

CSC 436, Fall 2017

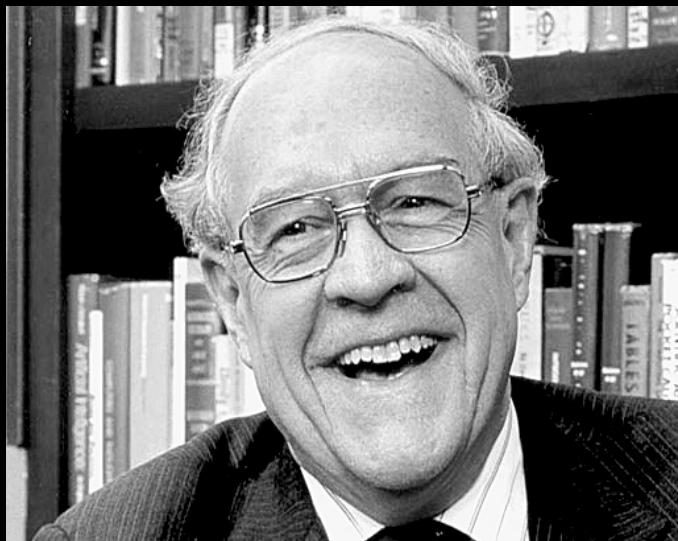
# Customer Needs and Wants

*Ravi Sethi*



# Inherent versus Incidental Impediments

Fred Brooks, in a 1986 essay on impediments to software



**“The hardest single part of building a software system is deciding precisely what to build. ... the most important function that software builders do for their clients is the iterative extraction and refinement of the product requirements. For the truth is clients do not know what they want.”**

# Requirements or Specifications?

Think *Customer Requirements* and *Product Specifications*

- ## Requirements

- A ***requirement*** is a constraint that must be met
- We use the term **requirement** by itself as an abbreviation for a “customer requirement” that must be met to solve a customer problem or achieve a customer objective
- The term may also be used in other contexts: e.g., functional requirements, performance requirements, interface requirements,

# Requirements or Specifications?

Think *Customer Requirements* and *Product Specifications*

- **Specifications**

- A *specification* is a complete, precise, verifiable description of a system or of an aspect of a system
- Example: Software Requirements Specification (SRS)

“A good SRS is (1) Unambiguous, (2) Complete, (3) Verifiable, (4) Consistent, (5) Modifiable, (6) Traceable, and (7) Usable during the Operations and Maintenance Phase.”\*

# Risk Factors for Software Projects

1998 study asked project managers

- Asked to identify and rank order risk factors
  - Three independent panels in Finland, Hong Kong, and the US
- Results were consistent
  - 7 of the 11 most serious risks related to either users or requirements
- Note
  - Project managers viewed risks that were outside their control as being more severe

# Risk Factors for Software Projects

## Top risk factors identified by project managers (1998 study)

1. Lack of top management commitment to the project
2. Failure to get user commitment
3. Misunderstanding the requirement
4. Lack of adequate user involvement
5. Failure to manage end-user expectations
6. Changing scope/requirements
7. Lack of required knowledge/skills in the project personnel
8. Lack of frozen requirements
9. Introduction of new technology
10. Insufficient/inappropriate staffing
11. Conflict between user departments

# Risk Factors for Software Projects

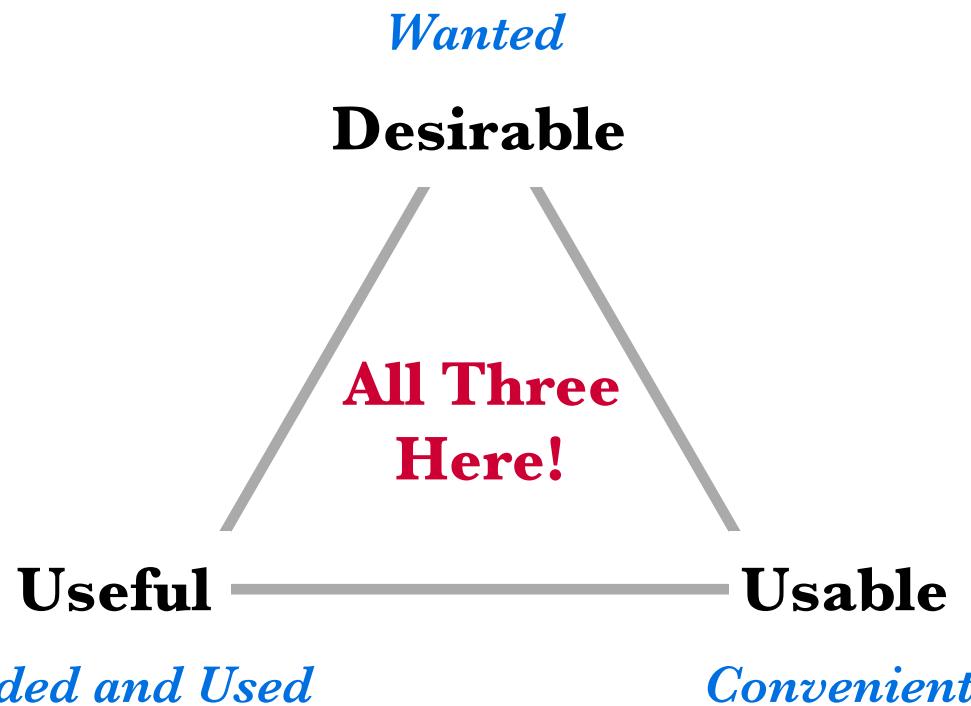
## Most are User-Related or Requirements-Related

1. Lack of top management commitment to the project
2. Failure to get user commitment
3. Misunderstanding the requirement
4. Lack of adequate user involvement
5. Failure to manage end-user expectations
6. Changing scope/requirements
7. Lack of required knowledge/skills in the project personnel
8. Lack of frozen requirements
9. Introduction of new technology
10. Insufficient/inappropriate staffing
11. Conflict between user departments

# Delight the Primary Customer

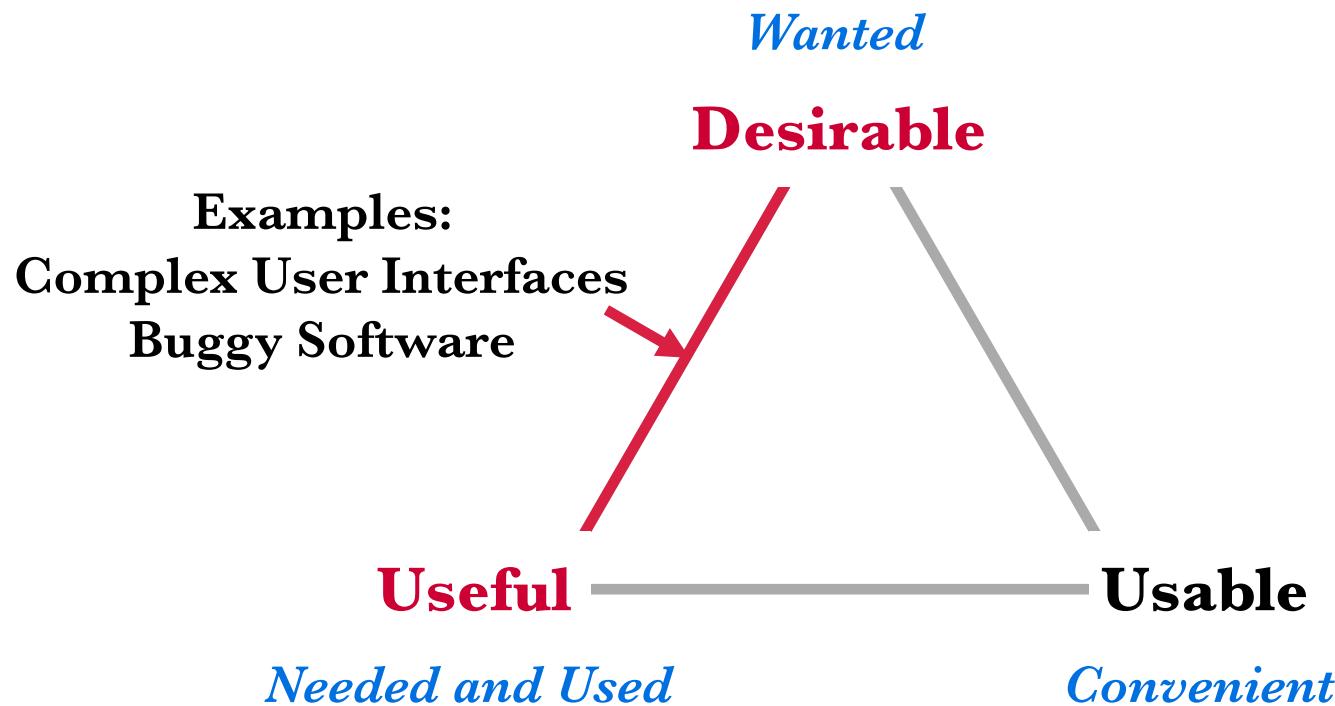
# Meeting Customer Needs

Great products are useful, usable, and desirable



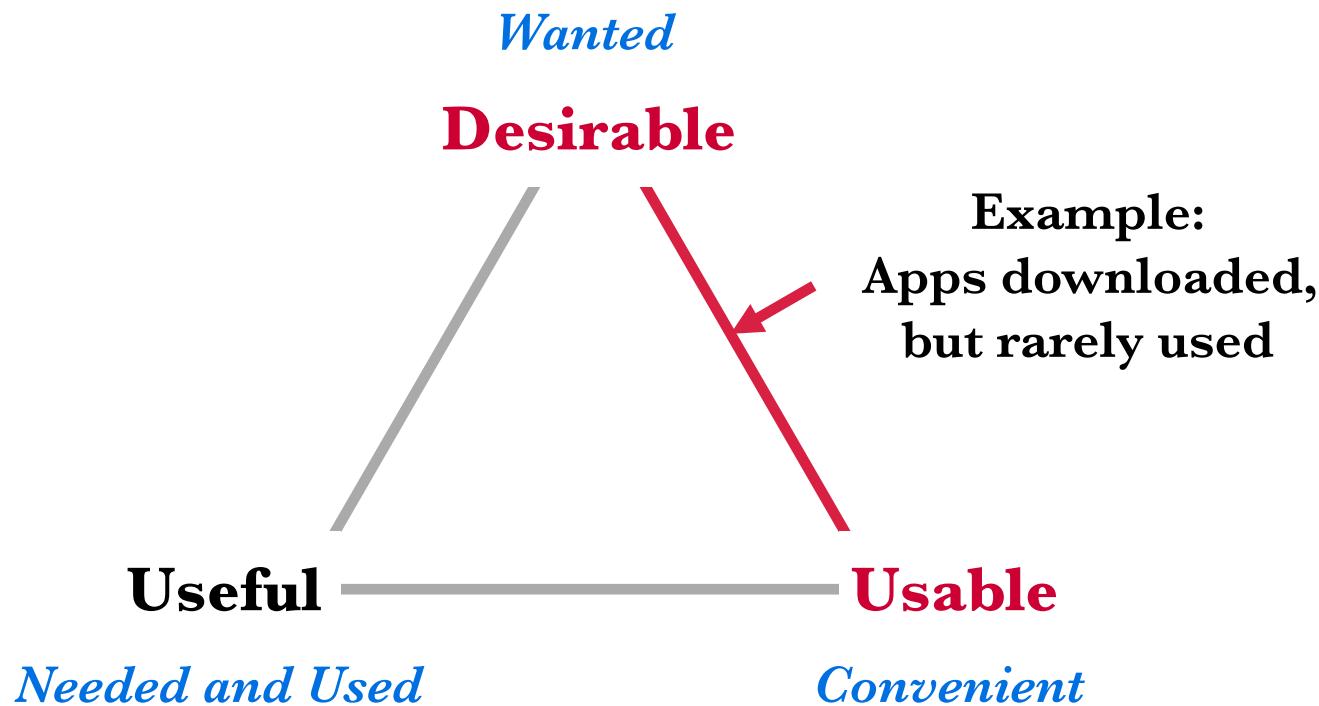
# Meeting Customer Needs

Desirable and useful, but not usable



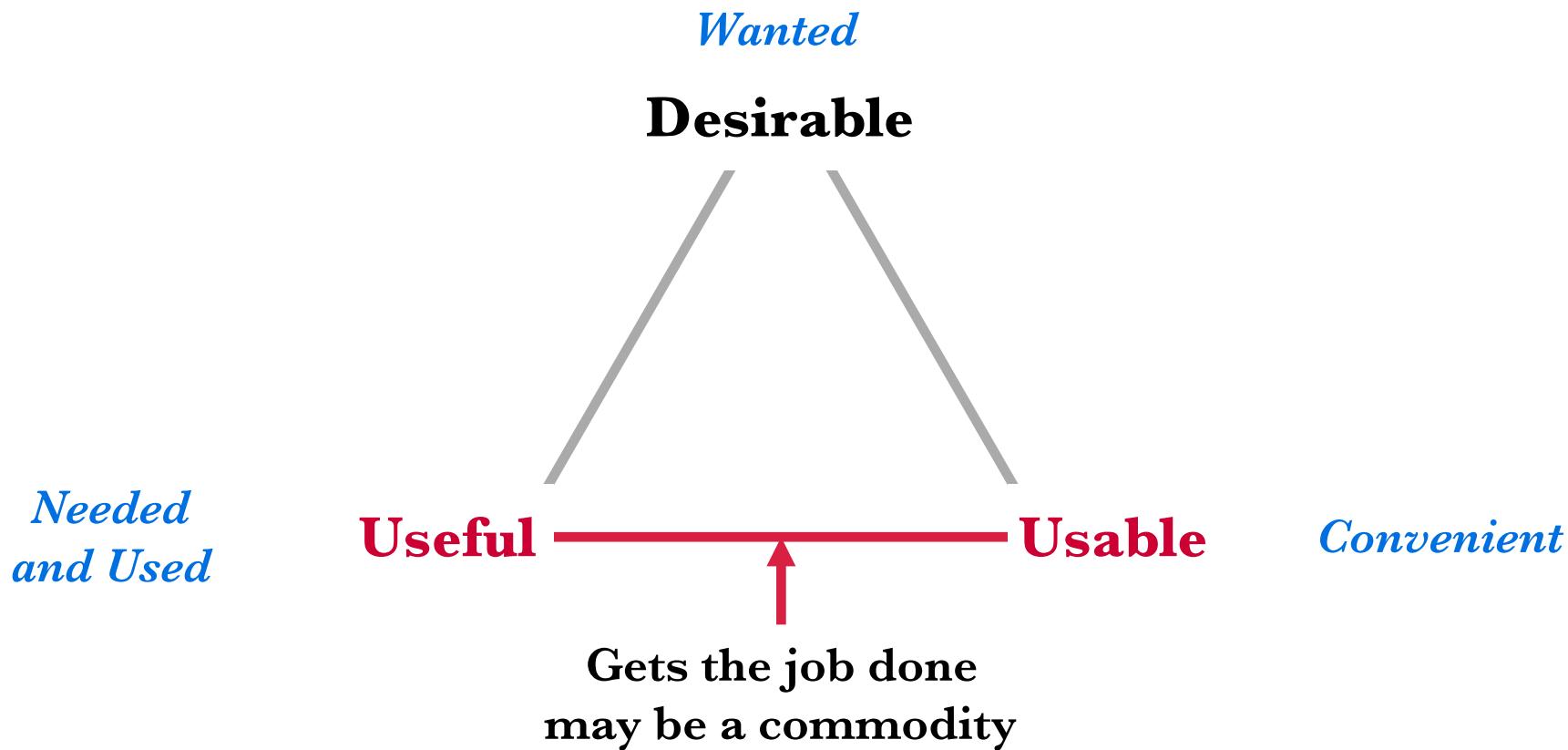
# Meeting Customer Needs

## Desirable and Usable, but not Useful



# Meeting Customer Needs

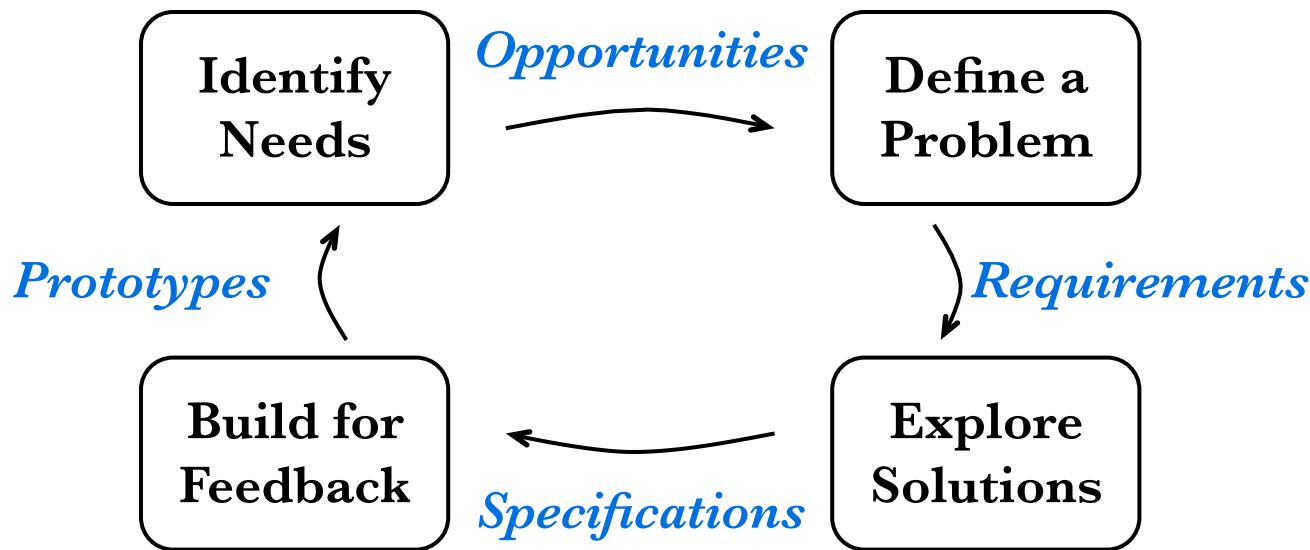
Useful and Usable, but not necessarily Desirable



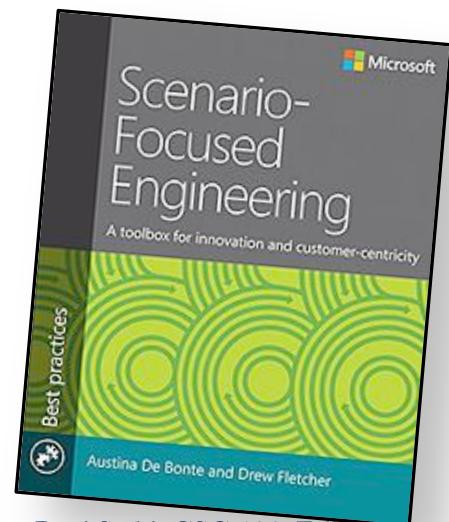
# Iterative Customer Interactions

# Fast Feedback Cycle

## Bringing Design Thinking to Engineers



- **Developed for teams at Microsoft**
  - Training taken by 22,000 engineers since 2008



# Identifying and Addressing Customer Needs

## Scenario-Focused Engineering

### PROBLEM DOMAIN

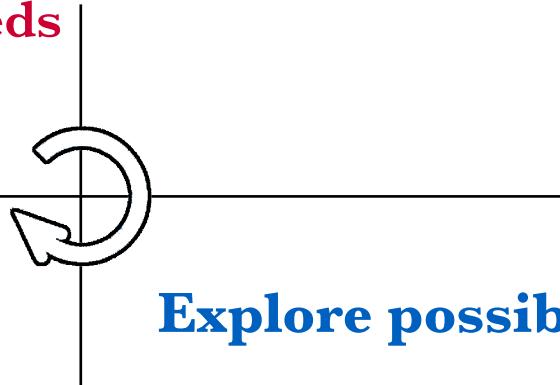
#### Identify needs and wants

- Focus on the primary customer
- Probe for the drivers for the needs
- Listen, observe, and empathize

#### Define a problem to solve

#### Build for early feedback

#### Explore possible solutions



### SOLUTION DOMAIN

# Identifying and Addressing Customer Needs

## Scenario-Focused Engineering

### PROBLEM DOMAIN

#### Identify needs and wants

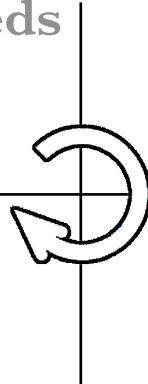
- Focus on the primary customer
- Probe for the drivers for the needs
- Listen, observe, and empathize

#### Define a problem to solve

- Prioritize stakeholder wants
- Define acceptance criteria
- Write user stories and scenarios

#### Build for early feedback

#### Explore possible solutions



### SOLUTION DOMAIN

# Identifying and Addressing Customer Needs

## Scenario-Focused Engineering

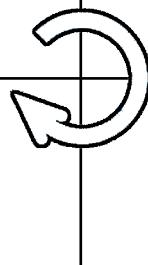
### PROBLEM DOMAIN

#### Identify needs and wants

- Focus on the primary customer
- Probe for the drivers for the needs
- Listen, observe, and empathize

#### Define a problem to solve

- Prioritize stakeholder wants
- Define acceptance criteria
- Write user stories and scenarios



#### Build for early feedback

#### Explore possible solutions

- Brainstorm, suspending disbelief
- Question assumptions
- Assemble potential solutions

### SOLUTION DOMAIN

# Identifying and Addressing Customer Needs

## Scenario-Focused Engineering

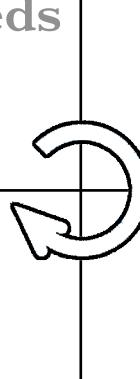
### PROBLEM DOMAIN

#### Identify needs and wants

- Focus on the primary customer
- Probe for the drivers for the needs
- Listen, observe, and empathize

#### Define a problem to solve

- Prioritize stakeholder wants
- Define acceptance criteria
- Write user stories and scenarios



#### Build for early feedback

- Validate the problem definition
- Evaluate options for solutions
- Deliver value with each increment

#### Explore possible solutions

- Brainstorm, suspending disbelief
- Question assumptions
- Assemble potential solutions

### SOLUTION DOMAIN

# Example: Portable Audio Player, 1979

## Precursor of MP3 players and iPods

- **Need: Listen to opera**

- Masaru Ibuki, co-founder of Sony liked to listen to opera on long trans-Pacific flights
- Sony had a high quality audio cassette device, but it was large and expensive

- **Result: Sony Walkman**

- Defined a category
- Sold 300 million units
- By Binarysequence - Own work, CC BY-SA 4.0,  
<https://commons.wikimedia.org/w/index.php?curid=40687158>



# Example: Portable Audio Player, 1979

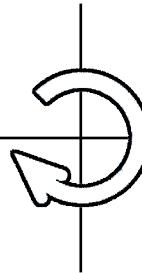
## Sony Walkman: Precursor of MP3 players and iPods

### Identify needs and wants

- Listen to opera on a long flight
- Lightweight for everyday use

### Define a problem to solve

- Lightweight portable music player
- Personal stereo audio



### Build for early feedback

- Prototype adapted an existing device
- Eventual product cost \$200

### Explore possible solutions

- Drop the ability to record
- Lightweight headphones

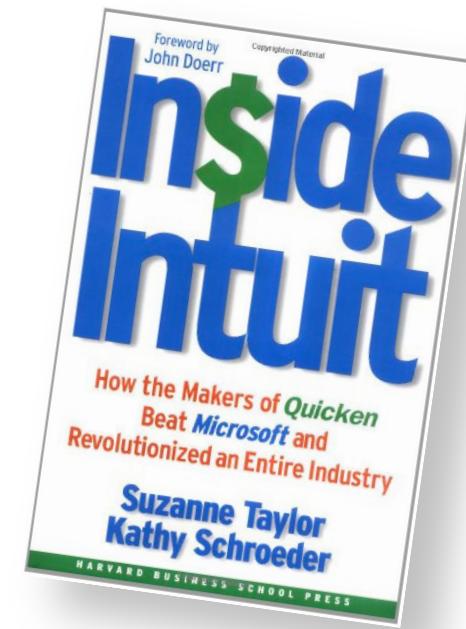


# Levels of Needs

# Intuit: Founded by Scott Cook in 1983

## Intense focus on customers from the start

- MBA in hand, Cook joined Proctor & Gamble (P&G) to learn from “the masters of marketing”
- **String of product failures in the late 1990s**
  - Cook “reembraced a P&G fundamental—that new products should be based on actual customer behaviors, not on what customers said they wanted to do. ... Cook resolved that for future new product development, Intuit should rely on customer actions, not words.”



**Words and actions relate to the top to levels of the hierarchy**

- **Expressed**

- What people say and think — and what they want us to hear

- **Observable**

- What they do or use — can be inferred from actions and behaviors

- **Tacit**

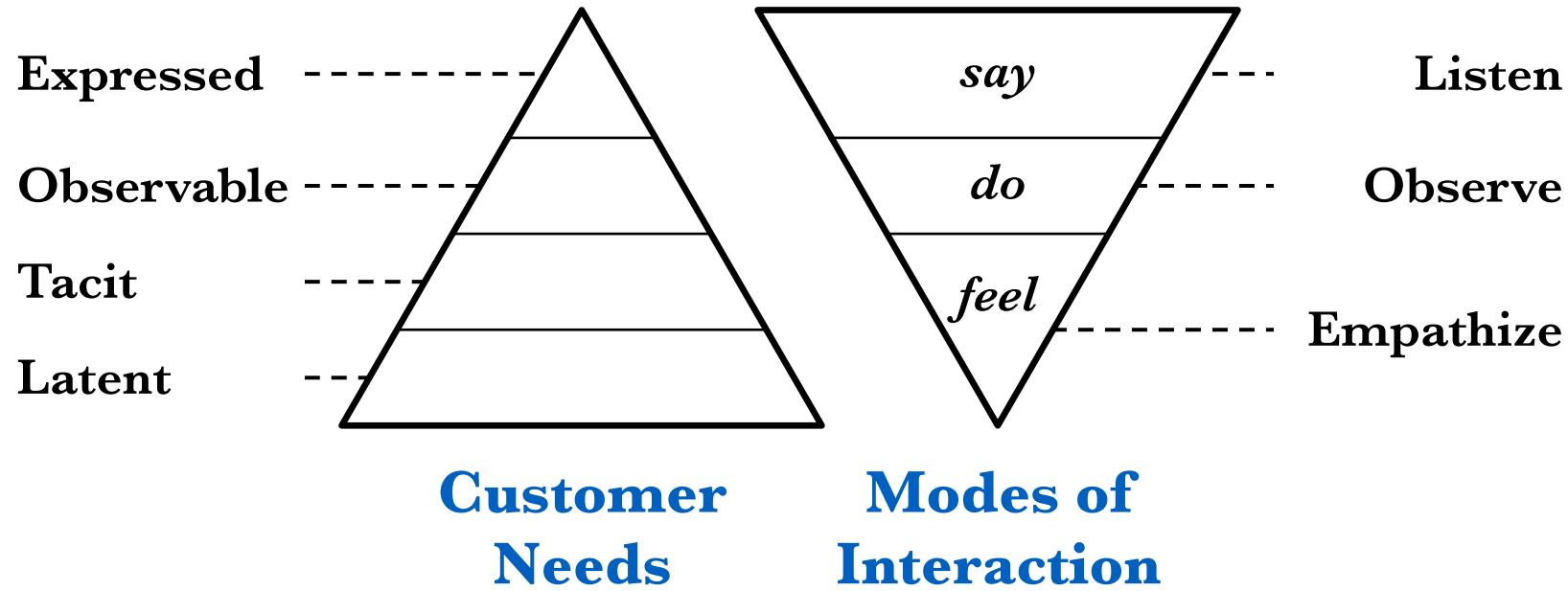
- Conscious need, but they can't find the words

- **Latent**

- “Didn’t know I needed it!”

# Levels of Needs

## Needs and Modes of Interaction for accessing needs



# What Customers Need and Want

## Techniques that get at Expressed and Observable Needs

	<i>Expressed</i>	<i>Observable</i>
<i>Qualitative</i>	Interviews	Observation
<b>Collected Data</b>		
<i>Quantitative</i>	Surveys	Usage Logs

**Customer Needs**

## Interviews & Observations: Case Study

*How Intuit created the QuickBooks app for the iPad*

*Recommended Reading:*  
*Rabinowitz [2014]*

<http://www.aiga.org/inhouse-initiative/intuit-quickbooks-ipad-case-study-app-design/>

---

AIGA | How Intuit Created ... +

www.aiga.org/inhouse-initiative/intuit-quickbooks-ipad-case-study-app-design/ Google

Most Visited Getting Started Latest Headlines Apple Yahoo! Google Maps YouTube Wikipedia News Popular

Log In Become a Member Follow AIGA: [f](#) [t](#) [in](#)

New here? I AM A ... SEARCH

FIND A JOB  
FIND A DESIGNER  
FIND A CHAPTER  
FIND A STUDENT GROUP

AIGA

INSPIRATION ABOUT AIGA WHY DESIGN? EVENTS & COMPETITIONS TOOLS & RESOURCES

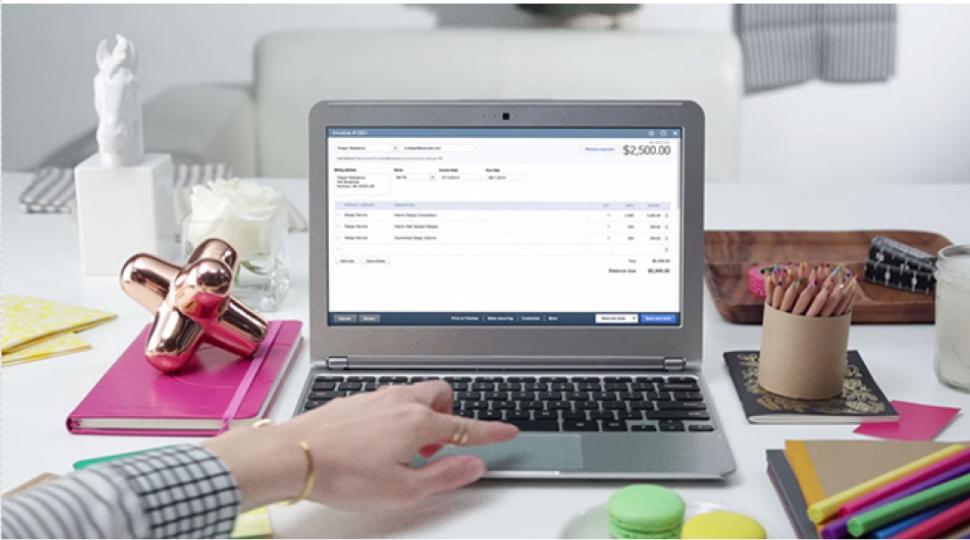
Home INitiative

# HOW INTUIT CREATED THE QUICKBOOKS APP FOR THE IPAD

Case Study by Dorelle Rabinowitz

Filed Under: Tags: *experience design, in-house design, interface design, user experience, INitiative*

Recommend (0)



Images courtesy of Intuit

CLIENT In-house for Quickbooks

AIGA in  
Presented by tcg® THE CREATIVE GROUP

4 Recommendations

## PROFESSIONAL DEVELOPMENT



AIGA Professional Development

AIGA members have opportunities to learn new skills, get advice on pressing career questions, hear insights from industry leaders and learn how to manage more effectively. Find out more about exclusive webinars, workshops, certificate courses and conferences.

Section: *Tools and Resources - Tags: professional development, continuing education*

# The QuickBooks App for iPad

## Target User: service-based sole proprietor

- **Vision: spend time on what you love, not on managing finances**
  - always connected and on the go 50% of the time.
  - since your data is anywhere you are.
- **Pain point: running a business out of the office**
  - frustrated, exhausted, and catching up on evenings and weekends
- **Solution: one tool for all your money needs**
  - synced data across devices to look more professional
  - be more efficient
  - no more losing critical data

# Intuit Process “Design for Delight”

Motto: do a few things really well

- **Step 1: Deep Customer Empathy**

- Understand their needs, goals and current tools using ethnography, interviews, and observation
- With available tools, important customer and job information can easily fall through the cracks

- **Step 2: Go Broad to Go Narrow**

- Agile, focused on building a usable product, story by story
- Discovered the need to share correct data from any access point
- Eager to innovate for mobile, but large compatibility constraints

- **Step 3: Rapid Experiments with Customers**

- From prototyping in the lab to alpha testing, iterated using customer feedback as primary criteria.

# The QuickBooks App for iPad

## Notes

- **Data to inform Agile backlog and design decisions**
  - Closely tracked usage data, app store ratings, and in-product feedback
  - Followed up with customers to understand their experience
  - Measured success using an embedded net promoter survey
- **Design: modular elements reusable across devices**
  - The iPad app is the foundation of the QuickBooks Online ecosystem
  - On the go, it's used in portrait mode; at home, in landscape
  - Gestures on the tablet, typing on the personal computer
- **Challenge: mobile solution without alienating existing users**
  - Serve new users and current users must recognize their data

# The QuickBooks App for iPad

Launched in the App Store on February 14, 2013

- **Results**

- Impressed with emotional response from new and existing users
- Existing users disappointed with limited scope of v1.0

- **Effectiveness**

- Too early to tell when the article was written
- Converting free trial users to paid subscriptions at a 5% rate

# Listening with Understanding

Carl Rogers, in a Harvard Business Review paper



**“there is one main obstacle to communication: people’s tendency to evaluate. Fortunately, I’ve also discovered that if people can learn to listen with understanding, they can ... greatly improve their communication with others.”**

## Capture the person's words verbatim

- **Truly Listening**
  - When the speaker feels you get the content and the emotion
- **Technique**
  - May feel awkward at first
  - Able to repeat back what you heard

# Customer Satisfiers and Dissatisfiers

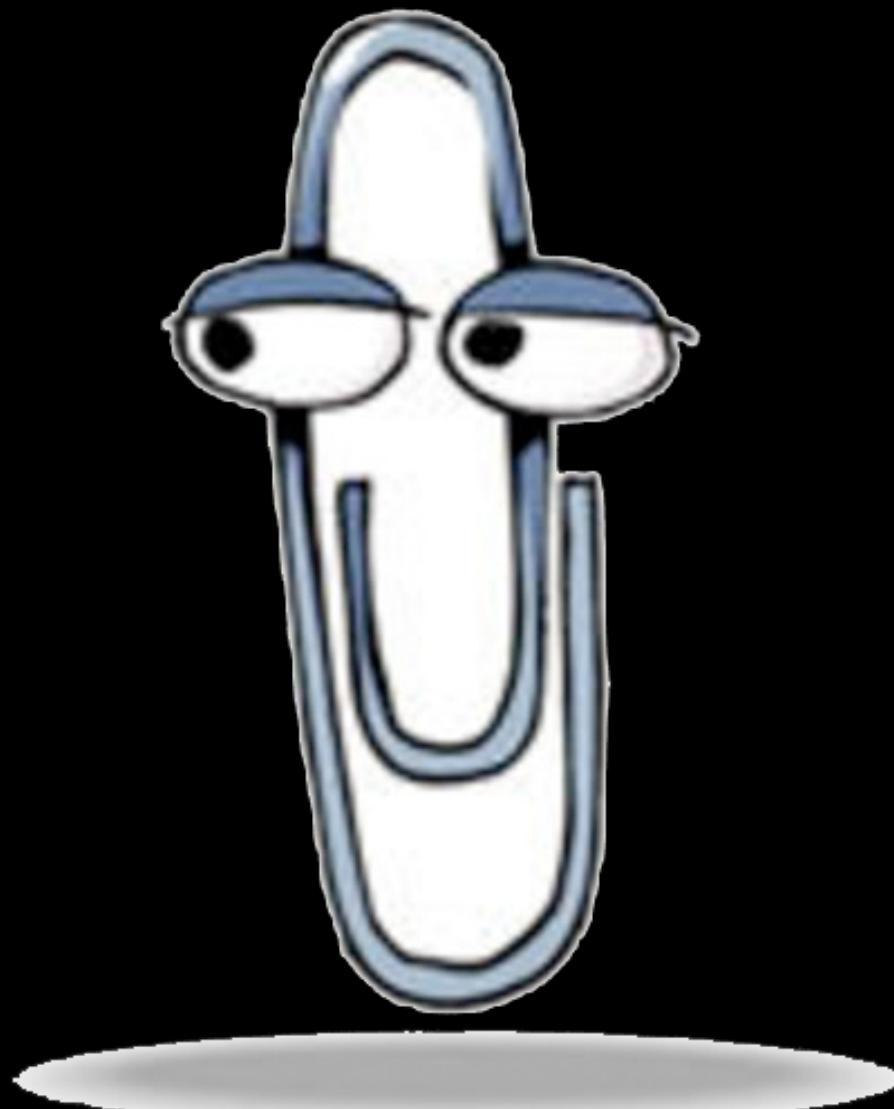
# Clippy, the Microsoft Office Assistant

“Even Early Focus Groups Hated Clippy.” – *Atlantic Headline*



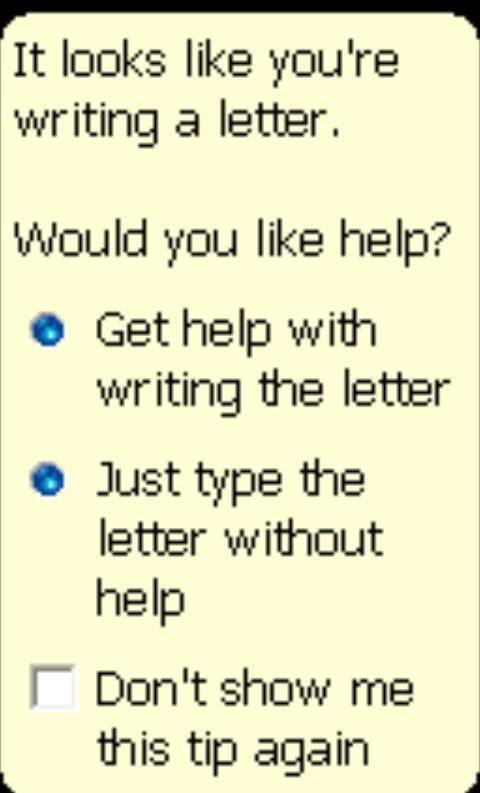
# One of the most unpopular features, ever

Anti-Clippy web site had 22 million hits in the first few months



# Clippy (1996-2007)

Popped up when you least expected it



# Job Satisfiers and Disatisfiers

# Different Factors for Satisfiers & Dissatisfiers

Frederick Herzberg studied job satisfaction in the 1950s

- **Job Satisfiers are tied to work, what people do**
  - job content, achievement on a task, nature of the task
  - recognition for doing, responsibility, professional advancement
- **Job Dissatisfiers to the work environment**
  - supervision, interpersonal relationships,
  - working conditions,
  - salary

# Different Factors for Satisfiers & Dissatisfiers

Frederick Herzberg studied job satisfaction in the 1950s

- **Improving the nature of the work alone**
  - Increases satisfaction, *but does not reduce dissatisfaction*
  - Working conditions and salary did not change
- **Improving working conditions alone**
  - Reduces dissatisfaction *but does not improve satisfaction*
  - The nature of the work did not change

# Customer Satisfiers and Dissatisfiers

Kano et al. applied Herzberg's work to products and features

- **Questionnaires: positive and negative questions**
  - If the product *has* this feature ...
  - If the product *does not have* this feature ...
- **Both forms of questions offered these options ...**
  - I would like it
  - I would expect it
  - I'm neutral
  - I can accept it
  - I would dislike it

## Kano Analysis

- **For simplicity consider 3 options, not 5**
  - I'd be satisfied
  - I'm neutral
  - I'd be dissatisfied

# Kano Analysis

## Classification of Features

*Satisfied*

**Feature is Built**

*Neutral*

*Dissatisfied*

	Attractor	Key
Reverse	Indifferent	Must Have
Reverse	Reverse	

*Satisfied    Neutral    Dissatisfied*

**Feature is Not Built**

# Kano Analysis

## Classification of Features

- **Key**

- The more the better, subject to the schedule and budget

- **Reverse**

- Detract from the product; e.g., Microsoft's Clippy Office Assistant

- **Attractors**

- Can differentiate a product from competitors; e.g., circa 2000, phones with web and email access were unexpected delights

- **Must Have**

- Features that are taken for granted; e.g., performance, reliability, security

Phones with web and email access were a novelty circa 2000

- **Results of Kano Analysis in Japan**

- Young people, like students, were very enthusiastic
- Older people were neutral, they did not see the need

- **Outcome**

- NTT DoCoMo I-mode phones rapidly gained market share



Three Views of an I-Mode Phone

# Kano Analysis: Degree of Sufficiency

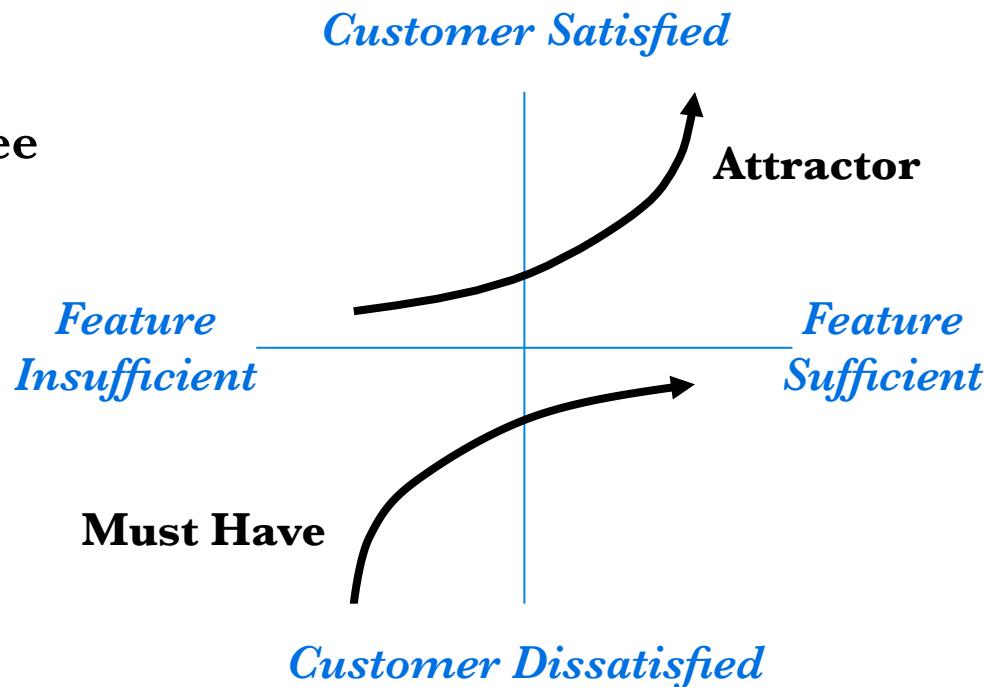
## Conceptual Diagram

- **Attractors**

- Satisfaction rises with the degree of sufficiency of the feature

- **Must Haves**

- Dissatisfaction decreases as the feature is built more fully



# Scenarios

*From Scenario-Focused Engineering*

# Writing a Scenario

## From Scenario-Focused Engineering

- **Scenarios answer four key questions**
  - Who are we building this solution for?
  - What specific need or opportunity does this customer have?
  - Why does the customer care about that?
  - How good does the solution need to be to delight the customer?

# Elements of a Good Scenario

## Acronym SPICIER

- **S: tells a narrative Story**
- **P: includes Personal details**
- **I: is Implementation-free**
- **C: is a Customer story, not a product story**
- **I: reveals deep Insight about customer needs**
- **E: includes Emotions and Environment**
- **R: is based on Research**

# Scenario Template

It's like writing the first and last paragraphs of a 10-page story

- **Title**
  - One-line headline
- **Introduction**
  - Who is the customer? What motivates him or her?
- **Situation**
  - What is the specific real-world context for the need or opportunity?
- **Outcomes**
  - What are the specific outcomes for the customer? Metrics for success?

## Clint Covington, Principal Program Manager Lead, MS Office

- **Problem: Customers weren't using Access database product**
  - It was difficult for them to get it to do what they wanted
- **Solution: create templates to jumpstart projects**
  - Sizable investment in identifying scenarios and creating templates
- **Metric: Downloads from Office Online**
  - Grew from 1.1 million to over 8 million in 18 months
  - “I felt great about our progress.”

# Success Metrics

## Changed approach to usage metrics

- **But do downloads measure usage?**
  - Downloads is a “vanity metric” – team centric, not customer centric
  - After 4 uses, asked “on a scale of 1 to 10, would you recommend this”
- **The Net Promoter Score (NPS) was “terrible”**
  - More demoters than promoters
  - Verbatim comments were brutal, as demoters described frustrations
- **Followed up with phone interviews**
  - Identified 3-4 design flaws and a few pain points
  - Fixed product and made training videos
  - The videos were the most popular help feature
  - NPS jumped 35 points (out of 100)