How to engage designers to ensure great products

Hugh Dubberly
Dubberly Design Office

**Great products fulfill at least three criteria.
They are effective, efficient, and engaging.**

— **Effective**: Meet important needs, in robust and reliable ways;
    they are low-maintenance, safe, **solid.**

— **Efficient**: Easy to learn and use—clear, consistent, and coherent;
    they unfold, adapt, and **accommodate.**

— **Engaging**: Considerate and inspiring—more than merely satisfying;
    they draw us in, enable "flow", and **delight.**

See Vitruvius, *De Architectura*, (circa 15 BCE), the earliest surviving treatise on architecture and design,
and ISO 9241 (circa 2005), which defines standards for ergonomics and HCI.

Great products come from iteration; they evolve—or more precisely, they co-evolve with changing internal and external environments.

That is to say: **all products are the result of conversations.**
A new product teaches its environment. The environment also teaches the product.

Some conversations are
with the product team.

Those conversations
cannot be controlled,
but they can be supported.
That is, managers,
engineers, and designers
can create conditions
in which the right
conversations flourish.

Product Managers

Engineers

Designers

Product

Users

Critics

Competitors
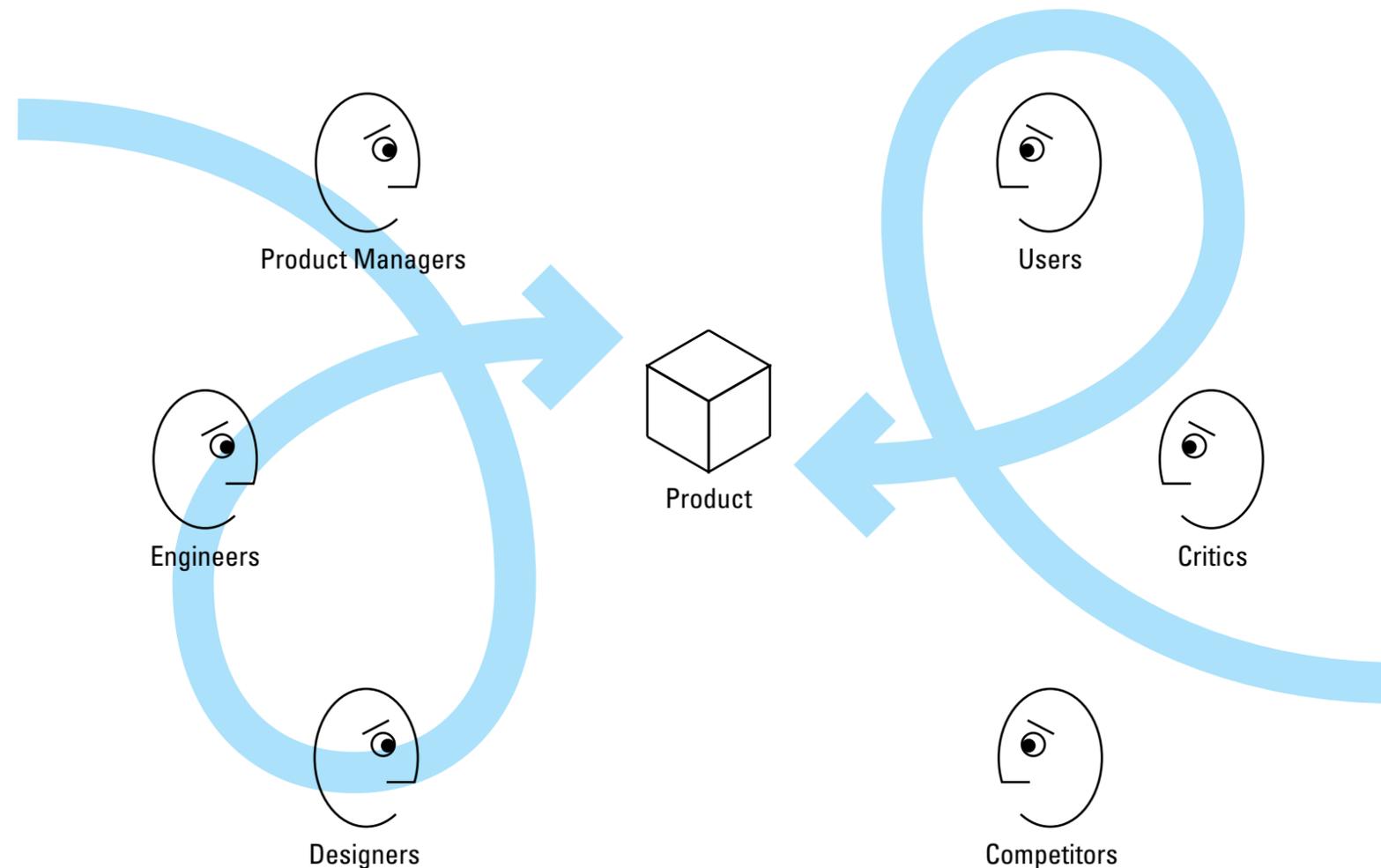
Some conversations are
with the marketplace.

Those conversations
are difficult to manage,
though smart, connected
products are creating new
possibilities.

Steve Jobs knew a thing or two about great products—and about design.

"In most people's vocabularies,
design means veneer. It's interior decorating.
It's the fabric of the curtains and the sofa.
But to me, nothing could be further
from the meaning of design.
Design is the fundamental soul
of a man-made creation
that ends up expressing itself in
successive outer layers of the product or service."

— Steve Jobs, *Fortune*, January 24, 2000

The partnership between Steve Jobs and Jony Ive is famous.
What's rarely discussed is what it means, what we can learn from it.

It was an on-going conversation that built a relationship and trust.

"*We had lunch together pretty much every day.*

He would spend many afternoons a week in the design studio, and we became very close friends."

—Jony Ive, *Financial Times*, March 13, 2015

The Jobs-Ives conversation is not unique; pretty much everywhere that you find really great design, you find such conversations.

Steve Jobs + Jonathan Ive = Apple

Ed Catmull + John Lasseter = Pixar

Tom Watson, Jr. + Eliot Noyes = IBM

Walter Paepke + Herbert Bayer = Container Corp.

Adriano Olivetti + Marcello Nizzoli = Olivetti

Artur & Erwin Braun + Dieter Rams = Braun

Max Dupree + George Nelson = Herman Miller

William Paley + William Golden = CBS

Frank Stanton + Lou Dorfsman = CBS

Hans Knoll + Florence Schust = Knoll

Martha Stewart + Gael Towey & Eric Pike = Martha Stewart

Pixar has made 15 hit movies—in a row—by design, not luck.
Founder Ed Catmull explains how in his book, *Creativity, Inc.*

Story meetings at Pixar—**conversations,** without laptops, supported by a dedicated war room, lots of sketches, a high-info-density physical environment.

Increasingly, venture capital (VC) and business consulting firms are bringing senior designers into their conversations.

– Google Ventures named Braden Kowitz "Design Partner"
– Khosla hired Irene Au, former head of design at Google
– Kleiner-Perkins hired John Maeda, former RISD President

– Accenture bought European service design firm Fjord.
– Deloitte bought design planning firm Doblin Group.
– McKinsey bought SF product design firm Lunar.

These firms didn't hire these designers to make wireframes;
they hired them to change the nature of their conversations.

The quality of the conversation depends on the relationship between a design group and the organization that it supports.
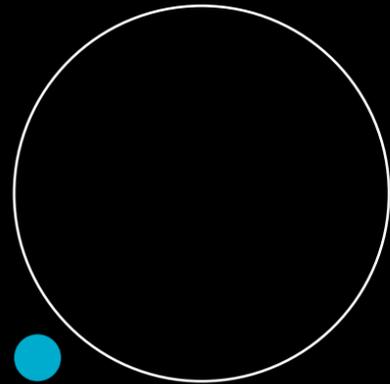
Separate

Design as external resource

Design thinking and methods have no continuous presence in the organization.

They are add-ons, limited to traditional problems: form, communication, function.

— Sabine Junginger, 2009
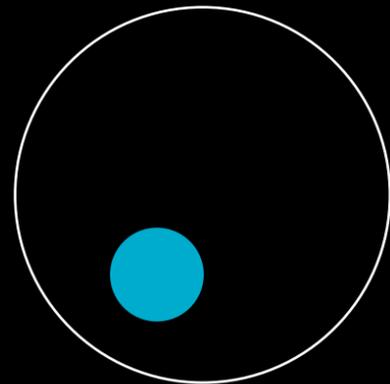
Peripheral

Design as part of the organization

Design thinking and methods practiced somewhere within the organization.

They apply to specific products and services.
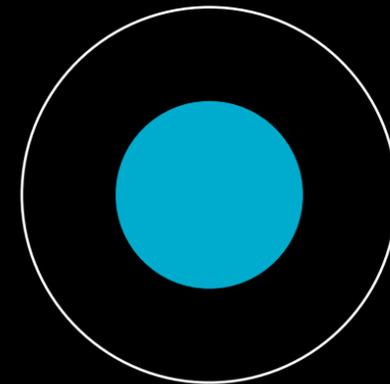
Central

Design at the core of the organization

Design thinking and methods are highly visible and take a central position.

They unify products and services across an organization; apply to corporate design and brand strategy.
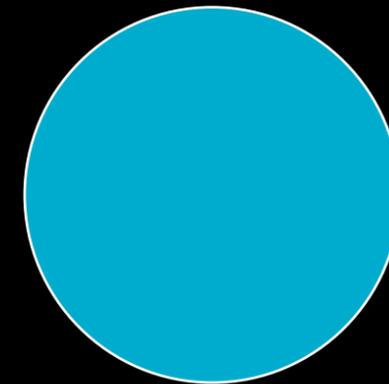
Integrated

Design integral to all aspects of the organization

Design thinking and methods are being applied at an organization's top level as means to inquire into a wide range of organizational problems with the aim to develop integrated solutions.

Traditional, industrial age management is hierarchical. Designers *add style after* a product has been engineered.

Manager Designer

Goals   Goals
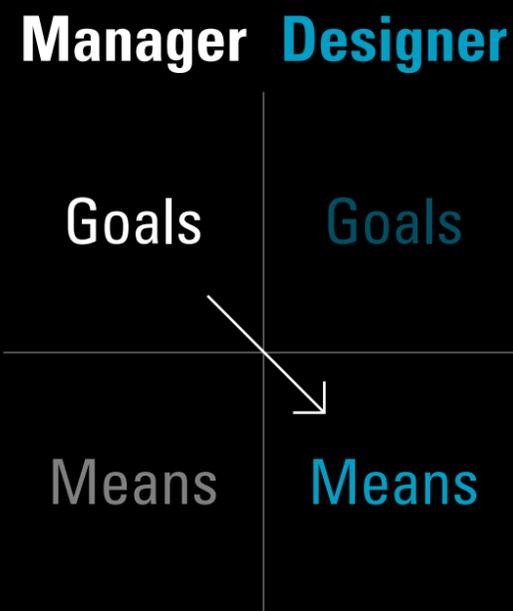
Means   Means

Controlling:
Manager tells designer
what to do + how to do it;
designer executes.

Manager Designer

Goals   Goals
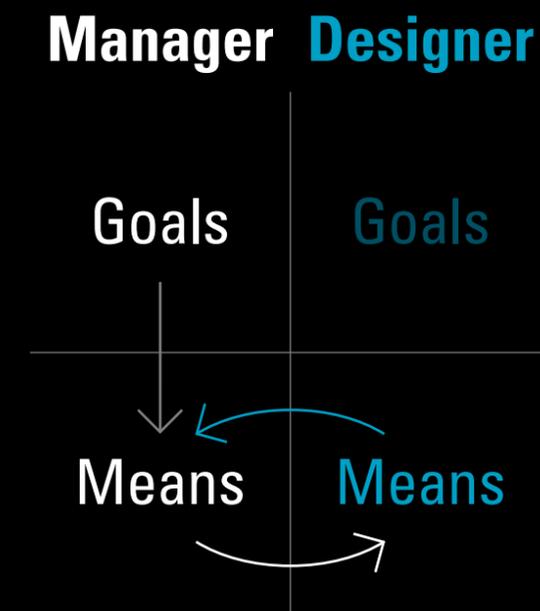
Means   Means

Mentoring:
Manager sets goals
but discusses means
with designer.

Emerging, information age management is more collegial. Designers *participate throughout* the entire development cycle.

Manager Designer

Goals $\rightarrow$ Goals
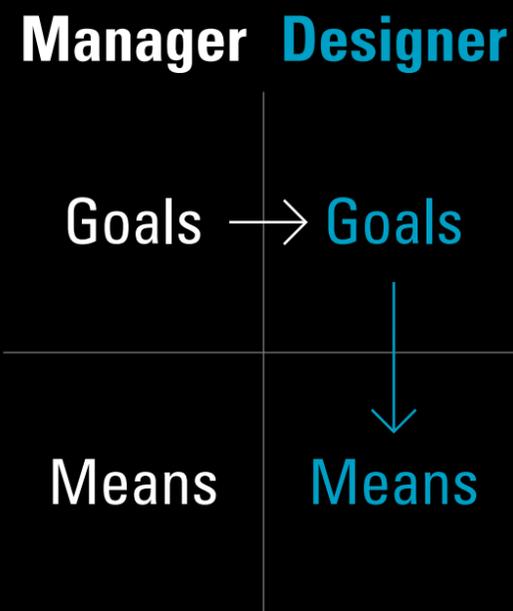
Means   Means

Delegating:
Manager sets goal
but leaves means
to the designer.

Manager Designer

Goals   Goals

Means   Means
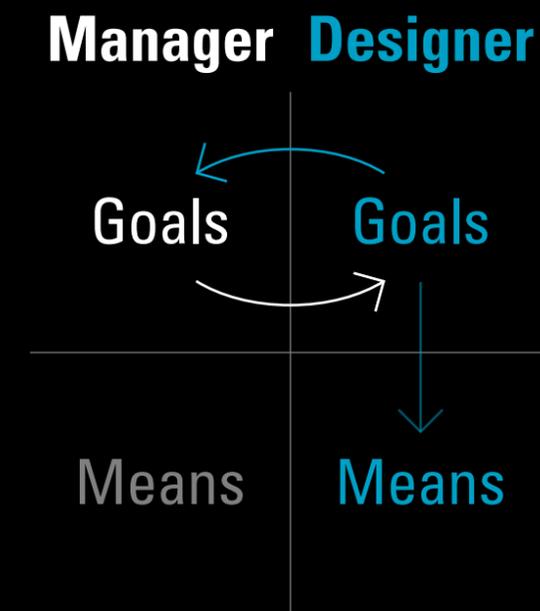
Collaborating:
Manager and designer
set goals together.

Sophisticated managers operating in an agile manner, **avoid waterfall handoffs** and include designers in daily scrums.
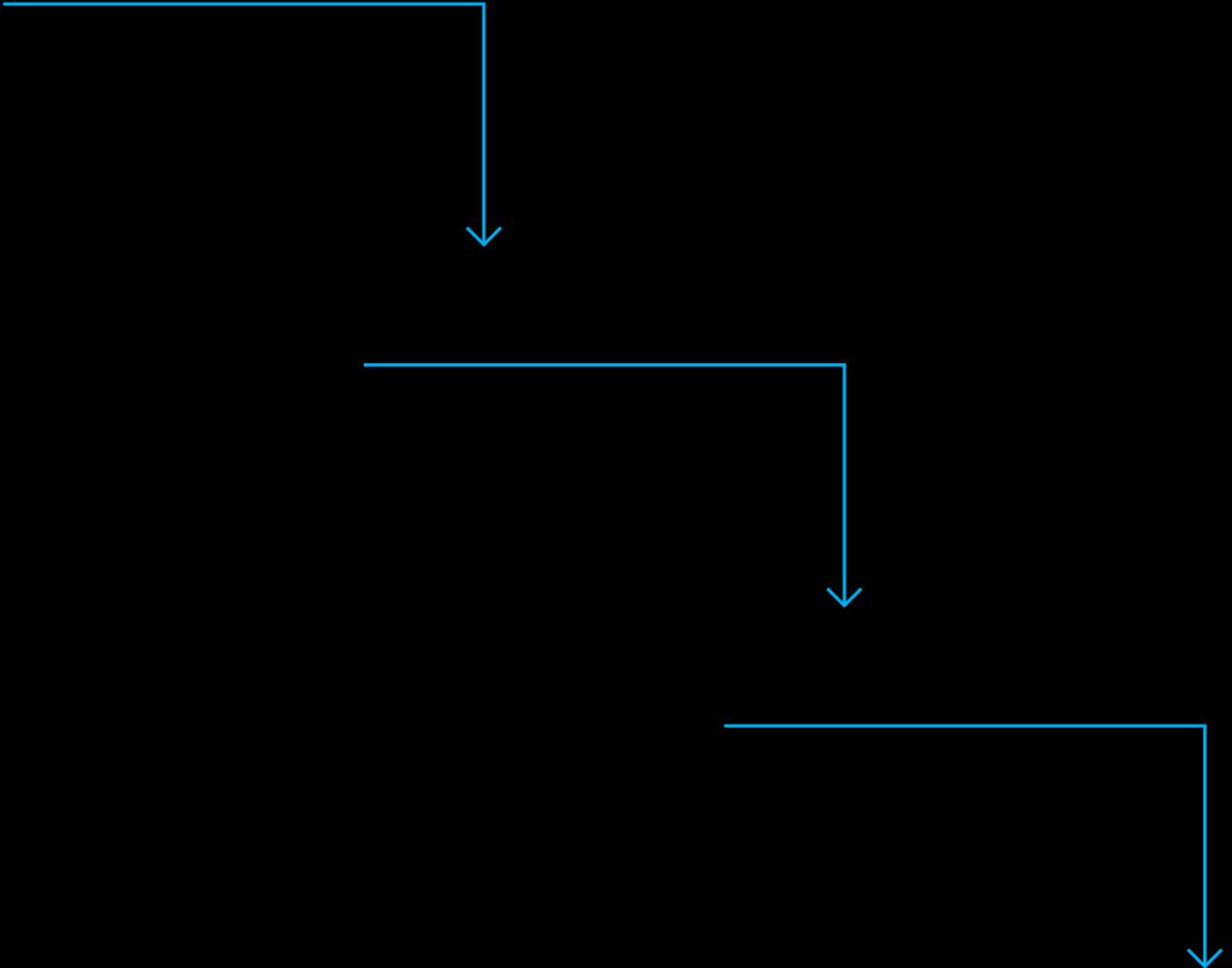
But agile managers often fall into the trap of focusing on wireframes, without building the necessary foundation.

If your designer isn't producing wireframes fast enough, it may be because you didn't make time to build the right scaffolding.

That is, you may not be having the right conversations.

The right conversations iterate shared vision and conceptual structures. That's where effectiveness, efficiency, and engagement are born.

"At its heart, software design is about creating virtual worlds in which users work, learn, and play.

Virtuality has two aspects:
1. Conceptual structure—the ideas and how they unfold, connect, and lodge in the mind
2. Feel—how things look and the other sensations we experience (crude or slick, bumpy or smooth, warm or cool)

The real issue is designing a consistent conceptual structure, one that fits the domain as much as possible, as comprehensively and comprehensibly as possible.

Consistency, completeness, and clarity are the objectives."

— Ted Nelson, the inventor of hypertext

"*For people to use a product successfully, they must have the same mental model (the user's model) as that of the designer (the designer's model). But the designer only talks to the user via the product itself, so the entire communication must take place through the 'system image': the information conveyed by the physical product itself.*"

— Don Norman, *The Design of Everyday Things*, 1988

"...most digital systems fail when they fail to provide a story, when there is a poor conceptual model."

— Don Norman
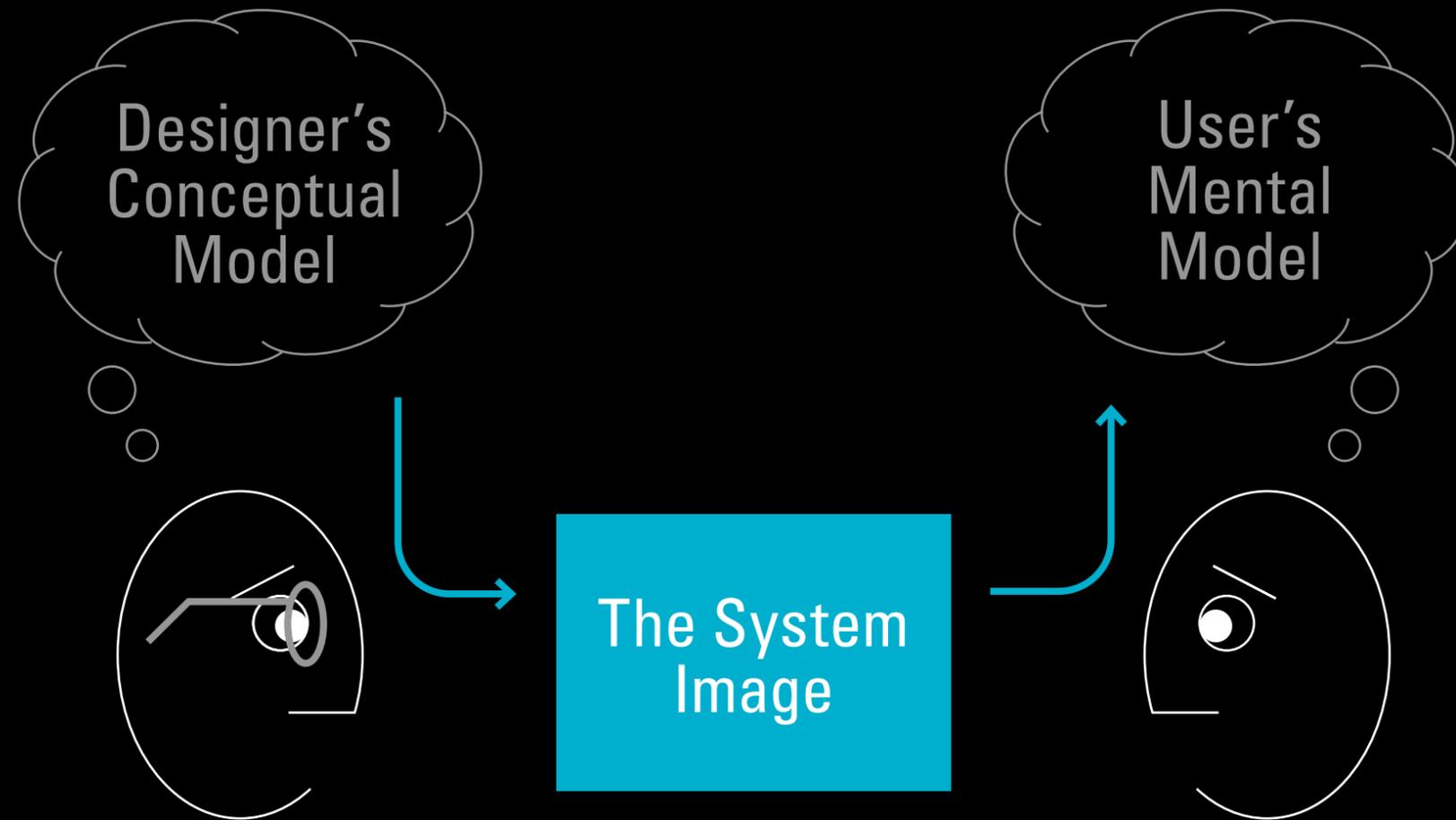
How to make conceptual models is explained in a wonderful new book
Conceptual Models: Core to Good Design, by Johnson & Henderson
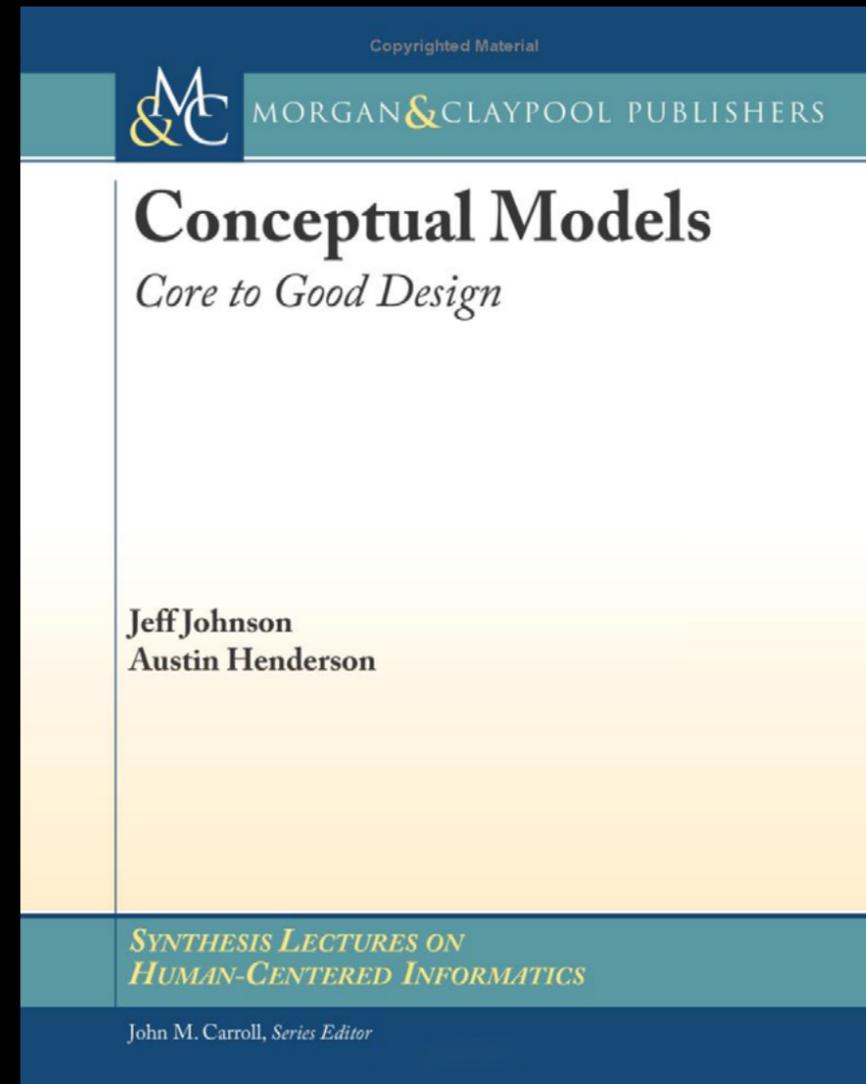
A conceptual model describes what a user needs to know in order to use your application successfully.

"*A conceptual model is a high-level description of an application. It enumerates all concepts in the application that users can encounter, describes how those concepts relate to each other, and how those concepts fit into tasks that users perform with the application.*"

— Jeff Johnson + Austin Henderson, *Conceptual Models: Core to Good Design*, 2012

As an example of a conceptual model Johnson + Henderson describe an alarm clock.

The clock **stores** the **current time** of day,
continually **updating** it to track the passage of time.

It **displays** the current time constantly.

Users can **set** the current time.

Users can set an **alarm** at a **specified time,** or no alarm.

When an alarm is set and the current time equals the set **alarm time,**
the **alarm** is triggered.

Users can **turn off** an alarm.

User conceptual models can be expressed in a number of ways. Node-link diagrams—or concept maps—can be very efficient.

Application   Includes   →   Main Data Object 1
can be
Created
Read
Updated
Displayed

Includes   →   Main Data Object 2
can be
Created
Read
Updated
Displayed

Example user conceptual model: an email client app

Accounts such as
- AOL
- Exchange
- Gmail
- Hotmail
- iCloud
- Ymail

include → Mail Boxes
can be
Added
Deleted
Logged in to
Logged out of

can be accessed by
Access includes
- Fetch and Push
- Auto and Manual

are → Primary (Default)
or
Secondary
can be
Designated

are associated with → Account Name
User Name
Email Address
User Name
Password
Server

Protocols
- SMTP for sending
- IMAP or POP for retrieving
- Exchange

for IMAP / POP → Port
Secure Connection
- Auto
- SSL/TLS
- STARTTLS

for Exchange → Domain
Secure Connection (yes or no)
can be
Configured

includes → Kindle email address
e.g., ron@kindle.com
is
Included w/Kindle purchase
Receives and converts docs
Automatically added to Contacts

Kindle Email Client or Other Clients

sends and receives → Messages
can be
Created
- New
- - Manually
- - From short cuts
- Reply
- Reply All
- Forward
- Sent
- Sent Again
Read (opened)
- As preview
- Full text
Read
- Text-to-speech
Updated (draft)
Deleted
- Locally
- Globally
- Singly
- En Masse
- Undo
Saved
- Locally
- To Cloud
Filtered
Sorted
Searched
- Locally
- Globally
- Within Filter
Moved
- Singly
- En masse
Flagged
Marked (read/not)
- Singly
- En masse

includes →

contain → Text
- Original
- Quoted
can be
Created
- By Virtual Keyboard
- By Microphone
  (voice-to-text)
Deleted
Formatted
Annotated
Spell-checked
Enlarged / Reduced

has → Format
could be
Hot Strings
- URLs (Auto-formed links)
- Dates (Calendar events)
- Names (Contacts)
- Phone numbers
- Addresses
can be
Auto-recognized

Annotations
- Highlights
- Notes
can be
Created / Deleted
Displayed / Hidden

could be → Plain Text
can be
Viewed + Created

RichText
can be
Viewed only

HTML Text
can be
Viewed only

Images
can be
Attached / Deleted
Displayed / Hidden
Saved
Loaded from remote source

sent to / uploaded from → Local memory
Cloud
Gallery
Quick Office
Documents to Go
Social network sites (later)

Attachments
can be
Attached
Deleted
Opened / Closed
Saved

sent to / uploaded from →

Meta-data
can be
Created / Deleted
- Automatic
- Manual
Displayed / Hidden
Used to Filter
Used to Sort

includes → Recipients
- Main
- CC
- BCC
Sender
Subject
Preview
Time Stamp (year, month, day, hour, minute)
Flag (on or off, depends on service)
Tags
Status (unread or read)
Response (none, reply, or forward)
Priority (low, normal, or high)
Routing Information

are stored in → Contacts List

Preferences
- Primary Account
- Notification
- - Sound
- - By Account
- - By Sender
- Preview Length
- Meta-data View
- Auto Download
- - Yes / No
- - Up to file size
- Number of Messages Stored
- Number of Days Saved

may include → Related Features
- Contacts
- Calendar
- Bookmarks
- Reminders (To-do's)

are organized in → Folders
can be
Created / Deleted
Opened / Closed
Moved

such as → All (unified in-box, default if multiple services)
Account (1 to n; mean = 6)
Inbox (Default, if only one service)
Drafts (Depends on service)
Sent (Depends on service)

Example user conceptual model: restaurant staff scheduling app

Users — have → **Profiles** — consist of →

Profiles
can be
Read
Updated

Picture
can be
Created
Read
Updated

Address
can be
Created
Read
Updated
— consist of →

Street Address
City
State
Country
ZIP
Lat + Long

Description
can be
Created
Read
Updated
Deleted

Privacy
can be
Created
Read
Updated
Public (Everyone)
Network (Connections only)
Private (Only me)

Accounts — consist of →
can be
Created
Activated
Updated
Deleted

First + Last Name
can be
Created
Read
Updated

Email Address
can be
Created
Read
Updated

Password
can be
Created
Reset
Updated

Mobile Number
can be
Created
Read
Updated

Language
can be
Created
Read
Updated

have → **Account Types**
can be
Switched
could be →

Individuals
can be
Created
Read
Updated
Deleted
— can be assigned as →

Owner
Administrator
Manager
Employee
— of →

Companies
can be
Created
Read
Updated
Deleted
— consist of →

Name
can be
Created
Read
Updated

Industry
can be
Created
Read
Updated

Size
can be
Created
Read
Updated

Account Holder
can be
Created
Read
Updated

can send → **Invites**
can be
Created
Read
Updated
Sent (Outgoing)
Pending (Incoming)
Accepted
Declined
— become → **Connections**
can be
Created
Read
Deleted
— could be →

have → **Skills** — consist of →
can be
Created
Read
Updated
Deleted

Name
can be
Created
Read
Updated

Rate Amount
can be
Created
Read
Updated

Rate Frequency
can be
Created
Read
Updated

Description
can be
Created
Read
Updated
Deleted

can have / can leave → **Recommendations**
can be
Created
Read

have → **Availability** — consist of / requires →
can be
Created
Updated
Repeated (Weekly)
Valid
Expired

Timeframe
can be
Created
Read
Updated

Comment
can be
Created
Read
Updated
Deleted

apply for / post → **Jobs**
can be
Created
Read
Updated
Invited
Posted (aka Advertised)
Applied to

Black = Entities (data type)
Cyan = Relation between entities
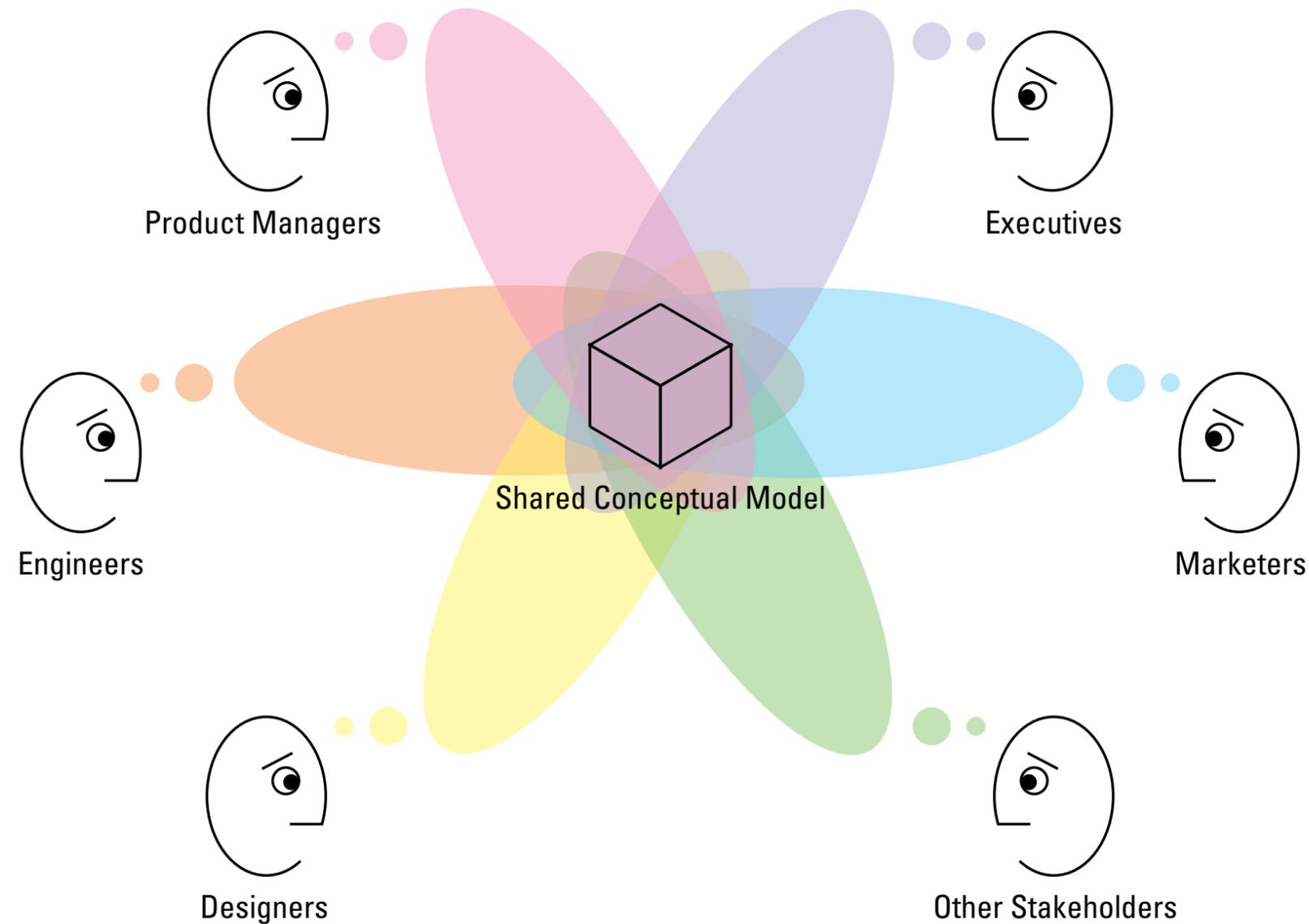Green = Action user can take on entity

23

Johnson + Henderson's conceptual model of a clock can be expressed in digital or analog interfaces.

At first the iPhone and Galaxy Nexus clocks look similar, but their conceptual models are more complex than the travel alarms.

In addition to communicating the conceptual model to users, designers also need to communicate with the product team.

Product Managers

Executives

Engineers

Marketers

Shared Conceptual Model

Designers

Other Stakeholders

The communication needs to go two ways; it needs to be a conversation.

That means designers don't own the model; the designer's role is to facilitate the conversation, by representing the model and prototyping.

In addition to user conceptual models, product teams benefit from producing a number of other artifacts, that support conversations.

These include:
– **An elevator pitch**—a summary of the product concept
– **Mission statements**—the purpose of the product, why it matters
– **Positioning statements**—how a product differs from competitors
– **User personas**—descriptions of users, their context, and their goals
– **Context maps**—representations of the situations in which a product is used
– **Systems maps**—representations of the infrastructure that supports the product
– **User stories**—descriptions of tasks users will undertake
– **Wireframes**—schematic representations of a product interface
– **Prototypes**—working versions of a product

Each artifact is what sociologist Susan Star calls a *boundary object*—objects that help bridge the boundaries between domains of knowledge.

The purpose of product team conversations is to improve products.

The better the conversation, the more the product team learns;
the more the product team learns, the better the product.

In a single iteration: understanding > agreement > action
Repeated many times: iteration > learning > better products
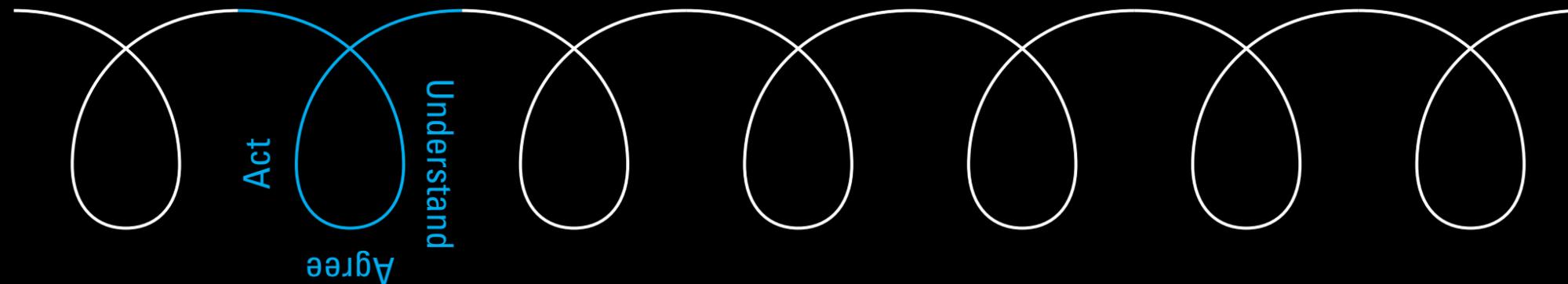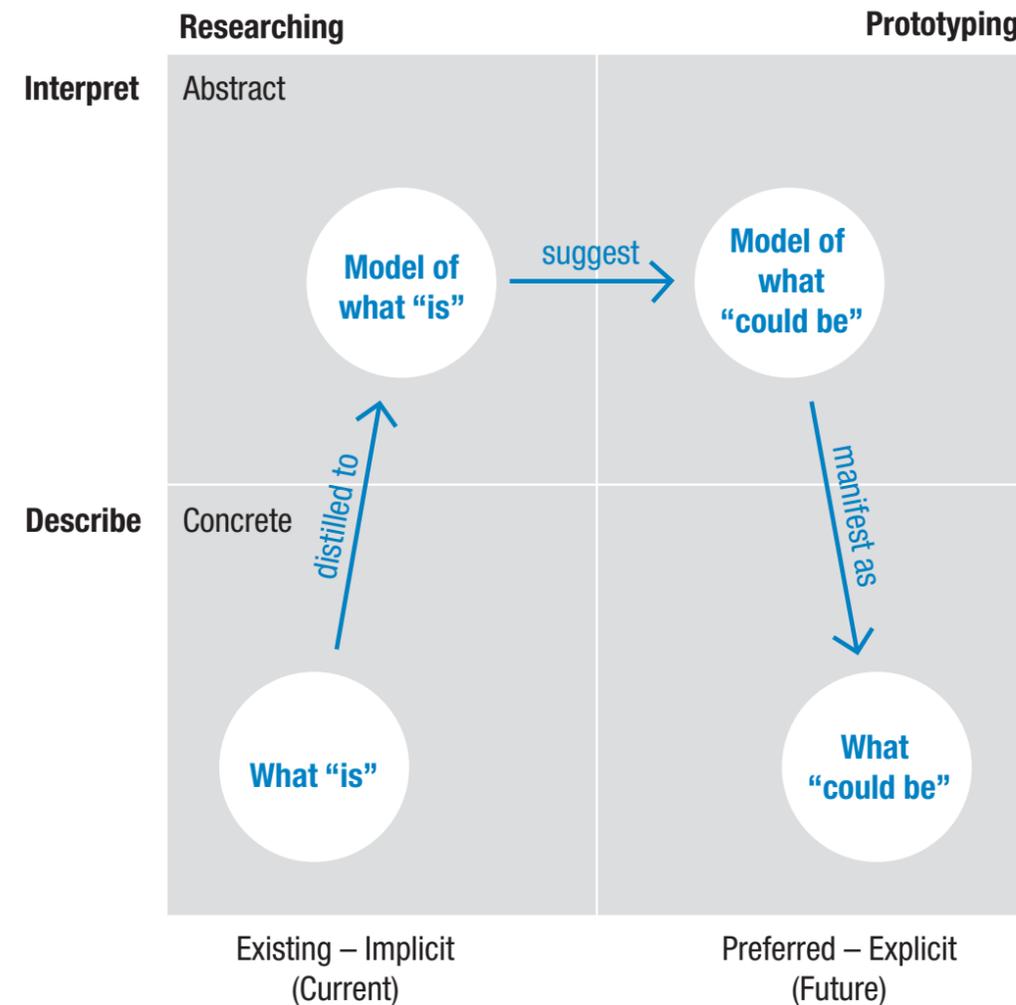
Herbert Simon framed design as the heart of all professions— design as knowledge work, design as learning.

> "*Everyone designs, who devises courses of action aimed at changing existing situations into preferred ones. The intellectual activity that produces material artifacts is no different fundamentally from the one that prescribes remedies for a sick patient or the one that devises a new sales plan for a company or a social welfare policy for a state.*"

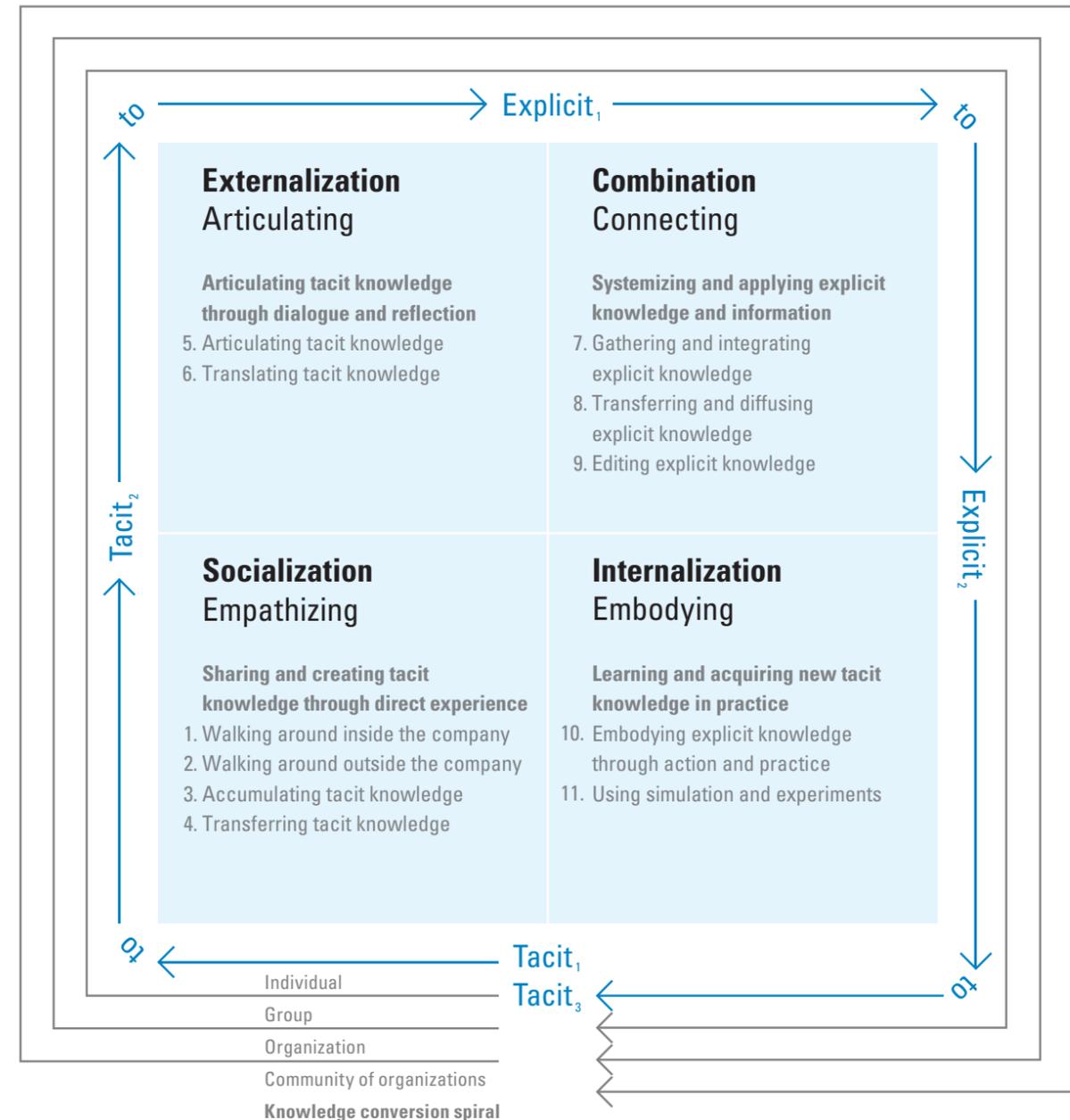— Herbert Simon, *Sciences of the Artificial*, 1969

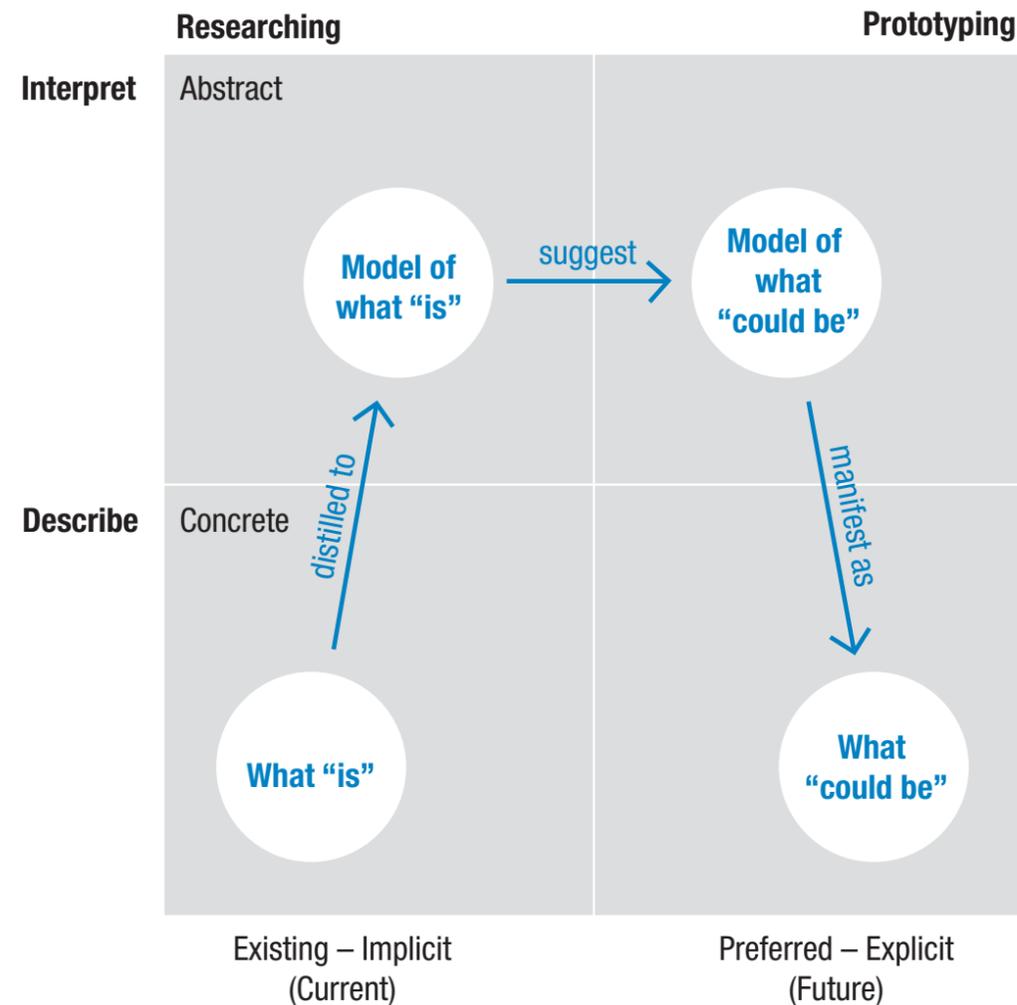# The Analysis-Synthesis Bridge Model shows how design crosses the gap between *what is* and *what should be*.

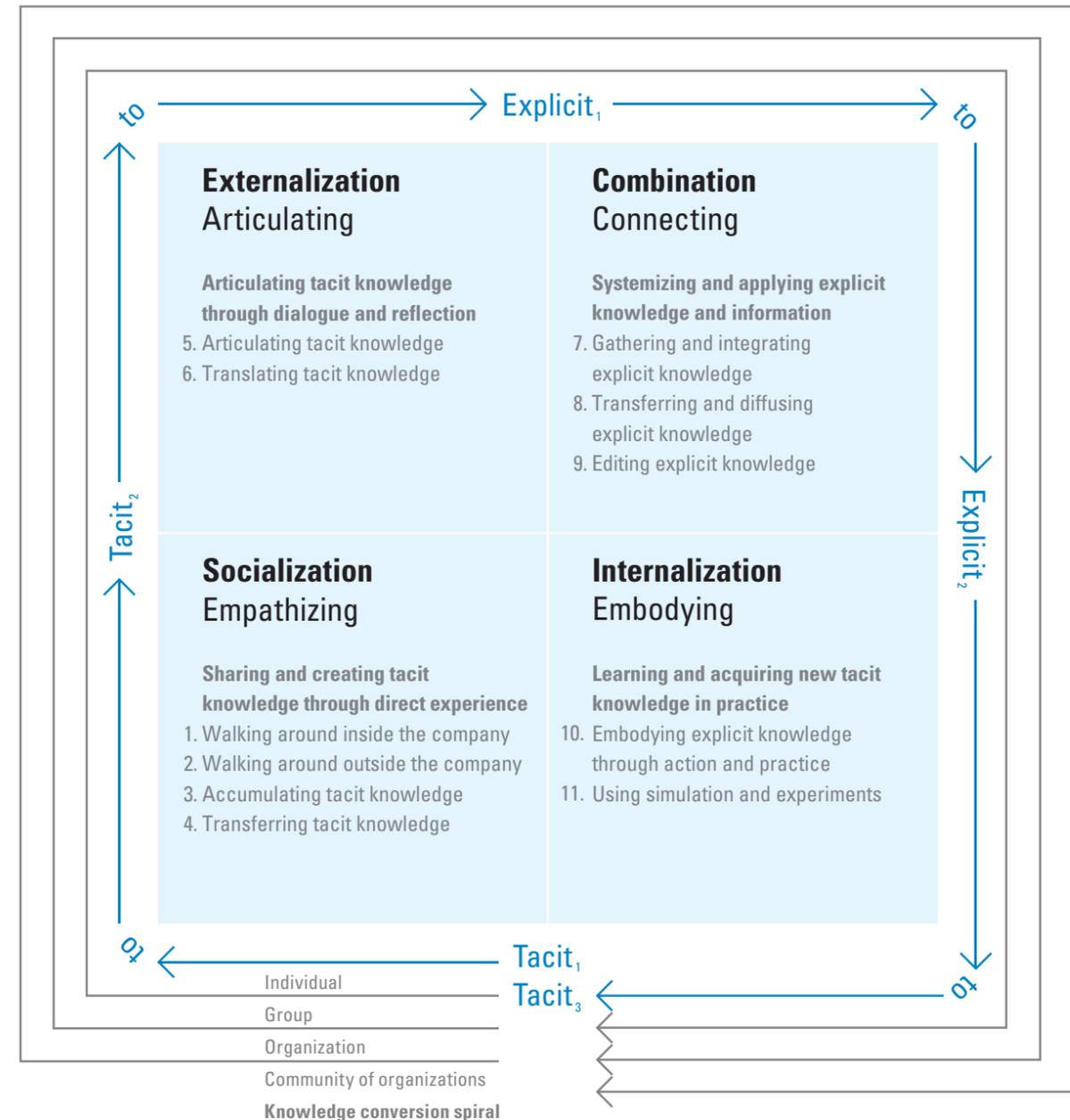Researching

Prototyping

Interpret — Abstract

Model of what "is"

→ suggest →

Model of what "could be"

↑ distilled to

↓ manifest as

Describe — Concrete

What "is"

What "could be"

Existing – Implicit
(Current)

Preferred – Explicit
(Future)

The SECI Model shows how organizations turn tacit knowledge into explicit knowledge, create new knowledge, and deploy it in operations.

Explicit₁

to → ← to

Externalization
Articulating

Articulating tacit knowledge through dialogue and reflection
5. Articulating tacit knowledge
6. Translating tacit knowledge

Combination
Connecting

Systemizing and applying explicit knowledge and information
7. Gathering and integrating explicit knowledge
8. Transferring and diffusing explicit knowledge
9. Editing explicit knowledge

Tacit₂   Explicit₂

Socialization
Empathizing

Sharing and creating tacit knowledge through direct experience
1. Walking around inside the company
2. Walking around outside the company
3. Accumulating tacit knowledge
4. Transferring tacit knowledge

Internalization
Embodying

Learning and acquiring new tacit knowledge in practice
10. Embodying explicit knowledge through action and practice
11. Using simulation and experiments

to ← ← to

Tacit₁
Tacit₃

Individual
Group
Organization
Community of organizations
Knowledge conversion spiral

Both models have the same basic structure—iterative loops— suggesting that *designing is* learning.

Researching   **Prototyping**

Interpret   Abstract

Model of what "is"   suggest   **Model of what "could be"**

distilled to   manifest as

Describe   Concrete

What "is"   **What "could be"**

Existing – Implicit (Current)   Preferred – Explicit (Future)

Analysis-Synthesis Bridge Model
Dubberly, Evenson & Robison (2008)

Explicit$_1$

Tacit$_2$ to   to   Explicit$_2$

Externalization
Articulating

Articulating tacit knowledge through dialogue and reflection
5. Articulating tacit knowledge
6. Translating tacit knowledge

Combination
Connecting

Systemizing and applying explicit knowledge and information
7. Gathering and integrating explicit knowledge
8. Transferring and diffusing explicit knowledge
9. Editing explicit knowledge

Socialization
Empathizing

Sharing and creating tacit knowledge through direct experience
1. Walking around inside the company
2. Walking around outside the company
3. Accumulating tacit knowledge
4. Transferring tacit knowledge

Internalization
Embodying

Learning and acquiring new tacit knowledge in practice
10. Embodying explicit knowledge through action and practice
11. Using simulation and experiments

to   Tacit$_1$ Tacit$_3$ to

Individual
Group
Organization
Community of organizations
**Knowledge conversion spiral**

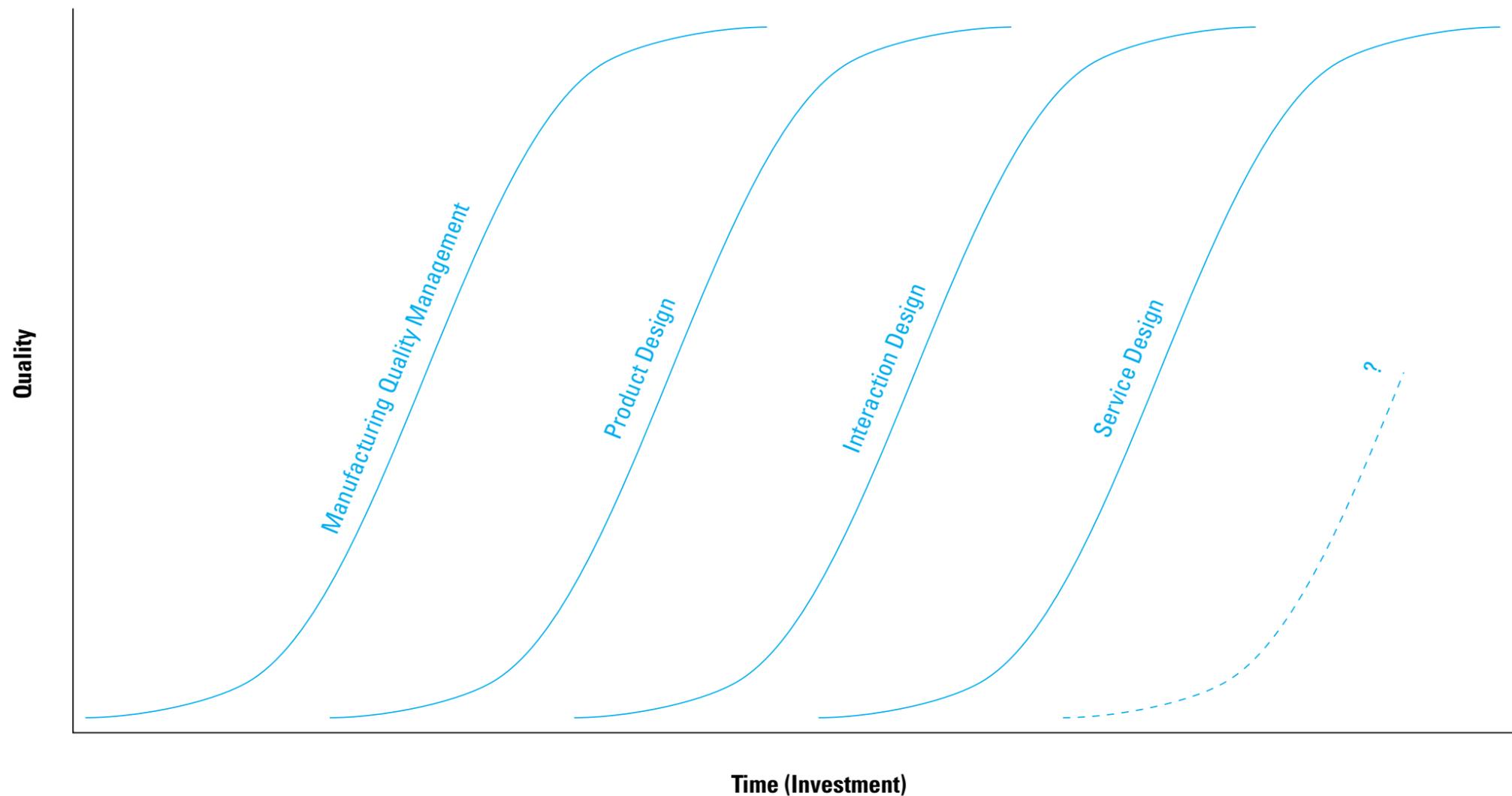**SECI model of knowledge create**
Ikujiro Nonaka (1995)

Learning happens at different scales—small + large.

– **Individuals** have insights, which they refine and share with colleagues, building support within an organization.

– **Companies** who master new skills first gain a lead over competitors, but competitors soon copy success and catch up.

– Eventually, knowledge becomes distributed throughout an **industry**— and innovative practices become standard operating procedure— table-stakes.

Learning is often described as a curve; learning curves can be seen as a series of waves.


Quality

Progress

Time (Investment)
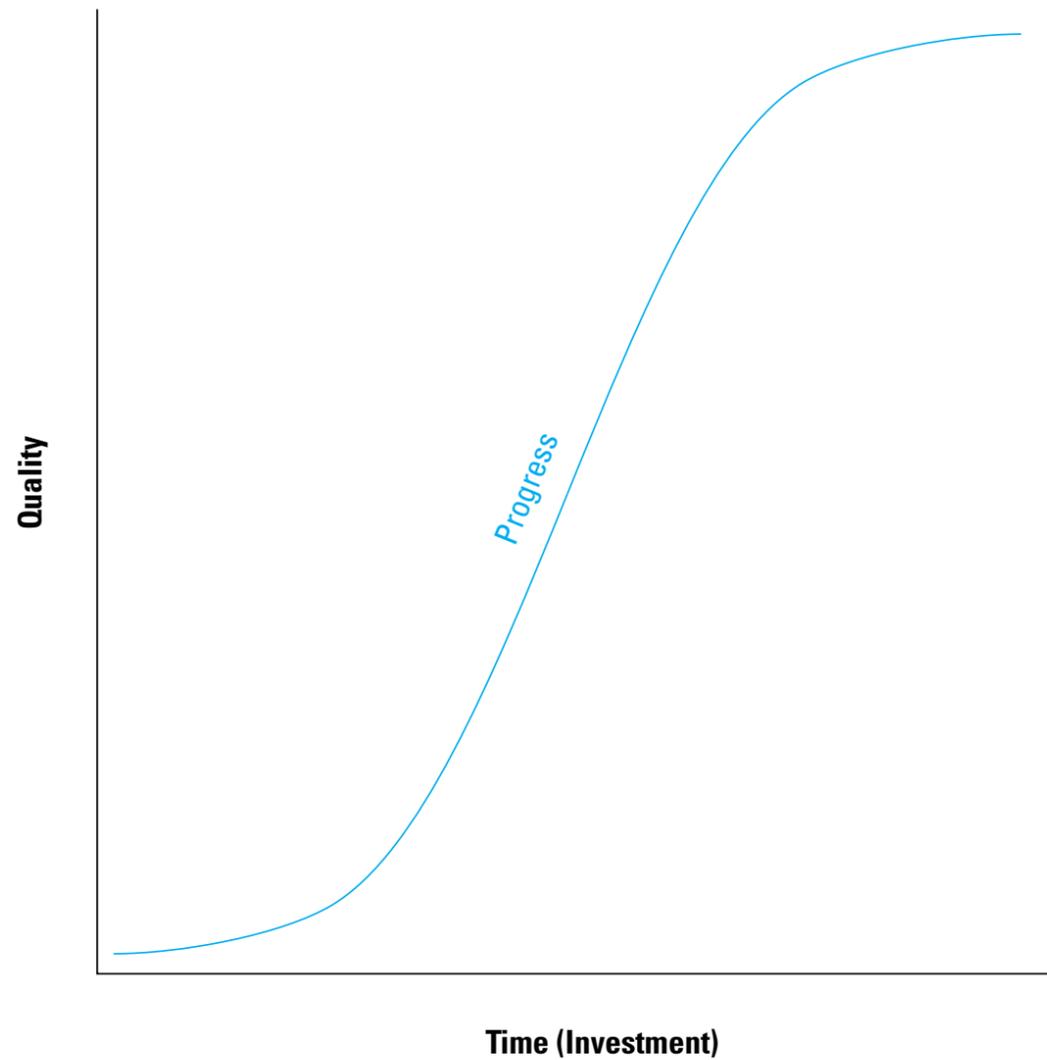
Over the last 40-50 years, product **innovation has occurred in 4 waves.** Climbing a curve early confers advantage; eventually it's a necessity.

Quality

Manufacturing Quality Management

Product Design

Interaction Design

Service Design

?

Time (Investment)

Apple's lead over Samsung has been across a series of learning curves.

Samsung responds quickly to external conversations—it's a follower.
Apple has generative conversations, iterating quickly—it's a leader.
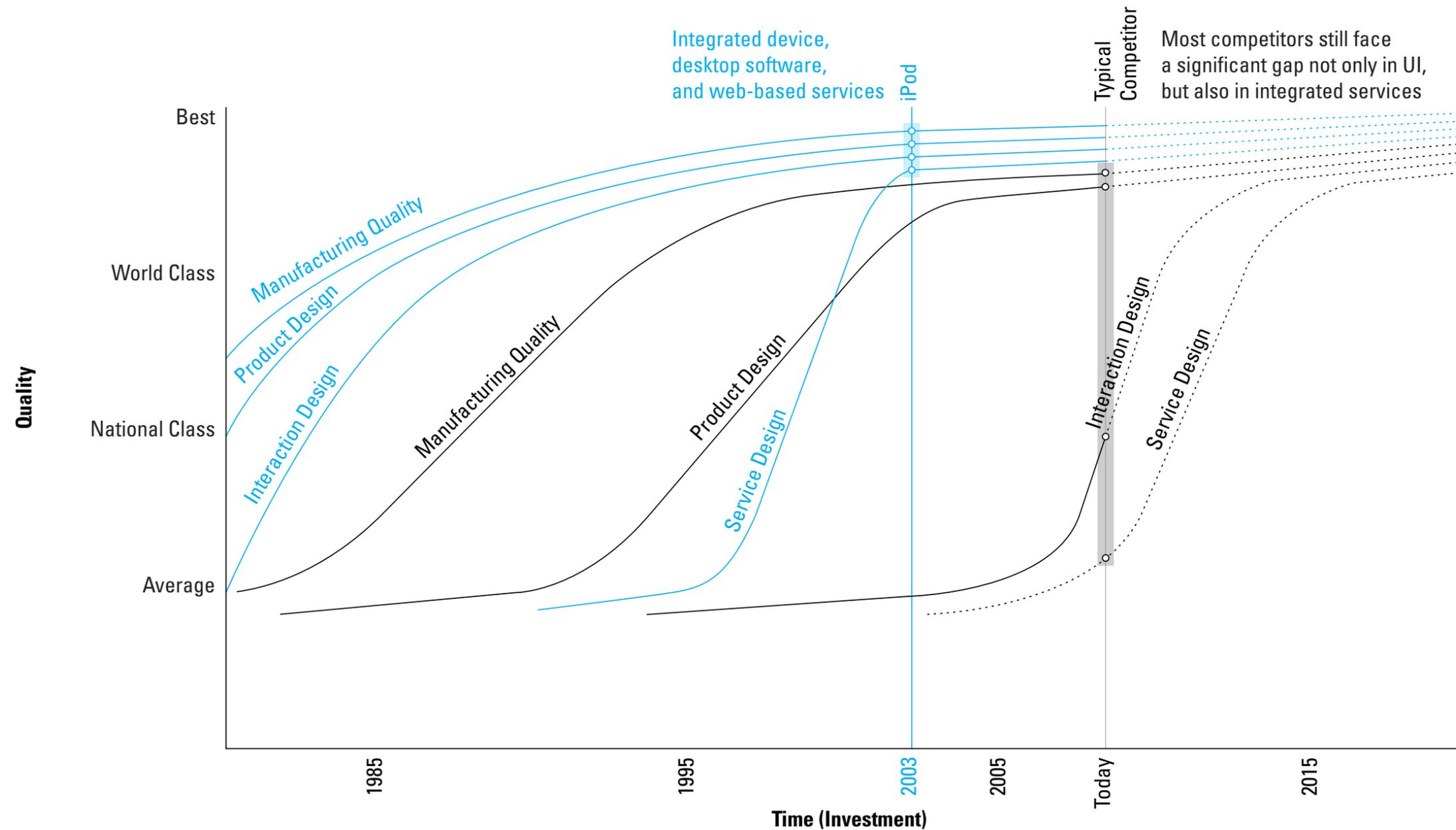
A final thought about the domain of design:
Design is not art; design is not science; design is politics.

Wireframes are an important step in movie-making and software design.
But if you begin and end the design conversation with wireframes,
the story will be lost; the project will lack coherence; and audiences will yawn.

The real issues, the issues that matter, are essentially political in nature.
What's important to us? What do we value? What are our values?

What aspects of our world do we wish to bring forward?
What do we want to conserve? What is our preferred situation? What *should* be?
What is the fundamental soul of this character, product, or enterprise?

These questions can only be answered through conversation.

Special thanks to
Zoe Hu
Paul Pangaro
Ryan Reposar

hugh@dubberly.com

Presentation posted at
www.dubberly.com/presentations/engage_designers.pdf