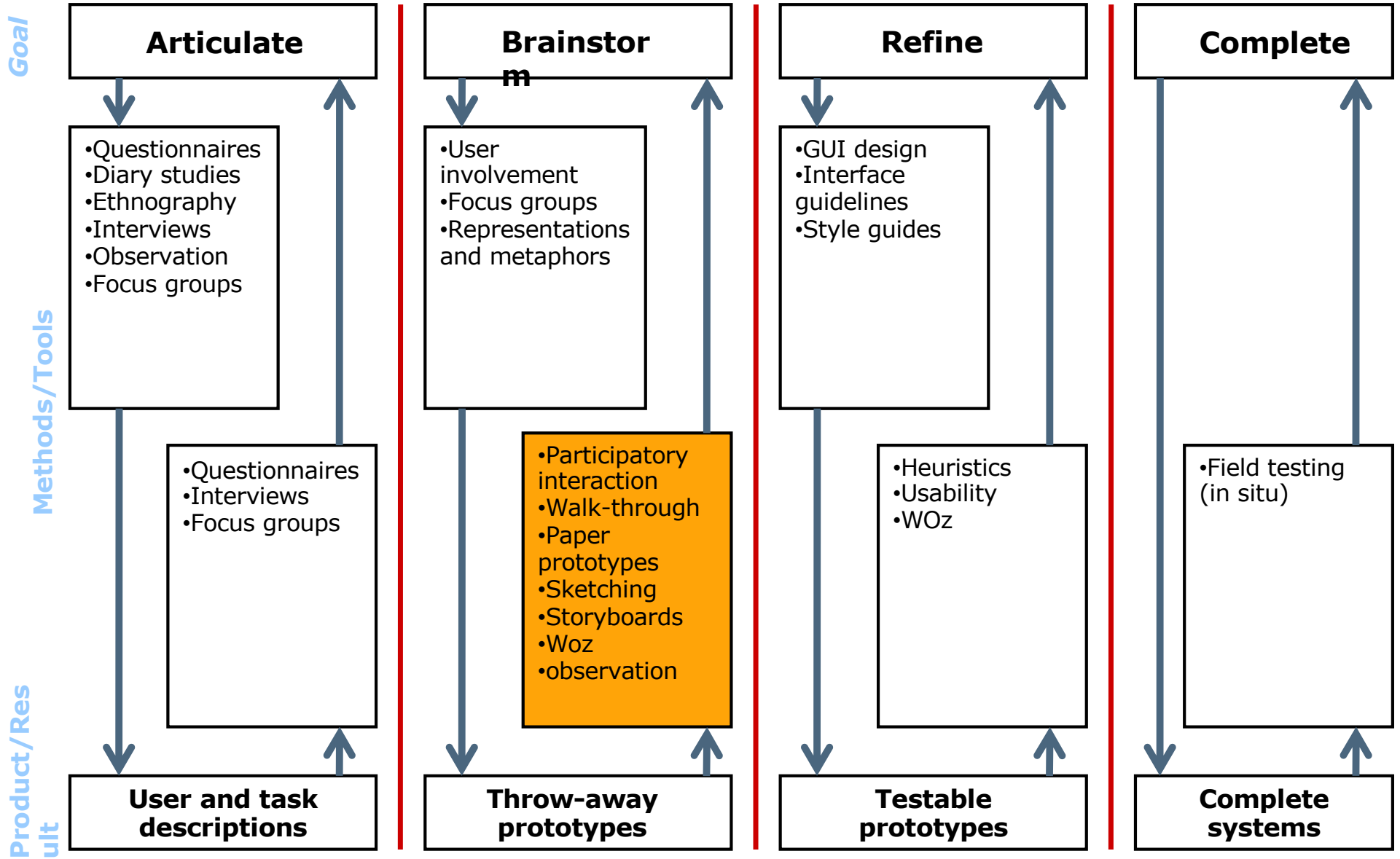
User Centered Design

[5] Prototyping & early evaluations

Prof. Denis Lalanne
Human-IST Institute, University of Fribourg
October 16th, 2018

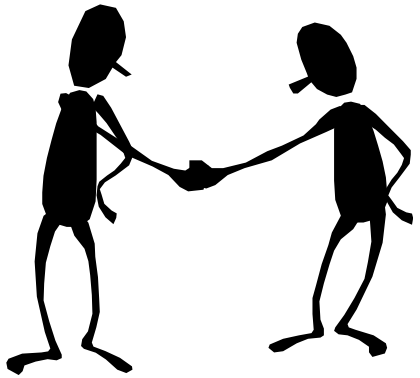
UCD Design Process



Content Overview

- Designing for and with the User
  - Persona, scenarios
  - Sketch, Storyboarding
  - Prototyping (low fidelity, high fidelity) / Wizard-of-Oz
- Evaluating Design with Users (qualitative methods)
  - Observations
  - Walk-through

User Centered Design



- Design is based upon a user's
  - Abilities and real needs
  - Context
  - Work
  - Task

- Golden rule of interface design : **Know The User!**
  - who are they? talk to them, watch them, read about them
- The process of design is a collaboration between designers and customers.
- The result of a good design is a satisfied customer !

A need for participatory Design

- Problems :
  - Wrong intuitions, Interview not precise
  - Poor knowledge of user to answer issues that come up during the design
- Designers should have access to representative users
  => Include usability early in the design / development process

Participatory Design

Involve users early in the design cycle

Up side

- users are excellent at reacting to suggested system designs
- users bring in important “folk” knowledge of work context
- greater buy-in for the system often results

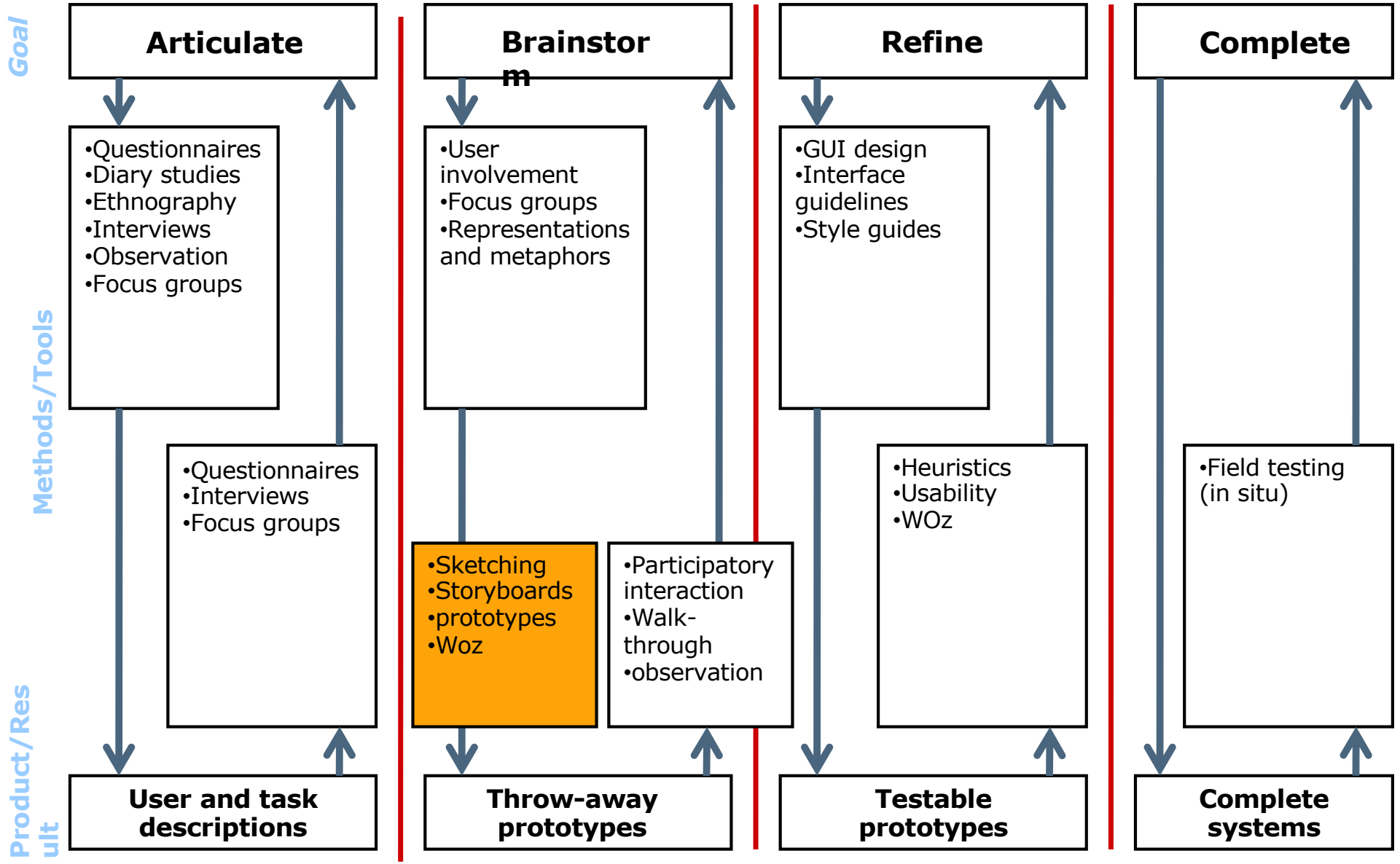
Down side

- hard to get a good pool of end users
- users are not expert designers
  - don’t expect them to come up with design ideas from scratch
- the user is not always right
  - don’t expect them to know what they want

Methods for involving the user

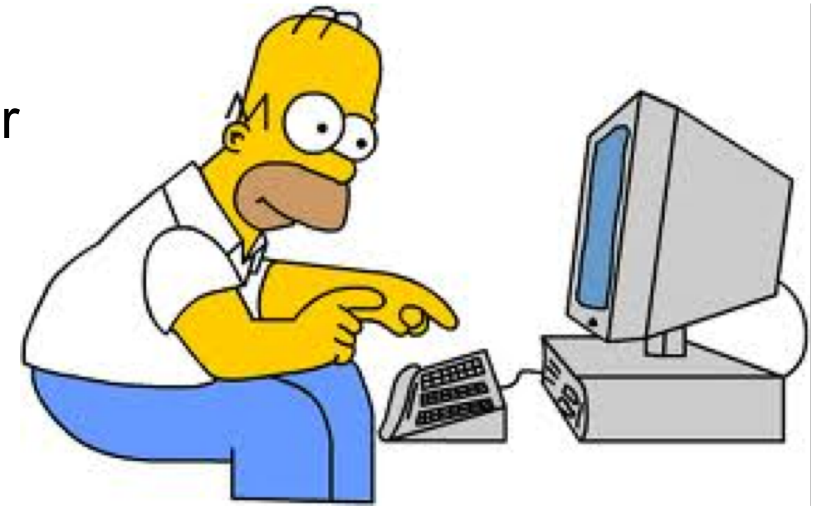
- At the very least, talk to users!!
- Contextual interviews + site visits
  - Interview users in their workplace, as they are doing their job
  - discover user's culture, requirements, expectations
- Explain designs
  - describe what you are going to do
  - get input at all design stages
- Important to have visuals and/or demos
  - people react far differently
  - **this is why prototypes are critical**

Design Methods & Low-fi prototypes



Persona

- A **Persona** is a description of a character that will use your system.
- It acts as a focus for design
- A precise descriptive model of the user
  - A composite archetype
- Details matter
  - Makes him/her ‘real’
  - Give him/her a name, a relevant life
- Use as substitute user
  - what would “Ginette” think



What is a Persona

- A Persona includes information such as:
  - **Personal profile**
    - ✓ name, age, sex, education, job, hobbies, family, socio-economic group, etc
  - **Role**
    - ✓ job role for work-centred sites
    - ✓ position in household for home-centred sites (eg mother)
  - **"Flavouring"**
    - ✓ back-story, what sort of house they live in, how long they've had their job, where their parents live, when they got married, etc.

- A good persona generally gets…
  - **"oh, I know someone just like that"**
  - A persona should be rich enough that they are a believable person.
  - The designer must be able to feel empathy

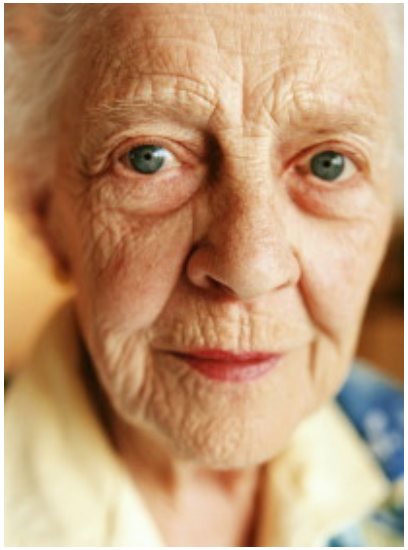
An Example: Ginette Pittet

Biography

- 78 years old
- Just moved to Fribourg from Lausanne
- A son in Bern, a daughter in Geneva
- Doesn't know people in Fribourg yet
- Hasn't been driving for a few years now
- Sometimes feels lonely
- Has a help come in once a week
- Would like to be able to read more

Health

- Has trouble sleeping from time to time. Will wake up in the early hours and often not get to sleep again for 2-3 hours
- A little arthritis in her hands
- Early cataracts, so less acute vision
- Can move about, perhaps not quite as quickly as she could 10 years ago
- Sometimes has a rest in the afternoon

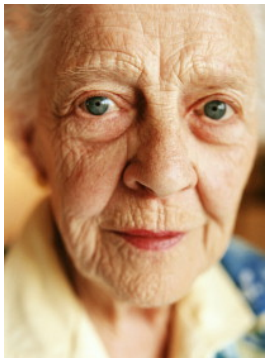


Technology

- Has never used a computer before, and is a little nervous about them
- Has a mobile phone, and instructions on how to use it from her son
- Uses the microwave to prepare many of her meals
- Uses a video recorder, but can't be bothered setting it to record things

A Persona has Goals

- A persona has goals they want to achieve
  - not tasks they wish to perform
  - tasks pre-suppose a solution, goals are invariant



- **Goals (example: Ginette Pittet)**
  - Not to be lonely
  - Keep in touch with her child and their families
  - Not to rely on someone else
- The personas direct the design towards a solution which genuinely achieve its goals (“would Ginette understand this?”)
- The next step is creating design, starting with **Scenarios**

Scenarios in Design

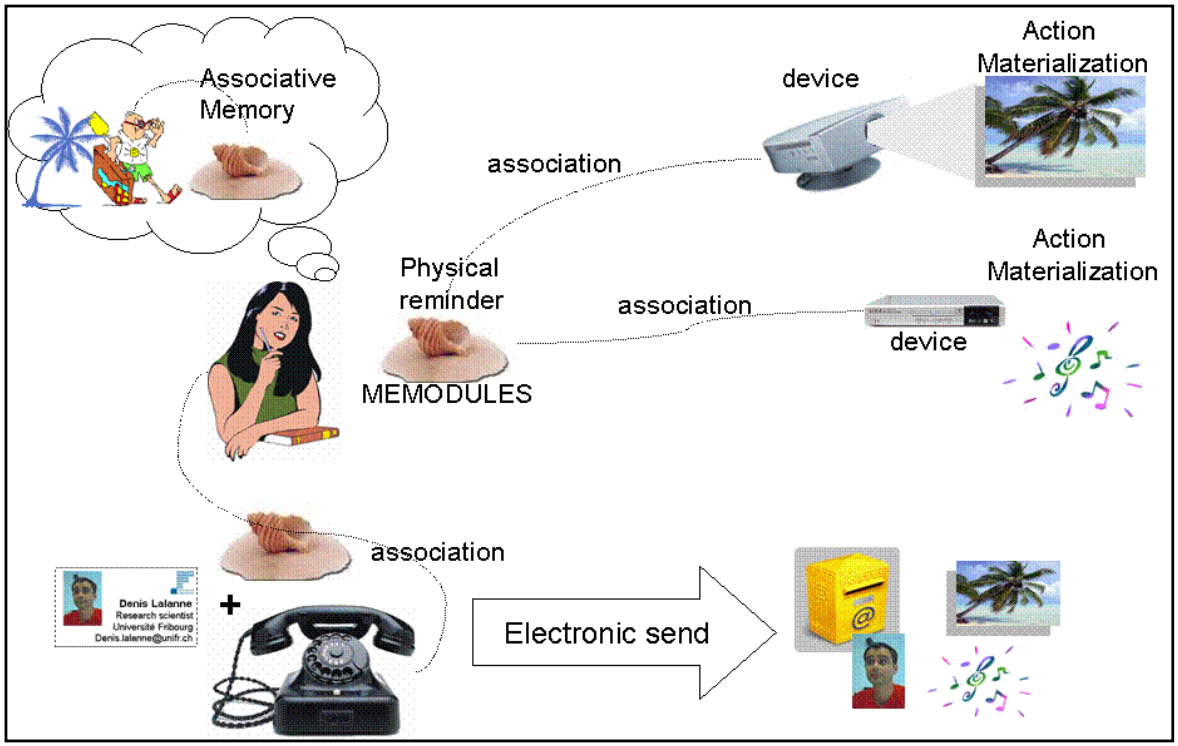
- A scenario is a walk through design, from the point of view of a specific persona.
- It's important that vague scenarios be completed before detail is added, since this forces **Breadth First** design.
- Step-by-step walkthrough
  - what can they see (sketches, screen shots, pictures)
  - what do they do (keyboard, mouse etc.)
  - what are they thinking?
- It is important to consider the context of use, e.g. home, car, etc.
- Use and reuse throughout design
  - Initially a scenario will be very vague, but as the design is filled in and refined it will become more and more specific.

Multi Level of details

- High Level Scenario (breadth first design)
  - Ginette is lonely, and would like to have a chat with someone. She finds out (or knows) who is about that she can chat with. She invites Robert round for a chat, and banishes the loneliness.
  - No mention of the implementation.
- Medium Level Scenario
  - Ginette is lonely…She looks at her device, and it shows her who is online. She sees that Robert is available, so she invites him for an online chat. Robert accepts her invitation, and they start chatting.
  - The implementation is implied.
- Low Level Scenario
  - Ginette is lonely… She looks at her tablet PC, which is switched on, and she can see in the address book that Robert, one of her contacts, is online, and has set his status message as “Want to chat?”. Ginette selects Robert, and then selects the chat program, and taps the start button. She then writes a short invitation to Robert “Hello Robert”…

Scenario example from MeModules

Sandra created a folder full of pictures from her last vacations in Corsica. She glues an RFID (Radio Frequency Identification) on a nice seashell she brought back from Corsica and asks the system to associate it with the picture folder. One week later she wants to show her friends a slideshow of his vacations. She places the seashell, together with a “jazz” MEMODULE, in a little hole next to the plasma screen, which activates the show. Denis who enjoyed the evening and the pictures gives his calling card to Sandra. The following day, Sandra places Denis’ calling card along with the seashell on top of the communicator and makes a brief gesture to send everything.

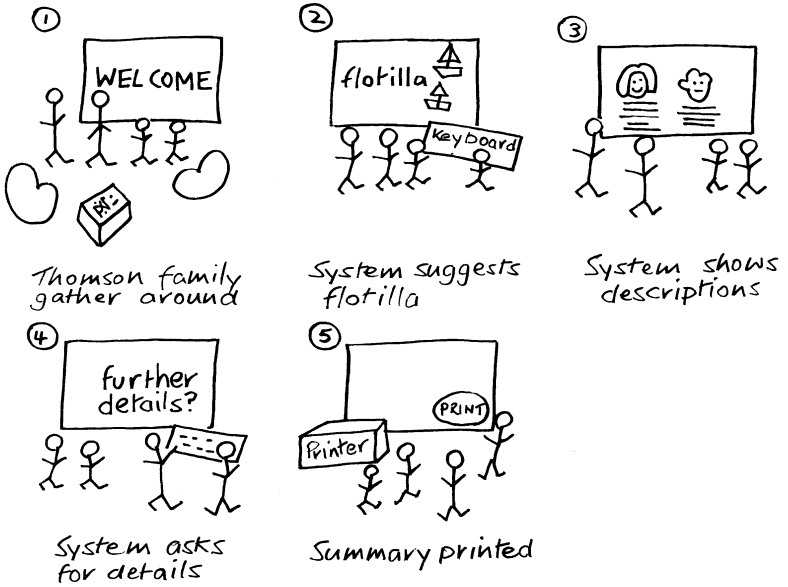


Use personas and scenarios to ..

- Test Design Solutions as they appear in the process
  - Would the persona understand the design?
  - Does the design help the persona achieve their goals?
  - These questions should be answered by the designer, based on their empathy with the persona
  - Keep focused. Once you've selected a Persona stick to it, don't allow yourself to switch to a different one mid way through.
- Communicate with others
  - designers, clients, users
- Express dynamics
  - screenshots – appearance
  - scenario – behaviour
  - ➔ storyboards!

Storyboarding

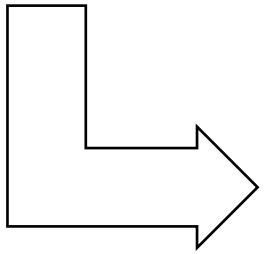
- A series of key frames as sketches
  - Together with scenarios
  - Showing how user progresses through a task using the planned system
- Early stages of interaction design
- Can be mockups of interface



From Hajin Choi. Student in Human-Computer Interaction at Carnegie Mellon University

Storyboards

- stories for design
  - communicate with others
  - validate models
  - understand dynamics
- linearity
  - time is linear - our lives are linear
  - BUT: don't show alternatives

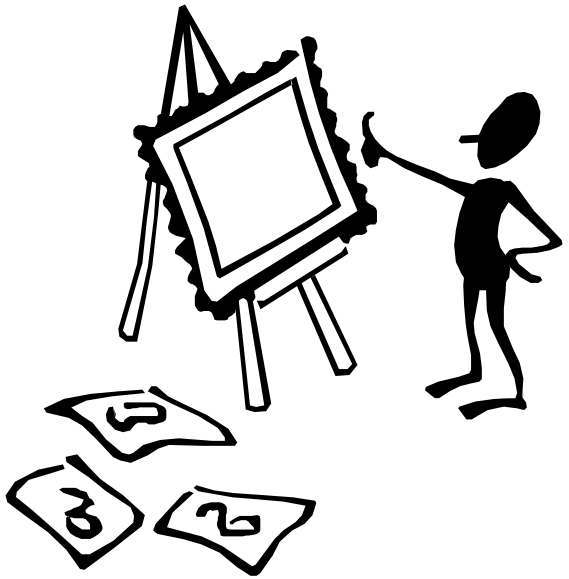


Spotlight: a paper sketch/storyboard
Credit: Sue-Tze Tan, Dept Industrial Design, University of Washington

From Design for the Wild, Bill Buxton

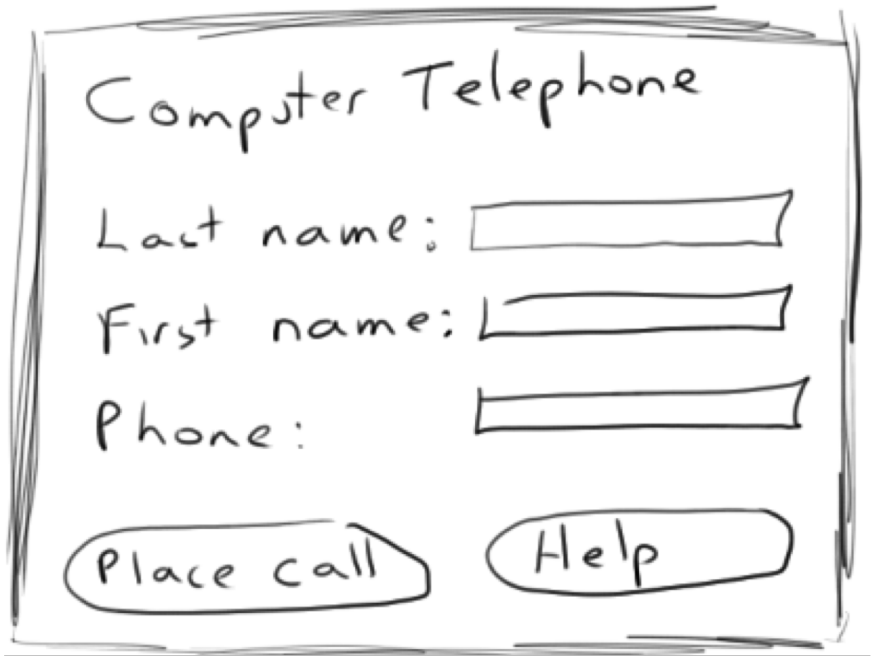
Low Fidelity Prototypes

- Paper mock-up of the interface look, feel, functionality
  - Quick and cheap to prepare and modify
- Purpose
  - Brainstorm competing representations
  - Elicit user reactions
  - Elicit user modifications / suggestions



Sketches

- drawing of the outward appearance of the intended system
- crudity means people concentrate on high level concepts
- **BUT**: hard to envision a dialog's progression



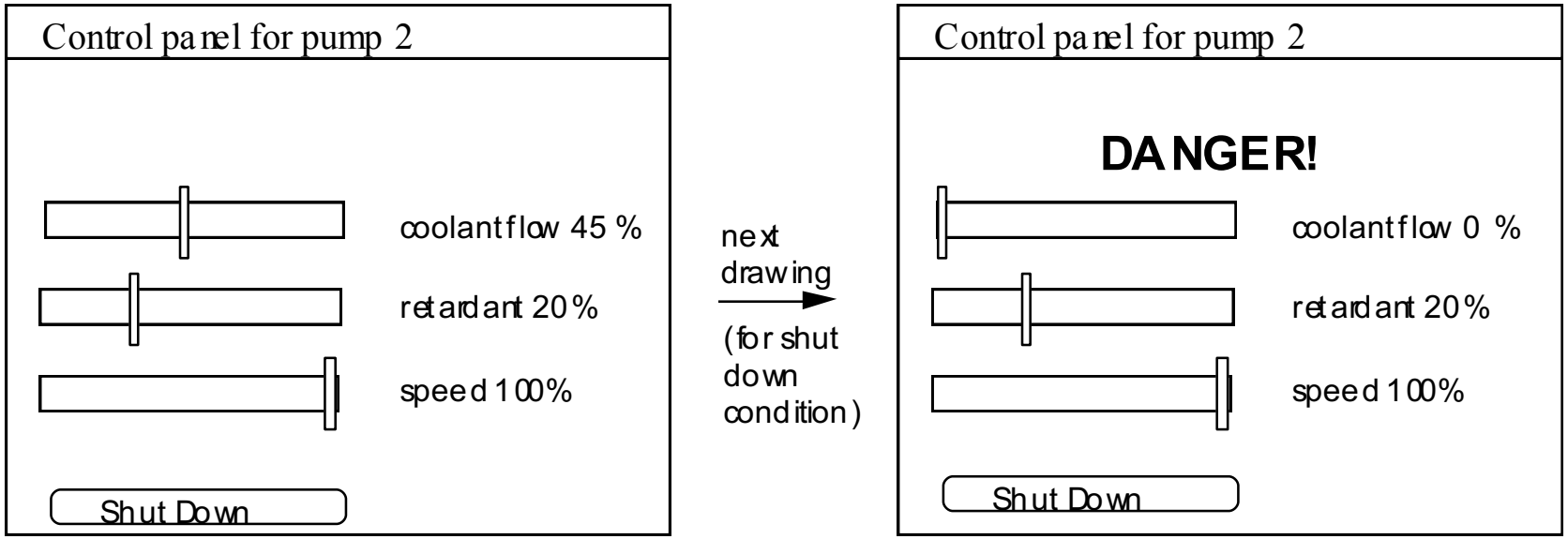
The attributes of sketches

- **Quick**
  - to make
- **Disposable**
  - investment in the concept, not the execution
- **Clear vocabulary**
  - rendering & style indicates it's a sketch, not an implementation
- **Consistency with state**
  - refinement of rendering matches the actual state of development
- **Suggest & explore rather than confirm**
  - value lies in suggesting and provoking what could be i.e., they are the catalyst to conversation and interaction

From Design for the Wild, Bill Buxton

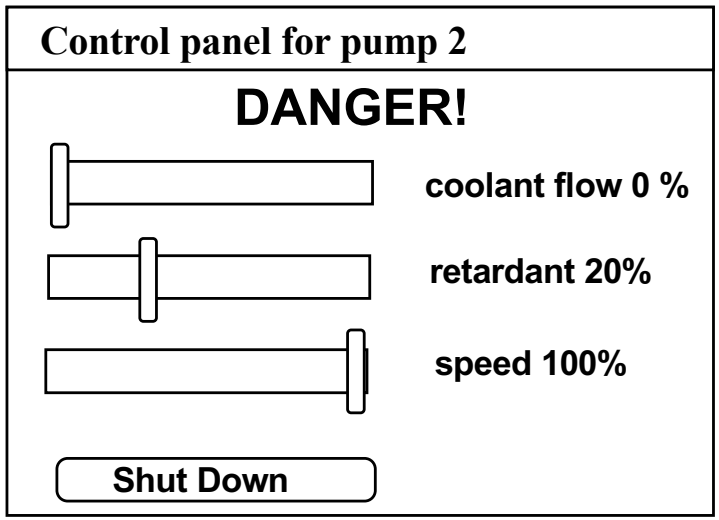
Painting/drawing packages

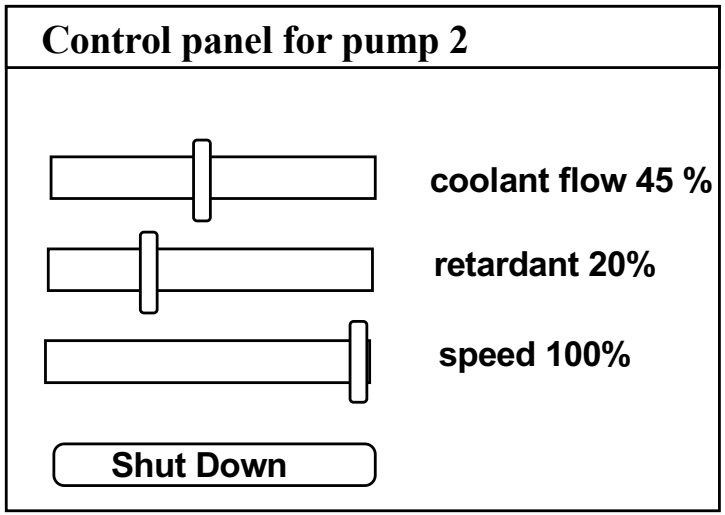
- draw each storyboard scene on computer
  - very thin horizontal prototype
  - The Look yes
  - **BUT**: does not capture the interaction “feel”



Scripted simulations

- create storyboard with media tools
  - scene transition activated by simple user inputs
  - a simple vertical prototype
- user given a very tight script/task to follow
  - appears to behave as a real system
  - **BUT**: script deviations blow the simulation

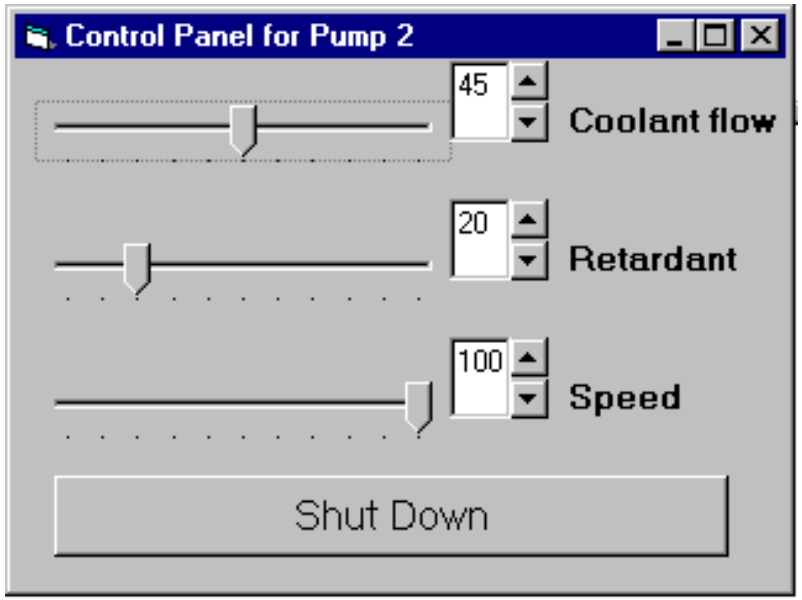




Interface builders

balsamiq

- Design tools for laying out common widgets
- Excellent for showing look and feel
  - a broader horizontal prototype
  - **BUT**: constrained to widget library
- Vertical functionality added selectively
  - through programming

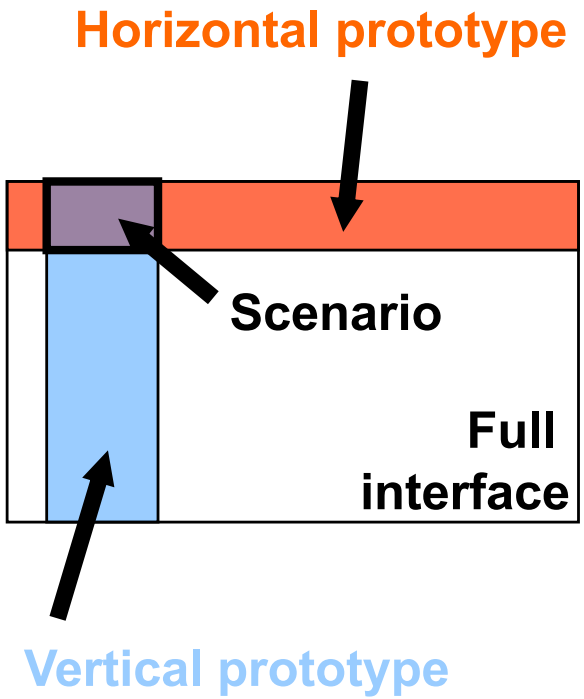


Medium fidelity prototypes

- Prototyping with a computer
  - simulate some but not all features of the interface
- Purpose
  - engaging for end users
  - provides sophisticated but limited scenario for the user to try
  - can test more subtle design issues
- Dangers
  - users reluctant to challenge designer
  - users reluctant to touch the design
  - management may think its real!

Compromises in Prototyping

- All prototypes involve compromises
  - Slow response, sketchy icons, limited functionality, scripted?
- Common types of compromise / limitation
  - Horizontal prototypes
    - ✓ the entire surface interface with no underlying functionality
    - ✓ a simulation; no real work can be performed
  - Vertical prototypes
    - ✓ includes in-depth functionality for only a few selected features
    - ✓ common design ideas can be tested in depth
  - Scenario
    - ✓ scripts of particular fixed uses of the system;
    - ✓ no deviation allowed



Nielsen, J. (1993) *Usability Engineering,* p93-101, Academic Press.

Wizard-of-Oz Prototyping

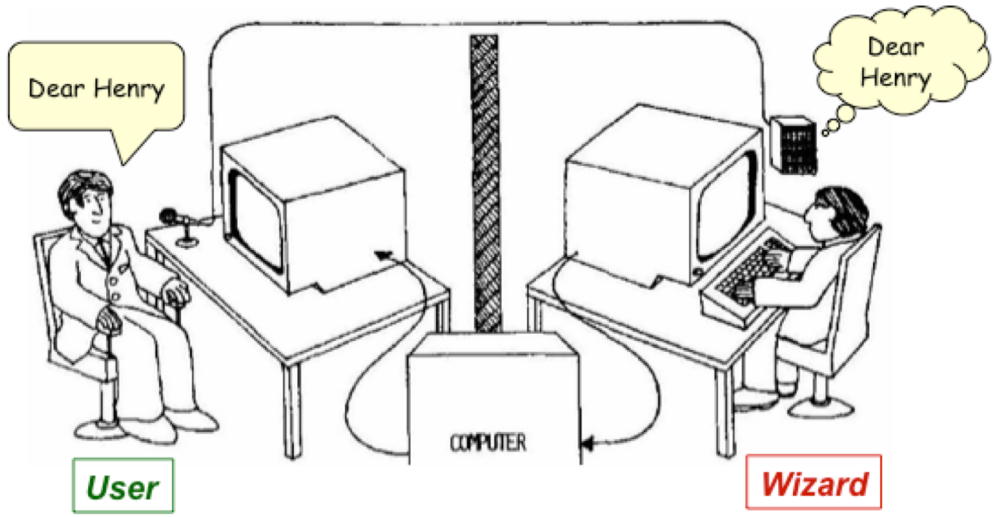
- The user thinks he is interacting with a computer but, in fact, a human operator is manipulating the interface and simulates the software's responses to the user.
- Usually done early in design to understand users' expectations.
- A low- to medium-fidelity prototyping method with limitations.

The last thing you should do when beginning to design an interactive system is write code
(Bill Buxton).

It is fidelity of the experience, not the fidelity of the prototype, sketch or technology that is important
(Bill Buxton)

Wizard of Oz

- Possibility to evaluate novel user interface concepts before the technology is mature enough.
- Human 'wizard' simulates system response
  - controls computer to simulate appropriate output
  - uses real or mock interface
  - wizard sometimes visible, sometimes hidden
    - ✓ "pay no attention to the man behind the curtain!"
- good for:
  - testing futuristic ideas & functionalities



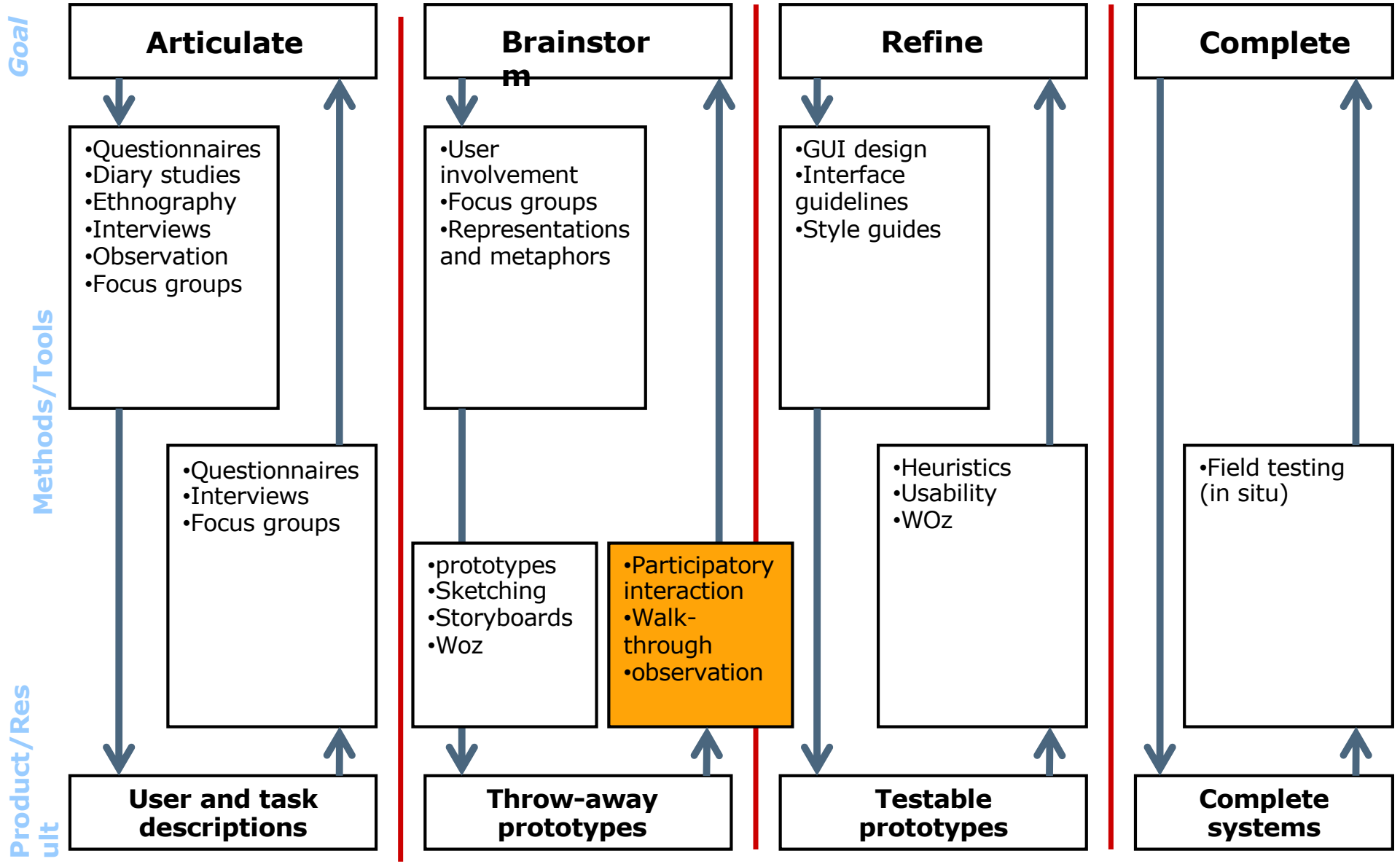
From Gould, Conti & Hovanyecz, Comm ACM 26(4) 1983

The listening typewriter, IBM 1984

WoZ examples

- Gesture recognition
  - Aim: Gestural interaction with regard to the comprehensiveness, comfort and learnability
  - 12 test participants (6 male, 6 female)
  - 2 conditions: standing vs sitting
  - Different tasks (e.g. moving the focus of a map of GoogleEarth from Europe to Australia).
  - Performance data (completion time) + usability questionnaire ISONORM 9241/10
  - Results indicated:
    - Performance measures do not differ between the two conditions
    - Users mentioned the risk of fatigue after long duration in standing condition
- IM2 (Archivus)

Qualitative evaluations with users



Evaluating interfaces with users

- Basic idea: directly involve people in the evaluation
  - They know their domain (usually better than you!)
- Type
  - Qualitative
    - ✓ HOW: observe users, gather explanations and opinions
    - ✓ OUTPUT: list of findings, needs, requirements
    - ✓ (+) ready explanation, easy solution, good to test concepts
    - ✓ (-) not measurable, hard to compare and track, chaotic process
    - ✓ **E.g. direct observations, query techniques, continuous evaluations**
  - Quantitative
    - ✓ HOW: measure efficiency (time), accuracy (errors), satisfaction
    - ✓ OUTPUT: measures
    - ✓ (+) measurable, can be tracked, repeatable, allows comparison
    - ✓ (+) Test theoretical questions, Gain insights over HCI processes
    - ✓ (-) hard methods, difficult to translate in solutions (more about findings)

Direct observations

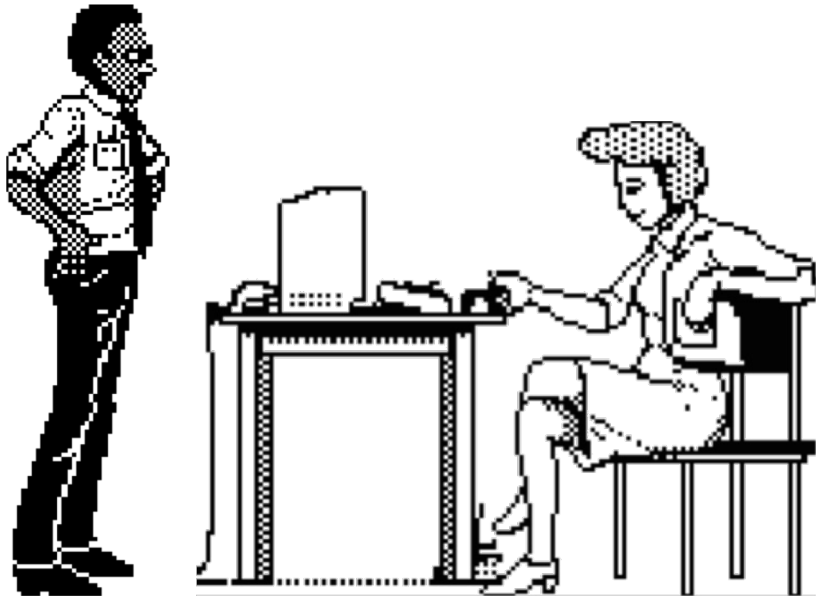
- Evaluator observes users interacting with system
  - in lab:
    - ✓ user asked to complete a set of **pre-determined** tasks
  - in field (field studies):
    - ✓ user goes through normal duties

- Value
  - excellent at identifying gross design/interface problems
  - validity depends on how controlled/contrived the situation is

Simple observation method

- User is given the task
- Evaluator just watches the user

- Problem
  - does not give insight into the user's decision process or attitude

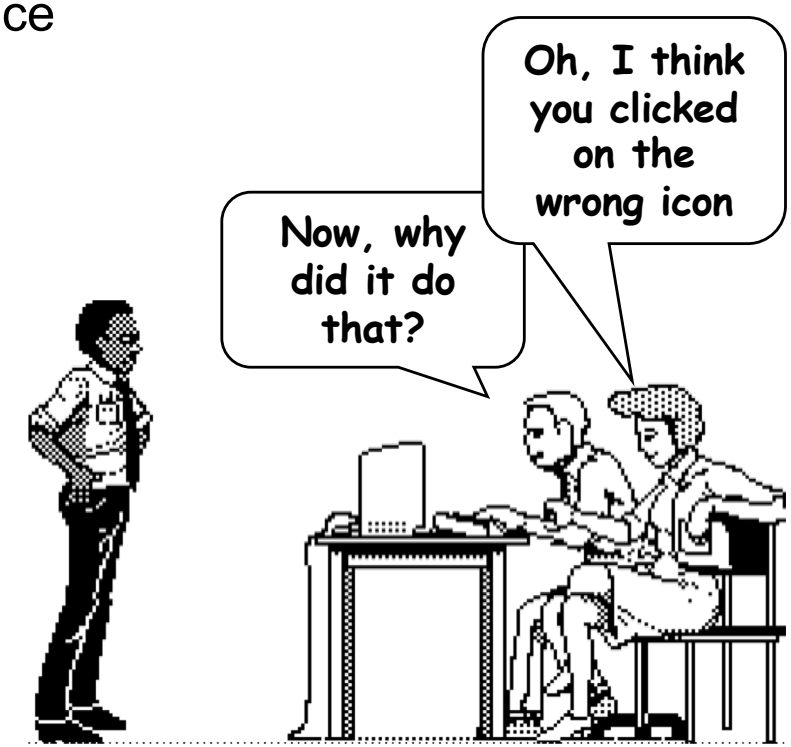


Think aloud method

- Users speak their thoughts while doing the task
  - what they are trying to do
  - why they took an action
  - how they interpret what the system did
  - Pros:
    - ✓ gives insight into what the user is thinking
    - ✓ most widely used evaluation method in industry
  - Cons:
    - ✓ may alter the way users do the task
    - ✓ unnatural (awkward and uncomfortable)
    - ✓ hard to talk if they are concentrating

Constructive interaction method

- Two people work together on a task
  - monitor their normal conversations
  - removes awkwardness of think-aloud
- Co-discovery learning
  - use semi-knowledgeable “coach” and novice
  - only novice uses the interface
    - ✓ novice ask questions
    - ✓ coach responds
  - gives insights into two user groups

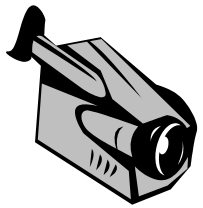


Recording observations

- Record user actions for later analysis
  - otherwise risk forgetting, missing, or misinterpreting events
  - paper and pencil
    - ✓ primitive but cheap
    - ✓ observer records events, comments, and interpretations
    - ✓ hard to get detail (writing is slow)

	General actions			Graph editing			Errors	
Time	text editing	scrolling	image editing	new node	delete node	modify node	correct error	miss error
09:00	X							
09:02				X				
09:05							X	
09:10					X			
09:13								

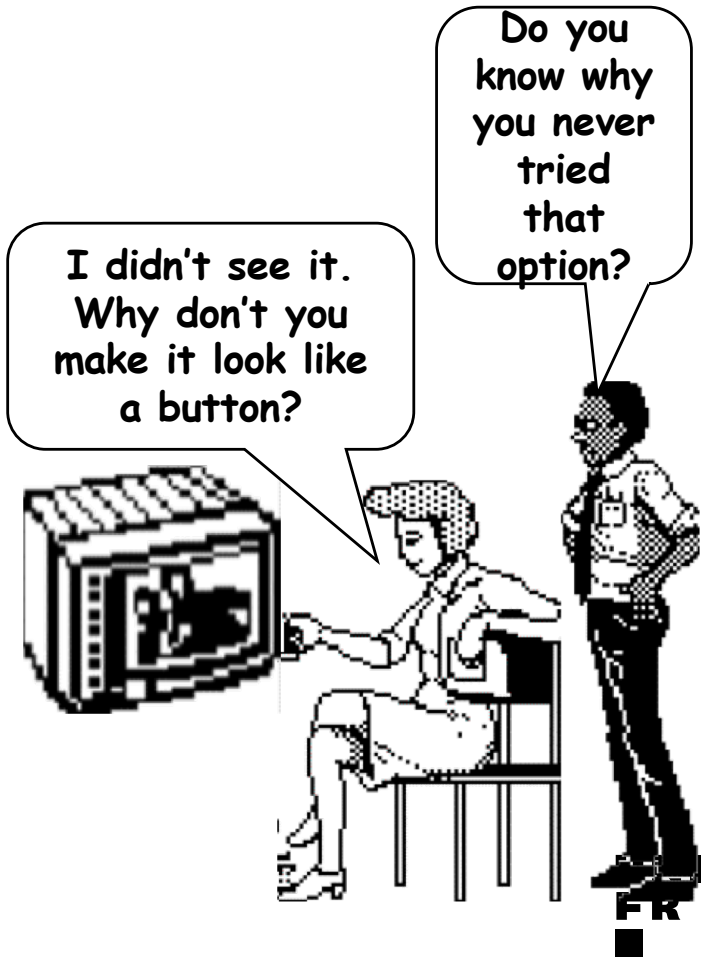
  - audio recording
    - ✓ good for recording think aloud talk
    - ✓ hard to tie into on-screen user actions



  - video recording
    - ✓ can see and hear what a user is doing
    - ✓ one camera for screen, rear view mirror useful…
    - ✓ initially intrusive
  - Logging, eye tracking, …

Retrospective testing interviews

- Critical incidence interviews
  - people talk about incidents that stood out
  - usually discuss extremely annoying problems with fervor
  - not representative, but important to them
  - often raises issues not seen in lab tests
- Post-observation interview to
  - perform an observational test
  - create a video record of it
  - have users view the video and comment on what they did
    - ✓ clarify events that occurred during system use
    - ✓ users often offer concrete suggestions



Questionnaires and Surveys

- Questionnaires / Surveys
  - preparation “expensive,” but administration cheap
  - can reach a wide subject group (e.g. mail, web)
  - results can be quantified
- But
  - only as good as the questions asked
- How
  - establish the purpose of the questionnaire
  - **do not ask questions whose answers you will not use!**
  - Online forms (e.g. surveymonkey)

Styles of Questions

- Open-ended questions
  - good for general subjective information
- Closed questions
  - makes questionnaires easy to fill in
  - can be easily analyzed
- Scalar
  - ask user to judge a specific statement on a numeric scale
- Multi-choice
- Ranked
  - useful to indicate a user's preferences
- Combining open-ended and closed questions
  - gets specific response, but allows room for user's opinion

Can you suggest any improvements to the interfaces?

Do you use computers at work:
✓ often O sometimes O rarely

Characters on the screen are:
hard to read easy to read
1 2 O 4 5

Rank the usefulness of these methods of issuing a command (1 most useful, 2 next most useful..., 0 if not used)
__2__ command line
__1__ menu selection
__3__ control key accelerator

Continuous Evaluation

- Monitor systems in actual use
  - good for seeing “real life” use
  - usually late stages of development
    - ✓ i.e. beta releases, delivered system
  - fix problems in next release
- User feedback
  - users can provide feedback to designers while using the system
    - ✓ help desks, forums, email
  - best combined with trouble-shooting facility
- A/B testing (quantitative type of evaluation in production)
  - E.g. google, color of links

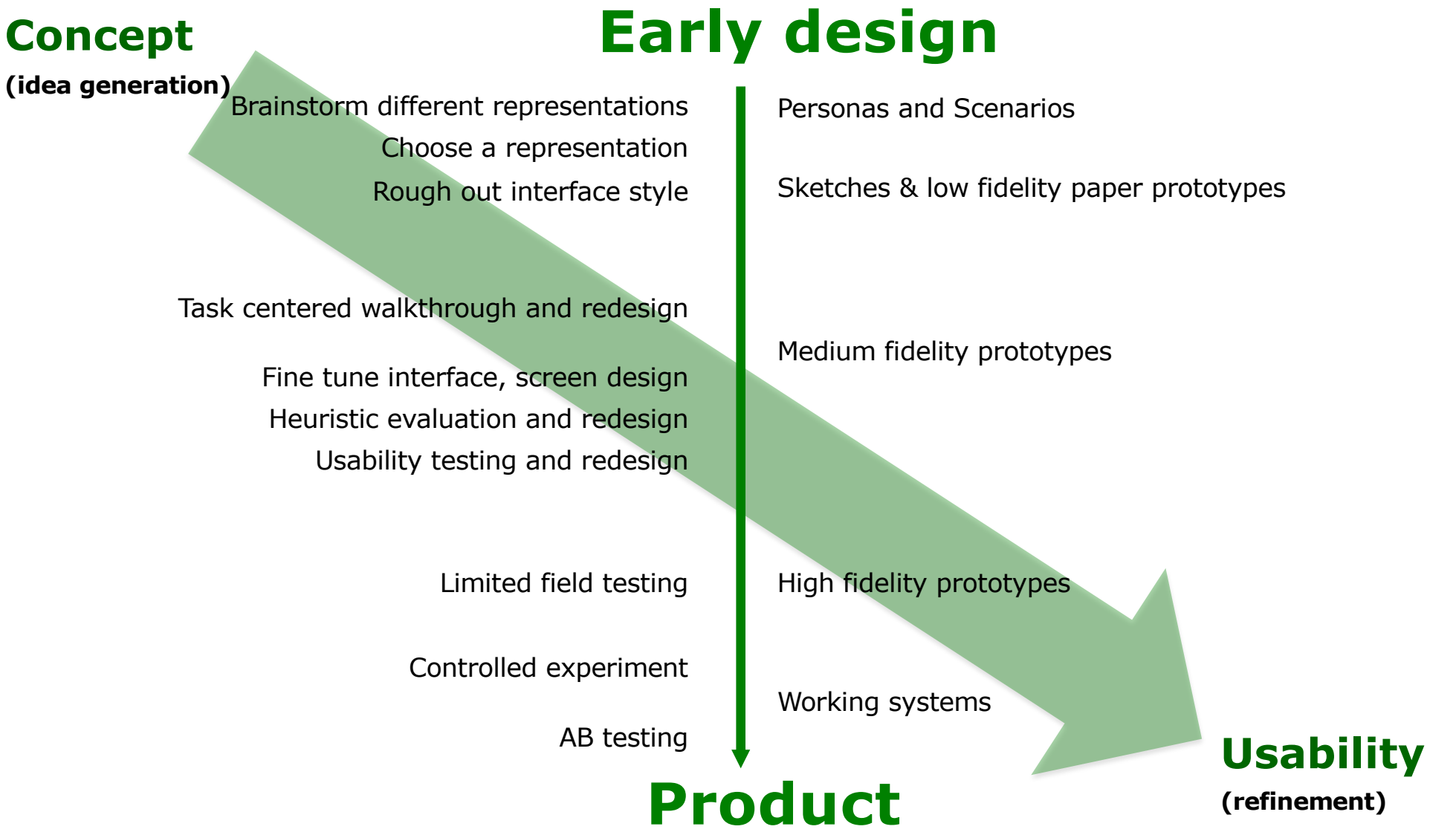
Wrap Up

Prof. Denis Lalanne
Human-IST Institute, University of Fribourg
October 16th, 2018

HUMAN-IST
HUMAN CENTERED INTERACTION SCIENCE & TECHNOLOGY

UNI FR

UCD: From concept to usability



Key Points / What You Should Know

- Participatory design
- Design methods:
  - Personas, Scenarios
  - Prototyping (sketches, low-fidelity, storyboarding, high-fidelity)
- Evaluation methods (with users):
  - Wizard-of-Oz
  - Differences between
    - ✓ Qualitative versus quantitative
    - ✓ Formative versus sommative
  - Qualitative methods to evaluate design, low fidelity prototype or product