Reminder

• Date for actual conference
• Deadlines are fixed
  – ICSE 2012: 408 submissions, why extending deadline?
  – ICSE 2009: 90% of submissions in the last hour (Samoa Time)

• Workshops and B-level conferences may extend deadlines
  – When too few papers submitted
  – Many requests for extensions
Why to write a paper

• Communicate new findings
  – publication = ultimate result of scientific research
  – research is never finished until it's published

• To let the community know about your work
  – Recognition
  – Contacts, fruitful collaborations

• Get feedback from peers
  – external, independent, frank (anonymous)
Encode a complex web of ideas

… as a linear stream of text.

HOW?
paper organization != research process
Criteria

• Significance
  – Motivate why the research is important or useful. Explain what problem it addresses

• Clarity
  – Organize the paper well and write clearly. Make sure you support your claims

• Novelty
  – Extend the frontier of knowledge. Explicitly relate your research to previous work

• Correctness
  – Critically evaluate and support your claims with proofs, an implementation, examples, or experiments.

Source: William Cook: Academic Writing
Anatomy of a paper

• Title
• Abstract
• Introduction
• (Background / Related Work)
• (Problem Statement)
• Body
• Evaluation
• Discussion
• Related Work
• Conclusion + Future Work
• References
Abstract

• Very brief summary of the paper
• Why is this work important, what was the motivation?
• Main contents, main results
• What is the contribution?
• Typically one of the last things to write

• => Is this paper relevant for the reader (and conference)?
Introduction

• What is the general problem? Why is it important?
• What is the specific problem? Why should the reader care?
• How is it different from prior work?
• What was the motivation for this work?
• What are the objectives/contributions? How is it new?
• What are the main results?
• What is the general approach/outline?
• Keep it short (approx. 1 column)
Background (if necessary)

• What is the necessary background to understand this work?
• In scientific papers usually very short.

• Know your audience!
• Only background that is really necessary!
Problem statement (if necessary)

• What is the specific problem? Why is it important?
• Example if necessary
• Sometimes necessary to tell the reader that there is a problem
The contribution

• Main part of the paper
• Describes the own approach, the innovation
• Readable, verifiable! Examples where necessary!
Evaluation / Proof

• Evaluation is critical
• What is the evaluation criteria?
• Case studies? Empirical studies?
• Does your innovation scale up? Does it solve real problems?
• Report experience
• Readable, verifiable! Can be assessed and replayed
• Separate data from interpretation
Discussion (if appropriate)

• Interpret results
• Advantages and Disadvantages
• (Comparison to related approaches)
• Threats to validity
Related Work

• What are others doing?
• How does this differ from your work? (is your approach better? are there trade-offs? synergies?)
• Also discuss the relationship to YOUR prior work

• Claims of contribution are more convincing in the context of related work

• Common reviewer comments:
  – “The paper omits important related work”
  – “The authors describe the related work but don't compare their work”
Conclusion and Future Work

• Summary
• Results, what has been achieved
• What's missing? New research questions?
• Bigger context, long-term goals?

• Clarify the contribution with respect to the promises in abstract, introduction, and evaluation
References

• Give credits to previous and contextual work
• Reference quotes, claims, previous results

• Only relevant, up-to-date references
• Prefer original source over secondary literature
• Prefer journal to conference to workshop to technical report to web pages

• Do not cite common knowledge (e.g., binary tree, propositional formula)
GETTING STARTED
Writing is Work

• Few people enjoy to write and revise
• Writing is part of a profession

• Academic writing != fiction (inspiration, creativity, art)
• Writing to convey information
• Clarity instead of artistic prose

• -> Learn and practice
• -> Welcome feedback and criticism
Why Learn to Write Well

• Poorly written paper:
  – ambiguity leads to misunderstanding
  – omissions frustrate
  – obscurity makes it difficult to reconstruct authors intentions
  – -> poor reviews, rejections
  – -> frustrated students
  – -> little impact

• Difficult to understand structure -> less focus on the content

• Even the best contribution is not convincing when it is difficult to understand

• Lazy presentation -> impression of unimportant work
Getting Started

• Just write

• Make an outline or slides
  – Discuss this outline with your peers/supervisor

• Make a schedule and stick to it

• No excuses

• Write first, revise later
Excuses

• I can’t find time to write (I would write more if I had the time)
  – Schedule a time, commit to it!
  – Reserve larger time slots
• I need to do more analysis first / read more papers first
  – Do it in your scheduled time! Measure progress.
• I need new computer/printer/software/...
  – ...
• Waiting till I feel like it / waiting for inspiration
  – Technical writing is work
  – Even novelists/poets reject notion of inspiration
• Writers block
  – Does not exist for technical writing
Time Schedule!!!

- 10.04.2014 Student Conference
Scheduled Writing

Boice 1990

- Abstinent (no non-emergency writing)
- Spontaneous (50 sessions, when inspired)
- Contingency Management (50 sessions, forced)
Motivational Tools

• Setting goals
  – Overall goals, project goals
  – Plan deadlines
  – Concrete goal for each day (writing first three paragraphs of discussion section, write at least 200 words, revise section 3, reconcile reference list, reread reviewers comments, …)

• Set priorities
  – Important vs Urgent

• Monitor progress
  – e.g. simple table: date, project, #words, goals met?
  – Calendar with entries on what have you done
• A paper is never “finished”
• Improve by rewriting
• Incrementally improve paper
First Steps

• Make an outline
• Or make a presentation
• Write first version, revise later
First Steps

\section{Introduction}

%SPL introduction. development of many variants in parallel, generation-compile
%many variants, testing etc \rightarrow novel approaches needed
%preprocessor currently common, discussion about alternative implementations, \{
whether longterm as well, tradeoffs, benefits, not discussed here
%type system for entire product lines (all variants are well typed), detection
%search for a simple solution, backward compatible, tool support, practical, s:
%formalization for java subset, proof with coq, implementation for full java as
in several SPLs by others
%own and other prior work
%summary contributions
First Steps

\section{Introduction}

A software product line (SPL) is an efficient means to create a family of domain-specific products from a core common code base, with features (increments in functionality relevant for stakeholders) from some assets that are common to the SPL. Instead of implementing each program from scratch modeling a domain with features, we can generate different variants, tailored to specific usage cases. There are several advantages of using SPLs: they allow for easier maintenance and more efficient testing, and they can be tailored to specific needs. However, there are also challenges, such as the need for robust testing and the potential for increased complexity.

**Many variants, testing etc \rightarrow novel approaches needed**

While the flexibility of SPLs to generate different tailored variants is an important strength, it comes at a price of increased complexity. Developers implement virtually millions of variants in parallel. Testing SPLs is not a trivial task, as a single product must be tested but potentially millions of different variants, in which a certain feature or feature combination is selected. The variants are never or rarely generated (e.g., only late after initial development of a variant), potential errors might go undetected for a long time, until they are generating, compiling, and running all variants is not feasible for most SPLs. Therefore, novel approaches are needed that check the entire SPL itself instead of individual variants.

**Preprocessor currently common, discussion about alternative implementations, \& whether longterm as well, tradeoffs, benefits, not discussed here**

**Type system for entire product lines (all variants are well typed), detection**
Typical Problems

• missing motivation (why is it important?)
• unclear goal, unclear contribution
• missing reasoning (“that’s the way I did it”)
• dead-end discussions, unused background
• unjustified claims
• missing cohesion
• bigger picture missing (just details)
• missing conclusions or results
• jargon, background missing
• related work missing