Editor's Note: How to Write Research Articles in Computing and Engineering Disciplines

Ivan Stojmenovic

WRITING contributions in a form that other people can understand is a very slow process. The purpose of this editorial is to serve as a guideline for making research presentations as easy to comprehend as absolutely possible. Perceived simplicity of any original idea is its ultimate advantage.

The key advice to a successful presentation is to repeat the description of main contribution four times: in the title, abstract, introduction, and text. That is, to make readable, appealing, and as complete as possible versions of the work using the order of 10, 100, 1,000 and 10,000 words. Each of them should be self-contained and complete to the greatest possible extent. This corresponds to the decreasing portion of readers for corresponding parts of the article. To the extent possible, each of these parts should address, in this order: the problem statement, existing solutions, the new solution(s), assumptions and limitations, analysis, simulation, and comparison with the best competing solutions.

Research is a problem solving exercise, differing it from development, implementation, or another type of work. Consequently, any research article should make a clear problem statement.

The selected title of an article should describe the main contribution and the essence of the basic ideas used. The abstract is the most important part of a research article. Instead of listing topics covered in the article, the abstract should convey the essential information found in the paper. The author should make an effort to claim the contribution properly at the most visible place, in the abstract. It should have the following structure (each part with flexible length):

1. A problem statement of the research under consideration.
2. A short list of existing solutions and what their drawbacks are, from the point of view of the above defined problem statement.
3. The essence of the proposed solution, and why it is expected to be better under the same conditions.
4. What type of analysis (theoretical, experimental, simulations, implementations, etc.) was done to show that the proposed solution is really better than any of the existing ones, from both the performance and the complexity points of view.
5. What the major qualitative and/or numerical highlights of the analysis are.

A clear abstract is the key to having the work properly credited in other people's work. Literature reviews of forthcoming papers could simply "cut and paste" the abstract there. The limited space is too often simply wasted by writing general sentences about the field and excessive explanations about the problem that should be part of the introduction.

This structure is also suitable for performance evaluation types of articles, where the problem is to determine the best protocol under various conditions. Existing performance evaluations are existing solutions. What are their drawbacks? Why is this evaluation novel, and what new insights about the protocols are gathered? How does performance evaluation data in this article compare to previous ones?

In brief, the introduction of a paper should present the same content, in the same order, as the abstract, with more space provided. There is also space to address some possibly additional items, such as a general overview of the field and motivation. The introduction should be sufficiently self-contained to comprehend the essence of contribution for people generally working in the area. They should be able to correctly understand what the important aspects of the contribution are, and how good the contribution is. The problem statement needs precise definition, but very technical definitions and statements should be avoided (and presented in later text). Instead, good intuition for the involved definitions or facts should be presented and even illustrated if desirable. Existing solutions and their criticism should be limited normally to only those directly relevant to the contribution. Conditions, context, assumptions, and limitations of the research done should be stated. Under what conditions and scenarios is the new solution the best one? The structure and content of the rest of the document is normally outlined at the end of an introduction in a single paragraph. It should also state the preliminary conference version of journal submission, if any. The introduction should therefore attempt to present a full version of the article in a concise, readable, and intuitively clear form.

Section 2 should give a full literature review. It should collect known results relevant to the problem stated, whether or not they are used in the proposed contributions. It is important to underline the need for a clear cut, *clear separation line*, between existing work and new ideas being presented in the paper. There are in fact three such separation lines: one in each of the abstract, introduction, and the most important one between the literature review section and the rest of the text.

In some cases, the paper may present minor variations, with major consequences, of an existing solution. In this case, the contribution may look large in the essence, but short in text. It is still advisable to separate these two, even if it means a single paragraph of text in describing novel ideas.

One of the major pieces of advice is to do a really *thorough literature review* on the suggested topic (which will be quite rewarding in the long run). This advice stems from a *scientific* view of doing research, where *the best* solution is searched for given *model and assumptions*. There also exists

For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org.

1045-9219/10/$26.00 © 2010 IEEE Published by the IEEE Computer Society

an *innovation* view of research, where a *good* solution is desired for a *practical* problem. In the scientific point of view, the new solution should be *compared* with competing solutions, the best existing solutions under particular assumptions, metrics, and models. In the innovation viewpoint, the emphasis is rather on the *validation* of the new idea, without comparing it with something else that exists. A very frequent problem, however, is to mix the two approaches by attempting to solve a practical problem by using its simple modeling (every model is incorrect, but some of them are useful), but evaluating it in a different "practical" model. Another issue is comparison with solutions that use different metrics and assumptions from the one used in new solution (unfair advantage).

Every discussed reference should be related to the stated problem and contribution in one of several ways: It does not exactly solve the same problem; it solves the same problem, but makes different assumptions about the system; it does not meet certain desirable properties; it has some additional limitations; or it makes the same assumptions, but does not work well under certain important conditions and scenarios that are the primary target of the new solution. A clear statement for each identified solution in this respect is recommended. The space allocated to describing existing solutions should also be "proportional" to its closeness to the new idea and assumptions. Some solutions do not need to be described at all, and a simple convincing statement of why they do not solve the problem at hand may suffice. Other solutions may need a brief description of the general philosophy of the solution before being able to make a similar statement. Otherwise, the solution is a candidate to be a competing one, and requires more attention and space. Such existing solutions need clear, concise descriptions of how they work so that readers can understand a comparison. They are targets for "defeat" by analytical and/or experimental comparisons. There might be a clear reason why a particular competing solution is inferior to the newly proposed one. Inability to "defeat" a particular solution certainly leaves a negative impression on readers.

In summary, the literature review should be a critical one, focused around desired outcome and contribution relevant. It should discuss advantages and drawbacks of known solutions that are relevant to the problem studied, and also discuss the relevance of each reviewed item to the topic studied and newly proposed solutions.

The remaining sections should present new contributions (including conditions, assumptions, and limitations, where appropriate) and their analysis. That is, the very same items listed above should be presented in full, preferably in the same order. Assumptions refer to the simplifications made in the model used so that the solution can be easily understood, while preserving most properties of a realistic model and enabling easy theoretical and/or experimental tractability. Analysis could be analytical, by simulation, or by implementation. Analytical analysis could provide, for example, the proof of validity of the major ideas of the paper. It could lead to a rough estimation of the performance (e.g., message complexity or average/worst time complexity for computation), calculation of parameter values for simulation, and other relevant properties and findings.

After stating the input and the output of an algorithm (which should also be given a mnemonic name), the key idea should be described (clearly and concisely) before discussing steps.

One should always keep in mind that a figure may be worth a thousand words. Important new concepts, and new ideas, should be illustrated by examples and figures as appropriate, to help the reader in understanding them, and to demonstrate one's own understanding of these concepts. Examples should not be trivial, but meaningful and helpful. Figures with examples and diagrams with performance evaluation should not be overly repetitive. A new example is welcome if it offers something essentially different and insightful compared to previous ones. Similarly, additional performance diagrams are welcome only if they offer new performance data, substantially different from data in previous diagrams, for the selected set of parameter values. Repetitive diagrams offering similar value for the analysis should be omitted. The additional size in page length should be justified by the additional contribution, explanation and insight made.

Captions deserve special attention. Reading only the figure captions of the paper should almost substitute for the first rough reading of the entire paper. In the case of simulation diagrams, parameter values and protocol names must be clearly visible and/or listed in the caption. Captions should include title, description of one or more phenomena that deserve attention, explanation (essential reason for observed behavior), and possibly the implication for the protocol/system design.

It is a very difficult task to find a new solution that is best in all circumstances. The primary task in the simulation part of an article is to identify assumptions, metrics, models, and parameter values for which the new solution is better than existing ones. The authors should search for scenarios in which their solution is the best. One should not be overly optimistic about new ideas and make unfounded claims. A smaller but justified claim is better than a large unfounded one.

One of the key pieces of advice is to include all of the possible criticisms of your own idea and contribution directly in the article. It is much better that authors criticize their own work and demonstrate good judgment than to leave such "pleasure" to the examiners and referees. Authors should show that they are in full control of the problem, solutions, and their performance.

Some people read only the abstract and the conclusion. Thus, important things missing in the abstract should be placed in the conclusion section. It could state what has been achieved by the current research, and could discuss and reiterate major advantages and drawbacks of the new solution. The most important part of the conclusion section is to list future work that can be done using the results of the current article. This may offer readers some open problems to study, and such feasible problems could lead to later citations of the article. Sometimes the space can be used to in fact briefly outline some ideas that the author intends to develop further.

One recommendation is to follow a + - + pattern in the introduction and the main text. That is, to start with positive enthusiastic comments about new work and the contribution, then become realistic and list all the drawbacks and limitations, but then finish on a positive note, with a clear statement about the value of the new contribution.

It is important to check if the article has an overall flow, a smooth transition from topic to topic, from familiar information to new information. Within the abstract, introduction, or main text, repetitions should be avoided. One statement and its description should be given in a single, most suitable place in that part of the article.

Writing should be clear and concise. It helps in revealing flaws or new ideas. Shorter and more direct phrases should be used wherever possible. Each concept or algorithm should have a descriptive name. Terms should be normally defined before using them, and should be used precisely and consistently, ambiguities should be avoided, the text should discuss how related concepts are different and/or similar, and passive voice should be avoided. Do not use words like "obviously" or "clearly," which may insult the reader's intelligence.

Finally, after the scientific presentation is deemed acceptable, it is time to pay more attention to the language used and the overall appearance. It is very important to use proper English grammar and sentence structure, and avoid slang and colloquialisms. Misprints must be corrected.

Precision in writing is not easily attained, but one always begins by using the correct word at the proper place and by carefully constructing each sentence. English descriptions and English text has preference to mathematical symbolism wherever possible, for smoother reading. Keep the formulas to a minimum and avoid symbols if ordinary language will do as well. Mathematical symbolism is by its nature intimidating, even to mathematicians. Another good reason to avoid math formalism is the impact of possible misprints. A single misprint anywhere in a fully mathematical formula and the expression can have disastrous consequences for the interpretation and understanding, not only of that particular formula, but the rest of the text. The reader may even be unable to continue reading the article. In some cases, the best approach is to give a math expression followed by its "decoding" with analogous statements in English. However, often it is simply not easy to avoid math symbolism without loosing precision.

The advice so far was applicable to technical papers in general, including "systems" papers. "Systems" papers should discuss the reality, lessons learned, and choices made. Does the paper describe something that has actually been implemented? If so, how has it been used, and what has this usage shown about the practical importance of the ideas? Otherwise, do the ideas justify publication now? What did the authors learn and what should the reader learn from the paper? How generally applicable are these lessons? What were the alternatives considered at various points, why were the choices made the way they were, and did the choices turn out to be right? How realistic are assumptions? Does the formal model, if presented, give new information, and is it supported by any deep theorem?

Writing a good paper is a hard work, but you will be rewarded by a broader distribution and greater understanding of your ideas within the community. Please refer to http://www.computer.org/portal/web/csdl/transactions/tpds for a full version of this editorial.

Ivan Stojmenovic
Editor-in-Chief

SINGIDUNUM
JOURNAL
OF APPLIED SCIENCES
ECONOMICS, MANAGEMENT, TOURISM, INFORMATION TECHNOLOGY AND LAW

SINGIDUNUM JOURNAL 2012, 9 (1): 42-50
ISSN 2217-8090
UDK 001.81:[007:004]
Original paper/Originalni naučni rad

HOW TO WRITE RESEARCH ARTICLES IN COMPUTING AND ENGINEERING DISCIPLINES

Ivan Stojmenović[1,*], Veljko Milutinović[2]
[1]University of Ottawa, SITE, Canada
[2]University of Belgrade, Department of Computer Engineering,
73 Bulevar Kralja Aleksandra Street, Belgrade, Serbia

Abstract:
This article advocates a general way of presenting research articles on any topic and in any field related to computer science, information technology, and engineering disciplines. The key advice to a successful presentation is to repeat the description of main contribution four times: in the title, abstract, introduction (or chapter 1) and in the text. That is, to make readable, appealing, and as complete as possible versions of the work using the order of 10, 100, 1000 and 10,000 words. This corresponds to the decreasing portion of readers for corresponding parts of the article. To the extent possible, each of these parts should address, in this order: the problem statement, existing solutions, the new solution(s), assumptions and limitations, analysis, simulation and comparison with best competing solutions.

Key words:
computer science,
original paper,
abstract,
title.

MOTIVATION FOR WRITING AND READING THIS ARTICLE

The major goal of this paper is to serve as a guideline for the organization of research presentations in written form. The major purpose of the entire effort is to make research presentations as easy to comprehend as absolutely possible. "The process of clarifying your thinking, of which writing papers is one aspect, is a valuable part of improving your research" (Ernst, 2005). The same structure is valid for thesis work, as well as for conference and journal publications, or technical reports for research sponsors.

The motivation to write this article came from the observation that the vast majority of papers in journals, conference proceedings, and the majority of observed oral presentations do not follow the basic structure, as outlined in the abstract here. In most cases, research results are obscured by poor presentation. It is not possible to quickly understand, either the essence of the presented contribution, or the most important research details. This is happening despite the presentation advice appearing to be quite natural and effective for easy understanding of contributions made in a given article. "If you do not write well, why should readers believe you were any more careful in the research itself?" (Ernst, 2005).

[*] E-mail: stojmenovic@gmail.com

Writing contributions in a form that other people can understand is a very slow process. Excellent presentation also brings some risks for the evaluation, due to "noise in the refereeing system" (Ernst, 2005). A novel idea, well presented, can be claimed to be too simple or even trivial by reviewers, and the judgment is subjective. This author believes in simplicity of any original idea as its ultimate advantage. A number of papers describe new ideas only via programming codes, or complicated diagrams, without presenting clear concise descriptions and/or figures and illustrations on how they work. This style could potentially bring favorable opinions from reviewers with similar attitude on presentation, or without real desire to understand the details. A well-understood idea also brings the risk of being identified as already existing by reviewers. When it is novel and if well understood, it also becomes easier to identify the drawbacks and criticize it. If a problem statement has a new name, different from the name well-known in the community (meaning that the literature review is not done properly), it becomes difficult to identify proper referees for the article and/or to properly judge its contribution. In summary, there are also reasons not to apply some pieces of advice from this article, if the publication of an article is the sole motivation.

This article places particular attention to four key parts of a research report: the title, abstract, introduction, and to the main body of text. Each of them should be self-contained and complete to the greatest possible extent, because there are four different types of readers. Among those who will ever see any part of a particular work, perhaps an estimated 80% (the numbers are subjective, based on experience of authors) will see only the title, 15% will read the abstract (and possibly parts of the conclusion in search for information missing in the abstract), 4% will also read the introduction, and the remaining 1% will meaningfully go through the whole paper (some of them may fully read and analyze it, depending on their particular interests and skills, content value, and its presentation).

"A paper should communicate the main ideas of your research early and clearly" (Ernst, 2005). Clear and appealing text in one part increases the chances that a reader will go to the next part and eventually use and cite the work. The readers could be important persons: thesis examiners, reviewers of conferences and journal submissions. They all have limited time to spend on a particular article and time should be used wisely, to make the greatest possible reading and understanding progress for a given amount of time. Reading time normally grows with the clarity and usefulness of text and presentation. Misleading title narrows the readership. If the abstract does not clearly state the contribution made, and fails to attract the reader, then the reader may not advance to the introduction (or the first chapter of a thesis). Sometimes it is beneficial to convince an important reader (examiner, reviewer) that there is no need to study much the main body of the research report, if the introductory text properly presented the contribution and gave a clear picture of the coming text. "A reader who understands the structure and big ideas can better appreciate the details" (Ernst, 2005).

RESEARCH AND PROBLEM SOLVING

Is research a problem solving exercise? Are there some research activities that do not solve any problems? We believe that this is not the case. If there is no problem to be solved than the related activity could be a development, implementation, or another type of work. Consequently, any research article should make a clear problem statement.

We were frequently faced with articles, even grant proposals, that in fact do not have any real problem to be solved. An example is description of a "novel" software architecture whose evaluation remains fully subjective, without defining any problem to solve. This does not mean that proposing a new architecture is not a valid research problem. However, it so only when the integral parts of doing research are addressed in the article: what problem is resolved by the design of a new architecture, what existing architectures are, what are assumptions and limitations made, how new architecture compares with existing ones *etc.*

Another example is research article (Stojmenović, 2000) on teaching recursion in the first computer science course. Existing textbooks suggested complicated or no proofs that recursive Fibonacci numbers and binomial coefficients algorithms have exponential time complexities (existing teaching solutions), which were proven on final exams to not be well understood by students. Article (Stojmenović, 2000) described an elegant proof in two lines, easily understood by all students. The proof does not have to be original itself; the main novelty is in applying this proof in the classroom. In general, the identification and application of existing concepts and techniques to new domains and problem statements is a research activity and represents contributions.

SELECTION OF 'TITLE'

The selected title of an article should enable the expert to figure out the essence of the basic idea(s) and the main contribution(s), even without reading the paper (Milutinović, 1996). Further, the title should induce the reader to think deeply over the philosophy of the contribution described in the paper (Milutinović, 1996). An example is the title of Kalosha *et al.* (2008). It states the problem (georouting), the area (sensor networks), desired properties (beaconless, guaranteed delivery) and essence of new idea (select and protest based). Proper title leads to more hits in Google Scholar searches. An imprecise title may bring wrong referees or examiners for the work, and less appreciation.

ABSTRACT: PROTECTING THE CONTRIBUTION

In our opinion, the abstract is the most important part of a research article. Suppose that one has five minutes of time for an oral presentation, or a few hundred words of written space to protect a new idea and contribution and gain deserved recognition. Note that in many cases abstracts are offered freely while the whole paper needs time (and sometimes also financial resources) to download or acquire. Examples are CiteSeer and Springer databases. How would this time or space be used wisely?

The time/space limitation is almost naturally enforced. Researchers may not pay attention to an article if the abstract does not make convincing statements. Instead of listing topics covered in the article, abstract should convey the essential information found in the paper. The author should make an effort to claim the contribution properly at the most visible place, and not expect readers to do so. Consequently, a good idea may not be noticed by the research community, and those who reinvent it at a later time will get credit instead of the initial inventor (Milutinović, 1996).

The authors generally agree with the five part structure of the abstract as described by Milutinović (1996), as follows (using a simple and concise language):

a) Problem statement of the research under consideration;
b) A short list of existing solutions and what their drawbacks are, from the point of view of the above defined problem statement;
c) Essence of the proposed solution, and why it is expected to be better under the same conditions;
d) What type of analysis was done to show that the proposed solution is really better than any of the existing ones, from both the performance and the complexity points of view;
e) What the major numerical highlights of the analysis are. Here we recommend also some qualitative (non-numeric) highlights that may be more feasible and helpful in many cases.

The length of each part should be flexible. For example, the proposed solution could be the first one (a) simple statement with few words for (b), while the new idea (c) may take about half of the abstract to express with sufficient intuition.

The abstract should be written for researchers that are familiar with the research area, and can grasp the contribution easily. Some of them could have worked on the same or related problems. A clear abstract is the key to having the work properly credited in other people's work. One should envision literature reviews of forthcoming papers by other researchers. They could simply "cut and paste" the abstract into their articles, which then serves as an extended patent protection. This could be even done semi-automatically, as others do not really need to read or understand the whole article, to cite it properly. If an abstract is without proper content, citations to the article in other papers may not be very informative either. Should they be? It is the author's obligation in the first place to properly describe the idea with limited space, without expecting that someone else will do this later on behalf of the authors.

Examiners and reviewers will especially appreciate such an abstract to have a friendly start with the text, and obtain a clear picture in a very short period of time. Misleading abstracts are unfortunately quite common practice in research literature. The limited space is too often simply wasted by writing general sentences about the field and excessive explanations about the problem, that should be part of the introduction or Chapter 1. It is best to first answer the above five questions, then see whether there is space left to say anything else.

This structure is also suitable for performance evaluation type of articles. In a performance evaluation based article, the problem is to determine the best protocol under various conditions. Existing performance evaluations are existing solutions. What are their drawbacks? Why is this evaluation novel, and what new insights about the protocols are gathered? How does performance evaluation data in this article compare to previous ones?

Survey type of articles however may have a different presentation style. A survey should describe all relevant solutions, classify them according to assumptions made and some properties (that is, present a taxonomy), and draw some conclusions. The contribution of a survey article should still be clarified. It could be the first survey on a given topic or problem, or may present sufficient novel material not covered in existing tutorials, or it could give a taxonomy not seen previously. The survey may also properly describe existing solutions, in a clear and concise manner, compared to existing surveys. A statement of gain the reader will obtain by reading the survey or tutorial article is needed in the abstract. Detailed instructions for writing survey articles can be found on the website of the second author.

CONTENT OF INTRODUCTION OR CHAPTER 1

In brief, the introduction of a paper intended to be published in a journal or conference, or Chapter 1 of a master or doctoral thesis, should present the same content (summary of the article), in the same order, as the abstract, with more space provided. Normally it is about ten times longer as a rough approximation. There is also space to address some possibly additional items, such as motivation.

While an introduction is normally a single section, Chapter 1 should have separate headings to address the structure and natural flow in the explanation. The abstract may be sufficient for an expert in the field or researchers who have worked on the same problem. The Introduction or Chapter 1 should suffice to comprehend the essence of contribution for people generally working in the area. They should be able to correctly understand what the important aspects of the contribution are, and how good the contribution is.

In our view, the introduction or Chapter 1 should present sufficient information, and be sufficiently self-contained, so that important readers and evaluators do not need to read the rest of text, being assured in the contribution made, and validity of the text to follow. If they are pressured with shortage of time and the need to make a decision on the value of the article, or a citation and reference to it in their own work, let this part give answers to all important possible questions, and earn appreciation. Readers and especially followers of the specific research directions will appreciate such a style and will prefer mentioning such work rather than the work of someone else who remained unclear in the introduction and the article was never read due to the lack of time or loss of confidence.

Here are recommended sections, or subsections, or parts of an introduction.

- General overview of the field (basic facts needed to tune the reader to the thesis or paper);
- Problem statement (precise definition and importance);
- Very technical definitions and statements should be avoided (and presented in later text) and instead, good intuition for the involved definitions or facts should be presented and even illustrated if desirable;
- Existing solutions and their criticism (limited normally to only those directly relevant to the contribution of the thesis), a motivation for doing research on the topic should be stated;
- Contributions (proposed solutions; why they are expected to be better; essence of the idea(s) used in proposed solutions);
- Conditions, context, assumptions and limitations of the research done;
- Analysis (theoretical, experimental, simulations, implementations, *etc.*) done in the thesis; under what conditions and scenarios is the new solution the best one?
- In case of thesis work, it is recommended to add specific statements about the contribution of the thesis author to the thesis work, and the contribution of the thesis to the research field. This is normally done by listing all existing and intended journal and conference publications out of the thesis, which include their authors and titles, and references to proper sections in the thesis.
- The structure and content of the rest of the document is normally outlined at the end of an introduction or Chapter 1 in a single paragraph.

Some sections should normally present the highlights with pointers to later sections and chapters that provide details. The introduction should attempt therefore to present a full version of the article in a concise, readable and intuitively clear form.

LITERATURE REVIEW: THE NEXT SECTION OR CHAPTER

Chapter or section 2 should give a full literature review. Many articles present it at the end of the paper, leaving the reader to wonder about the actual contribution until the very end of the text. The literature review (or related work section) should collect all known results relevant to the problem stated, whether or not they are used in the proposed contributions. No additional literature review shall be added in later chapters, where the text could only refer to well known results (*e.g.* those covered in the undergraduate computer science program such as Dijkstra's shortest path algorithm, sorting algorithms *etc.*) which can be reasonably assumed to be public knowledge for the particular field. In these cases, they should not be described (that is, if there is a need to describe how a well known shortest path algorithm works, then this should be done in the literature review section).

It is important to underline the need for a clear cut, clear separation line, between existing work and new ideas being presented in the paper. There are in fact three such separation lines: one in each of the abstract, introduction (or Chapter 1) and the most important one between the literature review section and the rest of text. That line should be very sharp.

In some cases, the paper may present minor variations, with major consequences, of an existing solution. In this case, the contribution may look large in the essence, but short in text. This author still advocates to separate these two, even if it means a single paragraph of text in describing novel ideas.

This approach however may not receive positive feedback from reviewers that do not share this philosophy. Separation lines are then needed inside the text itself as a compromise. The main problem with placing separation lines inside the text, rather than by placing content in different sections, is that the separation between existing and new ideas easily remains unclear if the presentation is not very careful. Of course, the assumption here is that the primary intention was to be honest with the contribution made in the article.

One of the major pieces of advice is to do a really thorough literature review on the suggested topic. In most observed cases, however, this does not occur. This is also not really advocated by most supervisors. It is also not emphasized in other papers seen on how to write research articles. We have even seen *PhD* theses without any literature reviews on the topic. This advice steams from a scientific view of doing research, where the best solution is searched for given model and assumptions. There exist also an innovation view of research, where a good solution is desired for a practical problem. In the scientific point of view, the new solution should be compared with competing solutions, the best existing solutions under particular assumptions, metrics, and models. In the innovation viewpoint, the emphasis is rather on the validation of the new idea, with/out comparing it with something else that exists. A very frequent problem, however, is to mix the two approaches, by attempting to solve a practical problem by using its simple modeling (every model is incorrect but some of them are useful) but evaluating it in a different practical' model (Stojmenović, 2008). Another issue is comparison with solutions that use different metric and assumptions from the one used in new solution (unfair advantage) (Stojmenović, 2008).

While on one hand doing a thorough literature review could prove to be dangerous for the new idea in mind, it is expected to be very rewarding in the long term, as it opens the views to different models, assumptions, different problem statements, and offers material for new contributions with much greater compensation for potentially lost contribution. In fact, ideas are normally credited to original authors anyway, not to those that duplicate it. One of the identified problems is that the existing practice is to merely cite a paper on the subject of study (which happens to be very close to the stated problem and possibly presenting a competing solution), without thoroughly studying it, or describing it properly, which opens the door for an even more dangerous duplication, one that eliminates the excuse of possible independent work.

It is very easy for a reviewer or even examiner to save his time by observing a missing important reference, and claiming that that the particular reference may solve the same problem in a better way. That may or may not be true, but some decisions are not recoverable (*e.g.* in the case of conference submissions).

The list of references is important for the selection of reviewers, especially when submitting to a journal. Editors normally check the reference list in search of reviewers working on same or similar problem. If the list appears "tiny" on the exact problem statement, editors may search for reviewers on Google Scholar for instance. Reviewers selected that way may be subjective in their evaluation if their important relevant work was not cited in the submission ("the paper does not cite me, therefore something is wrong here").

For every discussed reference, it is very important to relate them to the stated problem and contribution in one of several ways: it does not exactly solve the same problem, it solves the same problem but makes different assumptions about the system, it does not meet certain desirable properties (*e.g.* it is not real-time solution), it has some additional limitations, or it makes the same assumptions but does not work well under certain important conditions and scenarios that are the primary target of the new solution. A clear statement for each identified solution in this respect is recommended. The space allocated to describing existing solutions should also be proportional to its closeness to the new idea and assumptions. Some solutions do not need to be described at all, and a simple convincing statement of why they do not solve the problem at hand may suffice. Other solutions may need a brief description of the general philosophy of the solution before being able to make a similar statement. Otherwise the solution is a candidate to be a competing one, and requires more attention and space. Such existing solutions need clear, concise descriptions of how they work, so that readers can understand a comparison. They are targets for defeat by analytical and/or experimental comparisons. There might be a clear reason why a particular competing solution is inferior to the newly proposed one. Inability to defeat a particular solution certainly leaves a negative impression on readers.

In summary, the literature review should be a critical one, focused around desired outcome and contribution relevant. It should discuss advantages and drawbacks of known solutions that are relevant to the problem studied, and also discuss the relevance of each reviewed item to the topic studied and newly proposed solutions.

THE REMAINING CHAPTERS OR SECTIONS

The remaining sections/chapters should present new contributions (including conditions, assumptions, and limitations, where appropriate) and their analysis. That is, the very same items listed above should be presented in full, preferably in the same order. Assumptions refer to the simplifications made in the model used, so that the solution can be easily understood, while preserving most properties of a realistic model and enabling easy theoretical and/or experimental tractability. Analysis could be analytical, by simulation or implementation. Analytical analysis could provide, for example, the proof of validity of the major ideas of the paper. It could lead to a rough estimation of the performance (*e.g.* message complexity for communication among sensors or average/worst time complexity for computation in sensor processors), calculation of parameter values for simulation, and other relevant properties and findings.

"When presenting an algorithm, first state what the output is and preferably the key idea, before discussing steps" (Ernst, 2005). Pseudo-code description, if used, should include the mnemonic name for the algorithm, its input and output. Such pseudocodes should be preceded in text with clear concise descriptions of same algorithm.

One should always keep in mind that a figure may be worth a thousands words. Important new concepts, and new ideas, should be illustrated by examples and figures as appropriate, to help the reader in understanding them, and to demonstrate one's own understanding of these concepts. This author found most mistakes in student's understandings by simply asking them to give different types of examples. The same is with readers. Examples should not be trivial, but meaningful and helpful.

Figures with examples, and diagrams with performance evaluation, should not be overly repetitive. A new example is welcome if it offers something essentially different and insightful compared to previous ones. Similarly, additional performance diagrams are welcome only if they offer new performance data, substantially different from data in previous diagrams, for the selected set of parameter values. Repetitive diagrams offering similar value for the analysis should be omitted or moved to an appendix of a thesis. The overall contribution is not evaluated by the overall length of a thesis or paper, but by its technical content. In other words, the additional size in page length should be justified by the additional contribution, explanation and insight made.

Captions deserve special attention, which is neglected in a typical written presentation(Milutinović, 1996). Reading only the figure captions of the paper should almost substitute the first rough reading of the entire paper. In case of simulation diagrams, parameter values and protocol names must be clearly visible and/or listed in the caption. Captions should include title, description of one or more phenomena that deserve attention, explanation (essential reason for observed behavior) and possibly the implication for the protocol/system design.

It is a very difficult task to find a new solution that is best in all circumstances. The primary task in the simulation part of an article is to identify assumptions, metrics, models and parameter values for which the new solution is better than existing ones. The authors should search for scenarios in which their solution is the best. More details on simulation can be seen in (Stojmenović, 2008). One should not be overly optimistic about new ideas and make unfounded claims. A smaller but justified claim is better than a large unfounded one. Bigger claims open bigger doors for attacks by referees and examiners. Referees may easily turn down the complete idea because of an unsupported large claim, but can also easily accept even a minor contribution if it is well documented and proven.

One of the key pieces of advice is to include all the possible criticisms of your own idea and contribution directly in the article. It is much better that authors criticize their own work and demonstrate good judgment than to leave such pleasure to the examiners and referees. Authors should show that they are in full control of the problem, solutions, and their performance.

ON CONCLUSIONS AND REFERENCES

Some people read only the abstract and the conclusion. Thus important things missing in the abstract should be placed in the conclusion section. It could state what has been achieved by the current research, and could discuss and reiterate major advantages and drawbacks of the new solution. The most important part of the conclusion section is to list future work that can be done using the results of the current article. This may offer readers with some open problems to study, and such feasible problems could lead to later citations of the article. Sometimes the space can be used to in fact briefly outline some ideas that the author intends to develop further. The ownership of some other possible solutions, not fully explored, to the same or a relevant problem, or subject of your forthcoming different article, can be protected by outlining them briefly in the conclusion section, possibly even with reference to an upcoming article.

One recommendation is to follow a +-+ pattern in the introduction and the main text. That is, to start with positive enthusiastic comments about new work and the contribution, then become realistic and list all the drawbacks and limitations, but then finish on a positive note, with a clear statement about the value of the new contribution. It is important that the reader finishes reading the article with a positive impression. He might be writing a follow up report afterwards.

OVERALL FLOW AND APPEARANCE

It is important to check if the article has an overall flow, a smooth transition from topic to topic, from familiar information to new information. Within each of the abstract, introduction, or main text, repetitions should be avoided. One statement and its description should be given in a single, most suitable place in that part of the article.

Writing should be clear and concise. "Writing more clearly will help you think more clearly and often reveals flaws or ideas that had previously been invisible even to you" (Ernst, 2005). "Use shorter and more direct phrases wherever possible" (Ernst, 2005). Each concept or algorithm should have a descriptive name. Terms should be normally defined before using them, and should be used precisely and consistently. Ambiguities should be avoided. The text should discuss how related concepts are different and/or similar. Avoid passive voice. Do not use works like "obviously" or "clearly", which my insult reader's intelligence (Ernst, 2005).

Finally, after the scientific presentation is deemed acceptable, it is time to pay more attention to the language used and the overall appearance. It is very important to use proper English grammar and sentence structure, and avoid slangs. Misprints must be corrected. One always expects very professional referees and examiners. Their opinion is partially subjective, compared to, say, an evaluation expected from a knowledgeable robot. A good approach in extracting a positive impression for the subjective part of the overall evaluation is by showing the overall care taken in writing the article.

RELATED WORK ON PRESENTING RESEARCH ARTICLES

This literature review is limited to items that are not elaborated elsewhere in this article. There is plenty of advice for writing research articles that can be found on the Internet. However, most of it deals primarily with language, grammar and formatting issues, and does not go deeply into discussing how to properly and effectively present the essence of the contribution made in a given article.

Instructions for preparing transparencies and oral presentations (Milutinović, 1996) are complete and we found no need to elaborate on them further here. A particular feature in article written by Milutinović (1997) is to use semantics based layout strategy on transparencies.

Alba (2002) presented some brief comments and advice addressing the following problems that students and new researchers face: structure the document, formatting guides, content, readability, electronic edition and diffusion. The suggested structure of the document is: Introduction, Problems (this part includes literature review), Resolution methods, Experiments, Results, Conclusions, and References. Resolution methods should stress the novelty of the method and approach, a specific non-ambiguous explanation of the method (*e.g.* pseudocodes), mathematical or formal issues of proposed techniques, parameters and most important decisions made to select these methods or techniques, plans to solve the problems with proposed appropriate methods, and expected results after having done so. The experiments section should present the goals and sets of experiments, parameters, algorithms and problem instances intended for use (preferably in table form), measures, statistical analysis, and criteria to judge the value of the results, steps to follow to get the results with justification.

Woodford (1999) lists the overall steps in writing a journal article, dissertation or grant proposal, in a brief note, while the full text is available in his book. It begins from asking whether the time is right for writing, to analyzing and possibly answering the examiner's remarks. Ernst (2005) discusses rejection: "In most cases, reviews offer an opportunity to improve the work". To avoid reputation of submitting "a half-baked work", only submit to obtain new information (including concerns not predicted in advance).

Recommendations for writing articles also depend on the research field. Sherrill (2003) briefly described recommendations for writing articles in the chemistry discipline. The parts of a paper are: Abstract, Introduction, Methods (or Theoretical Methods), Results and Discussion, and Conclusions. Similar structure for papers can be found in other fields, *e.g.* psychology.

The structure of a research paper is less rigorous in mathematics field. Normally papers are collections of theorems and proofs, and every known proof from other sources is cited in the text where needed. Here are a few of pieces of advice by Cheney (2000) that are found to be quite relevant for computing and engineering articles (*e.g.* description of algorithms, protocols and systems). "Mathematics is preeminent in its striving for absolute precision in its formal written text. Precision in writing is not easily attained, but one always begins by using the correct word at the proper place and by carefully constructing each sentence. We also advise against the use of slang, colloquialisms, and other non-standard linguistic devices" (Cheney, 2000).

"Use English descriptions and English text in preference to mathematical symbolism wherever possible." (Cheney, 2000). "It makes for smoother reading. Mathematical symbolism is by its nature intimidating, even to mathematicians. There is nothing so daunting as having to read a page of formulas! Keep the formulas to a minimum and avoid symbols if ordinary language will do as well. There may be cases where, for good reason, one wishes to violate this rule." (Cheney, 2000).

Another good reason to avoid math formalism is the impact of possible misprints. A single misprint anywhere in a fully mathematical formula and the expression can have disastrous consequence for the interpretation and understanding, not only of that particular formula but the rest of the text. In some cases, the reader is even unable to continue reading the article. On the other hand, a single misprint in an English sentence, even an awkward English sentence, allows the reader to automatically correct and continue reading. In some cases, the best approach is to give a math expression followed by its decoding with analogous statements in English. Sometimes it is simply not easy to avoid math symbolism without loosing precision.

WRITING GOOD SYSTEMS PAPERS

The advice so far was applicable to technical papers in general. To address the readership of IEEE Transactions on Parallel and Distributed Systems, we address some specific problems in "systems" papers, following closely (Levin and Redell, 1983). They listed thirty-odd questions to help writing a better technical paper. Questions related to originality of ideas, focus, presentation and writing style are close to the discussion here, often given from a complementary perspective. Questions specific to systems papers are about reality, lessons learned and choices made. Does the paper describe something that has actually been implemented? If so, how has it been used, and what has this usage shown about the practical importance of the ideas? Otherwise, do the ideas justify publication now? What authors learned and what should the reader learn from the paper? How generally applicable are these lessons? What were the alternatives considered at various points, why the choices were made the way they were, and did the choices turn out to be right? How realistic are assumptions? Does the formal model, if presented, give new information, and is it supported by any deep theorem?

According to Levin and Redell (1983), writing a good paper is a hard work, but you will be rewarded by a broader distribution and greater understanding of your ideas within the community of journal and proceedings readers.

REFERENCES

Alba, E. (2002) Writing a scientific/technical report. [online] Available from: www.lcc.uma.es/~eat/pdf/sw.pdf [acessed 27 June 2011].

Cheney, W. (2000) Advice for students writing reports, theses, and dissertations. Department of Mathematics, University of Texas at Austin. [online] Available from: www.ma.utexas.edu/users/cheney/ [acessed 21 August 2011].

Ernst, M. (2005) Writing a technical paper. [online] Available from: www.cs.washington.edu/homes/mernst/advice/write-technical-paper.html [acessed 19 April 2011].

Kalosha, H., Nayak, A., Ruehrup, S., Stojmenović, I. (2008) Select and protest based beaconless georouting with guaranteed delivery in wireless sensor networks. *IEEE INFOCOM*, 13-18 April 2008 Arizona.

Levin, R., Redell, D. (1983) How (and how not) to write a good systems paper. *ACM Operating Systems Review.* 17 (3), 35-40.

Milutinović, V. (1996) The best method for presentation of research results. IEEE TCCA Newsletter [online] Available from: http://tab.computer.org/tcca/news/sept96/bestmeth.pdf [acessed 03 August 2011].

Milutinović, V. (1997) A good method to prepare and use transparencies for research presentations. IEEE TCCA Newsletter. [online] Available from: http://tab.computer.org/tcca/news/sept96/bestmeth.pdf [acessed 16 June 2011].

Sherrill, D. (2003) Writing journal articles. Georgia Institute of Technology. [online] Available from: vergil.chemistry.gatech.edu/resources/writing-papers.pdf [acessed 13 May 2011].

Stojmenović, I. (2000) Recursive algorithms in computer science courses. *IEEE Transactions on Education.* 43 (3), 273-276.

Stojmenović, I. (2008) Simulations in wireless sensor and ad hoc networks. *IEEE Communications Magazine.* 46 (12), 102-107.

Woodford, P. (1999) Writing a journal article, dissertation, or research-grant proposal: Steps to for graduate students and others. *CBE Views.* 22 (1), 12-13.

PREZENTOVANJE REZULTATA U OKVIRU ORIGINALNOG NAUČNOG RADA IZ OBLASTI RAČUNARSKIH I TEHNIČKIH NAUKA

Rezime:

Ovaj članak daje opšte smernice u pisanju istraživačkih članaka u oblastima koje su vezane za računarstvo, informacione tehnologije i tehničke nauke. U uspešnom članku je najvažnije da se ponovi opis glavnog doprinosa četiri puta: u naslovu, apstraktu, uvodu (ili prvom poglavlju) i u tekstu. To znači da se napišu čitke i što potpunije verzije rada upotrebom 10, 100, 1.000 i 10.000 reči. To odgovara i smanjenju broja čitalaca pomenutih delova članaka. Koliko god je to moguće, svaki od ovih delova treba da sadrži, ovim redom: formulisanje problema, postojeća rešenja, novo rešenje (rešenja), pretpostavke i ograničenja, analizu, simulaciju i poređenje sa najboljim konkurentskim rešenjima.

Ključne reči:
informatika,
originalni naučni rad,
apstrakt,
naslov.

Received: September 15th, 2011
Correction: October 22nd, 2011
Accepted: November 15th, 2011

A Good Method
to Prepare and Use Transparencies
for Research Presentations

Veljko Milutinović

Department of Computer Engineering,
School of Electrical Engineering,
University of Belgrade,
POB 35-54, 11120 Belgrade, Serbia, Yugoslavia

Phone: +381-11-769872, Fax: +381-11-769872
E-mail: vm@etf.bg.ac.yu
WWW: http://galeb.etf.bg.ac.yu/~vm/

Σ

Introduction

- Tuning the audience to the subject
- Presenting the forest to the audience
- Preventing misunderstandings of any kind

☺

The Ten Compile-Time Commandments

- Introduction (to tune the reader)
- Problem statement (and why it is important)
- Existing solutions (and their criticism)
- Proposed solution (and its essence)
- Conditions and assumptions (of the analysis to follow)
- Details of the solutions to be compared ($1 + k$)
- Formal analysis (or analytical modeling)
- Simulation analysis (to show performance)
- Implementation analysis (to show complexity)
- Conclusion (from the performance/complexity viewpoint)

The Magnificent Seven

- Elements of the set
- Operations
- Functions
- Axioms
- Lemmas
- Theorem(s)
- Corollaries

Font Size and Line Count: Bad

- Simple superscalar going after high clock rate; four instructions fetching from an 8Kbyte I cache, and placed in one of two I buffers (four instructions in each one).
- Issue goes from the buffers (in-order, no instructions bypassing at all); one must be empty before the other one can be used (simpler design, but slower issue).
- Branches predicted using a 2-bit history table associated within the I cache controller; I issue stalled if 2nd branch encountered before the 1st branch resolved.
- After fetch and decode, instructions are arranged according to the functional unit they will use; issue follows after the operands are ready (from registers or via bypassing).
- Four FUs: int with shifter, int with branchevalor, FP adder, FP multiplier; integer instructions update registers in-order; FP instructions update files out-of-order; not all FP exceptions are precise.
- On-chip primary caches: 8Kbytes each (I + D, dm for 1cp cache access); a 6-entry MAF with MissAddresses/TargetRegs for up to 21 missed loads (merge).
- On-chip secondary cache: 96Kbytes (shared, 3w/sa); in most of the current microprocessors this cache memory is off the chip.

Font Size and Line Count: Good

- Simple superscalar (four)
- Branch prediction issues (2-bit history table)
- On-chip caches (primary and secondary)

;A picture is worth a 1000 words

Automatic Line Breaks: Bad

- Fixed/variable allocation scenarios based on the home property (page manager): DSM + DSIO system approaches
- Writes get satisfied on distance or locally, depending on what brings better performance and smaller complexity
- Good if reads and writes are interleaved with similar probabilities of occurrence

Semantic Line Breaks: Good

- Fixed/variable allocation scenarios
  based on the home property (page manager):
  DSM + DSIO system approaches
- Writes get satisfied on distance or locally,
  depending on what brings better performance
  and smaller complexity
- Good if reads and writes
  are interleaved
  with similar probabilities of occurrence

The Ten Run-Time Commandments

- *Do not hesitate* to redo the presentation desk prior to session start, in order to optimize your efficiency and comfort.
- *Make sure* that you look good during the presentation.
- *Let* the introductory transparency be there while you are being introduced.
- *Check* the screen after each new transparency.
- *Point* at the wall, not at the transparency.
- *Avoid* the SOS words.
- *Watch* your time!!!
- *Remember*, discussion is where you show who you really are.
- *Do not promise*; just suggest the audience to ask by e-mail.
- *Reload* the introductory transparency immediately after you finish.

Conclusion

- Performance versus complexity
- Who will benefit
- Newly open problems…

☺

How to write a great research paper

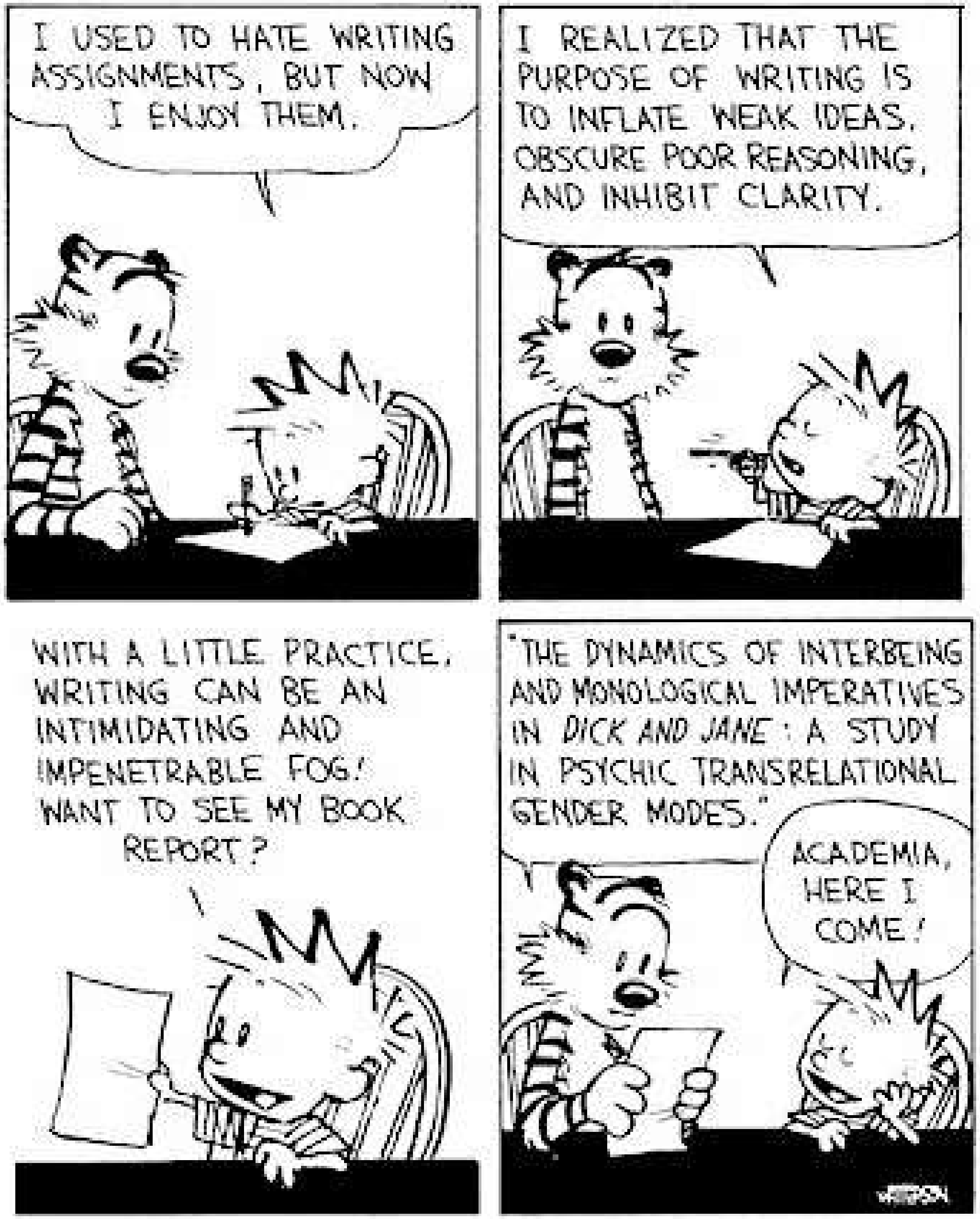
Simon Peyton Jones

Microsoft Research, Cambridge

Why bother?

Fallacy

we write papers and give talks mainly to impress others, gain recognition, anc get promoted

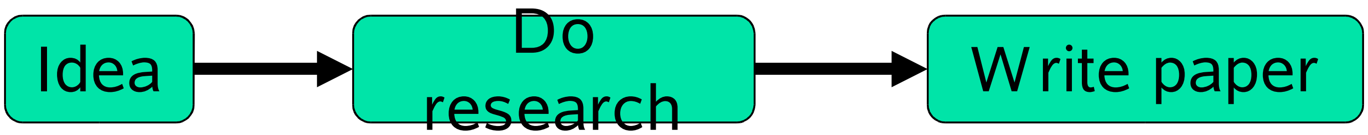

I USED TO HATE WRITING ASSIGNMENTS, BUT NOW I ENJOY THEM.
I REALIZED THAT THE PURPOSE OF WRITING IS TO INFLATE WEAK IDEAS, OBSCURE POOR REASONING, AND INHIBIT CLARITY.
WITH A LITTLE PRACTICE, WRITING CAN BE AN INTIMIDATING AND IMPENETRABLE FOG! WANT TO SEE MY BOOK REPORT?
"THE DYNAMICS OF INTERBEING AND MONOLOGICAL IMPERATIVES IN DICK AND JANE: A STUDY IN PSYCHIC TRANSRELATIONAL GENDER MODES."
ACADEMIA, HERE I COME!
WATTERSON

Papers communicate ideas

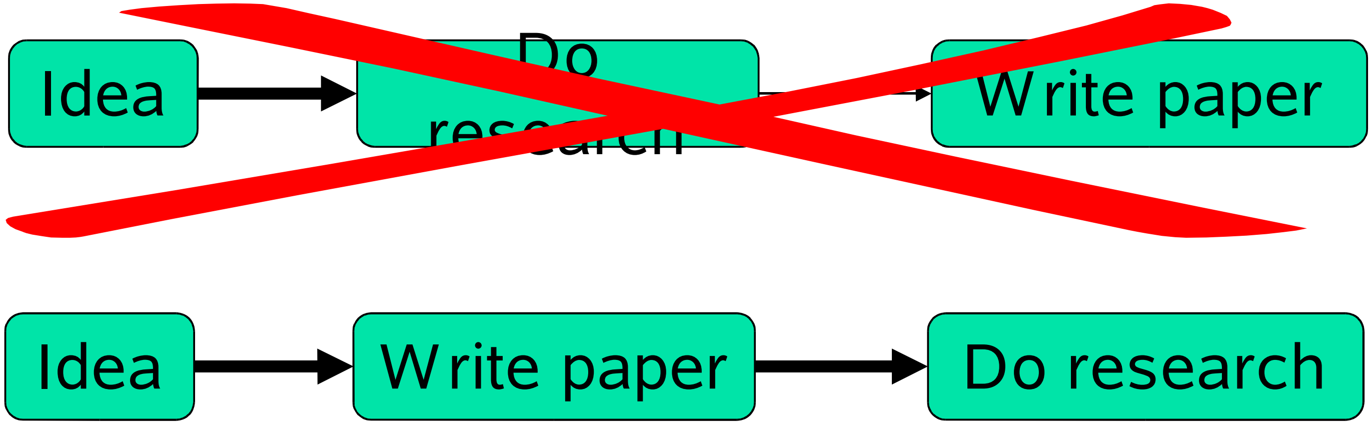
- Your goal: to infect the mind of your reader with **your idea**, like a virus
- Papers are far more durable than programs (think Mozart)

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

Writing papers: model 1



Writing papers: model 2



~~Idea → Do research → Write paper~~

Idea → Write paper → Do research

- Forces us to be clear, focused
- Crystallises what we don't understand
- Opens the way to dialogue with others: reality check, critique, and collaboration

Do not be intimidated

Fallacy You need to have a fantastic idea before you can write a paper or give a talk. (Everyone else seems to.)

Write a paper,
and give a talk, about
any idea,
no matter how weedy and insignificant it may seem to you

Do not be intimidated

Write a paper, and give a talk, about any idea, no matter how insignificant it may seem to you

- **Writing the paper is how you develop the idea in the first place**
- It usually turns out to be more interesting and challenging that it seemed at first

The purpose of your paper

The purpose of your paper is...

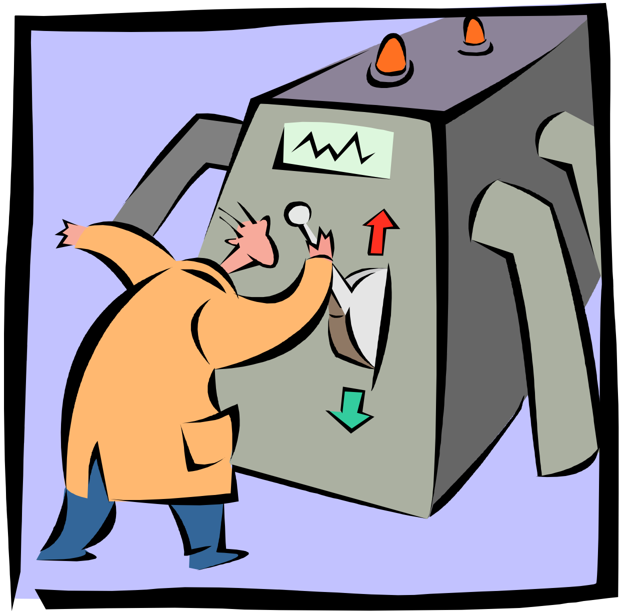
To convey your idea

...from your head to your reader's head

Everything serves this single goal

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

Conveying the idea

- Here is a problem
- It's an interesting problem
- It's an unsolved problem
- **Here is my idea**
- My idea works (details, data)
- 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

- Abstract (4 sentences)
- Introduction (1 page)
- The problem (1 page)
- My idea (2 pages)
- The details (5 pages)
- Related work (1-2 pages)
- Conclusions and further work (0.5 pages)

The abstract

- I usually write the abstract last
- Used by program committee members to decide which papers to read
- 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

Example

1. Many papers are badly written and hard to understand
2. This is a pity, because their good ideas may go unappreciated
3. Following simple guidelines can dramatically improve the quality of your papers
4. Your work will be used more, and the feedback you get from others will in turn improve your research

Structure

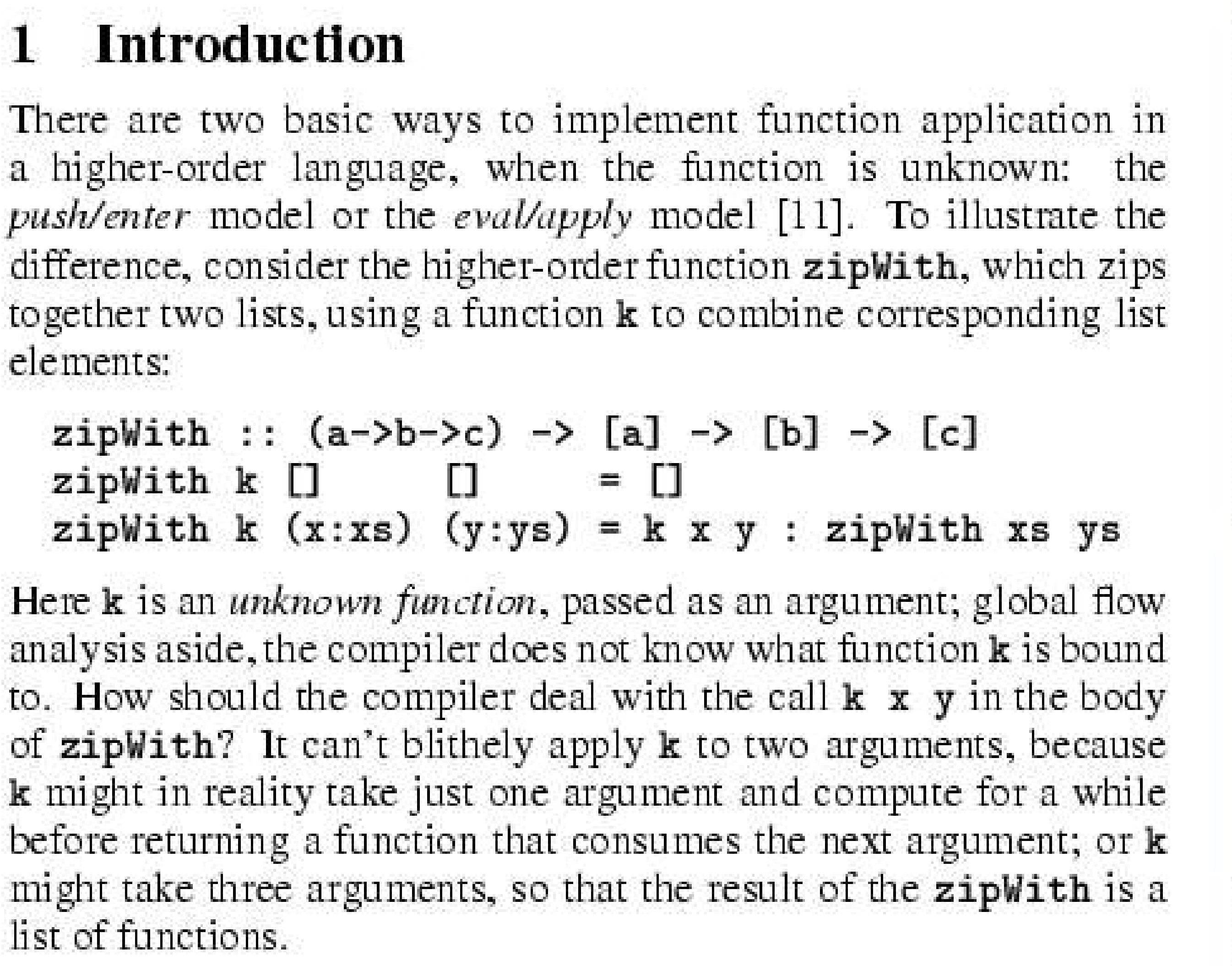
- Abstract (4 sentences)
- **Introduction** (1 page)
- The problem (1 page)
- My idea (2 pages)
- The details (5 pages)
- Related work (1-2 pages)
- Conclusions and further work (0.5 pages)

The introduction (1 page)

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
```

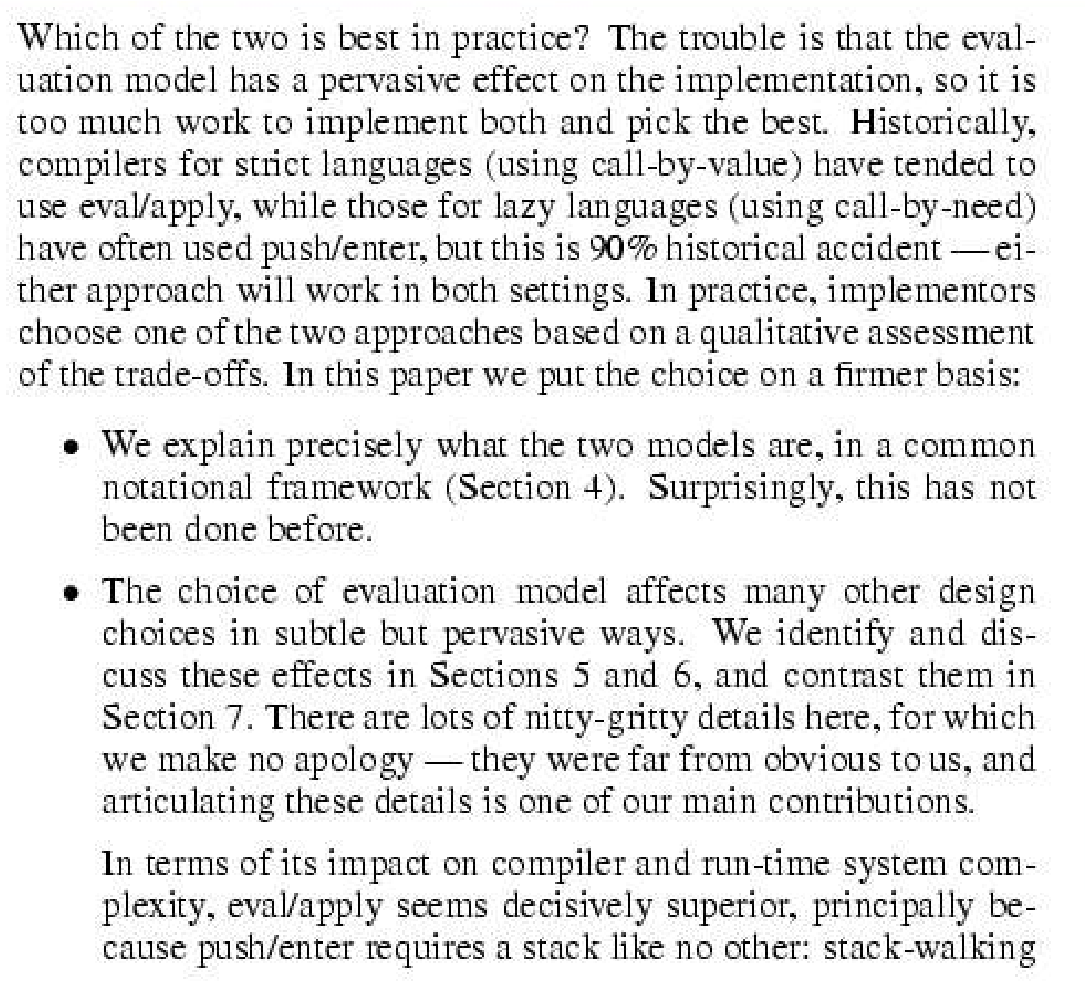
Here **k** is an *unknown function*, passed as an argument; global flow analysis aside, the compiler does not know what function **k** is bound to. How should the compiler deal with the call **k x y** in the body of **zipWith**? It can't blithely apply **k** to two arguments, because **k** might in reality 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"

State your contributions



Which of the two is best in practice? The trouble is that the evaluation model has a pervasive effect on the implementation, so it is too much work to implement both and pick the best. Historically, compilers for strict languages (using call-by-value) have tended to use eval/apply, while those for lazy languages (using call-by-need) have often used push/enter, but this is 90% historical accident — either approach will work in both settings. In practice, implementors choose one of the two approaches based on a qualitative assessment of the trade-offs. In this paper we put the choice on a firmer basis:

- We explain precisely what the two models are, in a common notational framework (Section 4). Surprisingly, this has not been done before.
- The choice of evaluation model affects many other design choices in subtle but pervasive ways. We identify and discuss these effects in Sections 5 and 6, and contrast them in Section 7. There are lots of nitty-gritty details here, for which we make no apology — they were far from obvious to us, and articulating these details is one of our main contributions.

  In terms of its impact on compiler and run-time system complexity, eval/apply seems decisively superior, principally because push/enter requires a stack like no other: stack-walking

Bulleted list of contributions

Do not leave the reader to guess what your contributions are!

Contributions should be refutable

We describe the WizWoz system. It is really cool.	We give the syntax and semantics of a language that supports concurrent processes (Section 3). Its innovative features are...
We study its properties	We prove that the type system is sound, and that type checking is decidable (Section 4)
We have used WizWoz in practice	We have built 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.

No "rest of this paper is…"

- Not:

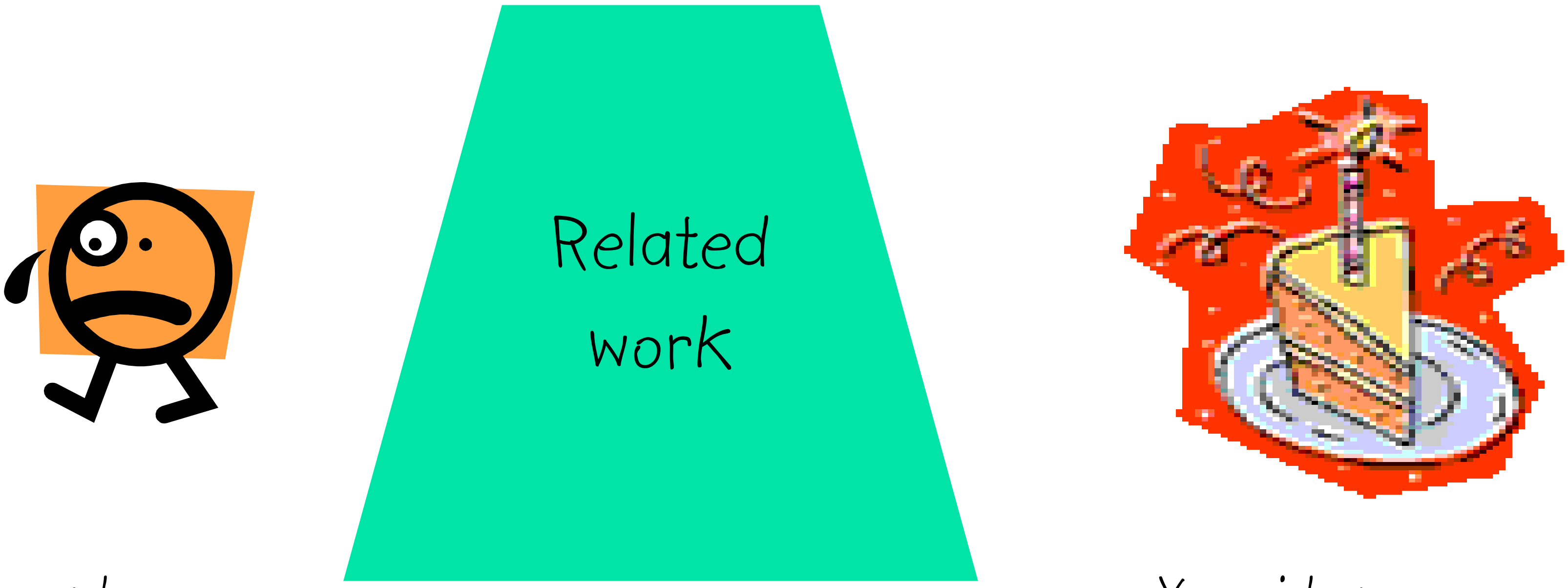
> "The rest of this paper is structured as follows. Section 2 introduces the problem. Section 3 … Finally, Section 8 concludes".

- Instead, **use forward references from the narrative in the introduction**.
  The introduction (including the contributions) should survey the whole paper, and therefore forward reference every important part.

Structure

- Abstract (4 sentences)
- Introduction (1 page)
- The problem (1 page)
- My idea (2 pages)
- The details (5 pages)
- Related work (1-2 pages)
- Conclusions and further work (0.5 pages)

No related work yet!



Related work

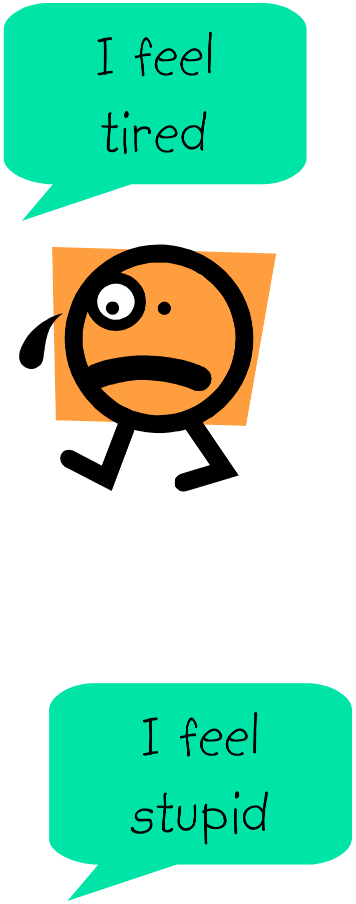
Your reader

Your idea

We adopt the notion of transaction from Brown [1], as modified for distributed systems by White [2], using the four-phase interpolation algorithm of Green [3]. Our work differs from White in our advanced revocation protocol, which deals with the case of priority inversion as described by Yellow [4].

No related work yet

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



Instead…

Concentrate single-mindedly on a narrative that

- Describes the problem, and why it is interesting
- Describes your idea
- Defends your idea, showing how it solves the problem, and filling out the details

On the way, cite relevant work in passing, but defer discussion to the end

The payload of your paper

Consider a bufircuated semi-lattice D, over a hyper-modulated signature S. Suppose pi is an element of D. Then we know for every such p_i there is an epi-modulus j, such that $p_j < p_i$.

- Sounds impressive…but
- Sends readers to sleep
- In a paper you MUST provide the details, but FIRST convey the idea

The payload of your paper

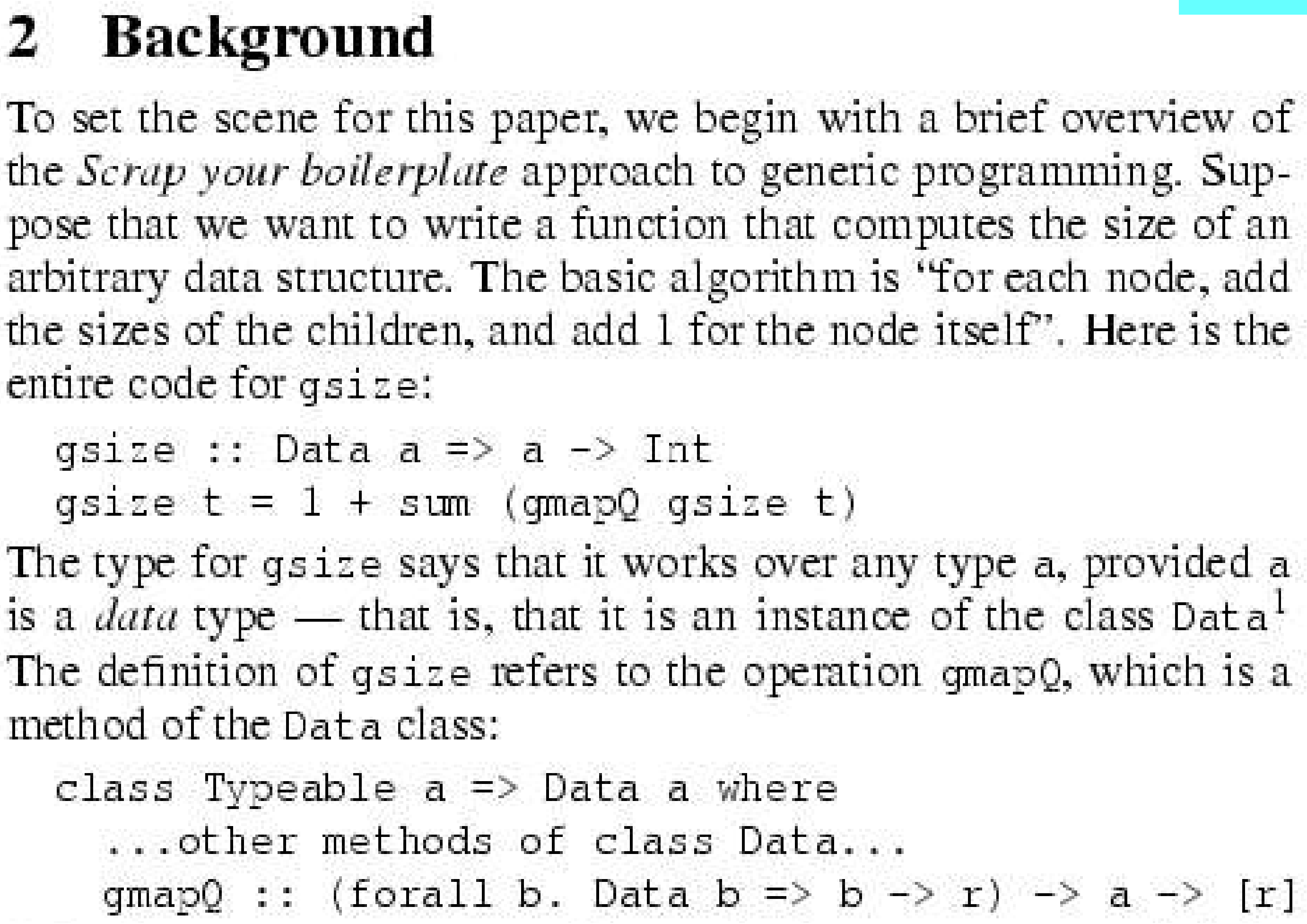
Introduce the problem, and your idea, using

EXAMPLES

and only then present the general case

Using examples

The Simon PJ question: is there any typewriter font?



2 Background

To set the scene for this paper, we begin with a brief overview of the *Scrap your boilerplate* approach to generic programming. Suppose that we want to write a function that computes the size of an arbitrary data structure. The basic algorithm is "for each node, add the sizes of the children, and add 1 for the node itself". Here is the entire code for gsize:

```
gsize :: Data a => a -> Int
gsize t = 1 + sum (gmapQ gsize t)
```

The type for gsize says that it works over any type a, provided a is a *data* type — that is, that it is an instance of the class Data[1] The definition of gsize refers to the operation gmapQ, which is a method of the Data class:

```
class Typeable a => Data a where
  ...other methods of class Data...
  gmapQ :: (forall b. Data b => b -> r) -> a -> [r]
```

Example right away

Conveying 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

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

- Abstract (4 sentences)
- Introduction (1 page)
- The problem (1 page)
- My idea (2 pages)
- The details (5 pages)
- **Related work** (1-2 pages)
- Conclusions and further work (0.5 pages)

Related work

Fallacy

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

The truth: credit is not like money

Giving credit to others does not diminish the credit you get from your paper

- Warmly acknowledge people who have helped you
- Be generous to the competition. “In his inspiring paper [Foo98] Foogle shows.... We develop his foundation in the following ways...”
- Acknowledge weaknesses in your approach

Credit is not like money

Failing to give credit to others can kill your paper

If you imply that an idea is yours, and the referee knows it is not, then either

- You don't know that it's an old idea (bad)
- You do know, but are pretending it's yours (very bad)

Making sure related work is accurate

- A good plan: when you think you are done, send the draft to the competition saying "could you help me ensure that I describe your work fairly?".
- Often they will respond with helpful critique
- They are likely to be your referees anyway, so getting their comments up front is jolly good.

The process

- Start early. Very early.
  - Hastily-written papers get rejected.
  - Papers are like wine: they need time to mature
- Collaborate
- Use CVS to support collaboration

Getting help

Get your paper read by as many friendly guinea pigs as possible

- Experts are good
- Non-experts are also very good
- Each reader can only read your paper for the first time once! So use them carefully
- Explain carefully what you want ("I got lost here" is much more important than "wibble is mis-spelt".)

Listening to your reviewers

Every review is gold dust

Be (truly) grateful for criticism as well as praise

This is really, really, really hard

But it's really, really, really, really, really, really important

Listening to your reviewers

- Read every criticism as a positive suggestion for something you could explain more clearly
- DO NOT respond "you stupid person, I meant X". Fix the paper so that X is apparent even to the stupidest reader.
- Thank them warmly. They have given up their time for you.

Language and style

Basic stuff

- Submit by the deadline
- Keep to the length restrictions
  - Do not narrow the margins
  - Do not use 6pt font
  - On occasion, supply supporting evidence (e.g. experimental data, or a written-out proof) in an appendix
- Always use a spell checker

Visual structure

- Give strong visual structure to your paper using
  - sections and sub-sections
  - bullets
  - italics
  - laid-out code
- Find out how to draw pictures, and use them

Visual structure

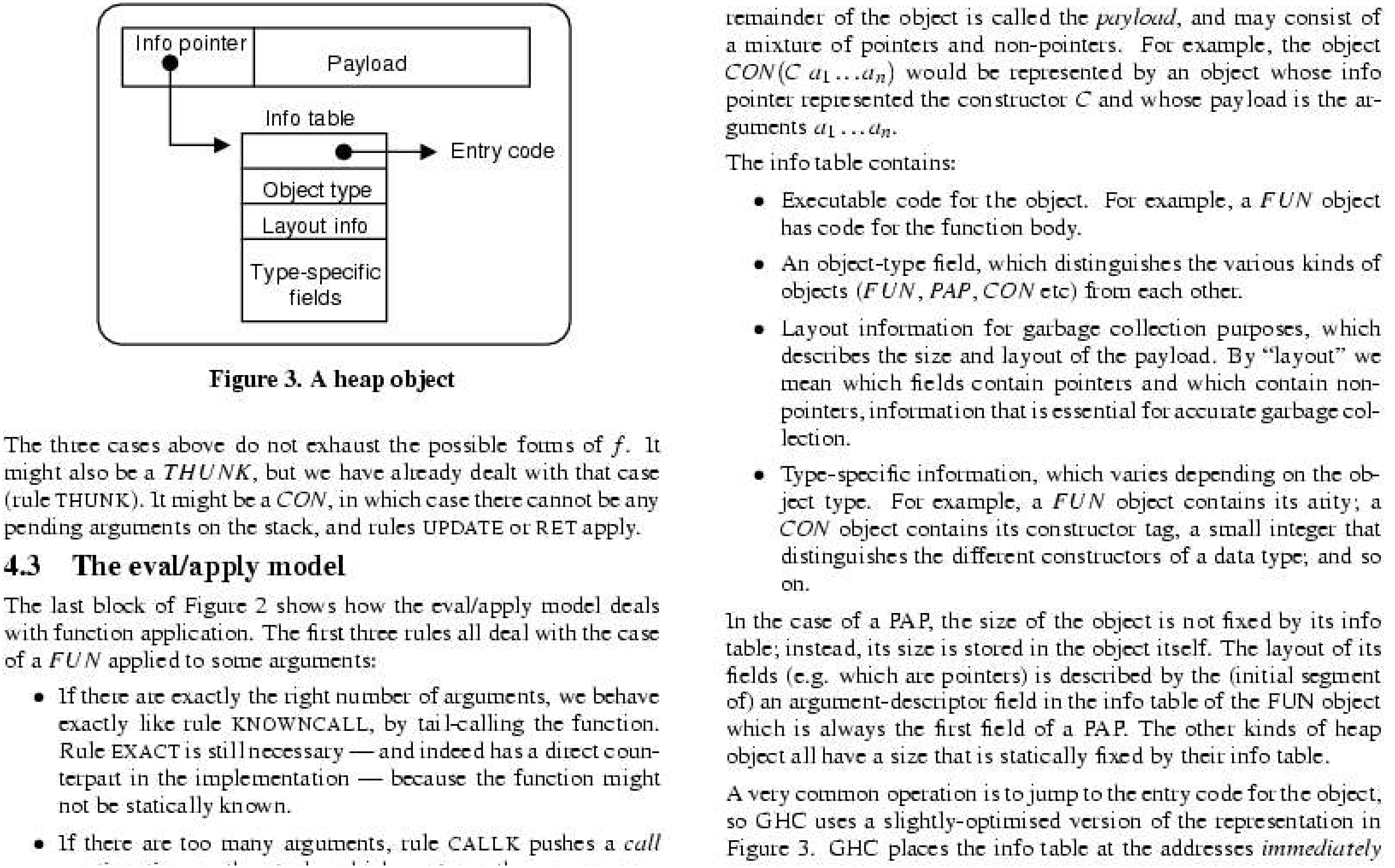

Info pointer
Payload
Info table
Entry code
Object type
Layout info
Type-specific fields

Figure 3. A heap object

The three cases above do not exhaust the possible forms of f. It might also be a *THUNK*, but we have already dealt with that case (rule THUNK). It might be a *CON*, in which case there cannot be any pending arguments on the stack, and rules UPDATE or RET apply.

4.3 The eval/apply model

The last block of Figure 2 shows how the eval/apply model deals with function application. The first three rules all deal with the case of a *FUN* applied to some arguments:

- If there are exactly the right number of arguments, we behave exactly like rule KNOWNCALL, by tail-calling the function. Rule EXACT is still necessary — and indeed has a direct counterpart in the implementation — because the function might not be statically known.
- If there are too many arguments, rule CALLK pushes a *call*

remainder of the object is called the *payload*, and may consist of a mixture of pointers and non-pointers. For example, the object $CON(C\ a_1 \ldots a_n)$ would be represented by an object whose info pointer represented the constructor C and whose payload is the arguments $a_1 \ldots a_n$.

The info table contains:

- Executable code for the object. For example, a *FUN* object has code for the function body.
- An object-type field, which distinguishes the various kinds of objects (*FUN*, *PAP*, *CON* etc) from each other.
- Layout information for garbage collection purposes, which describes the size and layout of the payload. By "layout" we mean which fields contain pointers and which contain non-pointers, information that is essential for accurate garbage collection.
- Type-specific information, which varies depending on the object type. For example, a *FUN* object contains its arity; a *CON* object contains its constructor tag, a small integer that distinguishes the different constructors of a data type; and so on.

In the case of a PAP, the size of the object is not fixed by its info table; instead, its size is stored in the object itself. The layout of its fields (e.g. which are pointers) is described by the (initial segment of) an argument-descriptor field in the info table of the FUN object which is always the first field of a PAP. The other kinds of heap object all have a size that is statically fixed by their info table.

A very common operation is to jump to the entry code for the object, so GHC uses a slightly-optimised version of the representation in Figure 3. GHC places the info table at the addresses *immediately*

Use the active voice

The passive voice is "respectable" but it DEADENS your paper. Avoid it at all costs.

NO	YES
It can be seen that…	We can see that…
34 tests were run	We ran 34 tests
These properties were thought desirable	We wanted to retain these properties
It might be thought that this would be a type error	You might think this would be a type error

"We" = you and the reader

"We" = the authors

"You" = the reader

Use simple, direct language

NO	YES
The object under study was displaced horizontally	The ball moved sideways
On an annual basis	Yearly
Endeavour to ascertain	Find out
It could be considered that the speed of storage reclamation left something to be desired	The garbage collector was really slow

Summary

If you remember nothing else:

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

A good starting point:

"Advice on Research and Writing"

http://www-2.cs.cmu.edu/afs/cs.cmu.edu/user/mleone/web/how-to.html