Metaphors in Software Engineering

Klaas van den Berg
Software Engineering Group
Farewell Lecture - 01 July 2010
How it Started in 1964
A Challenging Journey

1964  Campus THT – Electrical Engineering *BSc/MSc Material Science*
  - Prof. Jonker, Wim Hulscher, Cock Lodder, Jan Fluitman, …

1975  University of Zambia / Tanzania *Lecturer EE - NUFFIC*

1981  HTS - Technical College Enschede *Lecturer - from Hardware to Software*

1986 University of Twente – Informatica/CAP/SETI *Functional Programming*
  - Prof. Duijvestijn, Hans van Berne, Gerrit van der Hoeven, …

1995 University of Twente – *PhD Software Measurement & Functional Programming*
  - Prof. Duijvestijn, Norman Fenton, Pim van den Broek, …

1996 University of Twente – EWI/SE *Software Engineering Group*
  - Projects: AOSD-NoE, Darwin, QuadREAD
  - *TRESE: Mehmet Aksit … and many colleagues and students*
My Themes

Measurement
  - Quantification of Properties

Models
  - Abstractions of Reality

Why
  - Better systems, better solutions
  - Better understanding of the problems
A method for accurately determining lattice parameters using electron diffraction in a commercial electron microscope

by J. C. Lodder and K. G. v. d. Berg, Department of Electrical Engineering, Twente University of Technology, Enschede, The Netherlands

SUMMARY
In this paper an electron diffraction method is discussed by which the lattice constants of polycrystalline thin films can be accurately determined (0.1%).

The method involves the sequential examination of a standard material and the unknown material mounted on separate grids. The error which can arise through the possible difference in height of the two grids can be corrected by means of height and tilting adjustments. The analytical approach of the dependence of the appearance of the diffraction pattern on the dependence is given in terms of the lattice constants of the unknown specimen.

1. START
One can start by examining the standard material with a Philips CM10 electron microscope. The unknown material is mounted on a standard grid, and an electron diffraction pattern is taken. The same pattern is then taken for the unknown material.

Fig. 1. Electron micrographs of an evaporated manganese ferrite film, Mn₃Fe₂₋₅O₄₋₅, about 500 Å thick. (a) the as-evaporated film, (b) the same film after annealing for 3.5 h at 750 °C in a partial oxygen-pressure of 10⁻¹⁰ atm, (c) electron diffraction pattern of the as-evaporated film, (d) electron-diffraction pattern of the annealed film.

Thin Solid Films, 9 (1972) 363-375

Here \( \lambda \) is the wave-length of the electrons for the particular accelerating voltage, \( L_0 \) is the effective camera length, i.e. the distance from the specimen to the photographic plate. With the measured ring diameter \( D \) of the unknown specimen and obtained camera constant \( 2 \lambda L_0 \), formula (1) can be used to calculate the \( d \)-spacings of the unknown specimen.

A correction is introduced with respect to the simple formula (1) as the effective camera length \( L_0 \) depends on the ring diameter \( D \) (Andrews et al., 1967). The corrected camera-constant \( 2 \lambda L \) can be calculated from the following relation:

\[
2 \lambda L = 2 \lambda L_0 \left( 1 + \frac{3D^2}{L_0} \right)
\]
Software Engineering

Software Engineering
- How to design, construct and maintain software systems

Quality
- Productivity, Usability, Performance, Maintainability ..
- Quality Models & Measurement

Software Development
- Processes / Products: to design / the design

A Picture is Worth a thousand Words
Software Engineering

Some Problems

Change
  - Technology
  - New Requirements

Crosscutting
  - Modularization based on Concerns
  - Other Concern is present in all Modules

Trade-off
  - Conflicting Quality Properties
Change: Software Evolution

Darwin Project: MRI Scanners - Philips Medical Systems

- Modeling Software of MRI Scanner
- Measure and Improve Evolvability of Software
Change: Software Evolution

Darwin Project: MRI Scanners - Philips Medical Systems

- Modeling Software of MRI Scanner
- Measure and Improve Evolvability of Software
Models & Measurement

Framework (Hughes)
- Ivan Kurtev PhD Thesis

- My PhD Thesis

No Measurements without a Goal
No Models without a Goal

*Everything is a Model* (Bezivin)
Topics in Software Engineering Group

Empirical Software Engineering
- Validation of Measurements
- Controlled Experiments

Model-Driven Engineering
- Models / Metamodels / Meta-metamodels (PhD Ivan Kurtev)
- Modeling Requirements / Traceability (PhD Arda Goknil)

Aspect-Oriented Software Development
- Problem: Crosscutting (PhD Chema Conejero)
- Solution: Composition Filters
Composition Filters

Mehmet Aksit: Camera Filters
Composition Filters

Lodewijk Bergmans: Shape Sorter

(arriving messages)

(filter pattern)

(filter type)

(rejected messages)

(message does not match)

(message continues to next filter)

(message matches)

(message is modified, continues to next filter)

(message matches)

(message is dispatched)
Metaphors & Analogies

- Composition Filters – Cameras/Toys
- Software Architecture – Buildings/Bridges
  - Water Pump – Heart
  - Electron Orbits – Planetary Orbits
  - Life/Career – Journey
  - Computer – Brains

- Desktop, Folders, Recycle Bin, Threads, Patterns, Objects, Messages, Evolution, Life Cycle, Retirement, Garbage Collection, Code Smell, Bugs, ...
  (Metaphors in Computer Science - Colburn & Shute)

- Metaphors are fundamental to human thought

  *Metaphors are Everywhere*  (Richardson)
Metaphors Can Be Considered Harmful

Edsger W. Dijkstra, EWD1036-25

And this concludes my technical excursion into the reason why operational reasoning about programming is "a tremendous waste of mental effort" and why, therefore, in computing science the anthropomorphic metaphor should be banned.

Not everybody understands this sufficiently well. I was recently exposed to a demonstration of what was pretended to be educational software for an introductory programming course.
Metaphors

Metaphors are fundamental to human thought

A Metaphor is Worth a thousand Pictures

Linguistics, Cognitive Psychology, Philosophy of Science
Many competing theories: War on Metaphor’s (a meta-metaphor)

Metaphors are Everywhere
Everything is a Model

Metaphors are Models
Extend Ivan’s Framework
Metaphors & Models

- Source (familiar domain) - Target (unfamiliar domain)
- Semantic Mappings / Commuting Diagram / Pushout

Open Issue

- Metaphors in Model-Driven Engineering
- *Models-Measurements-Metaphors*
Metaphors for Software & Software Engineering

Dinosaur
Mehmet Aksit in Study Guide for Master Students in CS
Software & Software Engineering as Mountain Hiking
Software & Software Engineering as Mountain Hiking

Cockburn

- Training & Tools
- Individual – Team
- Planned – Improvised
- Challenging – Fun
- Dangerous ?? !!
Software Engineering can be Dangerous
Software Engineering is Team Work
My Team
Trade-Off Analysis

Quality Factors in Software Engineering

- Adaptability
- Maintainability
- Functionality
- Usability
- Performance
- Security
- Portability
- Cost
- .....

Trade-Off Analysis
Trade-Off Analysis

Quality Factors in Mountain Hiking

- Weight
- Volume
- Comfort
- Nutrition
- Navigation
- Safety
- Cost
- ...

...
Measurement in Mountain Hiking
Tradeoff-Analysis in Mountain Hiking
Hartkeks
Reflection
Software Engineering: A Challenging Journey

Only the top counts

The journey is the destination

It is not the mountain we conquer but ourselves
Conclusion

- Metaphors are omnipresent in Computer Science
- Metaphors are fundamental for human understanding
- Metaphors can be considered harmful
- Metaphors theories could be beneficial for Model-Driven Engineering
Future Work

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

After climbing a great hill, one only finds that there are many more hills to climb

Nelson Mandela
Acknowledgements

Many, many people ....

- Research / Teaching
- Colleagues / Students
- Courses / Supervision
- Study Advisor / Educational Committees
- Workshops / Conferences
- Projects / Companies
- Secretaries / Services / BOZ / Helpdesk / HD / Library
- ....

THANK YOU ALL
References

References