

# How to write a great research paper

---

Markus Strohmaier, Seminar on "Writing Scientific Articles"

adapted from and largely based on slides by:

Simon Peyton Jones  
Microsoft Research, Cambridge



# Why invest in Writing?

## Writing papers is a skill

---

- Many papers are badly written
- Good writing is a skill you can learn
- It's a skill that is worth learning:
  - You will get more brownie points (better grades, more salary, etc)
  - Your ideas will have more impact
  - You will have better ideas

Increasing importance





The purpose of your paper

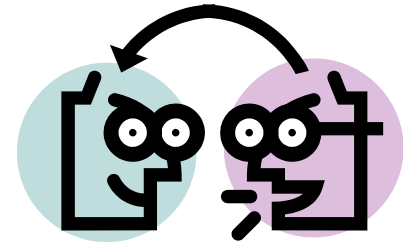
---



# Papers communicate ideas

---

- Your goal: to infect the mind of your reader with **your idea**, like a virus
- Papers are far more durable than programs (think Mozart)



The greatest ideas are (literally)  
worthless if you keep them to  
yourself



# The Idea

## Idea

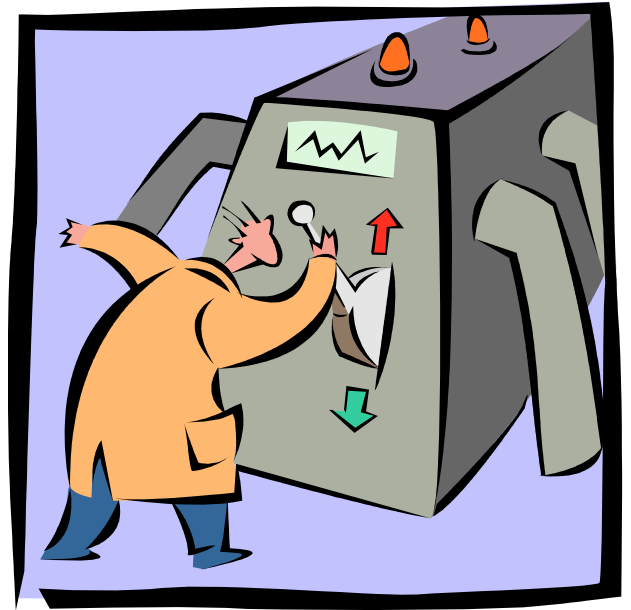
A re-usable insight,  
useful to the reader

- Figure out what your idea is
- Make certain that the reader is in no doubt what the idea is. Be 100% explicit:
  - "The main idea of this paper is...."
  - "In this section we present the main contributions of the paper."
- Many papers contain good ideas, but do not distil what they are.



The purpose of your paper is not...

To describe  
the WizWoz  
system



- Your reader does not have a WizWoz
- She is primarily interested in re-usable brain-stuff, not executable artefacts

# Your narrative flow

- Here is a problem
- It's an interesting problem
- ~~It's an unsolved problem~~
- **Here is my idea**
- My idea works (using X: details, facts)
- Here's how my idea compares to other people's approaches

I wish I  
knew how  
to solve  
that!

I see how  
that  
works.  
Ingenious!





# Structure (6 pages total, English)

---

- Title (1000 readers)
- Abstract (4 sentences, 100 readers)
- Introduction (0.5-1 page, 100 readers)
- The problem (0.5-1 page, 10 readers)
- My idea (0.5-1 pages, 10 readers)
- The details (2 pages, 3 readers)
- Related work (1-2 pages, 10 readers)
- Conclusions (0.5 pages, 50 readers)





# The abstract

---

- Four sentences [Kent Beck]
  1. State the problem
  2. Say why it's an interesting problem
  3. Say what your solution achieves
  4. Say what follows from your solution



# Structure (6 pages total, English)

---

- Title
- Abstract (4 sentences)
- **Introduction** (0.5-1 page)
- The problem (0.5-1 page)
- My idea (0.5-1 pages)
- The details (2 pages)
- Related work (1-2 pages)
- Conclusions (0.5 pages)



# The introduction

---

1. Describe the problem
  2. State your contributions
- ...and that is all!



# Describe the problem

## 1 Introduction

There are two basic ways to implement function application in a higher-order language, when the function is unknown: the *push/enter* model or the *eval/apply* model [11]. To illustrate the difference, consider the higher-order function **zipWith**, which zips together two lists, using a function **k** to combine corresponding list elements:

```
zipWith :: (a->b->c) -> [a] -> [b] -> [c]
zipWith k []      []      = []
zipWith k (x:xs) (y:ys) = k x y : zipWith xs ys
```

Here **k** is an *unknown function*, passed as an argument; global flow analysis aside, the compiler does not know what function **k** is bound to. How should the compiler deal with the call **k x y** in the body of **zipWith**? It can't blithely apply **k** to two arguments, because **k** might in reality take just one argument and compute for a while before returning a function that consumes the next argument; or **k** might take three arguments, so that the result of the **zipWith** is a list of functions.

Use an  
example  
to  
introduce  
the  
problem



# State your contributions

---

- Write the list of contributions first
- The list of contributions drives the entire paper: the paper substantiates the claims you have made
- Reader thinks “gosh, if they can really deliver this, that’s be exciting; I’d better read on”



# Contributions should be refutable

NO!	YES!
We describe the WizWoz system. It is really cool.	We give the syntax and semantics of a language that <b>supports</b> concurrent processes (Section 3). Its innovative features are...
We study its properties	We <b>prove</b> that the type system is sound, and that type checking is decidable (Section 4)
We have used WizWoz in practice	We <b>have built</b> a GUI toolkit in WizWoz, and used it to implement a text editor (Section 5). The result is half the length of the Java version.

We will evaluate your paper based on the contributions it claims.



# Structure

---

- Abstract
- Introduction
- ~~Related work~~
- The problem
- My idea
- The details
- Related work
- Conclusions and further work

# No related work yet

- **Problem 1:** the reader knows nothing about the problem yet; so your (carefully trimmed) description of various technical tradeoffs is absolutely incomprehensible
- **Problem 2:** describing alternative approaches gets between the reader and your idea

I feel stupid



I feel tired





# Structure (6 pages total, English)

---

- Title
- Abstract (4 sentences)
- Introduction (0.5-1 page)
- The problem (0.5-1 page)
- My idea (0.5-1 pages)
- The details (2 pages)
- Related work (1-2 pages)
- Conclusions (0.5 pages)



# Presenting the idea

---

- Explain it as if you were speaking to someone using a whiteboard
- **Conveying the intuition is primary**, not secondary
- Once your reader has the intuition, she can follow the details (but not vice versa)
- Even if she skips the details, she still takes away something valuable



## Putting the reader first

---

- **Do not** recapitulate your personal journey of discovery. This route may be soaked with your blood, but that is not interesting to the reader.



# The details: evidence

---

- Your introduction makes claims
- The body of the paper provides **evidence to support each claim**
- Check each claim in the introduction, identify the evidence, and forward-reference it from the claim
- Evidence can be: analysis and comparison, theorems, measurements, case studies



# Structure (6 pages total, English)

---

- Title
- Abstract (4 sentences)
- Introduction (0.5-1 page)
- The problem (0.5-1 page)
- My idea (0.5-1 pages)
- The details (2 pages)
- **Related work** (1-2 pages)
- Conclusions (0.5 pages)



# Related work

---

Fallacy

To make my work look good, I  
have to make other people's  
work look bad



# Structure (6 pages total, English)

---

- Title
- Abstract (4 sentences)
- Introduction (0.5-1 page)
- The problem (0.5-1 page)
- My idea (0.5-1 pages)
- The details (2 pages)
- Related work (1-2 pages)
- **Conclusions** (0.5 pages)



# Conclusions and further work

---

- Be brief.





# Summary

---

If you remember nothing else:

- Identify your key idea
- Make your contributions explicit
- Use examples