



like Jonathan Ive said in 'Objectified'

SATURN16
MAY
2-6
2016

SATURN 2016
12th ANNUAL SEI
Software Engineering
Institute
ARCHITECTURE TECHNOLOGY
USER NETWORK CONFERENCE
San Diego, CA USA

SKETCHNOTES @mjbradbert

Nitpicky little DETAILS are at the ♥ of great software design

SAID WITH A WINK ↘

"Design thinking people tend to like PEOPLE more than most software developers"

[DT] is suitable for domains where there is a lot of UNCERTAINTY

Keynote

DANIEL JACKSON

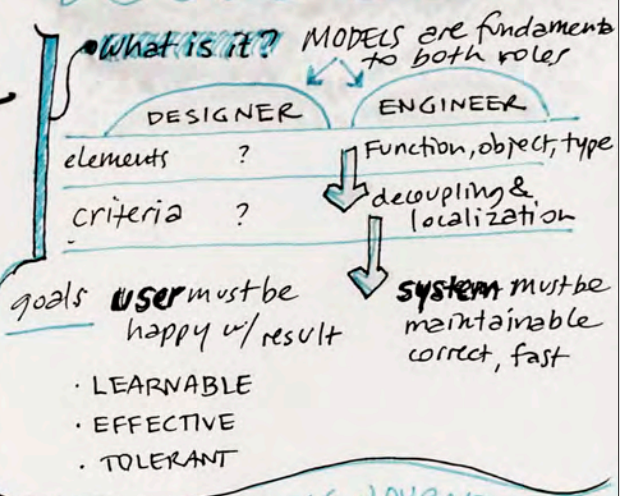
M. I. T. COMPUTER SCIENCE and ARTIFICIAL INTELLIGENCE LAB

Rethinking Software Design

EXAMPLES

- ▶ Google email categories
- ▶ Digital camera settings (CORRELATION of RESOLUTION w/ ASPECT RATIO)
- ▶ Fonts and styles: IN DESIGN (FONT GEEKS MUST KNOW EXACT NAME)

MINOR NIT PICKS?
CODING BUGS?



RICH CONCEPTS have LONG JOURNEYS often FROM THE PAST!

What characterizes an app?

CONCEPTS!

which define CLASSES

TEXT EDITOR → WORD PROCESSOR → DESKTOP PUBLISHING APP

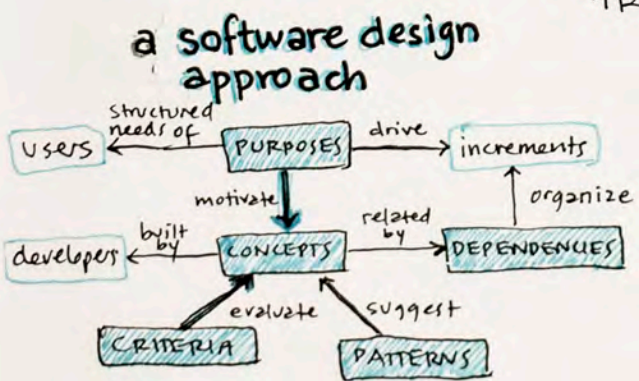
the OPERATIONAL PRINCIPLE a way to explain concept
Michael Polanyi "IF... THEN..."

- invented → GIVEN CONCEPTS (calendar event)
- analogical or new → INSTRUMENTAL CONCEPTS (hashtag, label)
- invented → ENABLING CONCEPTS (relative reference)

1 SKETCH first

2 MODEL

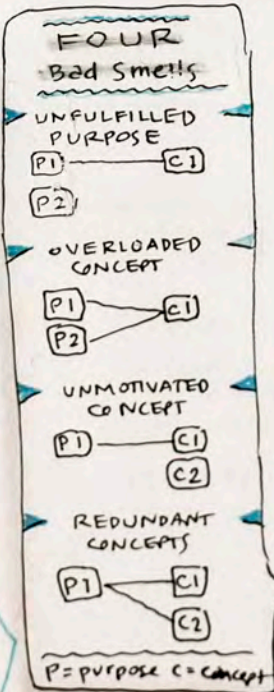
Sometimes there are PURPOSES, PRINCIPLES & MISFITS



LIKE HOW TRASH BEHAVES

FALSE CONVERGENCE two purposes looked the same

EMERGENT PURPOSE Users find second purpose for concept



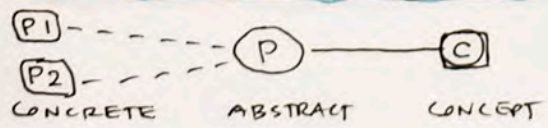
CONCEPT DEPENDENCIES
 $\langle c, c' \rangle \in \text{depends} \Leftrightarrow \forall a: \text{apps}$
 $c \in \text{concepts}(a) \Rightarrow c' \in \text{concepts}(a)$

- GENERIC CONCEPT PARTS
- name
 - purpose
 - sample uses
 - model
 - operations
 - related to
 - issues

3 EVALUATE COMPATIBILITY

FUNDAMENTAL PRINCIPLE
In a well-designed system: EACH CONCEPT IS MOTIVATED BY One Purpose

When OVERLOADING is acceptable



STORY: GITLESS a reworking of GIT
Santiago Perez De Ressa