Crash course
Paper writing and reviewing
for first year part-timers

Bas van Vlijmen

11 February 2013

Goal of this paper session

1. The main goal: Video me!
    1. I use this course for the UvA Teacher Training Program
    2. Meet our guest, biologist Rob de Heus
2. Preparation for Software Construction Paper + Review (who has experience with both?)
3. Guide you through self-study material on papers writing and reviewing
4. Practise with some small excercises
5. Overload you with sheets: one part we'll discuss, **others to browse through yourself (greyed out).**

Plan for this paper session

1. Positioning this session
2. Assignment for Software Construction
3. Topics in this session:
   A. What is a paper? Why write one?
   B. How to go about when writing a paper
   C. Writing, style
   D. Argumentation
   E. Literature
   F. Reviewing
4. Fill in evaluation form

1. Position of this paper session

- Looking back at paper sessions in block 1st semester (part-timers):
  - Session 1: Structure of papers, topics
  - Session 2: Fraude and plagiarism + finding Parnas' key papers by online search for appreciative and other statements
  - Session 3: read 5 papers, write 5-line summaries, and discuss relation to the course Software Architecture

Position of this paper session (cont.)

- How does this session fit in? What's the goal?
  - help you get going with paper assignment for Software Construction
  - point you to material for self-study
  - appreciate writing as a creative, intellectually challenging and fun
- Next (second) year…
  - the paper sessions of the second year will repeat the material
  - essay / paper for Software Evolution
  - proposal for your thesis
  - your thesis
- In your career:
  - not just do it, but also to know how to reflect on software engineering and use literature to your benefit. Share your insights with others.

- For inspiring examples texts/papers/essays that reflect on Software Engineering see:
  - part 1 of *Making Software*, edited by Andy Oram and Greg Wilson, O'reilly, 2011.
- This book will be used too in the course Software Process

2. Assignments for Software Construction

- Write a 3 to 5 page paper
- On the course website there's a list of topics:
  - '*Structured programming with or without gotos?*'
  - '*State Transactional Memory*'
  - et cetera.
- Review the papers of one of your fellow students
- Time is pressing… Deadline for the paper: Sunday, March 3

3. Crash course topics

A. What is a paper? Why write one?
B. How to go about when writing a paper?
C. Writing and style
D. Argumentation
E. Literature
F. Reviewing, reading, making annotations

A. What is a paper? Why write one?

- ....
- ....
- ....
- ....
- ....
- ....

What is a paper? Why write one?

- means of communication
- present NEW ideas, observations, facts, knowledge, insights
- position ideas, observations, facts, knowledge, insights
- debate
- archive ideas, observations, facts, knowledge, insights
- means to structure your thoughts
- document the scientific process
- building block in a larger scientific structure
- ….
- (…. more mundane: get and keep a job at the university; show you deserve a bachelor or master title)

Structural elements of a paper

- As discussed in previous sessions:
  - title
  - authors + affiliation
  - abstract
  - table of contents / figures / tables
  - introduction / motivation
  - research question / definitions
  - related work
  - method
  - exposition of work done, findings
  - conclusion
  - validation
  - future work
  - references
  - glossary
  - appendices

RUDY HIRSCHHEIM AND MARY LACITY

THE MYTHS AND REALITIES OF INFORMATION TECHNOLOGY SOURCING

...gers commiserate over the challenges of ...ior executives that, contrary to popular belief, ...cing isn't always a money-saving option.

outsourcing—the prac- assets, leases, staff, and ity for delivery of ser- unctions to third-party leniable trend ever since ion. In recent years, pri- zations worldwide have s of their IT functions, ace, British Petroleum, Manhattan Bank, Con- Bank, Enron, First City, General Dynamics, Inland Revenue, JP Morgan, Kodak, Lufthansa, McDonnell Douglas, South Australian Government, Swiss Bank, Xerox, and Commonwealth Bank of Australia. The IT outsourcing market, which was worth $76 billion in 1995, grew to over $120 billion in 1997 [5].

As the market evolves, a number of important aspects of these IT outsourcing decisions have been explored. These studies can be categorized as descriptive case studies and surveys of the current outsourcing practices, surveys of practitioners' perceptions of risks

Kees Vreugdenhil, Gerard Alberts, Pieter van Gelder — Waterloopkunde — NAW 5/2 no. 3 september 2001 — 267

Kees Vreugdenhil
Universiteit Twente, Faculteit Technische en Management, Afdeling Civiele Techniek, Postbus 217, 7500 AE Enschede
c.b.vreugdenhil@ctw.utwente.nl

Gerard Alberts
Wetenschap en Samenleving, Katholieke Universiteit Nijmegen, Centrum voor Wiskunde en Informatica, Postbus 94079, 1090 GB Amsterdam
g.alberts@uva.nl

Pieter van Gelder
Technische Universiteit Delft, Faculteit Civiele Techniek en Geowetenschappen, Postbus 5048, 2600 GA Delft
P.vanGelder@ct.tudelft.nl

Een eeuw wiskunde en werkelijkheid

Waterloopkunde

Onderzoekers van het Waterloopkundig Lab, de Deltadienst van Rijkswaterstaat, TNO en het Mathematisch Centrum bestudeerden in vrodige competitie het dynamisch gedrag van de Haringvlietsluizen. De uiteenlopende studies om de wiskunde in dienst te stellen van de waterhouwkunde, schaalmodellen, analogons, wiskundige modellen en rekenwerk, kwamen in een zogenaamde klap-luh samen. De beslissing bleef uit. Korte tijd later namen al deze inspanningen op de achtergrond door de revolutie in de richting van een heel nieuw type rekenmodellen.

In de twintigste eeuw veranderde in de natte civiele techniek de relatie tussen wiskunde en werkelijkheid ingrijpend. Eerst geleidelijk, later met grote omwentelingen. Manieren de civiel ingenieurs zich in de eerste helft van de eeuw los van het zoeken naar waarheid met wiskunde om te kiezen voor een meer pragmatische aanpak, de laatste jaren lijkt wondertlijk genoeg dan(k)j de computer het waarheidstreven teruggekeerd.

Foto p. 266. De doorbraak van de dijk tussen Ablasserdam en Papendrecht tijdens de watersnoodramp van 1953. Foto: Spaarnestad Fotoarchief.

Hydrodynamici en waterloopkundigen

Het wiskundig modelleren mag een betrekkelijk eeuwwise uitvinding zijn (zie [1], [2]), de toepassing van wiskunde op natuurlijke verschijnselen is veel ouder begonnen. Op het gebied van waterstromingen is er met name in de 18e eeuw een enorme ontwikkeling geweest. Voornaamstande wiskundigen als Bernoulli, Euler, d'Alembert en Laplace bereiken in de theorie van de hydrodynamica resultaten die tot vandaag van belang zijn, getuige de vergelijkingen van Laplace en van Euler. Deze resultaten waren nogal theoretisch en hadden hun beperkingen. Zo was de getijdentheorie van Laplace alleen geldig voor een oceaan die de hele aardbol bedekt – de bruikbaarheid voor concrete situaties als de Noordzee was dan beperkt. Een nog beruchter voorbeeld is de paradox van d'Alembert die zegt dat op een lichaam, geplaatst in een stroming – een potentiaalstroming – geen netto kracht wordt uitgeoefend. Dit is uiteraard in eenlig conflict met de dagelijkse waarneming.

In de loop van de negentiende eeuw begon zich een on derschei d at e tekenen tussen hydrodynamici en hydraulici. Hydrodynamici waren mensen als Laplace die mooie theorieën maakten zonder veel praktische waarde. Met hydraulici werden de mensen bedoeld die praktische waterloopkundige problemen oplosten [25], ingenieurs in de traditie van Vlierlingh, Leeghwater en Stevin. Zo deden Chézy en Manning waarnemingen aan de stromingsweerstand in een leiding, Darcy aan de stroming in de grond, Froude aan weerstand van varende schepen. Wiskunde kwam hier nauwelijks aan te pas, anders dan in empirische relaties die we tegenwoordig als 'curve fitting' zouden beschouwen. Ook tegenwoordig wordt deze aanpak nog wel gevolgd voor het beschrijven van moeilijk te doorgronden verschijnselen, zoals zandtransport in stromend water, zij het dat nu de gelijkvormigheidstheorie een wat betere theoretische basis biedt om vast te stellen welke dimensieloze combinaties van parameters een rol kunnen spelen.

In de negentiende eeuw stonden het theoretische zoeken naar waarheid en het oplossen van praktische problemen tegenover elkaar. In de twintigste eeuw kwam er een kentering en lukte het geleidelijk aan om verbindingen te leggen tussen de theorie van de hydrodynamica en de praktijk van de waterbouw. Vanuit de praktijk trad een zekere wetenschappelijkheid in. Vanuit de

On the Definition of a Theoretical Concept of an Operating System

J.A. Bergstra and C.A. Middelburg
Informatics Institute, Faculty of Science, University of Amsterdam,
Science Park 107, 1098 XG Amsterdam, the Netherlands
J.A.Bergstra@uva.nl, C.A.Middelburg@uva.nl

Abstract. We dwell on how a definition of a theoretical concept of an operating system, suitable to be incorporated in a mathematical theory of operating systems, could look like. This is considered a valuable preparation for the development of a mathematical theory of operating systems.

1 Introduction

Presently, operating systems are a hot topic in the sector of information and communication technologies. General-purpose operating systems that have been developed for desktop computers or laptop computers are not suitable for mobile devices, such as smartphones, personal digital assistants, personal navigation devices and e-book readers, due to the special needs of these devices, such as regulation of power consumption to prolong battery life and real-time responses for time-critical applications. Therefore, the increasing importance of mobile devices has triggered the development of many mobile operating systems. There is really a very strong competition going on among various major companies from the sector of information and communication technologies in a bid for the most successful mobile operating system (cf. [12]).

We expect that a theoretical understanding of the concept of an operating system will become increasingly important to the development of successful operating systems. However, it happens that in computer science since the introduction of the first operating systems more than fifty years ago no serious attention has been paid to the clarification of what is an operating system. Almost any explanation of the concept fails to capture the concept of an operating system satisfactorily. The existing theoretical understanding related to operating systems concerns details of the functioning of operating systems, such as scheduling the programs in execution and allocating resources to the programs in execution, and shows little coherence.

We became fully aware of the state of affairs outlined above only after an extensive search for publications on operating systems recently carried out by one of us, which is reported on in [14]. This state of affairs forms our motivation to have a closer look at the concept of an operating system. In this note, we dwell on how a definition of a theoretical concept of an operating system, suitable to

What and what ordering will probably be found in your Software Construction paper?

- title
- authors + affiliation
- abstract
- table of contents / figures / tables
- introduction / motivation
- research question / definitions
- related work
- method
- exposition of work done, findings
- conclusion
- validation
- future work
- references
- glossary
- appendices

To keep in mind …

- The form of the papers we write is ‘new’
- Emphasis on mathematical underpinning was matter of debate 80 to 50 years ago, even in something as Civil Engineering!
- 50 years ago a more essayistic style was quite common
- The amount of papers written per person per year was much lower in the past (… more time for reading and pondering)
- Books were more prominent
- Styles still differ per scientific discipline
- *So, the form is not that precisely defined. Don’t let the format hamper you to put your thoughts on paper.*

‘Papers’, a long tradition

- The republic of letters (Renaissance 15th century)
- Time of the first printing press (Gutenberg)
- Famous ‘paper’ author: Erasmus (1469-1536)
  - *The Adages of Erasmus*, William Barker ed., Univ. of Toronto Press, 2001.

The ‘invention’ of the Essay

- Michel de Montaigne (1533 – 1592)

- Background material:
  - *Les Essays*, Michel de Montaigne; has many translations
  - Amazon/google: Philosophy/history of Science
  - *De ontwikkeling van wetenschap* (in Dutch) by Gerard de Vries
  - *A New History of the Humanities* by Rens Bod

B. How to go about when writing?

- Impressed by those shiny papers in top journals?
- Scientific writing is most of the time a *step by step* process
- Polished papers have often come a long long way:
  - first presentation of a draft idea on a conference,
  - reworking the idea for a proceeding,
  - rewriting and extending the paper for a journal or a book chapter,
  - then a next paper on the same subject, slightly different...
- Peer feedback/review is given in each step: on contents, style, structure.
- *So... don't be afraid to jot something down and get feedback! Even for this assignment for Soft. Construction*

It starts with an appealing hunch

- Go to the list of subjects at https://github.com/software-engineering-amsterdam/software-construction/wiki
- Pick a subject that attracts you (can't be true!, must be true!, how come?, What's this about?)
- Pick something strong that gives direction when you get stuck in the details
- Read the suggested papers
- Then …

then…

- devise a draft title
- write a concise *draft* introduction comprising:
  - motivation
  - definition
  - contribution
  - plan
- Background material on the iterative writing process:
  - *Overtuigend schrijven*, Frans van Eemeren et al., ThiemeMeulenhof, 2011
  - *Academic writing in English, Jeanne van Loon et al., Uitgeverij Coutinho, 2011.*

Modes of writing

- You'll develop your own style of working
- But, these modes of working will iteratively pass in some order:
  - creative mode (jot down pieces of tekst, ideas)
  - structuring mode (ordering, rewriting)
  - reader mode (put yourself in the position of your reader, your audience)
  - vacuum cleaner mode (layout, consistency of word use, spelling, typo's, interpunction)

Parts of the intro: Motivation

- Why a motivation?
  - giving excuses for invested time in writing? No. (we don't care you need to write something for this master)
  - providing reasons for Tijs (the reader) to read your paper!

Definition

- What is the topic?
- What is not the topic?

- What is a good definition?
  - what's the audience?
  - what background knowledge could you expect?
  - what literature could you refer to?
  - find balance between introduction elaborations in others sections

Position

- What is your perspective? What do you think?
- In what sense is this paper “unique” or “special”?
- Position by relating your paper
  - to other papers (“related work”)
  - to what existed before, to current reality
  - to what the reader already knows
  - to an accepted contrary view

Contribution

- What new insights does the paper bring?
- What is the return on investment for reading?
- What are the lessons to learn?
- What are findings / conclusions?
- How {c,w,sh}ould your paper “change the world”?

- *Who knows examples of a papers with an outstanding contribution?*

Plan

- Plan of attack ("we'll first look at … for …")
- Outline of the paper
  - forward references to sections
  - explains global structure of the paper
- Helps the reader decide where to put energy

- A plan as part of the intro is not that relevant for your current assignment. Why?

C. Writing and style

Contents:

- title
- abstract
- introduction
- related work
- method
- work done, findings
- conclusion
- validation
- future work
- references
- appendices

Building blocks:

- (parts)
- (chapters)
- sections
- subsection
- subsubsection
- (…)
- **paragraphs**
- sentences
- words
- …

Paragraphs are containers for ideas

- Key features: unified, coherent, well-developed, flow
  - **unified**: when it deals with one topic.
  - **coherent**: when the reasoning and place of explanations are in a logical order
  - **well-developed**: when the paragraph contains enough concrete support for the main point raised in the topic sentence
  - **flow**: when sentences read easy one after the other (here's where style comes in)
- Length …
- We will now look at some examples:
  - simplicity
  - flow
  - coherence
  - connectors and puntuation
- See: Chapter 1 *Spring into technical writing*, chapter 4 *Academic writing in English*

Example 1 (simplicity)

- Do not confuse ‘difficult to understand’ with ‘deep thinking’
  - “An understanding of the causal factors involved in code cloning by programmers could lead to more effective prevention”
  - “We could prevent programmers to clone code more effectively if we new why they do”
- What is the difference?

○ Example 1, 2, 3 taken from: *Style - The Basics of Clarity and Grace*, Joseph M. Williams et al., Pearson Education, 2006

Example 2 (simplicity)

- Writing is telling a story
  - Make main characters subjects
  - Make important actions verbs
- "Once upon a time, as a walk in the woods was taking place on the part of Little Red Riding Hood, the Wolf's jump from behind a tree…"
- "Once upon a time, Little Red Riding Hood was walking in the forest, when the Wolf jumped from behind a tree…"

Example 3 (simplicity)

- "The argument of authors of code clone detection tools with respect to the harmfulness of copy and paste programming is based on their belief in the tendency of programmers of wanting to maintain each clone consistently"
- "The authors of code clone detection tools argue that copy and paste programming is harmful because they believe that programmers want to maintain each clone consistently"
- So, use subjects for actors and verbs for actions
- This makes your paragraphs less abstract and more concrete

Example 4 (flow)

- Chaining sentences:
  - "Programmers frequently introduce clones: exact copies of source code fragments. Later, they may forget to maintain such clones consistently"
  - "Programmers frequently introduce clones: exact copies of source code fragments. Such clones are often forgotten to be maintained consistently"
- Flow is especially important in longer paragraphs
- More tips and examples you find in:
  - *Overtuigend schrijven*, Frans van Eemeren et al., ThiemeMeulenhof, 2011
  - *Academic writing in English, Jeanne van Loon et al., Uitgeverij Coutinho, 2011*

Example 5 (coherence): What could be improved?

- "(1) Online gaming marks a new chapter in the progress of video games. (2) More and more video games have started to include network capabilities and its growth has followed the general trajectory of the computer industry – expansions of computer networks from small local networks to the internet. (3) Another crucial factor that led to the rapid development of online gaming was the growth of the internet itself and the increasing adoption of broadband connections. (4) Online games range from simple textbased games to games incorporating graphics to virtual worlds populated by thousands at one time."

  - Source: *Academic writing in English*

Example 6 (interpunction): What could be improved?

- “(1) Although female attendance of university studies is fairly high the ones who tend to be most successful in terms of career tend to be men. (2) The cause of this problem is mainly tradition; even today, women are expected to look after the children. (3) So companies are hesitant to hire women. (4) In this way a ‘glass ceiling’ is created; even for women who are very ambitious about their career and do not want to have children because a family is an obstacle in finding their way to the top. (5) Therefore a solution is needed to bring an end to prevailing stereotypes which prevent women from succeeding in scientific careers.”

  - source: *Academic Writing in English*

- “(1) Although female attendance of university studies is fairly high, the ones who tend to be most successful in terms of career tend to be men. (2) The cause of this problem is mainly tradition. Even today, women are expected to look after the children, so companies are hesitant to hire women. (4) In this way, a ‘glass ceiling’ is created even for women who are very ambitious about their career, and who do not want to have children because a family is an obstacle in finding their way to the top. (5) Therefore, a solution is needed to bring an end to prevailing stereotypes which prevent women from succeeding in scientific careers.”

Paragraph structure

- Topic sentence
- Reasons
- Examples
- Summary
  - *Academic writing. A handbook*

- Topic level
- Explanation level
- Illustration level
  - *Academic writing in English*

- Tell'm what you're going to tell 'm
- Tell'm
- Tell'm what you told them
  - *Spring into technical writing for engineers and scientists*

D. Argumentation

- Remember from your 'draft introduction':
  - motivation
  - position
  - contribution
- Here you phrased the claim (assertions, resarch questions, hypothesis) you made
- Argumentation = structured support for your claim + structured positioning/defense against of competing viewpoints

Why's argumentation so important in SE?

- Wicked problems
- [wikipedia] "'Wicked problem' is a phrase originally used in social planning to describe a problem that is difficult or impossible to solve because of incomplete, contradictory, and changing requirements that are often difficult to recognize"
- The term 'wicked' is used, not in the sense of evil but rather its resistance to resolution. Moreover, because of complex interdependencies, the effort to solve one aspect of a wicked problem may reveal or create other problems"
- Software Engineering 'is' a wicked problem
- Every part of software engineering probably too
- This is why argumentation is particularly interesting in our field

Example

- It has been clearly shown that the babies of mothers who drink a lot of coffee often have health problems in early pregnancy
- So, mothers should not drink coffee in early pregnancy, to prevent health issues of their babies
- What is wrong with this argument?
- What is an alternative explanation?
- When pregnancy hormone levels are low, the mother is not feeling sick and so she is able to drink coffee without feeling nausea. The low hormone levels are also indicative of health issues for the baby

Example

- “Code clones are parts of source code that when represented as an AST are equal to each other. We made a tool that can detect ASTs that are equal. Our tool detects all of these ASTs and no more, so it has maximum recall and precision”
- What is wrong with this argument?
  - This argument is true because the definition was made to fit the quality aspect. Trivially true and definitely uninteresting: circular.
- What do the authors want/need to change here?

Example

- “I have unit-tested every component to the bone. Let’s assume therefore that there are no more bugs in the components, so the system is bug-free”
- What’s wrong here?
- “My atoms are invisible. I am composed of atoms, so I am invisible”

Tips: Language

- Be aware when you are arguing
- When you encounter: …. if, when, then, because, why, consequentially, therefore, so, and, but, concluding, believe, assume, know, implies, ...
- Use these words to make explicit, step by step, what you are thinking
- Writing an argumentation often makes you discover that you do not understand yet. That's great!

Tips: Common sense

- Do not underestimate yourself
- Be your own audience
- Use your common sense and your curiosity:
  - O really?
  - But I saw the opposite happening yesterday!
  - Why? Why? Why? and … Why?

Tips: Assume

- Not all questions can be answered
- Stop where common sense allows you to assume
- For example: we (sometimes) assume that
  - better tools always lead to more efficiency
  - we have infinite amount of RAM
  - P != NP, etc.
- Can you give an example of a paper based on explicit assumptions?
- Famous example: steps towards proving Fermat's theorem (for n>2 there's no integer non–zero solution for $x^n + y^n = z^n$)
- Just state your assumptions, don't hide them!

Formalizing Argumentation

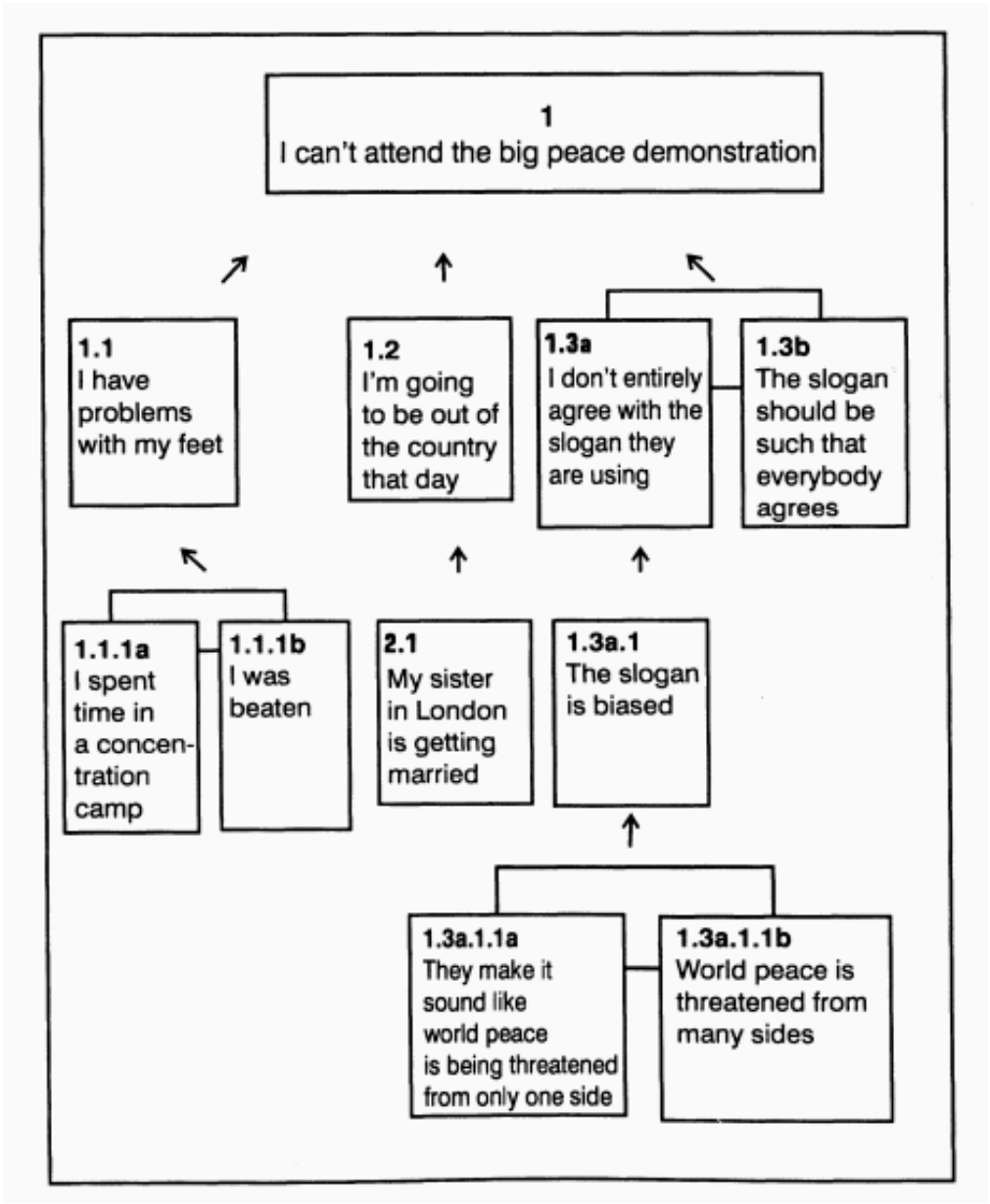
- Argumentation is a the heart of the scientific method
- Argumentation has a strong and historical relation with logic
- Take for instance, the famous syllogism:
  - All humans are mortal.
    Socrates is a human.
    So, Socrates is mortal.

Formalizing Argumentation

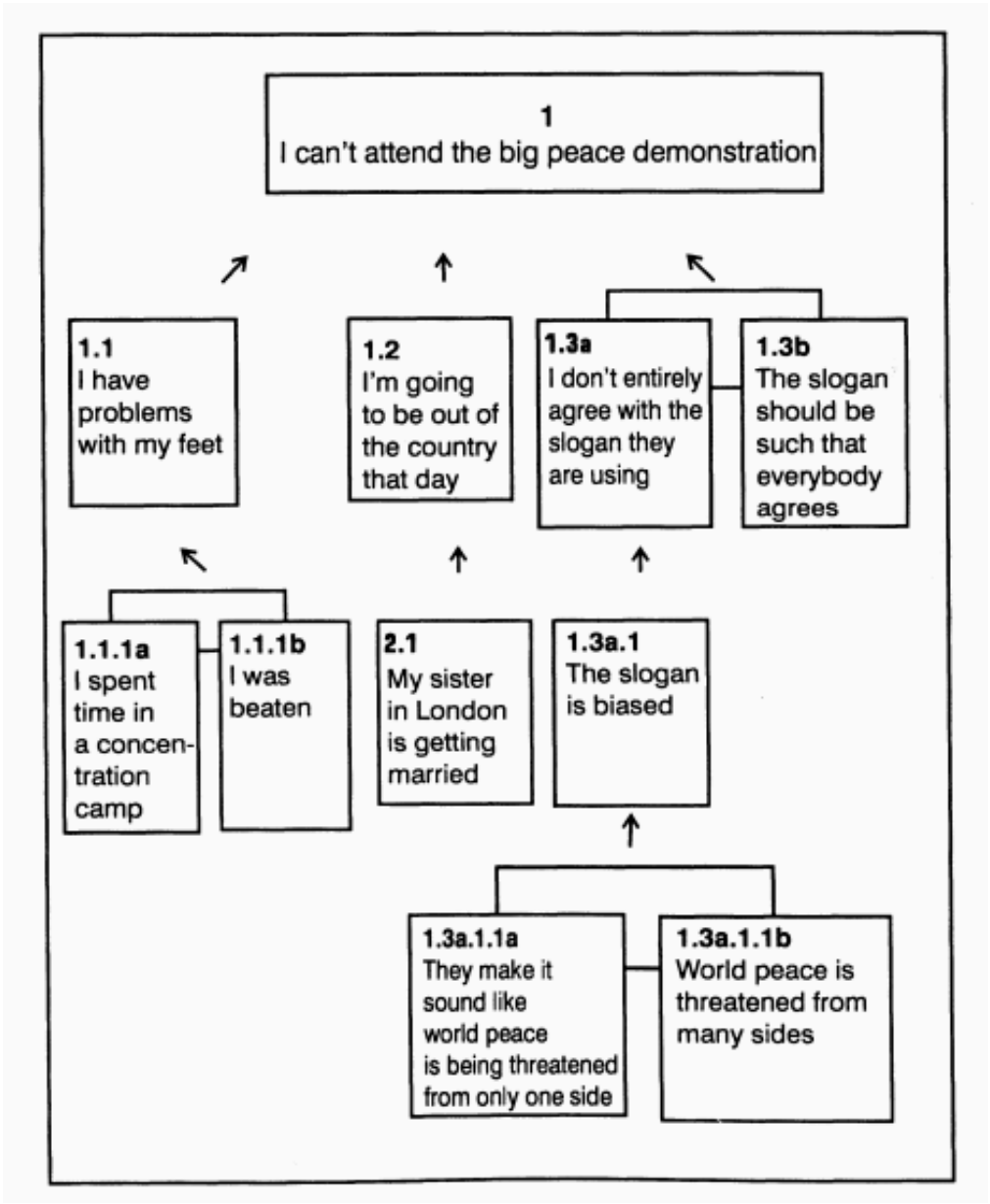
- Getting for observation A to conclusion B:
  - Find a plausible explanation to observed facts (abduction, used for finding hypotheses)
  - Logically arrive at a conclusion from earlier truths (deduction, used for proving)
  - Arrive at a general conclusion even though not all instances have been observed (induction, used for making sense of reality)
- Much more … Next year, in the source Software Testing, you'll see more of this.
- Background reading:
  - introduction and digressions in 'Logic in Action', www.logicinaction.org

Argumentation schemes

- A handy notation was proposed by Frans van Eemeren
- Can be used to understand existing papers
- Can be used to draft and refine your own argumentation
  - *Argumentation*, Frans van Eemeren et al., Routledge, 2001



- Coordinative (conjunctive, &&)
- Subordinative (implication, =>)
- Single argument (fact/assumption)
- Watch out for hidden assumptions
  - Chapter 5 of *Argumentation*



Do I know someone who makes these drawings when writing?

Well…

The power is not so much in this notation itself. It's found in being aware of its existence and the ideas behind it.

Fallacies

- Know how to recognize logical fallacies
  - accidental fallacies are just errors
  - intentional fallacies are rhetorical tricks
- Print …

E. Literature

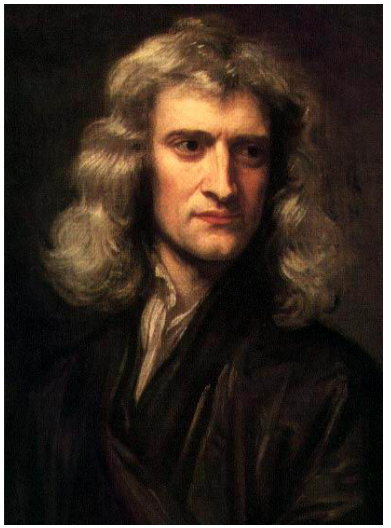
Searching and assessing literature. We discussed this in the first three papers, so I won't go into this…

- hot topics -> conferences, conferences proceedings
- more consolidated material, background reading -> journals, books
- Be aware of social phenomena around publications. Why is Parnas a star and receive his papers prices?
- Be aware of subjective steering of journals …
- Lessons:
  - Take care when judging papers. Peered reviewed paper says something about trustworthyness, but not all.
  - Famous papers? Don't think this makes it 'true'.
  - Ground your statements on more than one view.
  - Important journal? Not necessarily good papers.

Paraphrases, Quotations, References

- Why use paraphrases, quotations, references?
- Hook up 'your piece of writing to rest of the acedemic world' (Van Loon at al., 2001).

- A nice metaphor is what I found on Wikipedia : "Newton himself had been rather more modest of his own achievements, famously writing in a letter to Robert Hook in February 1676: 'If I have seen further it is by standing on the shoulders of giants.'"[1]



Godfrey Kneller's 1689 portrait of Isaac Newton.

- Nice to know who those 'giants' were. Nice to be able to check for yourself what someone else concluded/found/derived.

1. Quote taken from http://en.wikipedia.org/wiki/Isaac_Newton on February 9, 2013. Letter from Isaac Newton to Robert Hooke, 5 February 1676, as transcribed in Jean-Pierre Maury (1992) *Newton: Understanding the Cosmos*, New Horizons.

Examples

- The species was first described here [1].
- The species was first described in the scientific literature by the Czech physician and mycologist Julius Vincenz von Krombholz in 1828 [1].
- [2]presents a concise overview of the inner details of this Unix daemon.
- Lörbach et al. present a concise overview of the inner details of this Unix daemon in their handbook on daemon development [2].

Examples

- The term *software engineering* first appeared in the 1968 NATO [3] Software Engineering Conference [4] and was meant to provoke thought [5] regarding the perceived "software crisis" at the time [6].
- The term *software engineering* first appeared in the 1968 NATO Software Engineering Conference and was meant to provoke thought regarding the perceived "software crisis" at the time [4,5,6].

Examples

- Mike L. Lörbach and J. Wookpecker and Jan Bergstra and Harry G.H. Belafonte-La Peirre present a concise overview of the inner details of this Unix daemon in their handbook on daemon development [2].
- Lörbach et al. present a concise overview of the inner details of this Unix daemon in their handbook on daemon development [2].

Examples (different style)

- Studies by Corbett and Doig indicate that shortterm price savings continues to be a predominant reason for both offshore and domestic outsourcing (Corbett, 2005; Doig et al., 2001).
- Studies indicate that shortterm price savings continues to be a predominant reason for both offshore and domestic outsourcing (Corbett, 2005; Doig et al., 2001).
- Studies by Doig et al. (2001) and more recently by Corbett (2005) indicate that shortterm price savings continues to be a predominant reason for both offshore and domestic outsourcing.

Lessons

- Be consistent in the way you format your text
- Try to find a nice mix between the ways you put the format to work (see previous sheet where there are two solutions under bullet 2 and 3 for the ugly sentence under bullet 1)
- Find a right balance between readability, flow and references

  - Tip: look for examples in other papers (preferably from double checked peer reviewed journal)

(List of) references / Bibliography

- There are many styles to format bibliographies:
  - many journals have one or adhere to one (Harvard style)
- For the assignment it's not important which one you choose
- There are many tools to manage references:
  - the UvA sports Refworks
  - LaTeX + BibTeX
  - Endnote
  - etc.

Example

- Osborn, R., Baughn, C., 1990. Forms of interorganizational governance for multinational alliances. Academy of Management Journal 33 (3)
- Charles Perry, 1998. Processes of a case study methodology for postgraduate research in marketing. **European Journal of Marketing** 32 (9/10), 785–802
- Prahalad, K., Hamel, G., 1990. The core competence of the corporation. Harvard Business Review 79–90.
- Quinn, J., Hilmer, F., 1994. Strategic outsourcing. *Sloan Management Review* 35 (3), 43–55.
- Rosenhead, J., Elton, M., Gupta, S.Robustness and optimality as criteria for strategic decisions.
- Insights into services research: a research agenda. Production and Operations Management 12 (Summer (2)), 145–164.

Example

- Osborn, R., Baughn, C., 1990. Forms of interorganizational governance for multinational alliances. Academy of Management Journal 33 (3), 503–520.
- Perry, C., 1998. Processes of a case study methodology for postgraduate research in marketing. European Journal of Marketing 32 (9/10), 785–802.
- Prahalad, K., Hamel, G., 1990. The core competence of the corporation. Harvard Business Review 68 (3), 79–90.
- Quinn, J., Hilmer, F., 1994. Strategic outsourcing. Sloan Management Review 35 (3), 43–55.
- Rosenhead, J., Elton, M., Gupta, S., 1972. Robustness and optimality as criteria for strategic decisions. Operational Research Quarterly 23 (4), 413–431.
- Roth, A., Menor, L., 2003. Insights into services research: a research agenda. Production and Operations Management 12 (Summer (2)), 145–164. This article discusses how research in services changed over the years. The authors begin with presenting the findings of a grand survey over the past 20 years. Then they reflect upon this form a post-positivistic angle. Finally they formulate a research agenda.

Lessons

- Check references if you copy/paste one
- Be consistent in your formatting
- Give enough information to allow the reader to easily track down a source
- Don’t overdo it

F. Reviewing

- Warning: It looks like a paper, it reads like a paper, it must be a paper…
- Review a colleague's paper: what to look at?
- What elements does a review (often) have?
- Take a minute to draw up to lists:

Structural elements:

- ….
- ….
- ….
- ….

Qualities / aspects to assess:

- ….
- ….
- ….
- ….

Example 1 (taken from SCP)

"Overall work is average but the organization/presentation of the paper is ordinary. It seems that this paper was submitted without careful and thorough review of the work. Writing and references need correction and there are many flaws in write-up and need to be removed. The work still needs to be refined and presentation of the work also needs to be improved. There are several editing and language issues which need to be addressed in terms of technicality and contributions, I don't believe there are many! Proposed methodology need to be elaborated with more detail and how it is efficient must be stated.
Based on the above-mentioned comments, I strongly suggest that this paper be rejected."

My comment: did the reviewer read the paper?

Example 2 (taken from SCP)

--The authors propose a distributed middleware system to add presence awareness functions in mobile applications.

--The paper is clear and well organized, its results are interesting and potentially of practical value.

--I would have liked to see more emphasis on the reusability aspects of the approach. In fact, their middleware uses some patterns: Layers, Acknowlegement, and Context. The authors should think of recognizing more reusable and portable aspects since simplicity is one of their objectives.

--The paper uses several times the term HLMP but it is never defined. I found it to be a protocol proposed by the authors in a previous paper but this should be indicated explicitly.

--The English although clear needs improvement.

--A sequence diagram would have made clearer the dynamic aspects of Section 3.1.

--Figure 7: the yellow lines are hard to see and the text is in Spanish.

My comment: too concise, analysis missing.

Example 3 (taken from SCP)

- This paper presents the Verifast program verifier with relevant/industrial case studies showing its capabilities on realistic programs in both C and Java, which include both concurrency and OO features. Moreover, the article also describe the nitty-gritty (yet QUITE relevant) ``engineering'' sides of modelling in the large (i.e. handling of external libraries). The article is well-written, yet clarity is hampered at places by author's assumption the reader is thoroughly familiar with Verifast parlance (see specific details below).

- The paper provides a good contribution in showing how to use Verifast on larger examples. On the other hand, authors ought to be more candid about its limitations. For instance, what about total/partial correctness? The paper focus is mostly on absence of run-time errors, which tend to be easier (if not easy) to model/verify. Claims are in this sense a bit confusing: the paper talks about absence of run-time errors or data races but not functional correctness; yet Verifast also has pre/post annotations being checked.

- Moreover, object invariants are not even mentioned, despite the fact some (i.e. Spec#, SpaceInvader communities) argue they are essential for sound OO-specification. I couldn't find in the paper discussion about class invariants, which might pass as not so relevant to a general reader. I suggest authors add discussion in this respect on related work perhaps.

- Overall the authors fail to give a rationale of choice of when Verifast would be suitable for the user's problem. Instead, only Verifast's positive aspects are mentioned. I would be skeptical/careful to chose it... I think the paper would be improved if this rationale of choice, as well as design decisions behind the way supporting libraries are to be annotated effectively are given. Also, numbers used for evaluation are difficult to assess and more details on that is given below.

- ========== detailed comments ============

- p.1 l.28-31: maybe say explicitly these are checks over run-time errors rather than functional correctness, or have I misunderstood something?

- ----
- p.2 l16-19: this wholly inaccurate. Model checking has been used to verify a variety of large-scale industrial models, in particular in hardware. Software verification has also had significant examples: just look for any of the Grand Challenge / VSTEE … [etc..]

- ----
- [+ 20 more of these detailed comments and issues..]

My comment: this reviewer really read the paper, addresses the relevant aspects, gave many usefull comments to improve the paper.

Elements:

- Summary
- Analysis
- Comments
  - big issues
  - smaller issues
- Advice
- References used
- Scores:
  - relevance
  - readabilty
  - self assessment by reviewer (expert <-> novice)
- hidden part to editor

Aspects to assess:

- claim and argumentation
  - logic
  - proofs
  - underpinning
  - relevant references
- Contribution
  - what's the contribution
  - do we learn something
  - does it meet its promises
  - well motivated
- structure
- style, conciseness, typo's

References some with annotations

- Inspiring example papers:
  - *Making Software*, Andy Oram and Greg Wilson eds., O'reilly, 2011.
- Style:
  - *Stijlboek* (in Dutch), de Volkskrant ed., SDU Uitgevers, Den Haag, 4e editie, 2007.
  - *Schrijfwijzer* (in Dutch), Jan Renkema, SDU Uitgevers, Den Haag, 2002.
  - *Style - The Basics of Clarity and Grace*, Joseph M. Williams, revised by Gregory C. Colomb, Pearson Education, 2006.
- Language:
  - *The BBI combinatory dictionary of English – A guide to word combinations*, Morton Benson, Evelyn Benson and Robert Ilson, John Benjamins Publishing Company, 1993.
- Writing from A to Z:
  - *Overtuigend schrijven* (in Dutch), Frans van Eemeren et al., ThiemeMeulenhof, 2011
  - *Academic writing in English – A process-based approach*, Jeanne van Loon et al., Uitgeverij Coutinho, 2011.
  - *Spring into technical writing for Engineers and Scientists*, Barry J. Rosenberg, Addison-Wesley, 2005.
  - Academic writing *– A handbook for international students*, Stephen Bailey, Routledge, 2011.
- Argumentation:
  - *Argumentation*, Frans van Eemeren , Rob Grootendorst and Francisca Snoeck Henkemans, Routledge, 2001.
  - *Argumentatie* (in Dutch), Frans van Eemeren and Francisca Snoeck Henkemans, Noordhoff Uitgevers, 2011.
- Background reading:
  - 'Logic in Action', see: www.logicinaction.org.
- Historical context:
  - *The Adages of Erasmus*, William Barker ed., University of Toronto Press, 2001.
  - *Les Essays*, Michel de Montaigne, has many easily obtainable translations.
  - *De ontwikkeling van wetenschap – Een inleiding in de wetenschapsfilosofie* (in Dutch), Gerard de Vries, Wolters-Noordhoff, 1995.
  - *A New History of the Humanities*, Rens Bod, Oxford University press, in press.

Okay, what did we address today?

1. Present the goal
2. Positioning this session
3. Assignment for Software Construction
4. Topics in this session:
   A. What is a paper? Why write one?
   B. How to go about when writing a paper
   C. Writing, style
   D. Argumentation
   E. Literature
   F. Reviewing

Questions?

4. Forms please!