

---

# Reality Distortion in Communication

Matthias Rauterberg  
Department of Industrial Design  
Eindhoven University of Technology  
2007

1

Ten years after and still based on the presentation of Mark Weiser!

---

# Software Engineering for People

Mark Weiser  
Chief Technologist  
Xerox PARC

1997

[I replaced 'engineer' with 'scientist', 'marketeer' with 'designer', 'engineering' with 'technology', and 'software engineering' with 'design science'; the rest remains mainly the same from Mark's original presentation!]

2

## Most important thing Mark Weiser learned as an entrepreneur

---

- ...
- Problems with reaching agreement
- ...

3

## Problem with Agreement: Reality Distortion

---

- Why is it that designers and scientists rarely understand each other?
- Answer: they have different cultures for communicating importance.
  - Scientists and Designers have anathematic distortion fields.

4

## Reality Distortion varies

---

- Let 1.0 stand for a perfect match of reality to words
  - “The house appears white on this side.”

5

## Reality Distortion varies

---

- Scientists/engineers speak at 0.5-0.8
  - generally understated
  - more understated if more important
  - “The house was probably white.”
  - “We wish to suggest a software architecture for the living room in the context of ambient intelligence.”

6

## Reality Distortion varies

---

- Designers speak at 1.5-2.0
  - more if more important
  - “The house is dazzlingly white inside and out.”
  - “Outstanding product and service quality”

7

## Reality Distortion creates a listening

---

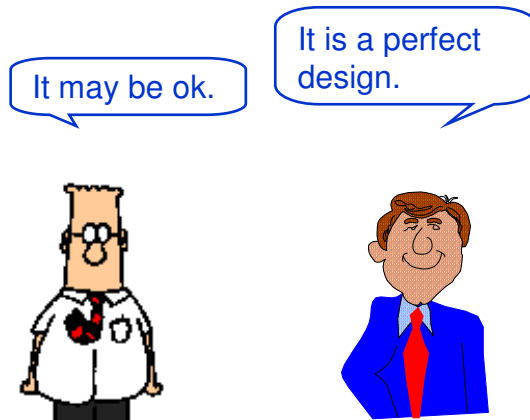
- Designer speaks at 1.2-2, and discounts everything heard by 0.8-0.5
- Scientist speaks at 0.5-0.8, and boosts everything heard by 1.2-2.0
- Can you see the problem?

8

## Reality Distortion

what are these two people thinking?

---



9

## Reality Distortion

---



*Scientist says:*  
This is a pretty good  
piece of technology.



*Designer hears:*  
???

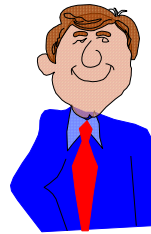
10

## Reality Distortion

---



*Scientist says:*  
This is a pretty good  
piece of technology.



*Designer hears:*  
This guy has no  
confidence, and his  
technology is pretty bad.

11

## Reality Distortion

---



*Scientist hears:*  
???



*Designer says:*  
This is really really  
important!

12

# Reality Distortion

---



*Scientist hears:*  
Either he's discovered cold fusion, or he's lying.

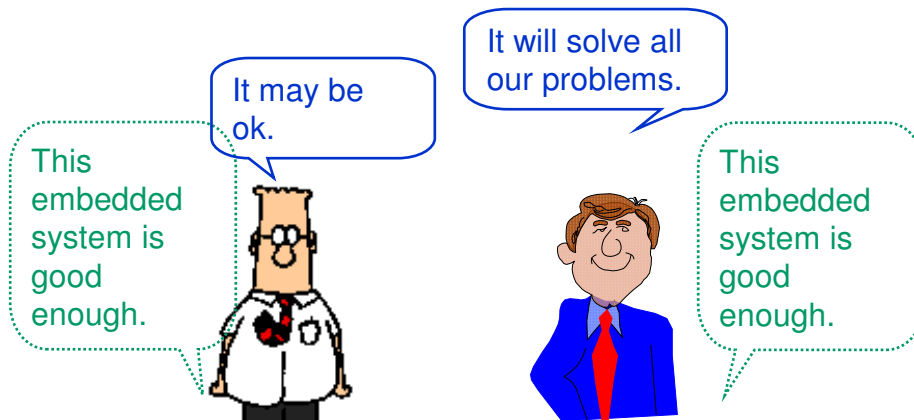


*Designer says:*  
This is really really important!

13

## Reality Distortion: the same thought can lead to different words

---



14

## Reality Distortion

---

Let  $S_S$  = speaking of scientist, about 0.7

Let  $L_S$  = listening of scientist, about 1.4

Let  $S_D$  = speaking of designer, about 2.0

Let  $L_D$  = listening of designer, about 0.5

Then  $L_S(S_S(x)) = x$ , and  $L_D(S_D(x)) = x$

**But  $L_D(S_S(x)) = 0.3x$ , and**

**$L_S(S_D(x)) = 2.8x$**

15

## What to do about Reality Distortion?

---

- Do you want to be understood?
  - speak into the culture of your listener
  - notice how you are heard
- Do you want to be understood?
  - apply the appropriate distortion field
  - ask questions to tune your distortion ratio
- Don't ever expect 1.0

16



## Problems of agreement in Design Science (DS)

---

- There are many kinds of reality distortion
  - seeming to agree but not really understanding
- What do we agree about in DS?
  - What are the foundations of our work?
    - Questioning their truth is almost unheard of.
    - We almost all are familiar with *and use* them.
  - What should all practitioners know?
    - Beyond technical skill in particular languages and systems

17

## On What Do We Agree: Some of Mark's Answers

---

- High level design languages, if appropriate, improve all aspects of a project except possibly running time.
- Dividing systems into modules improves implementation time and maintainability
- Documentation helps
- Adding people to a late project makes it later

18

## Forces against agreement

---

- Benefits to disagreeing
- Absence of pressure for agreement
- Benefits, when present, come slowly

19

## How can we understand DS better? (in industry and academia)

---

- More reflective practitioners
- More social science attention to technology and engineering processes

20

## Skill is not enough

---

- Knowledge of technology are crucial enablers ...
- But not always necessary nor sufficient
  - not necessary for many utilitarian and policy understandings
  - not sufficient for productive engineering
- Other community skills are required
  - students implicitly learn values, language, and culture whether we like it or not

21

## Learning is always joining a community

---

- Learning is not poured from faculty to student brains
  - especially not without a strong community in place
- All learning is within and about a community
  - *Lave and Wenger, 1991*

22

# Humanities

---

- “The goal is of computer science to make [computers] more useful and useable.”  
- Rick Weingarten, testifying to Congress, May 1996
- For people
- What are people like? This is the subject of humanities and social sciences
- Example: “The Tacit Dimension”

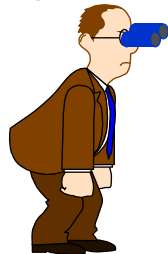
23

## The Tacit Dimension

from philosopher Polanyi's book by same name

---

- Imagine strapping toilet tubes to your eyes and walking around for two hours  
– lots of surprises



24

## The periphery informs and creates the center

---

- The periphery is the “tacit” dimension to thinking and understanding
- Technology tends to deliver information to the center, and strips off the periphery
  - causing flame-wars in email...
- The tacit is easy to forget...but without it nothing gets done

25

## The Flow State and the Tacit Dimension

---

- If you work uninterruptedly on a sufficiently interesting problem:
  - you forget where you are
  - time passes quickly
  - your unconscious (tacit) mind rallies to make you smart

26

## Academics are members of a foreign community

---

- They have not coded much, nor as a team
- They don't deeply know aesthetics or design
- Humanities (or customers) are not their life
- They hang out a lot with computer scientists and grad students

27

## Curriculum

---

- History of Technological Change  
*Electrifying America; Natural Monopoly and Universal Service; History of Technology Failures*
- History of the Modern Age  
*Cosmopolis; A Social History of Truth; The Pencil*
- Common Sense  
*The Tacit Dimension; The Psychology of Everyday Things; Elements of Style; Computer Projects That Failed; Mythical Man-Month*

28

## Agree to Agree

---

- Teach your graduate students, and your employees, to measure, and reflect, on themselves and others.
- Come to agreement on ethics, people, and tools
  - Tentative agreement is more important than being right!
  - When you know more, *then* disagree.

29