



CS 160: Lecture 3

Professor John Canny
Spring 2006

Administrivia

-  Please turn in project idea in class today.
-  Project groups will be announced to you by email by Monday when the next assignment will be handed out.

A design success story

- 📄 The Xerox Star was a landmark in computer design, the origin of the "WIMP" interface and the ancestor of the Apple Mac, MS Windows etc.
- 📄 Not only the device, but the **design process** that its developers followed was revolutionary.
- 📄 Today we'll cover this process in detail.



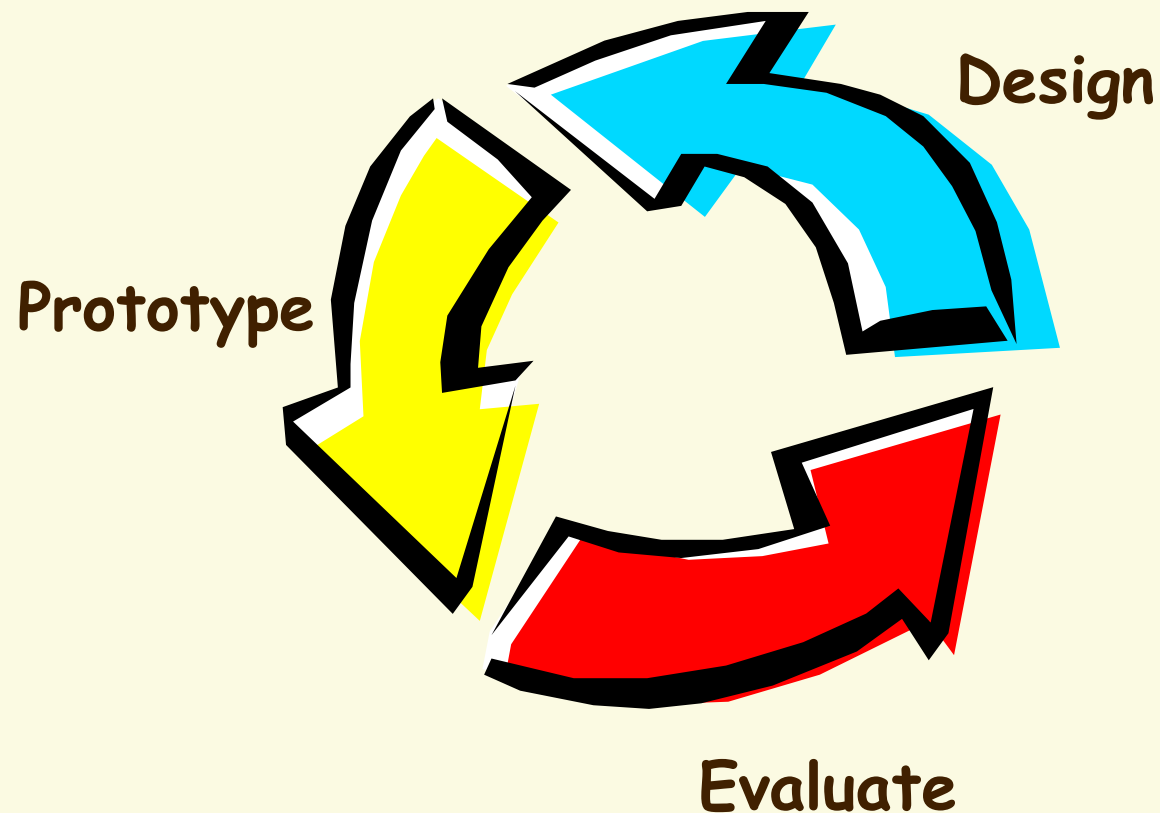
Good design lasts

- ☞ People often criticize modern computers for not going beyond the WIMP interface.
- ☞ But this misses the point: people and their office work practices haven't changed. The Star WIMP interface was an excellent solution, and remains so.
- ☞ UI design is *mostly evolutionary*, not revolutionary.
- ☞ Of course when you go outside the office and desktop (i.e. to smart phones) all best are off...



Human-Centered Design

 In a nutshell it's:



The Art of UI Design

Of course, there's more to it than that...

A soufflé is eggs, butter, milk & flour, but the difference between soaring and sinking is in the execution.

Same with UI design.



The Human-Centered Design Process

- Who is going to use the system?
- What are their characteristics, goals and desires?
- Choose representative tasks and analyze them
- Rough out a design (plagiarize as needed)
- Rethink the design - does it best address a need?
- Create a prototype
- Test it with users
- Iterate
- Build a production version (and ship it!)
- Track use
- Evolve the design

Human-Centered Design

- 📄 Who is going to use the system?
- 📄 What are their characteristics, goals and desires?
- 📄 Choose representative tasks and analyze them
- 📄 Rough out a design (plagiarize as needed)
- 📄 Rethink the design - does it best address a need?
- 📄 Create a prototype
- 📄 Test it with users
- 📄 Iterate
- 📄 Build a production version (and ship it!)
- 📄 Track use
- 📄 Evolve the design

Who is the user?

- 📄 It's one of the most important design questions you will ask - everything else follows from that.
- 📄 Remember that most users are different from you, in ways you may not realize:
 - * Test, don't guess
- 📄 Remember that the user is *paying* for the product, so give them something worthwhile.

Egoless design

- ❏ Cooper Interaction design emphasizes “egoless design”:
- ❏ You design for a customer, not yourself.
- ❏ Although good UI designs are visually pleasing, they are not works of art.
- ❏ Design is about realizing the customers goals and needs, not the designer's.



Human-Centered Design

- Who is going to use the system?
- What are their characteristics, goals, desires?**
- Choose representative tasks and analyze them
- Rough out a design (plagiarize as needed)
- Rethink the design
- Create a prototype
- Test it with users
- Iterate
- Build a production version (and ship it!)
- Track use
- Evolve the design

User characteristics: Personae

- Personae are *concrete* representations of the user group as individuals.
- Things to strive for in a good persona:
 - * Attributes (age, gender, occupation)
 - * Likes, dislikes
 - * Values and desires (or life's goals)
- A good persona is generative (of ideas) - like a good fictional character.

Personae

- 📄 You know it's a good persona if the design team is passionate about what the persona would or would not like.



User Characteristics

- Another way to understand the lives of users is to document them with informal photographs (IDEO and many other design firms do this).
- Things to look for:



Rituals



Sacred places

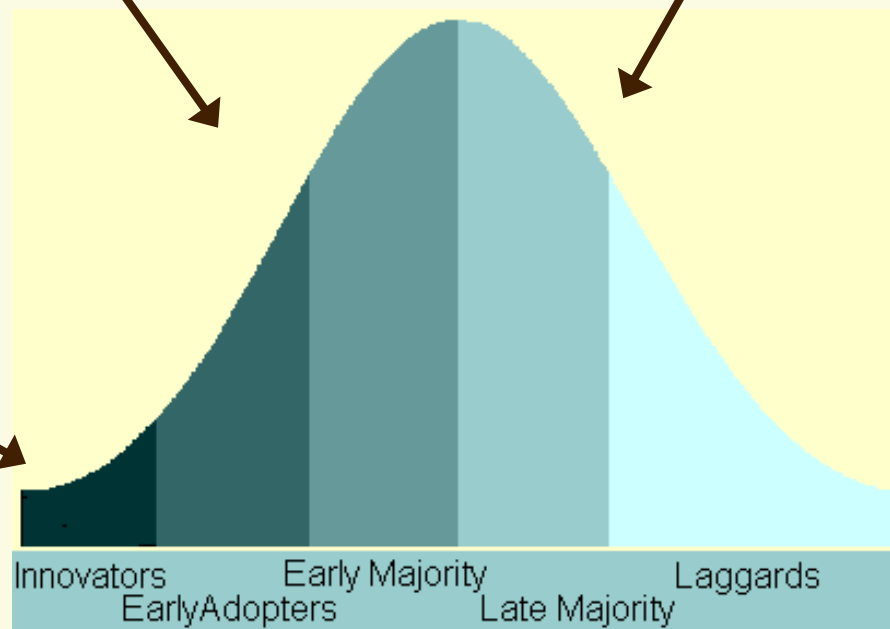
Relationships



How users are not like you

People here care a lot about features

People here want reliability, convenience
"no fuss or bother"



Time from product appearance to adoption →


Human-Centered Design

- Who is going to use the system?
- What are their characteristics, goals and desires?
- Choose representative tasks and analyze them**
- Rough out a design (plagiarize as needed)
- Rethink the design
- Create a prototype
- Test it with users
- Iterate
- Build a production version (and ship it!)
- Track use
- Evolve the design



Tasks

 Tasks are goal-directed behaviors like:

- * Finding a table in a restaurant
- * Composing an email message
 - + Searching an address book
- * Performing a web search
- * Getting money from an ATM...

 Tasks are the building blocks for user behavior, and we can study them with or without a design solution...

Task Analysis and Contextual Inquiry


-  You normally discover tasks during a structured observation/interview process called "Contextual Inquiry" (next time).
-  The "analysis" in task analysis provides more information to guide you in design. There are several approaches, which we'll talk about later.

Human-Centered Design

- Who is going to use the system?
- What are their characteristics, goals and desires?
- Choose representative tasks and analyze them
- Rough out a design (plagiarize as needed)**
- Rethink the design
- Create a prototype
- Test it with users
- Iterate
- Build a production version (and ship it!)
- Track use
- Evolve the design

Rough out the design

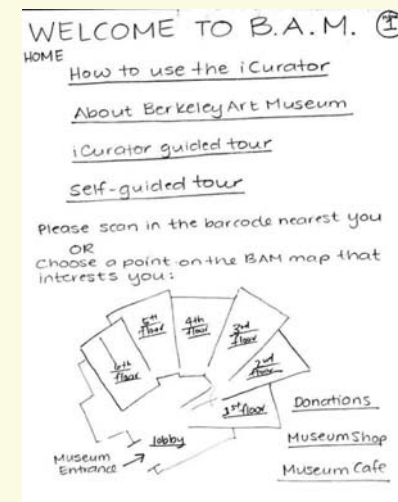
 *Sketch*, if sketching works for you.

 Collage - use actual clip art, cardboard, fabric etc.

 Use plastic clay...

Rough out the design

- 📄 Rough out your ideas in a *shared space* to negotiate them with other designers.
- 📄 Focus on high-level issues (what features are needed and why).
- 📄 Keep the task analysis and personae in mind when discussing features.



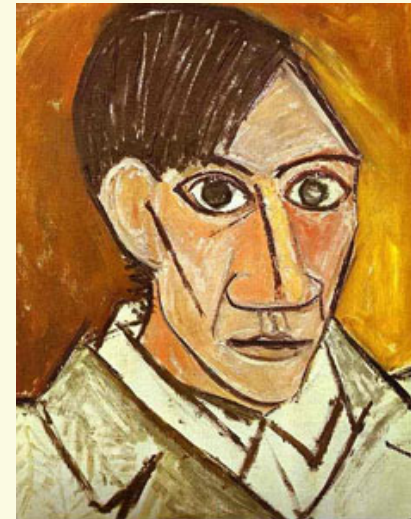
Plagiarism

Quote from a famous artist:

"Good artists borrow (from other artists),
but great artists steal!"

- Pablo Picasso

Works for UI design too!



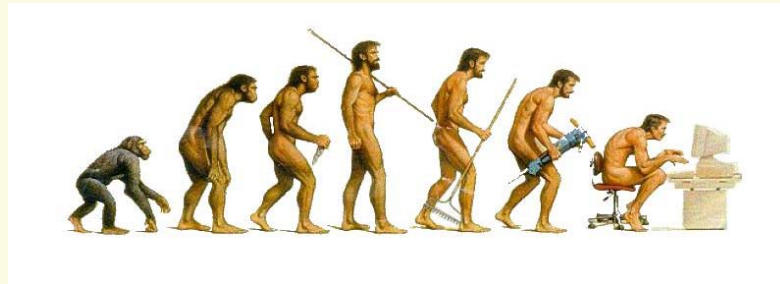
Innovation can be harmful!

Where *not* to innovate:

- ☞ Brake pedal on left, accelerator on right
- ☞ Steering wheel CW → right, CCW → left
- ☞ Analog clocks that go CCW
- ☞ Light switch up = on, down = off
- ☞ Keyboard layout: QWERTY, Dvorak
- ☞ Directory/file icons
- ☞ Typical contents of file/edit/view menus
- ☞ What scroll bars look like
- ☞ Active areas of windows for move and resize

Moral

Good UI design is an *evolutionary* process.



Ergo, its better to start from an advanced "species" (UI design) than a primitive one, even if its someone else's (Star→Mac→PC)



1/24/2006

Human-Centered Design

- Who is going to use the system?
- What are their characteristics, goals and desires?
- Choose representative tasks and analyze them
- Rough out a design (plagiarize as needed)
- Rethink the design**
- Create a prototype
- Test it with users
- Iterate
- Build a production version (and ship it!)
- Track use
- Evolve the design

Think about the design

- 📄 Don't get stuck on your original idea.
- 📄 Now that you have users, tasks and needs, *explore* some completely different solutions.
- 📄 **Rethink** your assumptions:
 - * Does this have to run on a PDA?
 - * Does it really require continuous net access?
 - * Will users really adopt this product (even if you like it)?
- 📄 Force yourself to sketch some designs that are *very different*.

Think about the design

- ☞ This is the phase to do *engineering analysis* if appropriate.
- ☞ For usability, automated systems are not very powerful, and there are few (GOMS, EPIC).
- ☞ Heuristic evaluation is a systematic method for human evaluation of an interface.
- ☞ Another method is "cognitive walkthroughs" explained later in Lewis and Rieman.
- ☞ More elaborate techniques include:
 - * scenario development
 - * role-playing

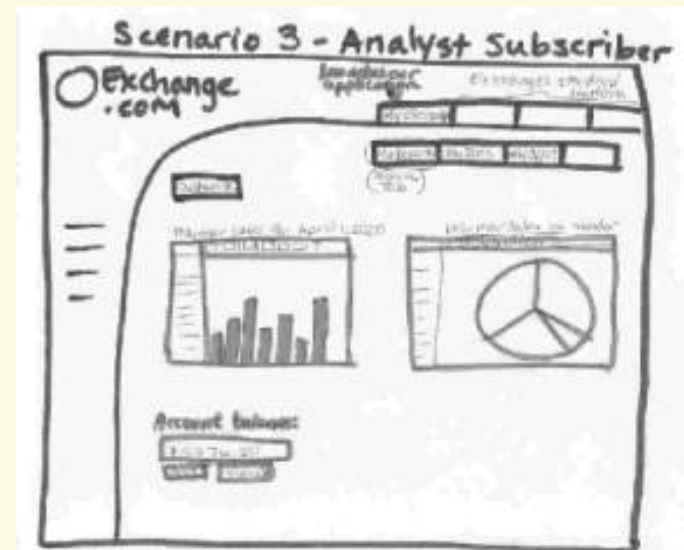
Break

Human-Centered Design

- Who is going to use the system?
- What are their characteristics, goals and desires?
- Choose representative tasks and analyze them
- Rough out a design (plagiarize as needed)
- Rethink the design
- Create a prototype**
- Test it with users
- Iterate
- Build a production version (and ship it!)
- Track use
- Evolve the design

Prototype the design

- ❏ Prototypes let you simulate a lot of detail of an interface.
- ❏ Informal (paper or digital sketch) interfaces keep designs more fluid - more changes happen
- ❏ They allow presentations to the user
- ❏ The "Wizard of Oz" method has the designer simulate the *behavior* as well as the *appearance* of the system



Wizard of Oz Technique

Faking the interaction.

- * Comes from the film "The Wizard of Oz"
- * The wizard was actually a "man behind the curtain"

Long tradition in computer industry

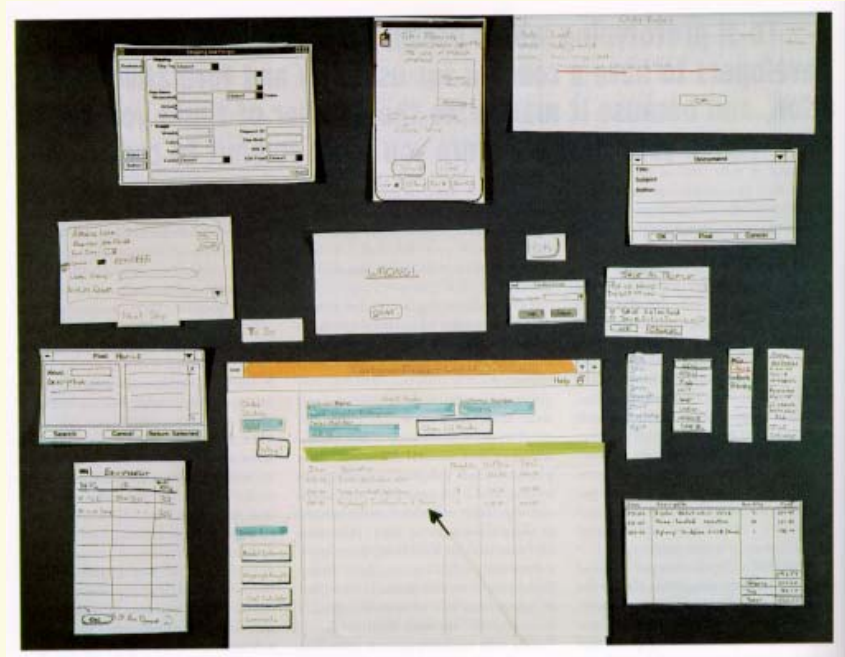
- * Prototype of a PC w/ a VAX behind the curtain!

Much more important for hard to implement features

- * Speech, vision & handwriting recognition

Wizard of Oz Technique

- One designer works as “the system” and moves around paper menus and dialogs, in response to user actions.
- The other designers observe and note problems.



Human-Centered Design

- Who is going to use the system?
- What are their characteristics, goals and desires?
- Choose representative tasks and analyze them
- Rough out a design (plagiarize as needed)
- Rethink the design
- Create a prototype
- Test it with users**
- Iterate
- Build a production version (and ship it!)
- Track use
- Evolve the design

Test the prototype

 User testing is one of the critical stages in design.

 Goal is to:

- * Discover *problems* as early as possible
- * Discover *other needs or features* from the users, i.e. needs analysis is not a one-shot deal

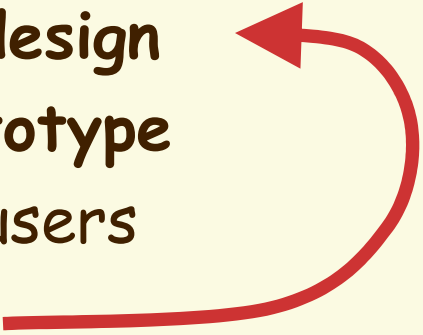
 What testing isn't for:

- * Proving that all your design decisions were right

Test the prototype

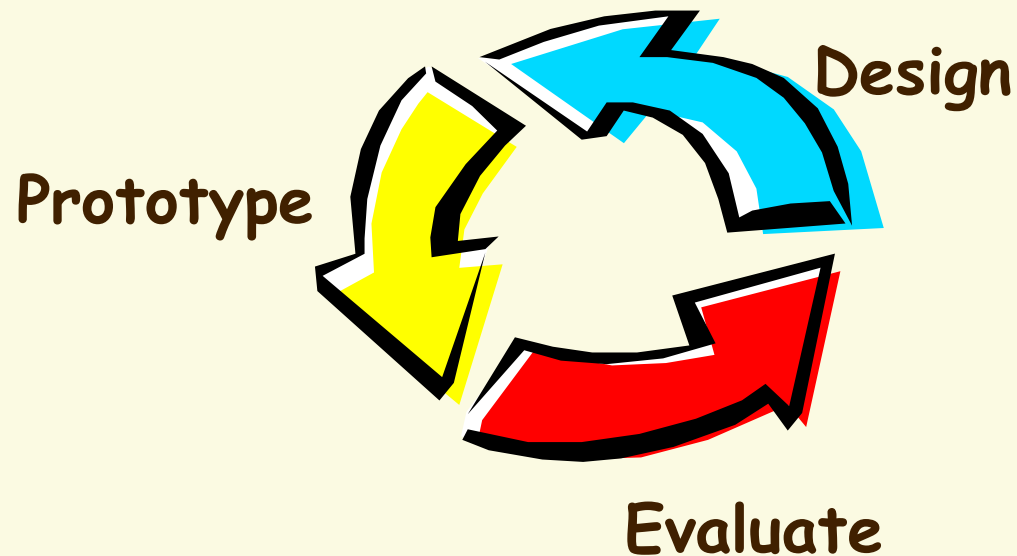
- ☞ User is asked to “think aloud” while performing the task.
- ☞ Testers observe user and makes notes about user actions (especially any problems) and what the user says.
- ☞ Testers prompt the user to explain something they said or did.
- ☞ Testers don't help users to do the task.
- ☞ Testers don't let users take shortcuts.

Human-Centered Design

- Who is going to use the system?
 - What are their characteristics, goals and desires?
 - Choose representative tasks and analyze them
 - Rough out a design (plagiarize as needed)
 - Rethink the design
 - Create a prototype
 - Test it with users
 - Iterate**
 - Build a production version (and ship it!)
 - Track use
 - Evolve the design
- 

Iterate!

- Testing will expose problems with various severity
- You can then attack those problems in order of severity - and work on features in order of value
- Beware of interactions between design elements - fixing one may break another



Human-Centered Design

- Who is going to use the system?
- What are their characteristics, goals and desires?
- Choose representative tasks and analyze them
- Rough out a design (plagiarize as needed)
- Rethink the design
- Create a prototype
- Test it with users
- Iterate
- Build a production version (and ship it!)**
- Track use
- Evolve the design

Build It!

- ☞ Some prototyping tools (IDEs or UIMS) allow you to move prototype code to production code - most do not, and this method is not recommended.
- ☞ When you move from prototype to production code, remember that commitments you make will be hard to undo - check everything first!
- ☞ Remember that UI code is at least half of all code for interactive systems. Allow enough time for development.

Build and Release

- 📄 Early releases (alpha and beta) allow yet more testing. Make sure you have good mechanisms in place to get developer/early adopter *feedback*.
- 📄 The time from “fully-working” code to “industrial-strength” code can be 6 months or more.
 - * Program defensively, anticipate and deal with errors inside *and* outside your system.
 - * Test at appropriate scale
 - * Introduce stress on the system (other apps, lots of users, unusual command sequences, undos etc.).
 - * Stress on testers would be a good idea - but hard to implement!

Human-Centered Design

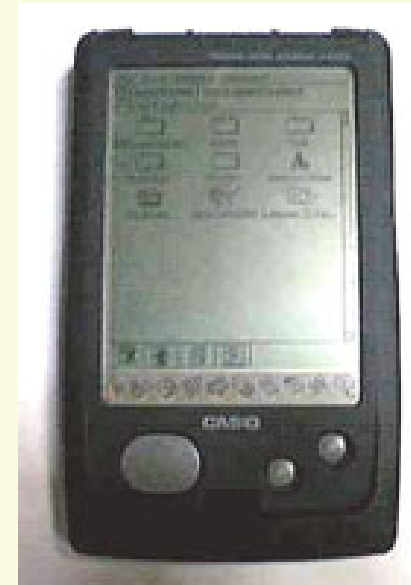
- Who is going to use the system?
- What are their characteristics, goals and desires?
- Choose representative tasks and analyze them
- Rough out a design (plagiarize as needed)
- Rethink the design
- Create a prototype
- Test it with users
- Iterate
- Build a production version (and ship it!)
- Track use**
- Evolve the design

Another bit of history

Q: What was the Zoomer?

A: The Palm Pilot's parent.

It failed in the marketplace.



Palm Pilot

- ☞ Intensive studies of Zoomer users began in 1994.
- ☞ Decided the PDA should be a paper replacement, not a PC replacement.
- ☞ Switched to graffiti.
- ☞ Shrunk to pocket size.
- ☞ Unveiled the Palm Pilot in 1994.



Tracking Use

- 📄 Interview real users, log their complaints and praise.
- 📄 Talk to maintenance and support staff.
- 📄 Put in logging and bug reporting software.
 - * Be very careful about privacy.

Toolbelt Design + Technology Probes

- ☞ There is a trend in design to build suites of interoperable tools that the customer can adapt (something like MS office + VBasic).
- ☞ Toolbelt design allows user evolution of the basic features of the design.
- ☞ New generations of the system can move user ideas into the core system.

- ☞ In other words, users can become your best designers.

Human-Centered Design

- Who is going to use the system?
- What are their characteristics, goals and desires?
- Choose representative tasks and analyze them
- Rough out a design (plagiarize as needed)
- Rethink the design
- Create a prototype
- Test it with users
- Iterate
- Build a production version (and ship it!)
- Track use
- Evolve the design**

Evolve the Design

- ❏ Real user feedback should help you figure out what needs to change.
- ❏ Its often a challenge to accept what you find, and act on it:
- ❏ The company's strategy and assumptions may have been wrong (Zoomer again).



Evolve the Design

📄 But remember that many truly successful products were 2nd or 3rd attempts:


- * Palm Pilot
- * Apple Mac
- * Windows 3.1

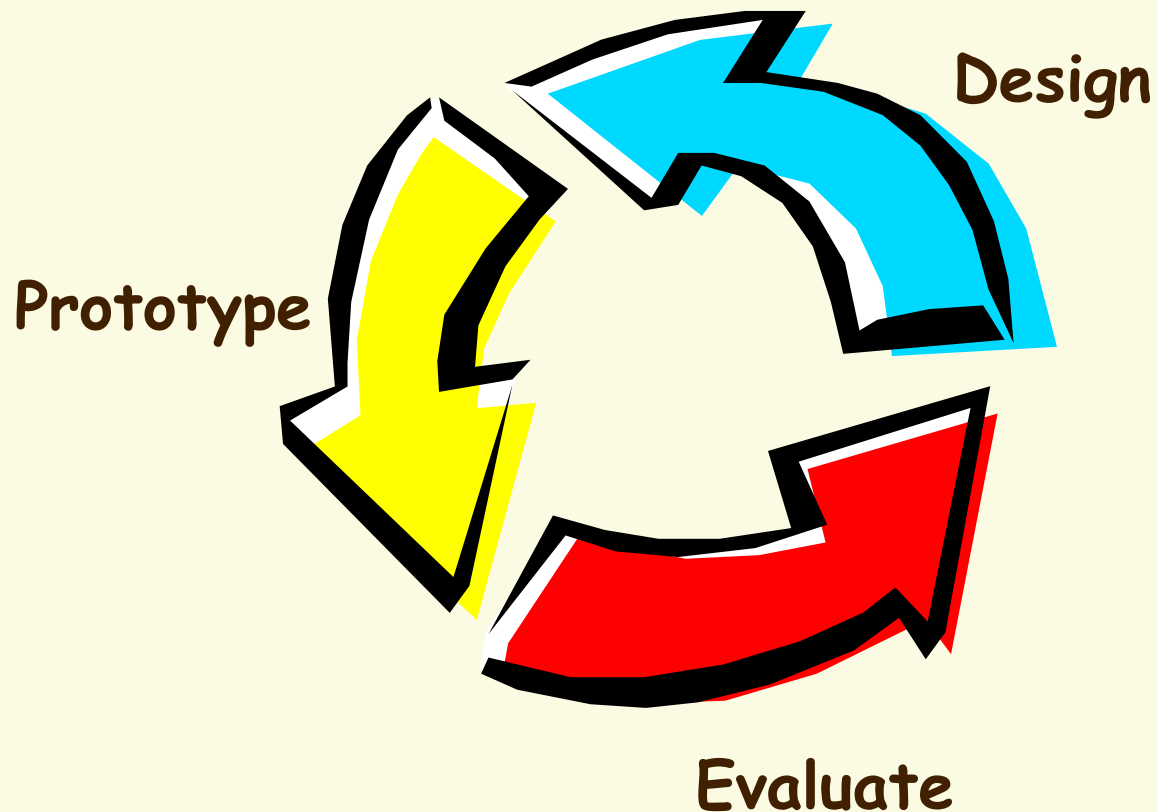


The Recipe again

- Who is going to use the system?
- What are their characteristics, goals and desires?
- Choose representative tasks and analyze them
- Rough out a design (plagiarize as needed)
- Rethink the design - does it best address a need?
- Create a prototype
- Test it with users
- Iterate
- Build a production version (and ship it!)
- Track use
- Evolve the design

Human-Centered Design

 In a nutshell:



Summary: Human-Centered Design

- 📄 This iterative design process has been “best practice” since the Xerox Star.
- 📄 Executing the entire process gives a very good chance of success.
- 📄 Skipping steps (e.g. tracking use) can lead to missed opportunities.