also an option. Compare:

> *The user can enter their preferences in the text area.*
> *Users can enter their preferences in the text area.*

A few expressions require some care.
Write *person-year* instead of *man-year*, *chair* instead of *chairman*, etc.
Words such as *middleman* and *straw man* must be accepted as is or avoided.

Sexism can be subtle. Generally avoid *Max Mustermann* unless you also have *Erika Musterfrau*. Do not refer to women by first name only. Thus write *Rózsa Péter* or *R. Péter* or *Péter*, not *Rózsa*.

Examples

Scientific and technical writing tends to be abstract.
To make your text accessible, use concrete language, examples, and diagrams.
Tell and show.

Your examples should **look realistic**.
If you write a Java tutorial, do not call your classes *`Foo`*, *`Bar`*, or *`MyClass`*.
In fact, never use *foo* or *bar* for anything. Be creative.

If you have to show some data as an example, use **real data**.

Running examples—examples that are recalled multiple times through a document—can be useful. You can choose them early on, once you have sketched your table of contents. The examples ought to be interesting.

Analogies

An **analogy** is a comparison between a concept being introduced and a familiar concept.

Analogies can aid understanding but can be tricky to get right—e.g.:

> *Finding the right element in an array is like finding a needle in a haystack.*

Examples of Analogies

*We may compare a man in the process of computing a real number to a machine which is only capable of a finite number of conditions $q_1, q_2, \ldots, q_R$ which will be called "*m*-configurations". The machine is supplied with a "tape" (the analogue of paper) running through it, and divided into sections (called "squares") each capable of bearing a "symbol".*[1]

SETHEO contains a powerful preprocessing module. Its incorporation is motivated by analogy to the action of a human reasoner, say a mathematician, who, before applying a specific proof method, first tries to analyse and reduce the respective problem according to more global aspects.[2]

[1] Alan M. Turing, "On Computable Numbers, with an Application to the Entscheidungsproblem," *Proceedings of the London Mathematical Society* s2-42(1), pp. 230–265, 1937.

[2] R. Letz, J. Schumann, S. Bayerl, and W. Bibel, "SETHEO: A High-Performance Theorem Prover," *Journal of Automated Reasoning* 8, pp. 183–212, 1992.

Opening Hooks

In your introduction, try to **engage the reader**, for example using a pun, a quotation, a question, an anecdote, or a provocative statement.

The first two paragraphs of Knuth's celebrated toilet-paper paper:

> *The toilet paper dispensers in a certain building are designed to hold two rolls of tissues, and a person can use either roll.*
>
> *There are two kinds of people who use the rest rooms in the building:* big-choosers *and* little-choosers. *A big-chooser always takes a piece of toilet paper from the roll that is currently larger; a little-chooser always does the opposite. However, when the two rolls are the same size, or when only one roll is nonempty, everybody chooses the nearest nonempty roll. When both rolls are empty, everybody has a problem.*[1]

[1] Donald E. Knuth, "The Toilet Paper Problem," *The American Mathematical Monthly* 91(8), pp. 465–470, 1984.

Positivity

Positive sentences tend to read better than their negative counterparts.
When processing the negative sentence *Do not press the red button*,
the brain performs two operations:

1. It conjures the positive sentence *Press the red button*.
2. It negates it.

Yet the unnegated sentence *Press the red button* hangs in the air.
Better say, *Only press the green button*.

Sometimes, we can use the contrapositive to eliminate negations. Compare:

> *If you do not register the listener,* `mouseClicked()` *will not be called.*
> *If you want* `mouseClicked()` *to be called, you must register the listener.*

Symmetry

Strunk advocates expressing **coordinate ideas in similar form**:

> This principle, that of parallel construction, requires that expressions of similar content and function should be outwardly similar. The likeness of form enables the reader to recognize more readily the likeness of content and function.

Examples:

> *The variable `name` stores the input file's name, whereas the variable `path` stores the file's absolute path.*
>
> *According to Christopher Hitchens, what can be asserted without evidence can also be dismissed without evidence.*

It would be a mistake to vary—e.g.:

> *What can be asserted without evidence can also be dismissed without proof.*

Example of Symmetry

For beginning programmers, tiny, artificial, and unrealistically expensive examples can be instructive models for how more realistic systems should—and should not—be structured. For more experienced programmers, such examples can serve as touchstones that provide handy, memorable points of reference—and that can guide them toward more maintainable, more performant solutions.

Example of Symmetry

For beginning programmers, { *tiny,* / *artificial,* / *and* / *unrealistically expensive* } *examples can be*

instructive models for how more realistic systems { *should* / *—and* / *should not—* } *be structured.*

For more experienced programmers, such examples can

serve as touchstones {
- *that provide* { *handy,* / *memorable* } *points of reference*
- *—and*
- *that can guide them toward* { *more maintainable,* / *more performant* } *solutions.*
}

Pitfalls

Pitfalls

Academese

This is jargon-heavy, pompous, needlessly complex academic writing—e.g., *The individual member of the social community often receives his information via visual, symbolic channels.* (In other words, *People read.*)[1]

Asymmetric coordination

Beware of unbalanced coordinations—e.g., *both in C# and Java*.
Write either *both in C# and in Java* or *in both C# and Java*.

Awkwardness

Various clumsy constructions should be avoided, such as sentences starting with a symbol or number, accidental word repetitions, and consecutive symbols—e.g., *If $x < y$, $y > x$.* (Change to *If $x < y$, then $y > x$.*)

[1] Richard P. Feynman, Ralph Leighton (contributor), and Edward Hutchings (editor), *"Surely You're Joking, Mr. Feynman!": Adventures of a Curious Character*, W. W. Norton, 1985.

Pitfalls

Bad taste

Inside jokes or otherwise bad jokes, obscure references, slang, and clichés are all in dubious taste. Make jokes only if you are sure that they will land.

Breach of conventions

Nonstandard terminology and notations can slow readers down or even confuse them.

Broken logic

Some phrases are incompatible, as in *e.g.*, `JFrame`, `JDialog`, *etc.* The leading *e.g.* ("for example") suggests an incomplete list, but *etc.* ("and the others") completes the list.

Pitfalls

Cultural references

References to movies, celebrities, soccer players, etc., will not be understood by some readers. They also date the text.

Exaggeration

It is impossible to overstate the challenges that we still face in our struggle to revise the computer science curriculum at our university.

Is it really impossible?

Excessive backward referencing

Phrases such as *as mentioned above* should not be overused.

Often, they are an indication that the text is poorly structured.

Pitfalls

Excessive forward referencing

Unless the text is a reference manual, when reading Section 2.1, the reader will not suddenly jump to Section 4.7. So why provide a forward reference?

Excessive metadiscourse

Metadiscourse refers to the text itself, not to the actual topic—e.g., *In this section, we review the literature related to object-oriented databases* or *As my work does not rely on these functions, I do not present the algorithms here. However, they can be found in Kim [40].*

Metadiscourse is occasionally useful but generally best avoided.

Foreign influence

If English is not your native language, it is easy to make certain mistakes. False friends are an issue—e.g., 'n!' is called *factorial* in English, not *faculty* (cf. *Fakultät* in German). Even comma rules differ across languages.

Pitfalls

Negativity

Negativity can often be recast into positivity. Compare:

Thus an equation such as $g(x) = f(x, y)$ would not be allowed.

Thus an equation such as $g(x) = f(x, y)$ would be forbidden.

Nominalization (also called "zombie nouns")

Nominalizations are "abstract nouns formed from verbs or adjectives through the addition of a suffix such as *-ance*, *-ence*, *-ity*, *-ness*, *-ion*, *-ment*, or *-ism*" (Sword). Compare:

The introduction of the well-formedness condition leads to a proliferation of occurrences of `wf` in the program.

If we need objects to be well formed, calls to `wf` will be required all over the program.

Pitfalls

Passivity

The passive voice is widespread in scientific writing,
especially in German but also in English.
Yet often the active voice is clearer and therefore preferable. Compare:
Several computational experiments using the new procedure were conducted.
We conducted several computational experiments using the new procedure.

Self-reference

The reader will feel more at home if you avoid needless references to yourself.
In technical documentation, avoid screenshots that reveal your user name.
"An author in his book must be like God in the universe,
present everywhere and visible nowhere" (Gustave Flaubert).

Pitfalls

Tics

"Everyone has pet phrases and constructions that grow wearisome with reuse. I once read through a thirty-page paper in which the expression 'the case that' occurred more than a hundred times (I counted)" (van Leunen).
So let your friends and colleagues read your texts. They will catch your pets.

Understatement (also called "hyperqualification")

This is the opposite of exaggeration—e.g.,
Partial orders are relatively simple mathematical objects. (Omit *relatively*.)