Finding out what users *really* need: “trial and error” and “sticky information”

Professor Eric von Hippel

MIT Sloan School of Management

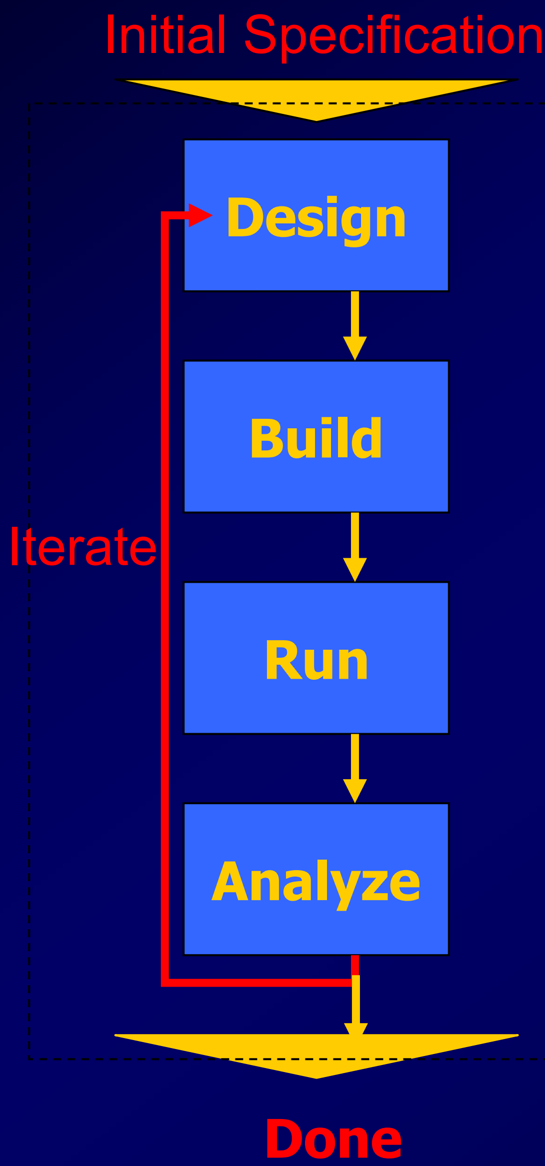
MITSloan

Innovation is:

1. A problem-solving process based upon directed trial-and-error

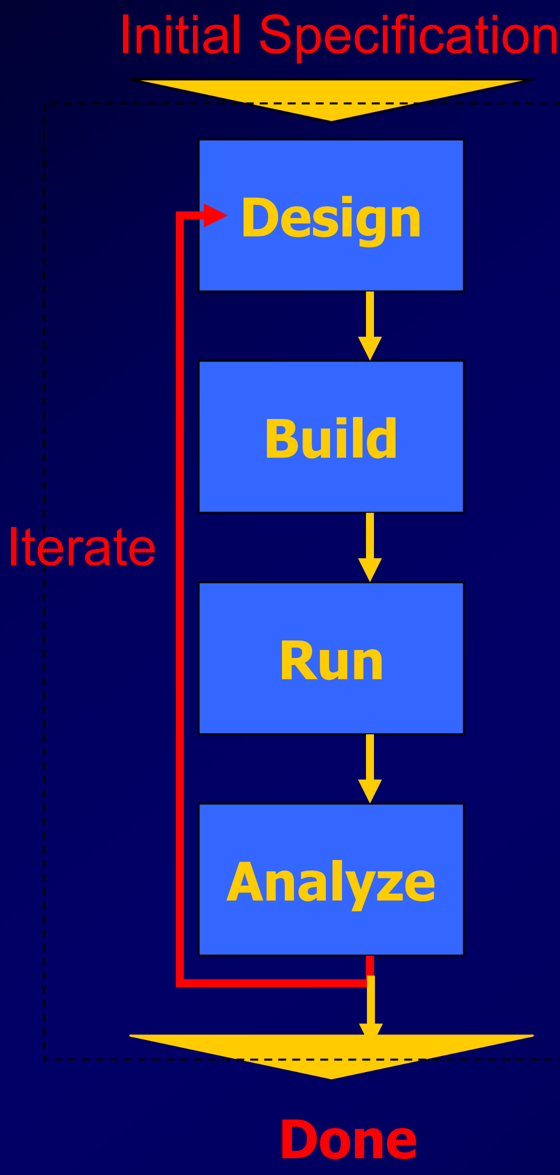
2. Carried out at the site of "sticky information"

Trial and error is THE fundamental problem-solving process



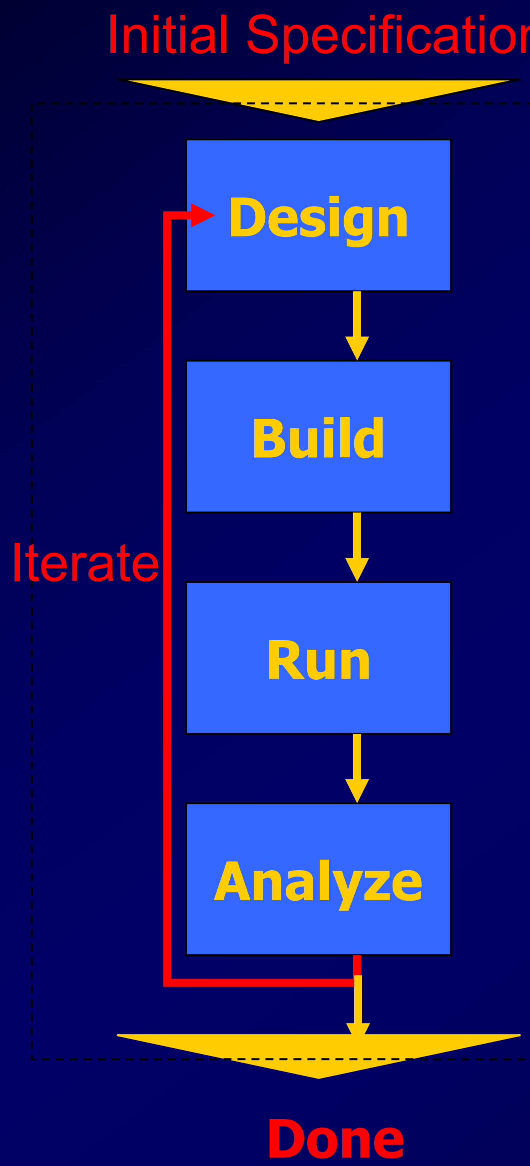
- Start by designing what you THINK will work (*directed* T&E)
- Develop a thought experiment, a simulation or a prototype
- Test your design in a real or simulated use environment
- Analyze what happened, revise your design, try again till satisfied

Directed trial and error is THE fundamental problem-solving process



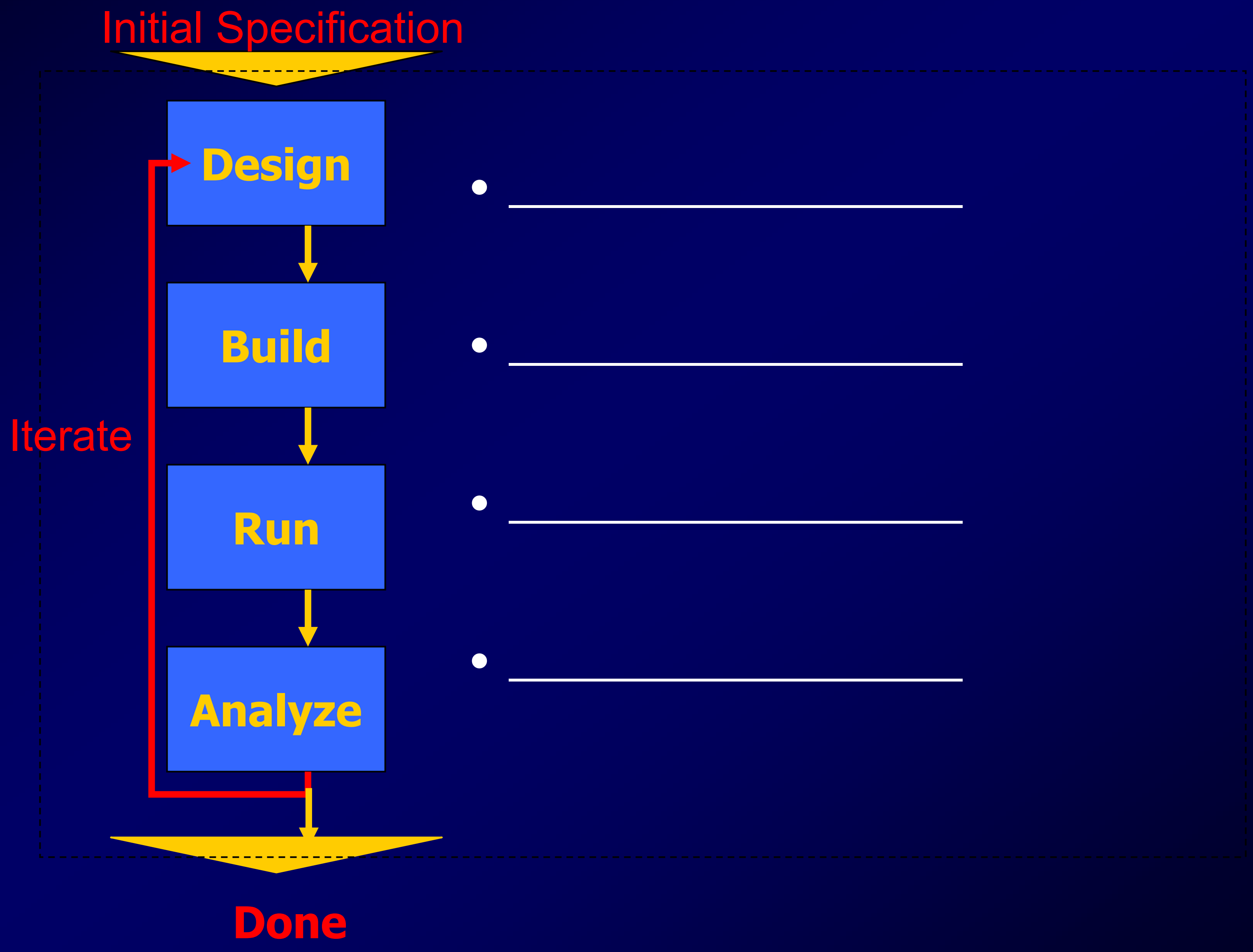
- I will outline a paper about X
- I will write a draft
- I will see if my room-mate likes it
- I will learn from the comments and revise the paper

Directed trial and error is THE fundamental problem-solving process



- I will design a rocket
- I will build it
- I will flight test it
- It crashed: I will analyze what went wrong so I can redesign

TRY YOUR OWN EXAMPLE



Innovators / problem-solvers require information about both a need and a solution approach

- **Need** information is usually found at user sites.
- **Solution** information is usually found at manufacturer sites.

But bringing full and accurate need and solution information together is often *VERY* difficult

Why? Because information is often "sticky" - very costly to transfer from place to place

- Information needed by developers may be ***tacit***
  - Can you <u>tell</u> your child how to ride a bike?
- A ***lot*** of information is often needed by developers
  - "You didn't tell me you were going to use the product *that* way!"

A result: user and manufacturer innovations differ in *kind*

Users tend to develop **Functionally Novel** innovations:

- The first sports-nutrition bar
- The first scientific instrument of a new type

Manufacturers tend to develop **Dimension of Merit Improvements:**

- A better-tasting sports-nutrition bar
- Improvements to an existing type of scientific instrument

Example of the impact of sticky information on the locus of innovation:

Fifty percent of all prescriptions written in the U.S. are written for "off-label" uses of prescription drugs

- **New prescription drugs are generally developed in the labs of pharmaceutical firms** – sites where much specialized information about drug development has been build up over the years.
- **Off-label applications are generally found by patients and physicians.** They apply the drugs many times under widely varying field conditions – and discover unanticipated positive (or negative) effects thereby. ("Doctor: this blood pressure medication you gave me is causing my hair to regrow!")

Studies show this effect clearly

Sample of 24 inventory control system innovations by 7-11 Japan and NEC

(For this diagram, see:

Ogawa, Susumu. *Does sticky information affect the locus of innovation? Evidence from the Japanese convenience-store industry*. *Research Policy* 26, 7-8, April 1998. Figure 1, p. 78.)

Product or service design tends to move to the site of the crucial sticky information

Manufacturer-Based Design (DOM products)

Manufacturer design tasks

- Have solution information
- Acquire *need info* from user
- Design product

← **User design task**: Need Info Source

User-Based Design (Functionally novel products)

Manufacturer design task: Solution Info Source →

User design tasks

- Have need information
- Acquire solution information
- Design product

Product or service design tends to move to the site of the crucial sticky information : EXAMPLE

Manufacturer-Based Design (DOM products)

Manufacturer design tasks — **User design task**

Video Game Engine ← Need Info Source

User-Based Design (Functionally novel products)

Manufacturer design task — **User design tasks**

Engine Interfaces → Video game mods

Product or service design tends to move to the site of the crucial sticky information : EXAMPLE

Manufacturer-Based Design (DOM products)

Manufacturer design tasks | **User design task**

DOM: Smoother-running skateboard wheels ← Need Info Source

User-Based Design (Functionally novel products)

Manufacturer design task | **User design tasks**

Tools and materials → The Skateboard

Sticky information and the (failed) "waterfall" innovation process

Manufacturer to user: "Specify what you want –
We will negotiate a contract to deliver exactly that!

The manufacturer labors to meet the specification. Delivers the completed product!

User to manufacturer: "Now that I try it out,
I find that this is NOT what I want!"
Manufacturer to user: "It IS what you contracted for!"

The problem: Due to sticky information the agreed-on specification *was not* complete and accurate

“Rapid prototyping” innovation processes as a solution

MFR ACTIVITY	**USER ACTIVITY**
	User provides initial specification
Manufacturer develops prototype	
	User evaluates and improves /changes specifications
Manufacturer incorporates changes	
	User iterates until satisfied

Learning by doing

Users have the advantage of problem-solving in their *own use environments* as they “do” a desired activity – they are “learning by doing.”

Examples:

- Airlines learn how to maintain their planes more efficiently as they do that work – they “go down the learning curve.”

- Skateboarders learn to do new things on their boards *as they skate.* They don’t go into the lab and do R&D – they are learning by doing

Learning by doing can be incredibly cheap for users *within* their own narrow niche of “doing”

(Photograph of windsurfers.)

Photograph courtesy of Lisa A. Devlin. Used with permission, http://www.windwardskies.com/

Learning by doing can be incredibly cheap for users *within* their own narrow niche of "doing"

Mountain bike innovation

- "When I do tricks that require me to take my feet off the bike pedals in mid-air, the pedals often spin, making it hard to put my feet back onto them accurately before landing."

  I added a foam ring around the pedal axle near the crank. This adds friction, and prevents the pedals from free-spinning when my feet are off."

Development of instant messaging at MIT

In 1987 MIT Lab for Computer Science had thousands of Athena workstations online and difficulties diffusing system administration information rapidly.

On-site programmers programmed the "Zephyr" instant message system. MIT students quickly begin to use Zephyr for general instant messaging.

Learning by doing can be incredibly cheap for users *within* their narrow niche of "doing"

Example: "I'm a mountain biker and a human movement scientist working in ergonomics and biomechanics. I used my medical experience to improve my mountain bike.

(Consider the cost if that person had not been a biker and had to learn the sport to innovate – or did not have medical training and tools "in stock.")

Exercise: think of your own instances of learning by doing

Example:

I worked out the quickest route from home to school.

1. Think about the process you used to determine the quickest route.
2. Notice the low incremental cost to you. For example, since your trip from home to school was a trip you took "anyway," the cost of each experiment was minimal. (It would cost much more to hire someone to do this experiment who did not have to take that trip "anyway.")

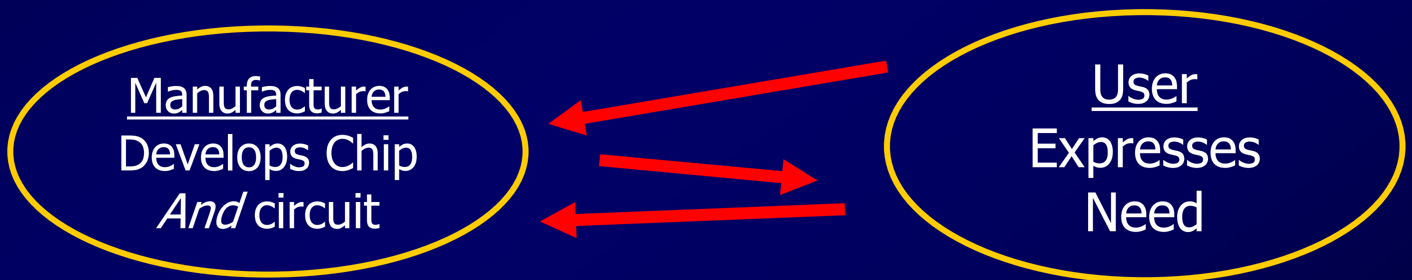
How can you reduce iteration?

Repeated shifts of problem-solving sites during product development can be very costly – what can you do to reduce the need for it?

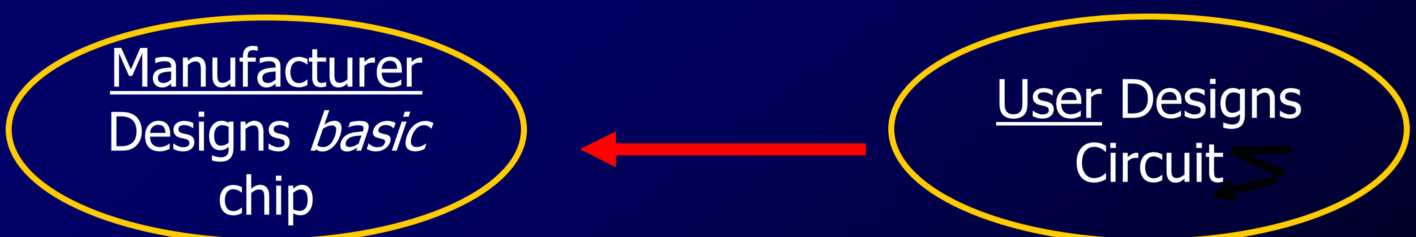
3. Reframe the initial product or service design problem which draws on two sticky information sites into sub-problems – each of which draws on sticky information location at only one site

Example: Custom Integrated Circuit Design

"Full custom" chip development procedure

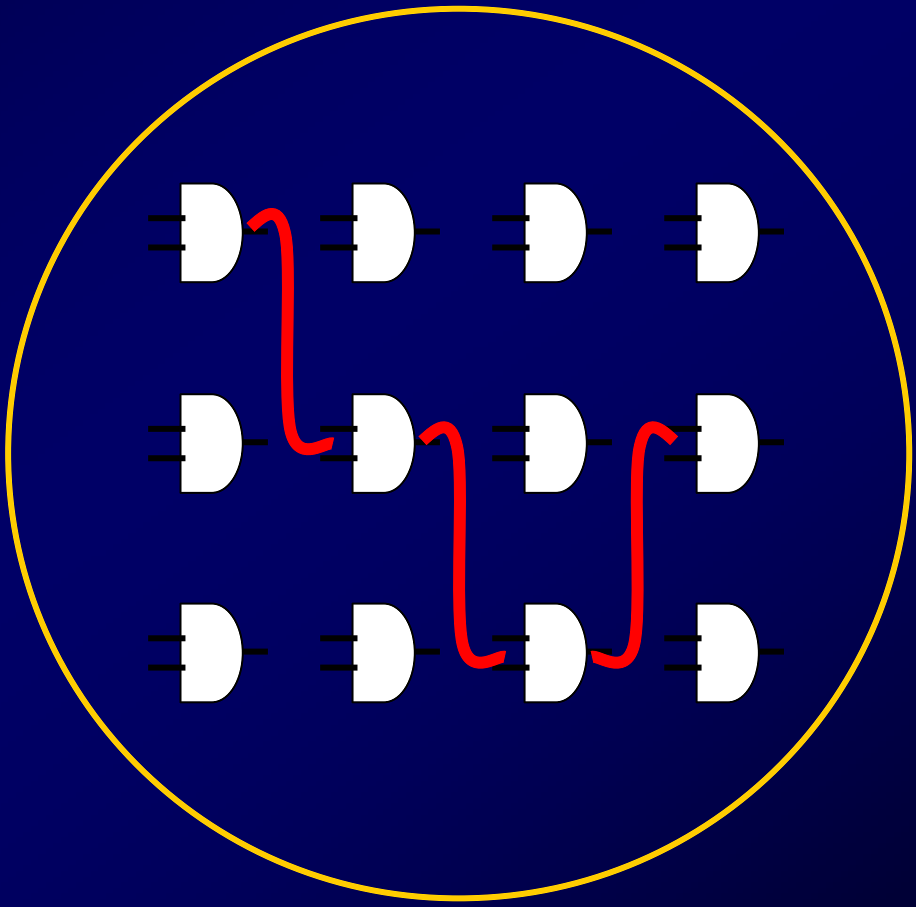


ASIC custom chip development procedure



Example

“Full-custom” IC Design vs “Gate Array IC Designs”



Shifting innovation to users

Economics of sticky information tends to shift the locus of problem-solving to users. For custom design projects, manufacturer information is standard from project to project but user need differs

Example:

Each ASIC design may require the same information from the ASIC manufacturer, but unique information from the ASIC user.