



How To Make (Almost) Anything Usable

Prof. Dr. Jan Borchers

hci.rwth-aachen.de/fabusability

Fab Academy Recitation • May 3, 2021

Usability

The art & science of designing stuff that works great for people

- More useful & convincing final project
- For your future startup
- Fab Fame™

Today

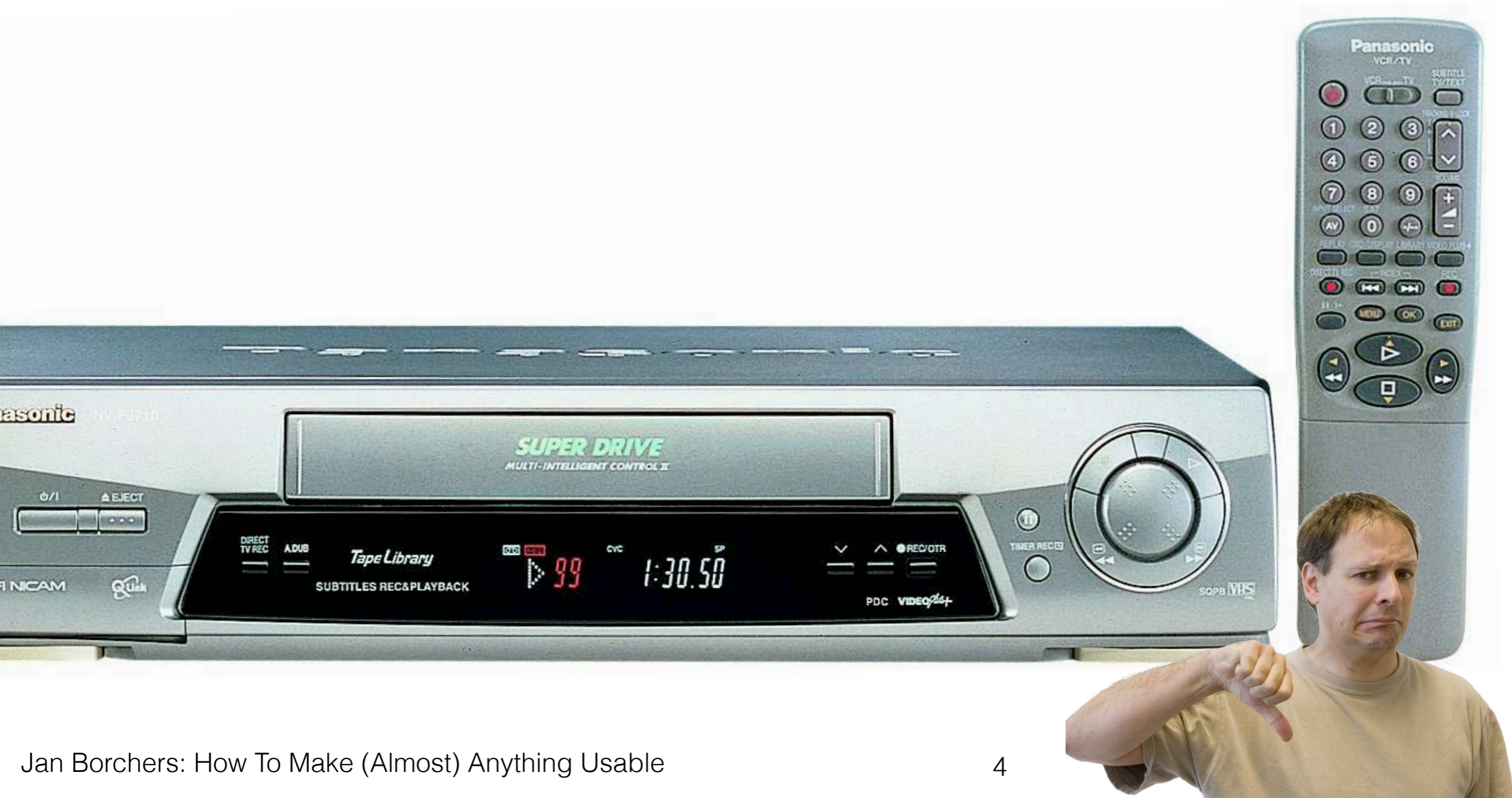
1. Golden rules



2. Process tips

I. Simplicity

User • Task • Context



I. Simplicity

User • Task • Context



2. Visibility and Feedback



Jan Borchers

2. Visibility and Feedback



Jan Borchers

2. Visibility and Feedback

1. Current state?
2. Available features?
3. How to access?



Jan Borchers

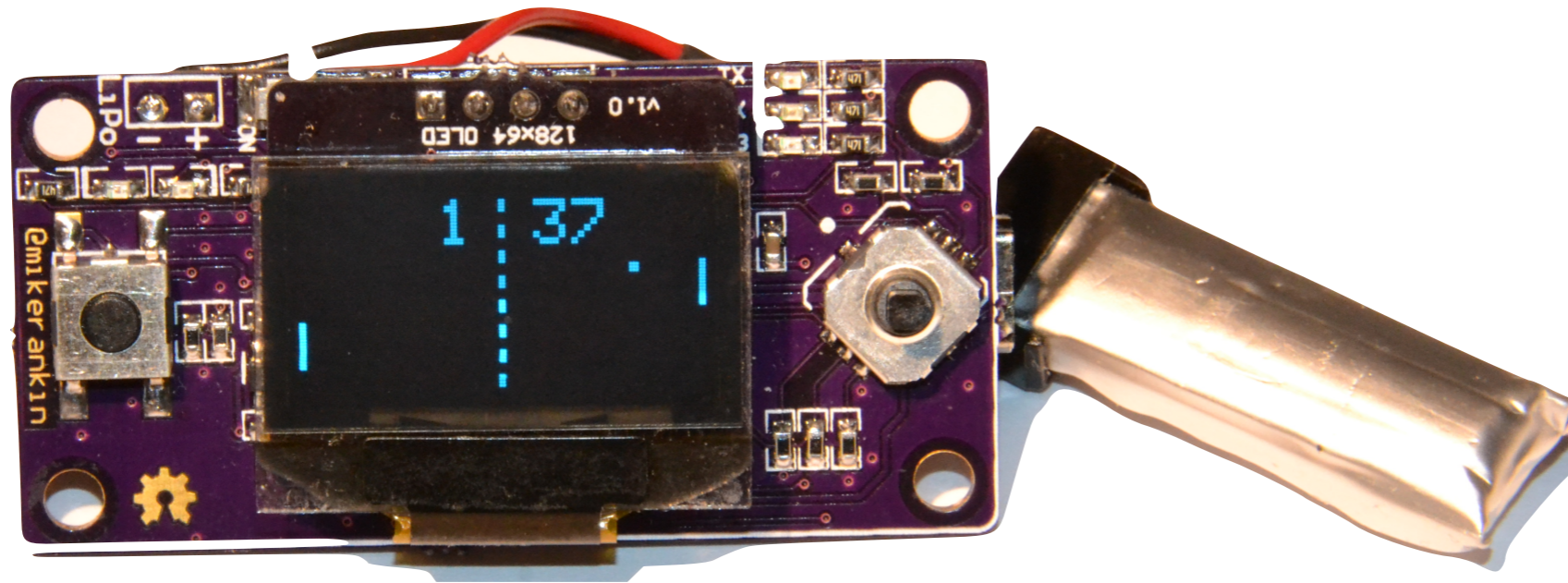
2. Visibility and Feedback

1. Current state ✓
2. Available features ✓
3. How to access ✓



Panasonic

2. Visibility and Feedback



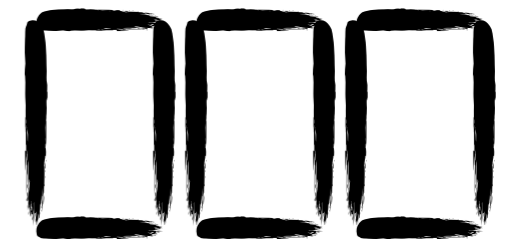
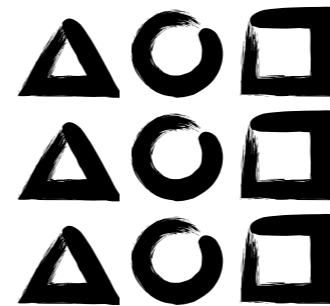
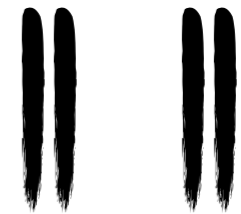
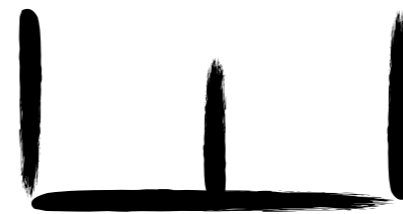
oomlout / C Sancho / Boringuser

2. Visibility and Feedback



Read:
Bret Victor's rant
on future of
interaction design

3. Gestalt Laws



Beyond

4. Natural Mappings



4. Natural Mappings



4. Natural Mappings



Panasonic

4. Natural Mappings



4. Natural Mappings



4. Natural Mappings



5. User's Language



6. Avoid Modes



(Almost) Anything Usable



6. Avoid Modes



7. Principle of Least Surprise



Your battery is now fully charged!

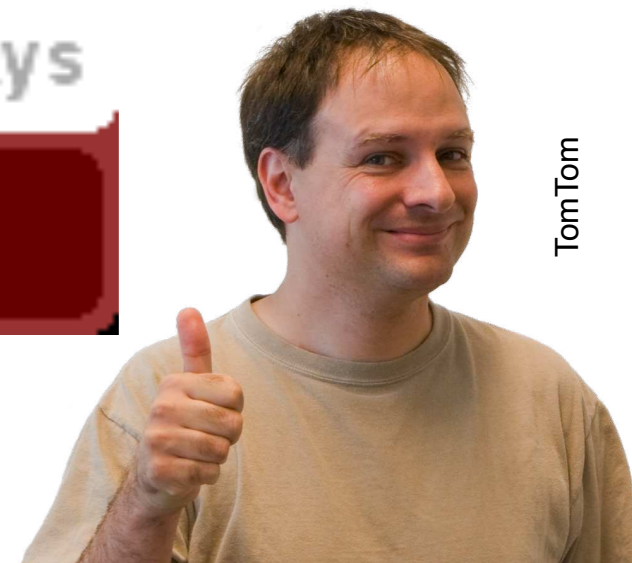
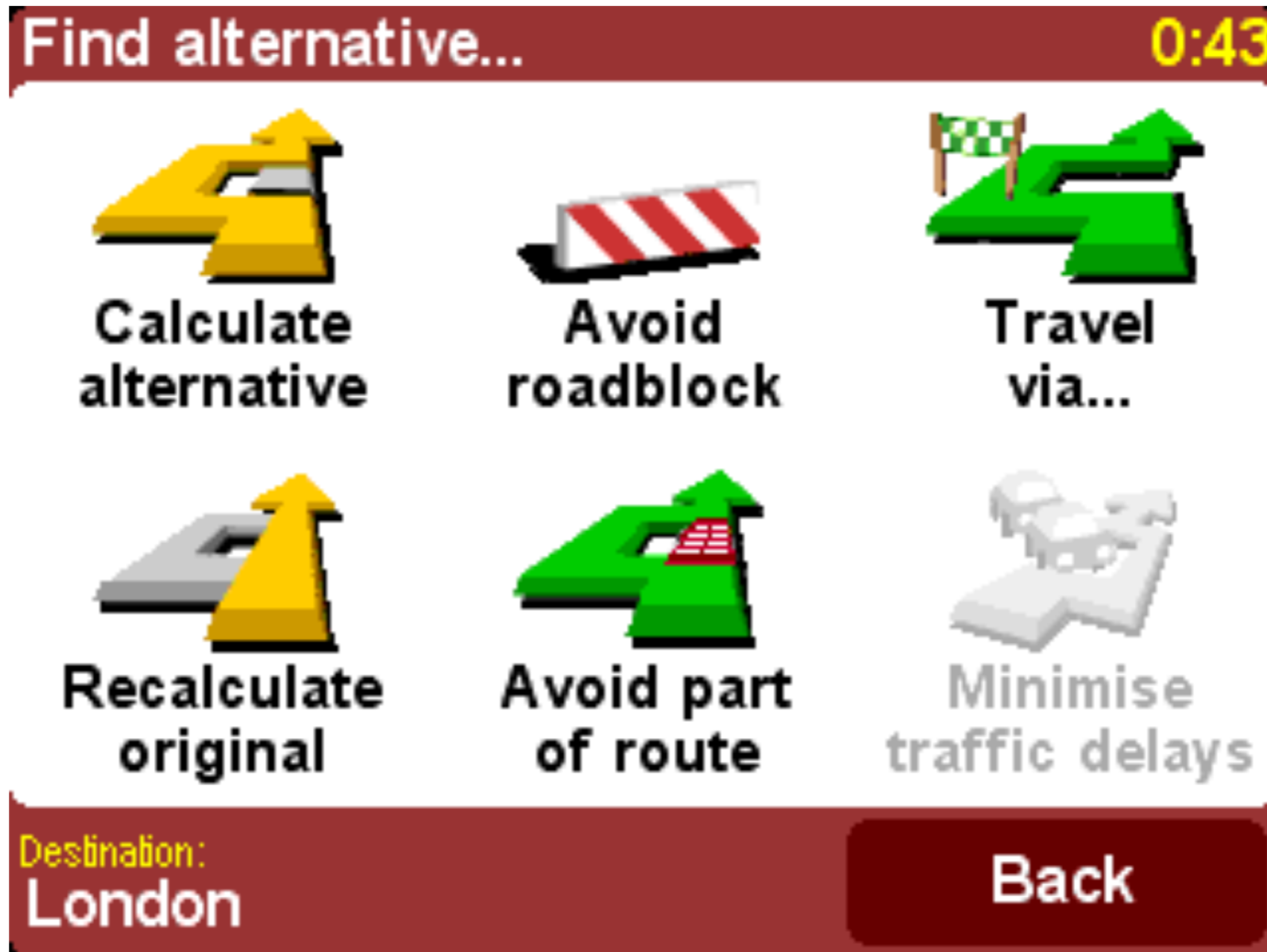


7. Principle of Least Surprise



Timeouts are evil!

8. Dialog, not Monolog



9. Tolerate Errors



10. Visual Design



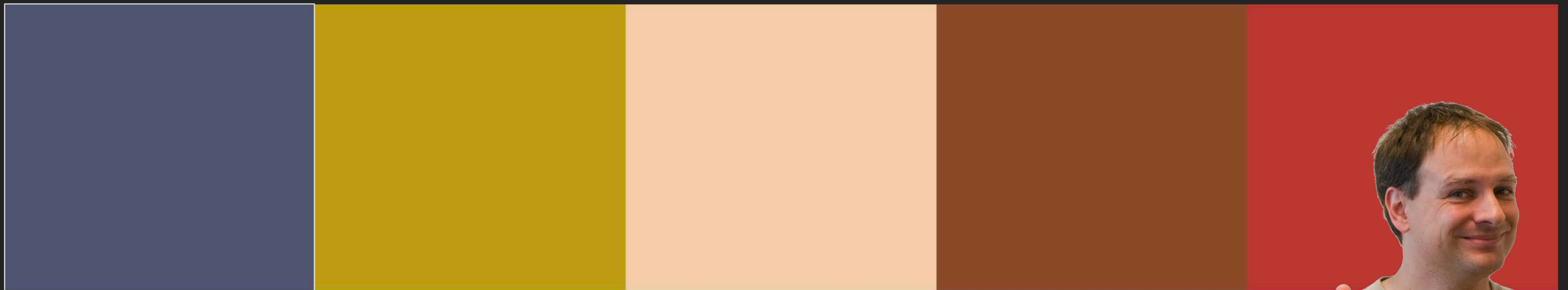
Omni



Save

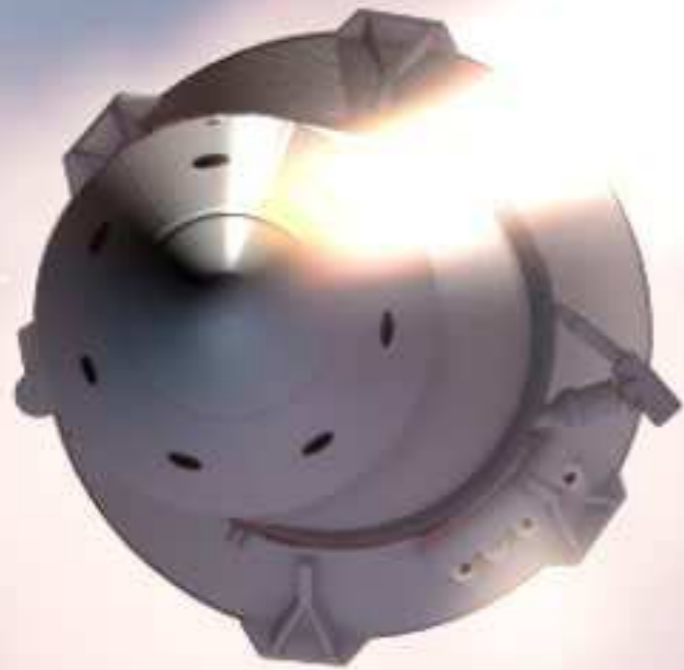
Color Mood

- Colorful
- Bright
- Muted
- Deep
- Dark
- Custom



Today

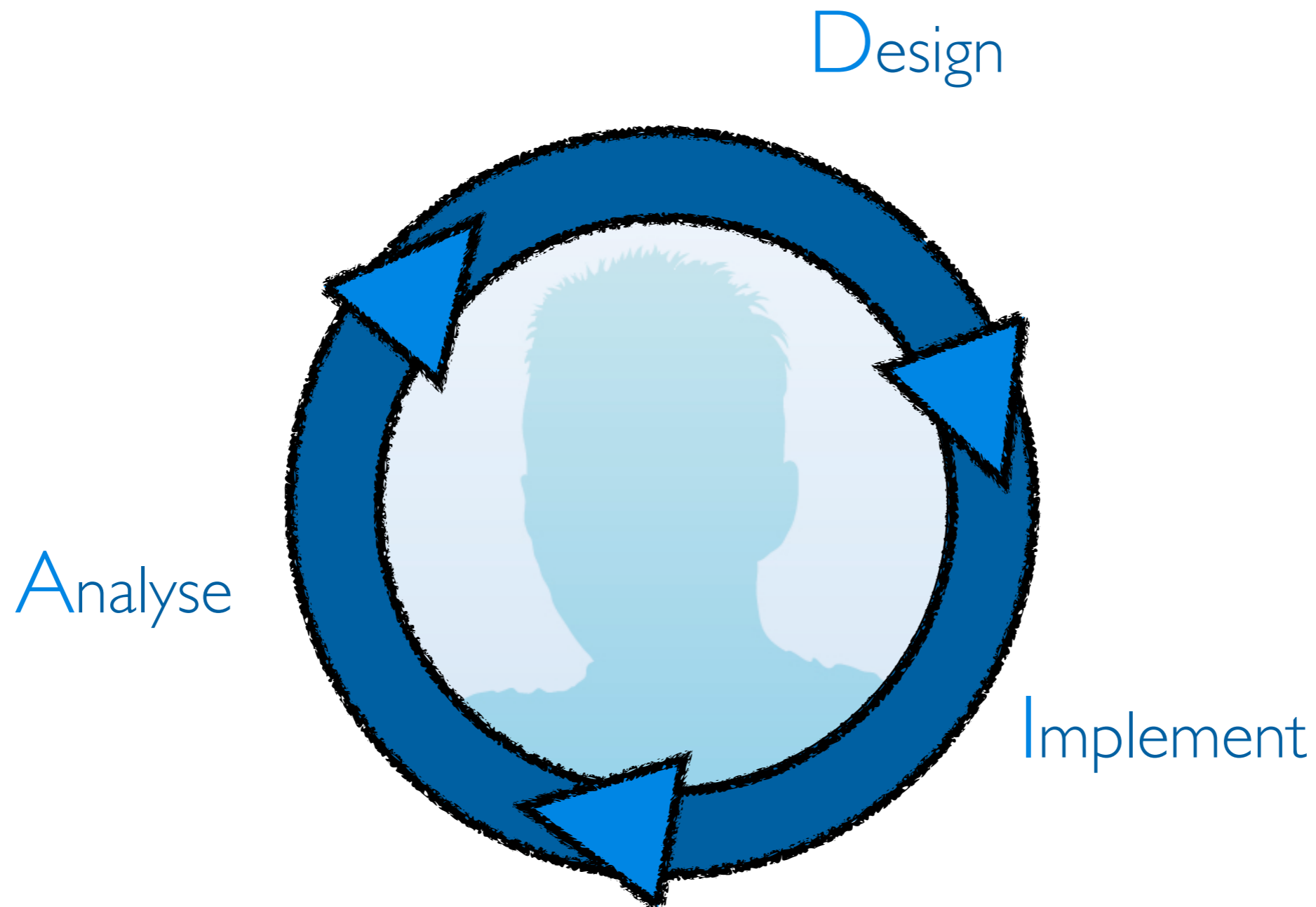
1. Golden rules



2. Process tips

The DIA Cycle

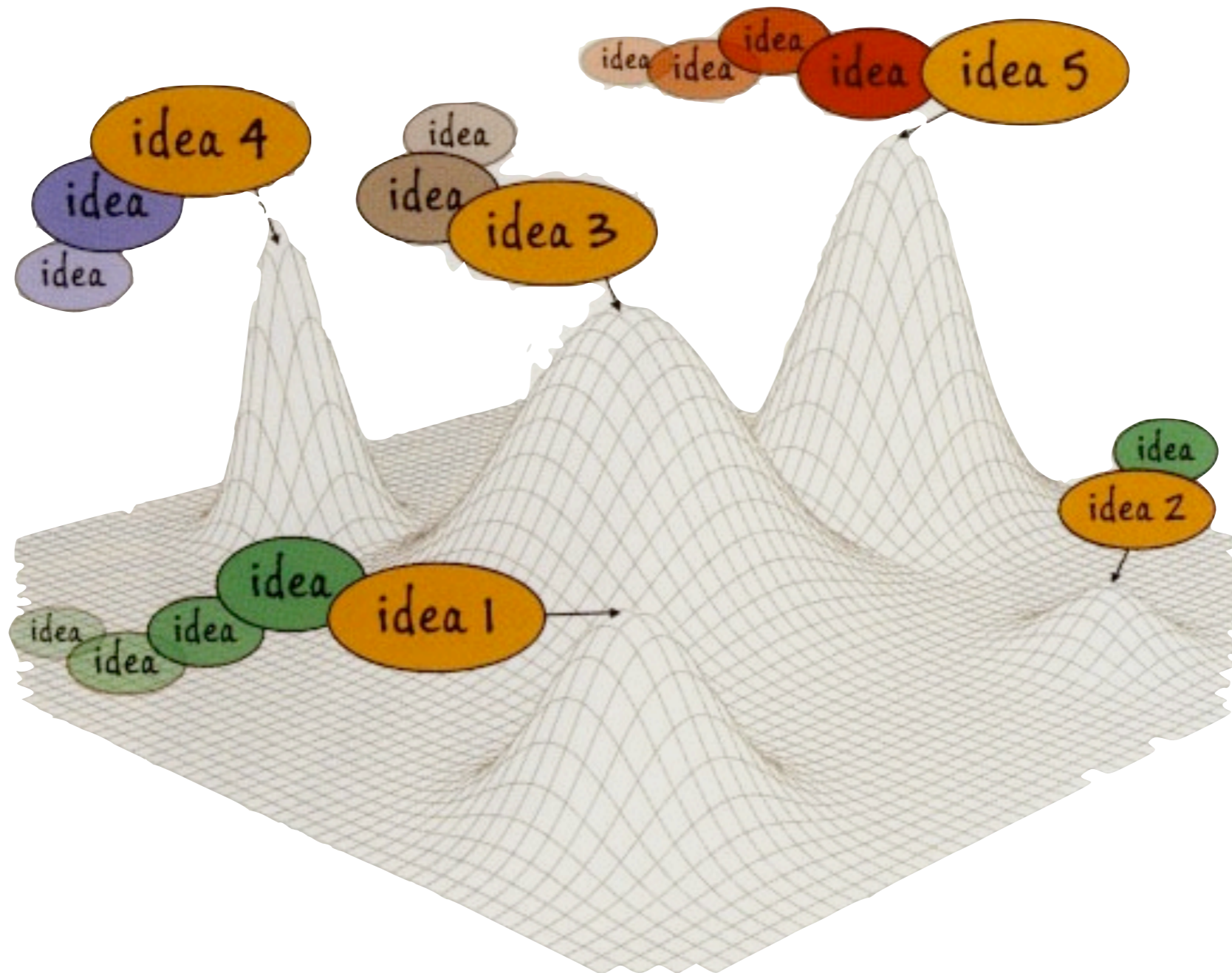
Design is Iterative & Agile



Observe and Ask First, Then Start Solving

- Are you looking for a problem for your solution?
- **What** problem to solve, not **how** to solve it
- **Users:** Who is it for? Other stakeholders?
- **Task:** What is their *actual* problem?
- **Context:** When & where does it occur?

Design: Search the Solution Space

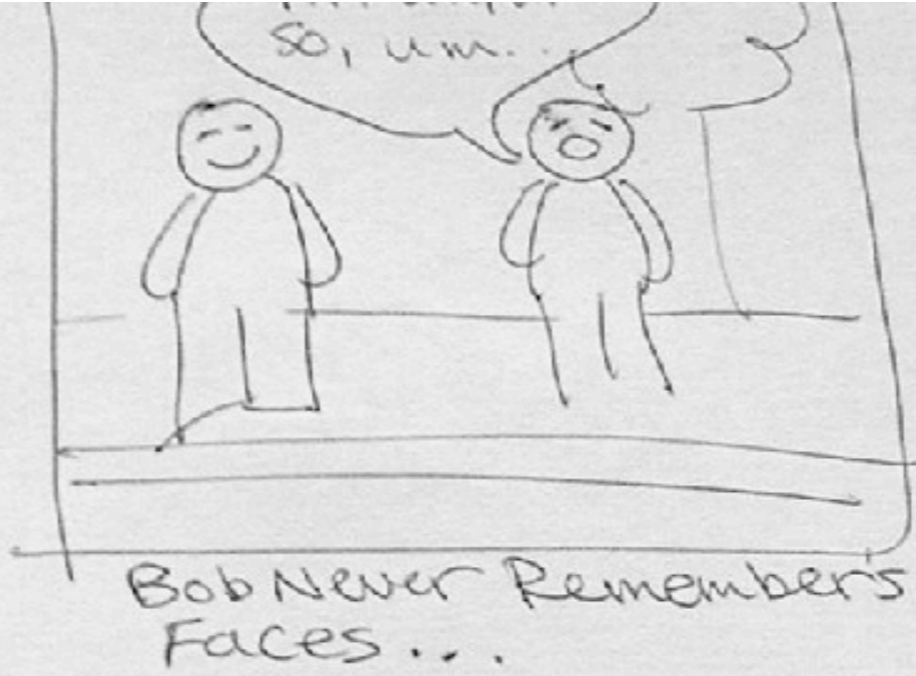


Implement: Prototype

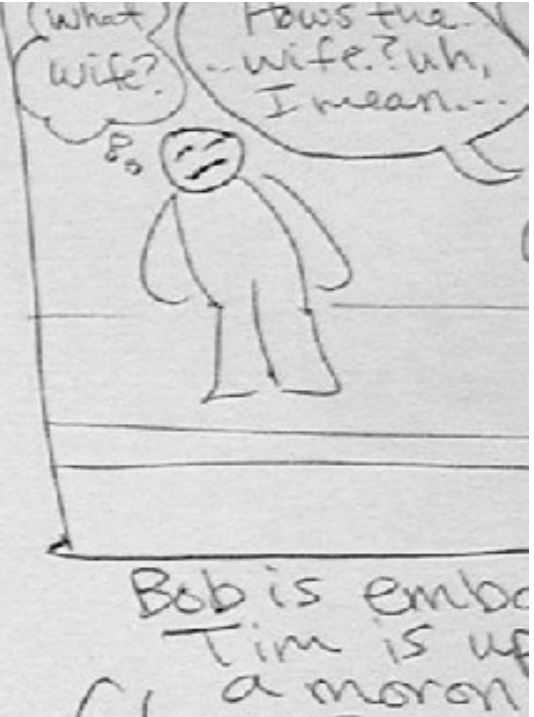
- Create quick prototypes to get feedback on from others, to improve and fill in your idea(s!). For each idea:
- Write one-paragraph success story, test it
- Then draw 3-panel success comic, test it
- Fake features with Wizard of Oz



Bob Walking somewhere



Bob Never Remember's Faces...

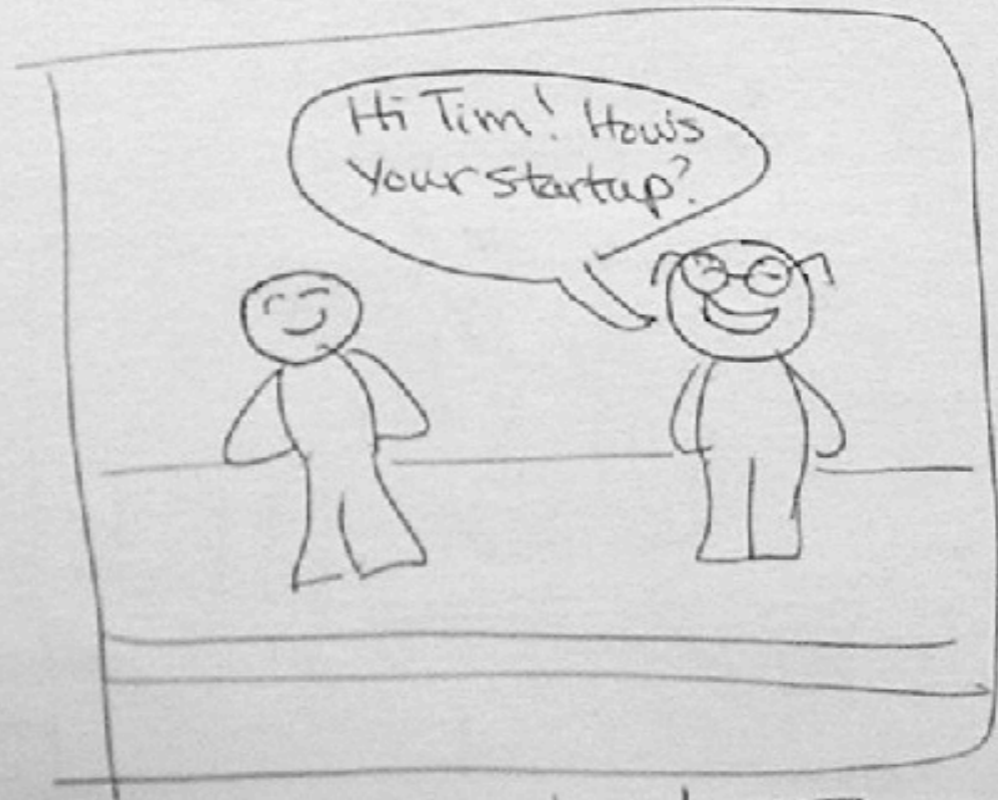


Bob is embar
Tim is up
a moron

If he had Recognition Glasses, ...



glasses register Tim's Name



Bob Remember's Tim Brown's Start-up, "Brown.com"



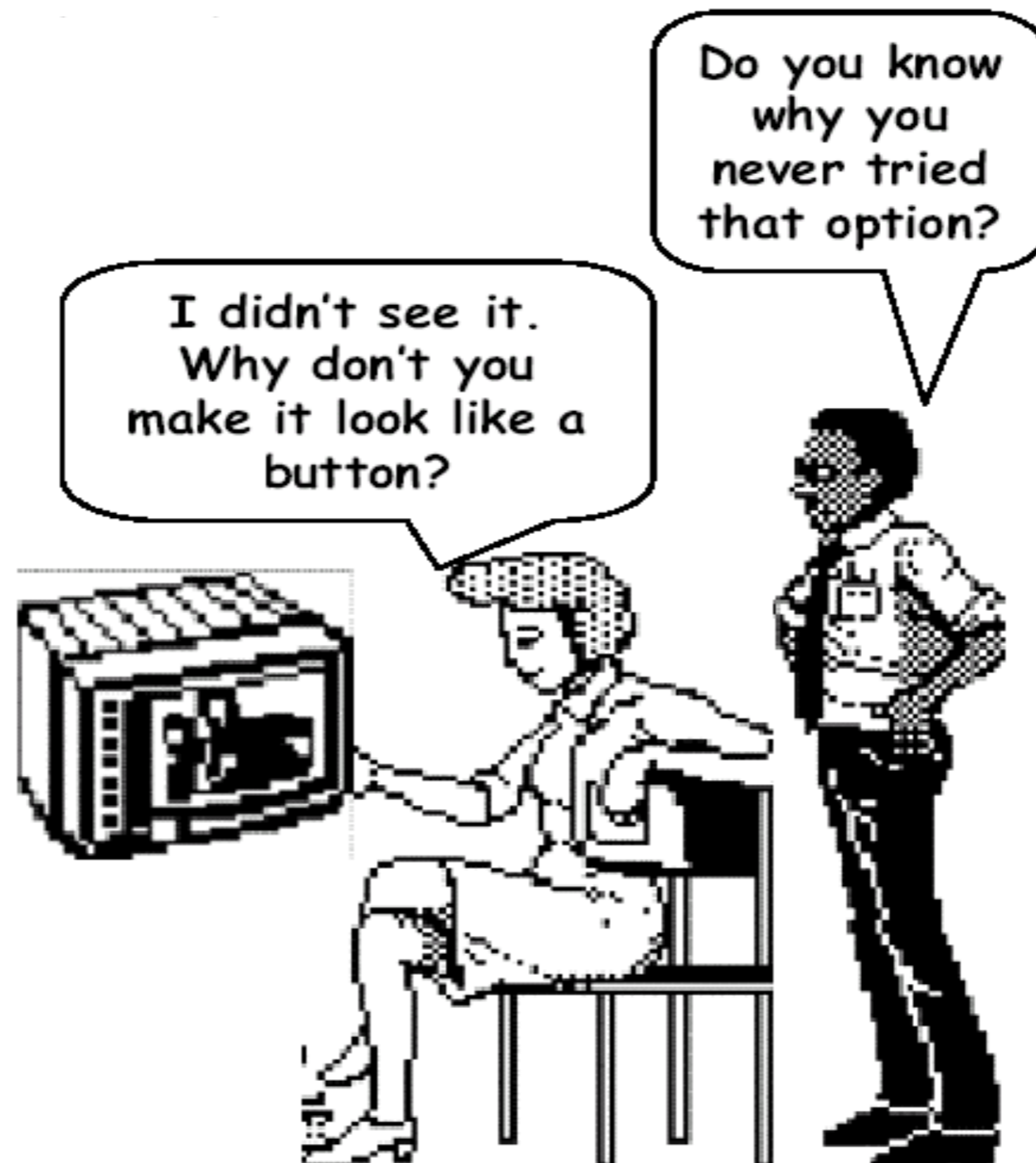
Bob is impor
needs
with

Analyze: Observe & Ask Constructive Interaction



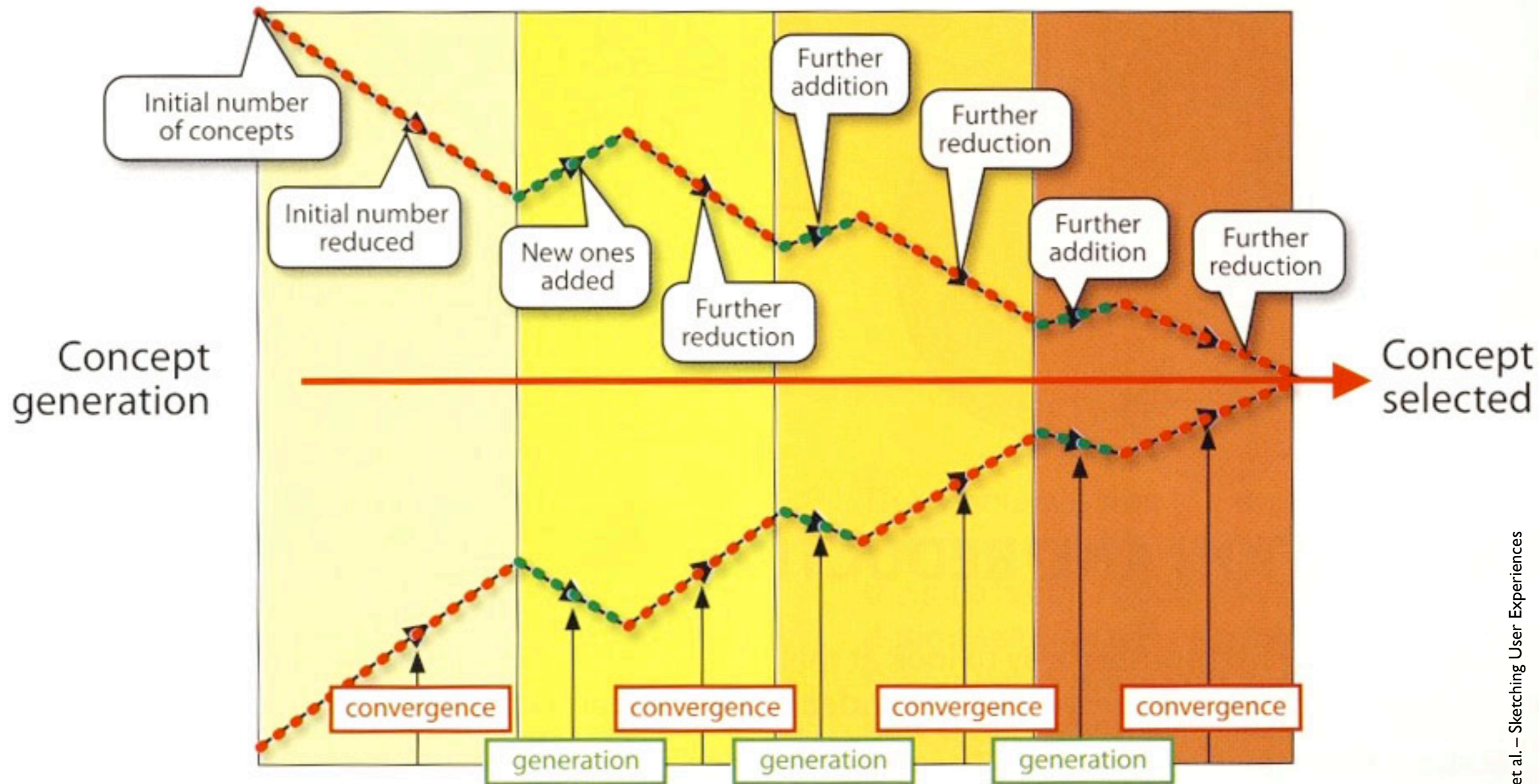
Saul Greenberg

Analyze: Retrospective Testing



Saul Greenberg

Iterate to Expand and Focus Solution



Iterative:

General

overall concepts

Iteration 1

exploratory

Iteration 2

clarification

Iteration 3

resolution

Granularity:

General

overall concepts

Coarse

significant alternatives

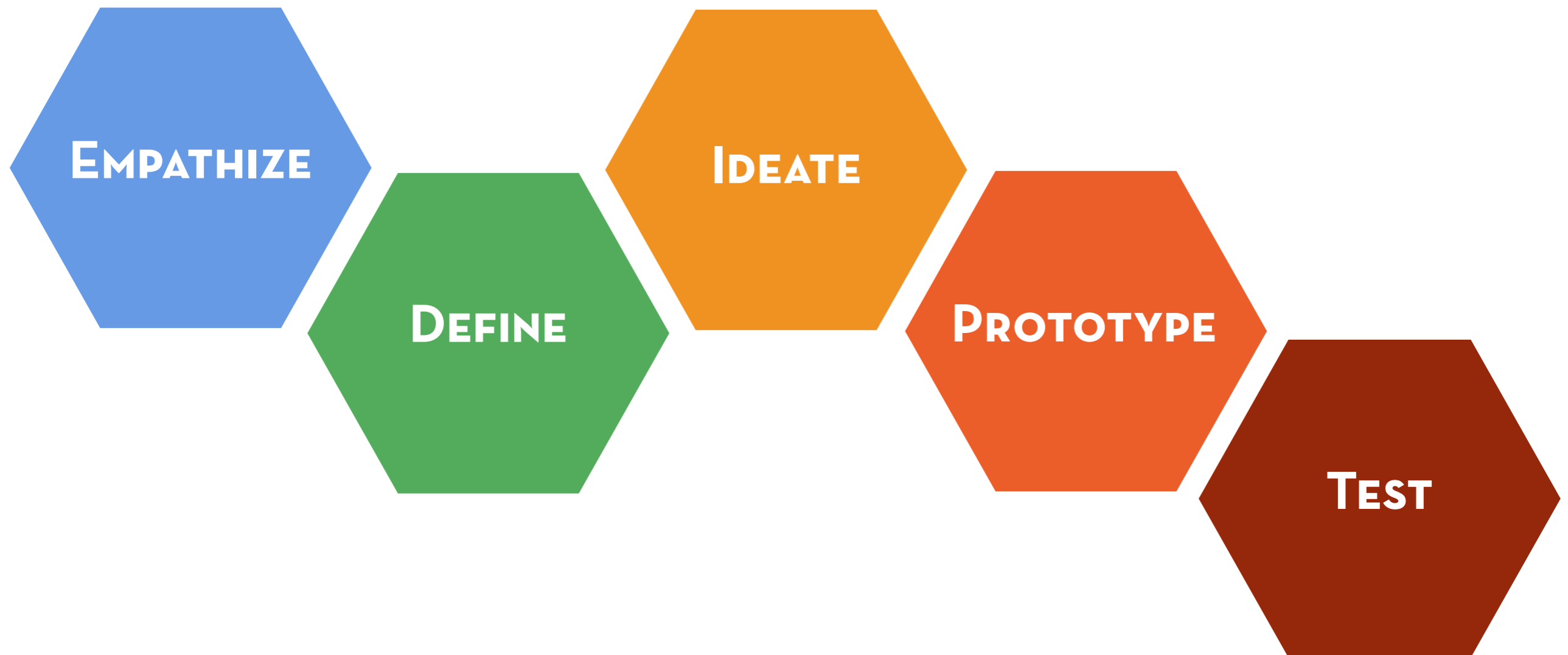
Medium

intermediate development

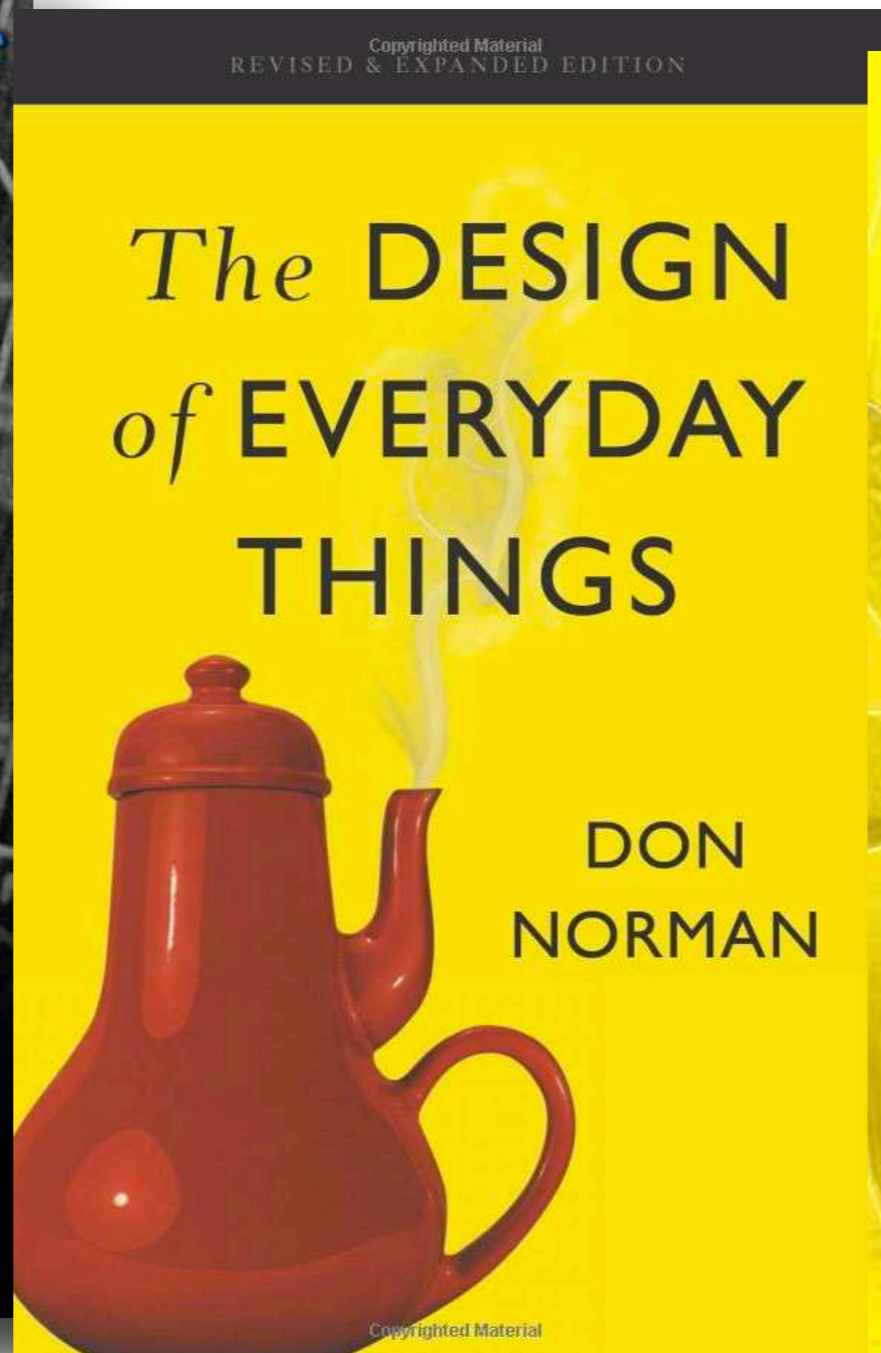
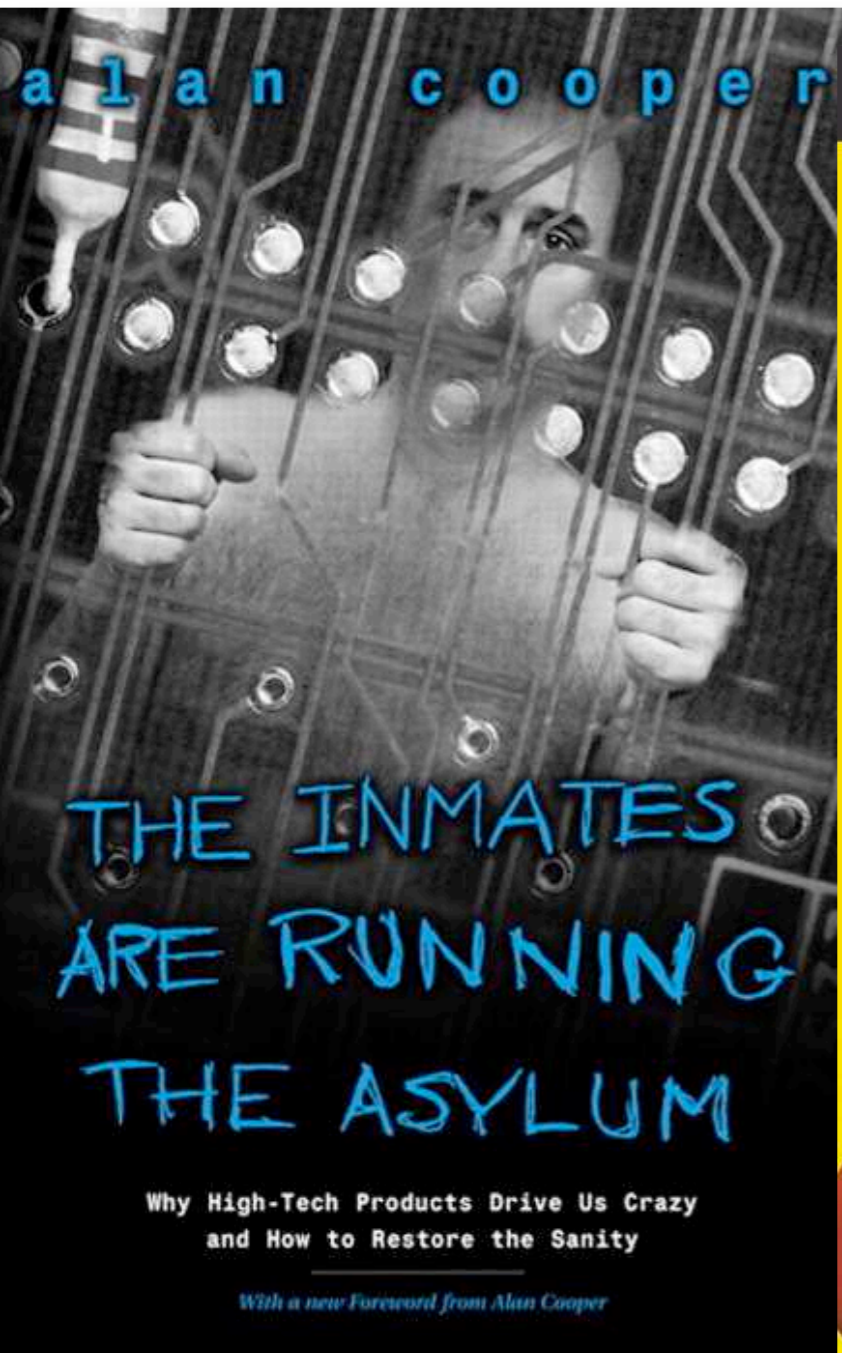
Fine

detailed refinement

Design Thinking



Literature



Why to Make (almost) Everything Useable

- Designing the user experience is a tradeoff
 - Perfect for one — OK for many
- I teach making skills to CS students
 - Iterative SW+HW prototyping is invaluable
 - Electronics = glue between form (mechanics) and function (software)

For Persistent Storage

Paying attention to your user experience pays off

Universal Golden Rules

Simplicity

Visibility & Feedback

Gestalt Laws

Natural Mappings

User's Language

Avoiding Modes

Princ. of Least Surprise

Dialogs

Error Tolerance

Visual Design

Process Tips: Iterative DIA Cycle (Design Thinking)

Design: Explore problem space first

Implement: Storyboard before building

Analyse: Observe and ask others



hci.rwth-aachen.de/fabusability

