Wicked Problems, Affordances, Service Systems:

Some perspectives for designing

David Ing
Aalto University and
the International Society for the Systems Sciences

Tongji University College of Design & Innovation
Shanghai, PRC
November 24, 2017
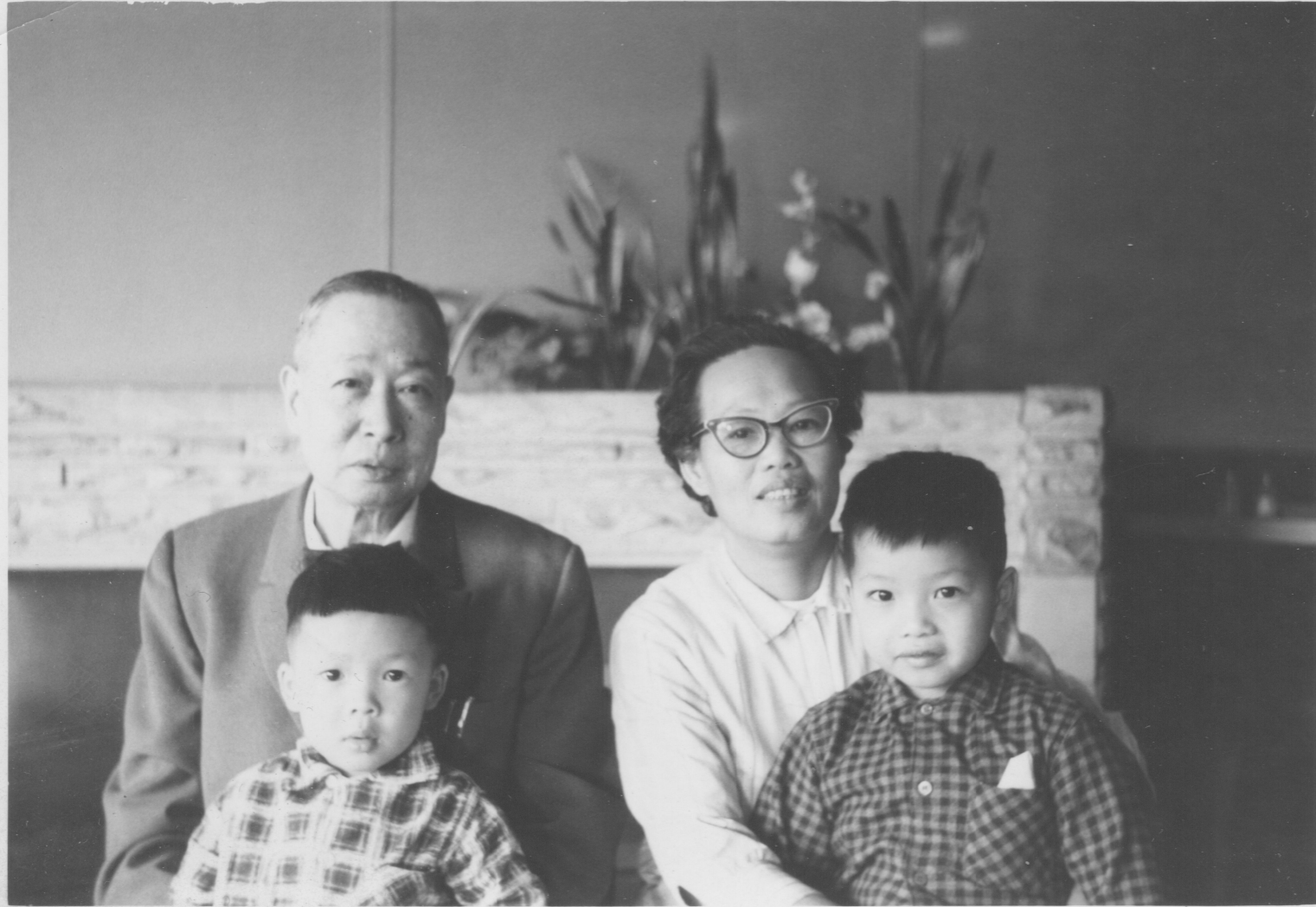
David Ing, Professional Experience

IBM Canada /
North America
(1985-2012; retired early)
Management consultant;
market development;
marketing scientist

Aalto U.
(2003-)
Since 2010, teaching in
master's program in
Creative Sustainability

U. of Toronto
Canadian Centre for
Marketing Information
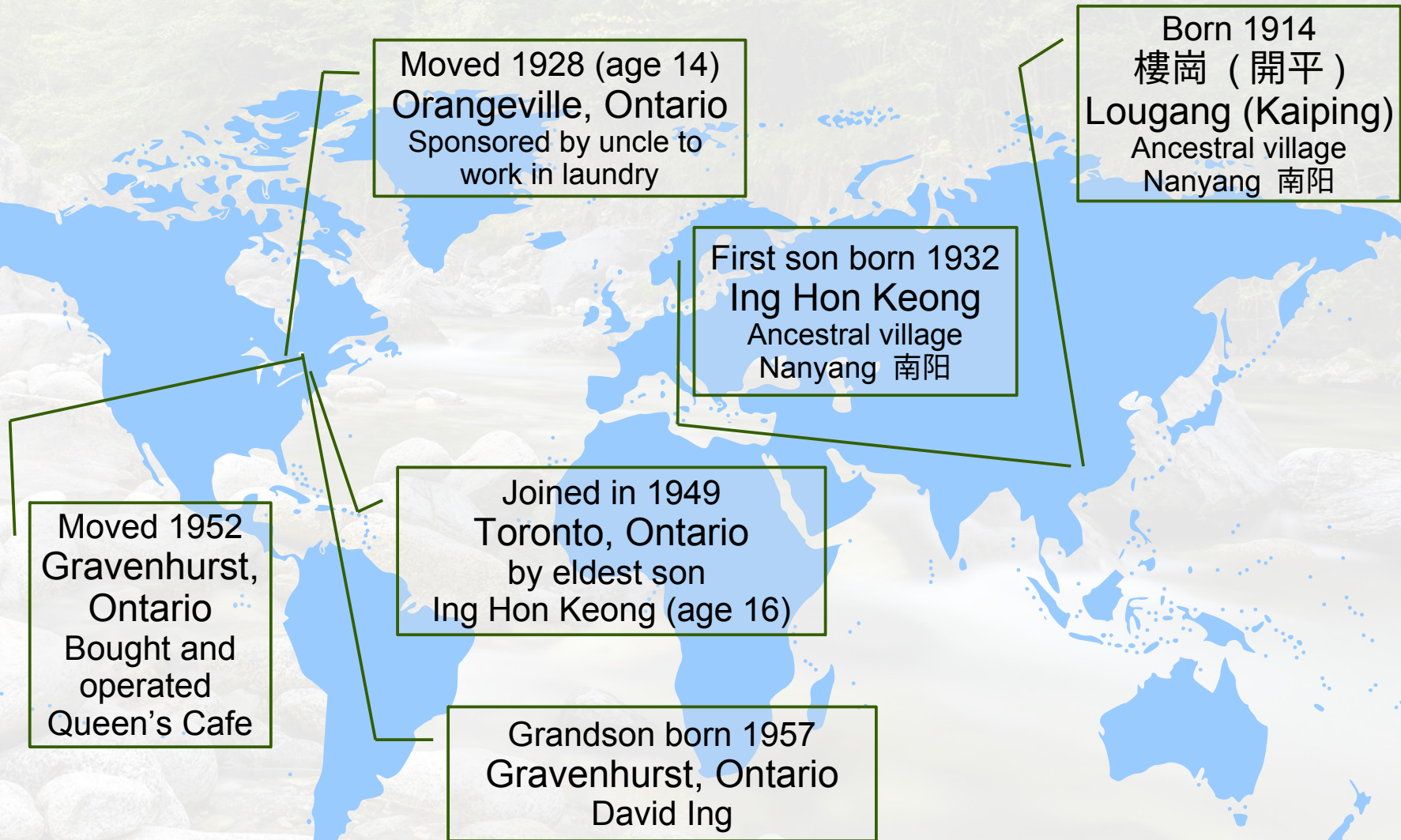Technologies (C^{2}MIT)
(cofounder 1990-1992)

International Society
for the
Systems Sciences
(President 2011-2012)

Circa 1964,
Gravenhurst, Ontario, Canada

Wicked problems, affordances, service systems

November 2017

Ing See Chong (Henry) (son and grandson)

Moved 1928 (age 14)
Orangeville, Ontario
Sponsored by uncle to work in laundry

Born 1914
樓崗（開平）
Lougang (Kaiping)
Ancestral village
Nanyang 南阳

First son born 1932
Ing Hon Keong
Ancestral village
Nanyang 南阳

Moved 1952
Gravenhurst, Ontario
Bought and operated
Queen's Cafe

Joined in 1949
Toronto, Ontario
by eldest son
Ing Hon Keong (age 16)

Grandson born 1957
Gravenhurst, Ontario
David Ing

Wicked problems, affordances, service systems

November 2017

David Ing, 2017

Coevolving Innovations

… in Business Organizations and Information Technologies

Christopher Alexander, Horst Rittel, C. West Churchman

At U.C. Berkeley in the 1960s, Christopher Alexander, Horst Rittel and C. West Churchman could have had lunch together. While disciplinary thinking might lead novices to focus only on each of pattern language, wicked problems and the systems approach, there are ties (as well as domain-specific distinctions) between the schools.

Circa 1968-1970: Christopher Alexander, Horst Rittel, West Churchman

Recent Posts

- Christopher Alexander, Horst Rittel, C. West Churchman
- Open Innovation Learning and Open Data
- Learning data science, hands-on
- Innovation Learning and Open Sourcing: IoT + Cloud + Cognitive
- Acts of representation with systems thinking (OCADU 2017/03)
- Service Systems Thinking, with Generative Pattern Language (Metropolia 2016/12)

Subscribe via e-mail

email address

Subscribe

Search

Search

Tweets by @daviding

David Ing
@daviding

Anshansicun: Whimsically residential area,... bit.ly/2jU

At Berkeley: Churchman, Rittel and Alexander taught in 1960-1970s

C. West Churchman (1913-2004)
- 1957 joined Berkeley, graduate programs in OR at School of Business Administration
- 1964-1970 Associate Director and Research Philosopher, Space Sciences Laboratory
- 1981-1994 retired, taught Peace & Conflict Studies

Horst Rittel (1930-1990)
- 1963 Berkeley College of Environmental Design
- 1974 both Berkeley and University of Stuttgart

Christopher Alexander (1936 -)
- 1963 Berkeley College of Environmental Design
- 1967 cofounder Center for Environmental Structure
- 1998 retired from university

Both Alexander and Rittel were part of what at the time was called the 'design methods' movement in architecture, worked and taught in the same building, and did talk and were seen walking off to have lunch together. Churchman was teaching in the Business School a few minutes down on the way to the center of campus.
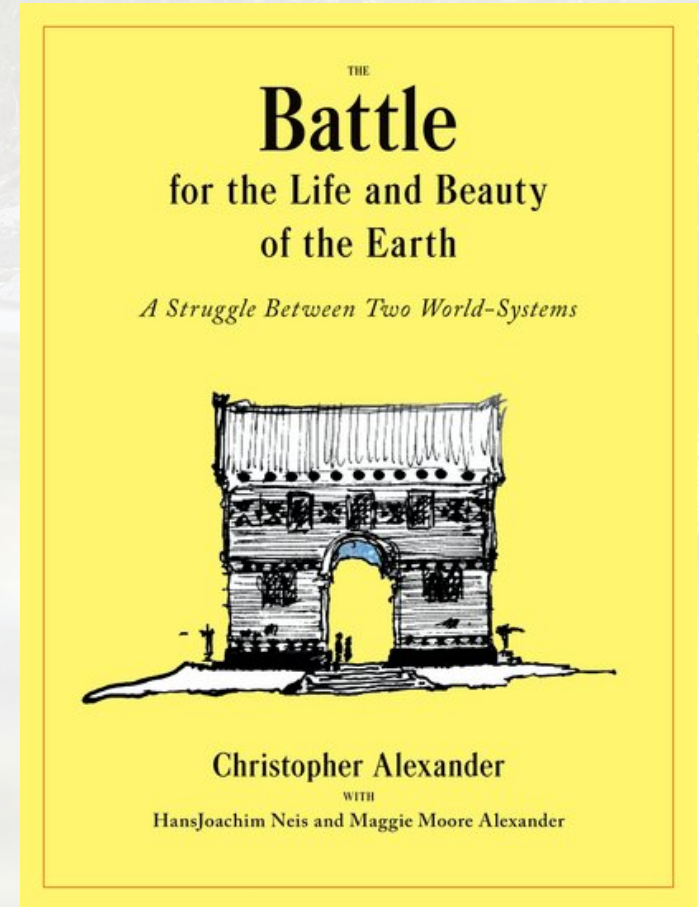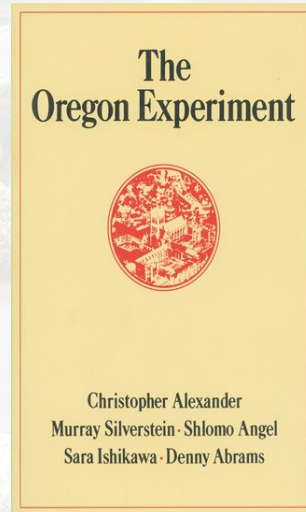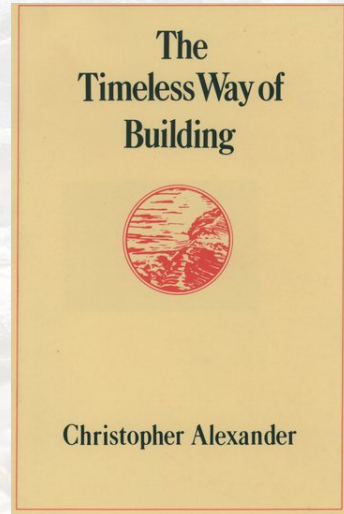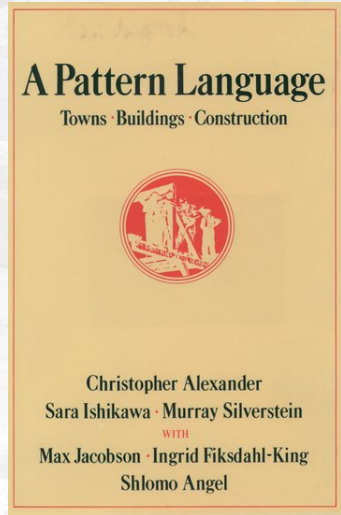
- *Thor Mann* (posted April 17, 2017)

Wicked problems, affordances, service systems   November 2017   David Ing, 2017

Agenda

1. **Solution to a problem in context ~ Wicked problems**

2. Functions in things ~ Affordances alongside interactions

3. Production systems ~ Service systems

Wicked problems, affordances, service systems   November 2017   David Ing, 2017

The writing of 1975-1979 by Alexander was prescriptive; the 2012 is reflections on practice

A Pattern Language
Towns · Buildings · Construction

Christopher Alexander
Sara Ishikawa · Murray Silverstein
WITH
Max Jacobson · Ingrid Fiksdahl-King
Shlomo Angel

The Timeless Way of Building

Christopher Alexander

The Oregon Experiment

Christopher Alexander
Murray Silverstein · Shlomo Angel
Sara Ishikawa · Denny Abrams

THE
Battle
for the Life and Beauty
of the Earth

A Struggle Between Two World-Systems

Christopher Alexander
WITH
HansJoachim Neis and Maggie Moore Alexander

Wicked problems, affordances, service systems

Here is a short and necessarily incomplete definition of a pattern:

A recurring structural configuration that solves a problem in a context, contributing to the wholeness of some whole, or system, that reflects some aesthetic or cultural value.[1]

Pattern Name: A name by which this problem/solution pairing can be referenced

Problem: The specific problem that needs to be solved.

Context
The circumstances in which the problem is being solved imposes constraints on the solution. The context is often described via a "situation" rather than stated explicitly.

Forces
The often contradictory considerations that must be taken into account when choosing a solution to a problem.

Solution: The most appropriate solution to a problem is the one that best resolves the highest priority forces as determined by the particular context.

Resulting Context
The context that we find ourselves in after the pattern has been applied. It can include one or more new problems to solve

Rationale
An explanation of why this solution is most appropriate for the stated problem within this context.

Related Patterns
The kinds of patterns include:
•Other solutions to the same problem,
•More general or (possibly domain) specific variations of the pattern,
•Patterns that solve some of the problems in the resulting context (set by this pattern)

The essential idea of a pattern language is: a *solution* to a *problem* in *context*

Every time a designer creates a pattern (or, for that matter, entertains any idea about the physical environment), he essentially goes through a three-step process.

He considers a PROBLEM, invents a PATTERN to solve the problem, and makes mental note of the range of CONTEXTS where the pattern will solve the problem. [....]

The format says that whenever a certain **CONTEXT** exists, a certain **PROBLEM** will arise; the stated **PATTERN** will solve the **PROBLEM** and there should be provided in the **CONTEXT**.

While it is not claimed that the PATTERN specified is the only solution to the PROBLEM, it is implied that unless the PATTERN or an equivalent is provided, the PROBLEM will go unsolved (Alexander, Ishikawa, & Silverstein, 1967, pp. 1–4).

Alexander, Christopher, Sara Ishikawa, and Murray Silverstein. 1967. *Pattern Manual.* Berkeley, California: Center for Environmental Structure

# "Dilemmas in a General Theory of Planning", (Rittel + Weber, 1973)

There are at least ten distinguishing properties of planning-type problems, i.e. wicked ones .... We use the term "wicked" in a meaning akin to that of "malignant" (in contrast to "benign") or "vicious" (like a circle) or "tricky" (like a leprechaun) or "aggressive" (like a lion, in contrast to the docility of a lamb). [....]

1. There is no definitive formulation of a wicked problem ....
2. Wicked problems have no stopping rule ....
3. Solutions to wicked problems are not true-or-false, but good-or-bad ....
4. There is no immediate and no ultimate test of a solution to a wicked problem ....
5. Every solution to a wicked problem is a "one-shot operation"; because there is no opportunity to learn by trial-and-error, every attempt counts significantly ....
6. Wicked problems do not have an enumerable (or an exhaustively describable) set of potential solutions, nor is there a well-described set of permissible operations that may be incorporated into the plan ....
7. Every wicked problem is essentially unique ....
8. Every wicked problem can be considered to be a symptom of another problem ....
9. The existence of a discrepancy representing a wicked problem can be explained in numerous ways. The choice of explanation determines the nature of the problem's resolution ....
10. The planner has no right to be wrong ....

In 1969, problem seeking was *architectural programming*, and problem solving was *design*

Programming is a specialized and often misunderstood term. It is "*a statement of an architectural problem* and the requirements to be met in offering a solution. While the term is used with other descriptive adjectives such as *computer* programming, *educational* programming, *functional* programming, etc., in this report, programming is used to refer only to architectural programming.

Why programming? The client has a project with many unidentified sub-problems. The architect must define the client's total problem.

problem seeking → problem solving → solution
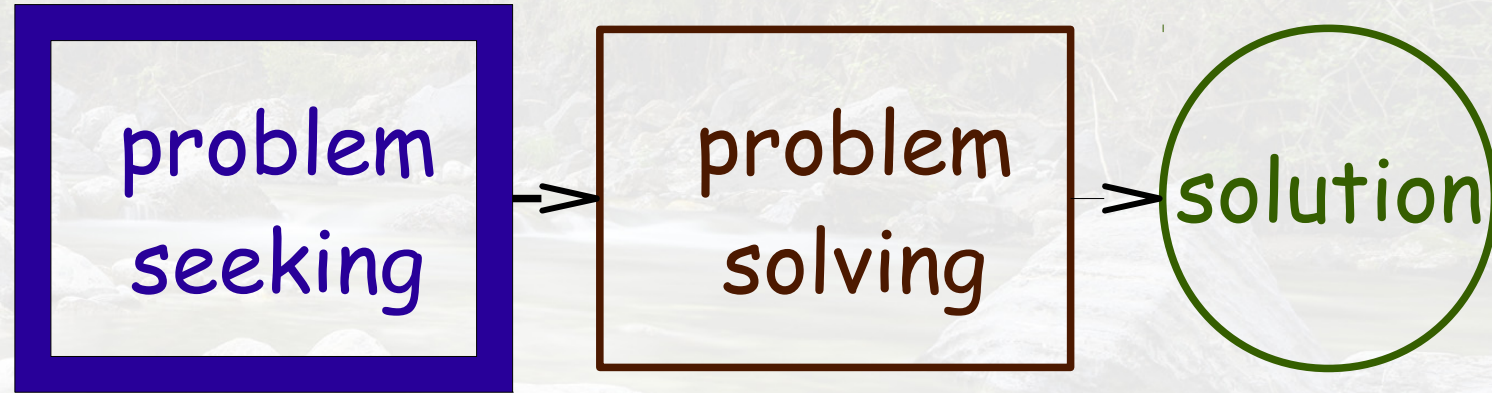
Design is problem solving; programming is problem seeking. The end of the programming process is a statement of the total problem; such a statement is the element that joins programming and design. The "total problem" then serves to point up constituent problems, in terms of four considerations, those of form, function, economy and time.

The aim of the programming is to provide a sound basis for effective design. The State of the Problem represents the essense and the uniqueness of the project. Furthermore, it suggests the solution to the problem by defining the main issues and giving direction to the designer (Pena and Focke 1969, 3).

Agenda

1. Solution to a problem in context ~ Wicked problems
   Problem seeking ~ Problem solving

Wicked problems, affordances, service systems   November 2017   David Ing, 2017

Agenda

1. Solution to a problem in context ~ Wicked problems

2. Functions in things ~ Affordances alongside interactions

3. Production systems ~ Service systems

Wicked problems, affordances, service systems   November 2017   David Ing, 2017

Systems thinking is a perspective on wholes, parts and their relations

containing whole

Function (non-living) or role (living)

structure

part A $_{(t)}$

part B $_{(t)}$

process

part A $_{(t)}$

part A $_{(t+1)}$
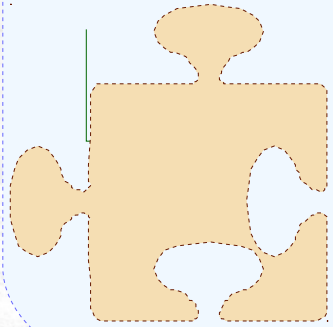
Function
"contribution of the part to the whole"
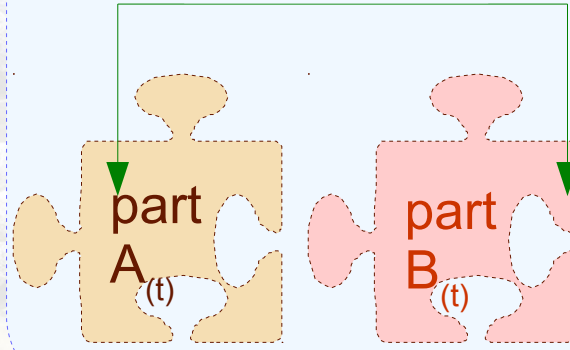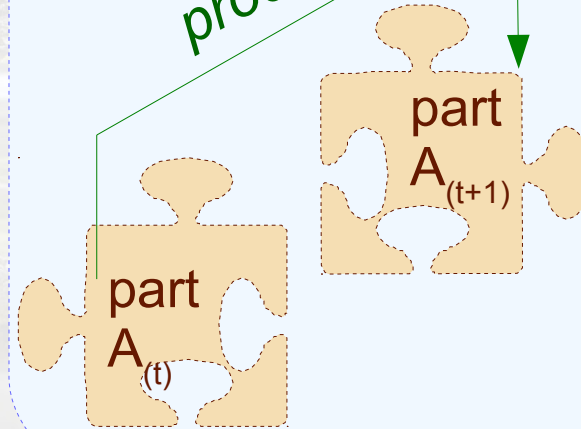
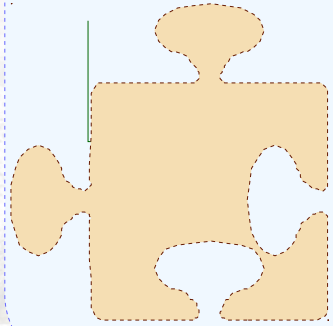**Structure**
"arrangement in space"

Process
"arrangement in time"

In authentic systems thinking, synthesis precedes analysis and the containing whole is appreciated

containing whole

Function (non-living) or role (living)

Synthesis precedes analysis
1. Identify a containing whole (system) of which the thing to be explained is a part.
2. Explain the behavior or properties of the containing whole
3. Then explain the behavior or properties of the thing to the explained in terms of its role(s) or function(s) within its containing whole.

Source: Ackoff, Russell L. 1981. *Creating the Corporate Future: Plan or Be Planned For. New York:* John Wiley and Sons. http://books.google.com/books?id=8EEO2L4cApsC.

Pacing layers emphasize coevolution and learning

SITE
This is the geographical setting, the urban location, and the legally defined lot, whose boundaries outlast generations of ephemeral buildings. "Site is eternal", Duffy agrees.

STRUCTURE
The foundation and load-bearing elements are perilous and expensive to change, so people don't. These are the building. Structural life ranges from 30 to 300 years (but few buildings make it past 60, for other reasons).

SKIN
Exterior surfaces now change every 20 years or so, to keep up with fashion or technology, or for wholesale repair. Recent focus on energy costs has led to re-engineered Skins that are air-tight and better-insulated.

SERVICES
These are the working guts of a building: communications wiring, electrical wiring, plumbing, sprinkler system, HVAC (heating, ventilation, and air conditioning), and moving parts like elevators and escalators. They wear out or obsolesce every 7 to 15 years. Many buildings are demolished early if their outdated systems are too deeply embedded to replace easily.

SPACE PLAN
The interior layout, where walls, ceilings, floors, and doors go. Turbulent commercial space can change every 3 years; exceptionally quiet homes might wait 30 years.

STUFF
Chairs, desks, phones, pictures; kitchen appliances, lamps, hair brushes; all the things that twitch around daily to monthly. Furniture is called mobilia in Italian for good reason.
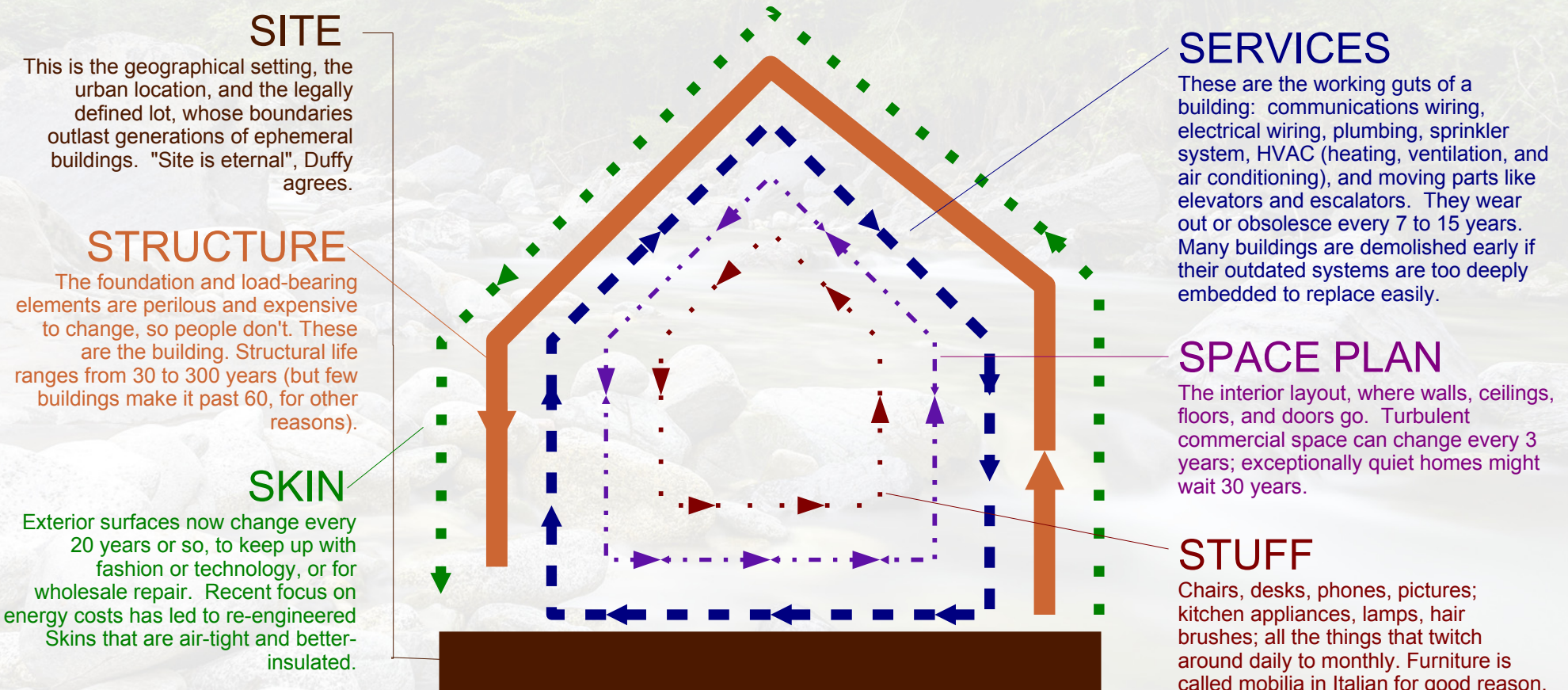
Source: Stewart Brand. 1994. *How Buildings Learn: What Happens after They're Built*. New York: Viking.

# Ask Not What's Inside Your Head, but What Your Head's Inside of

Stimulus – Response
(Behavioral Psychology)
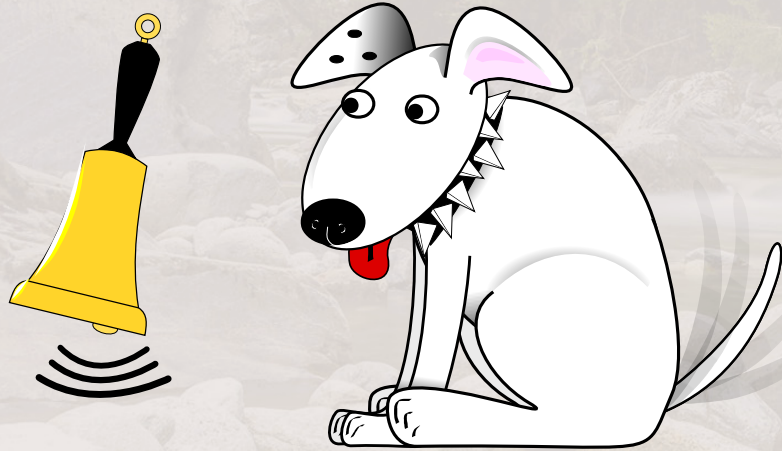
[In the 1950] psychophysics of perception … "givens" in the light to the eye could not support perceptual phenomena, but only elementary experiences such as sensations. [….] Succinctly put, the psycho-physical program was … traditional in considering perception to be a set of responses to presented stimuli (albeit "higher order" stimuli).

Ecological Approach to Perception
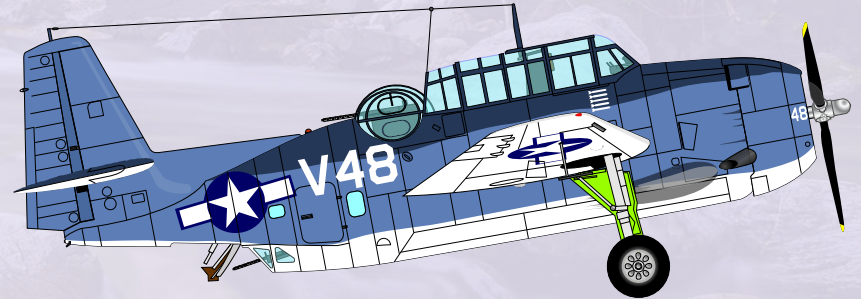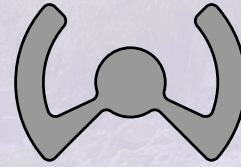
Over the last 10-15 years [James J. Gibson] has tried to develop enough theory … to demonstrate that direct perception is indeed plausible even if hordes of difficult details remain to be worked out. The … analysis of the optic array, stimulus organization, and the functional organization of perceptual systems are what Gibson oftens points to as radical features ….

William M. Mace 1977. "James J. Gibson's Strategy for Perceiving: Ask Not What's inside Your Head, but What Your Head's inside of." In *Perceiving, Acting, and Knowing: Toward an Ecological Psychology*, edited by Robert Shaw and John Bransford, 43–65.

After 2007, service systems have been recognized as the largest part of developed economies globally

A service system can be defined as "**a dynamic value-cocreation configuration of resources**, including people, organizations, shared information (language, laws, measures, methods), and technology, all connected internally and externally to other service systems by **value propositions**"
(Maglio, Vargo, Caswell, & Spohrer, 2009, p. 399).

The smallest service system centers on an individual as he or she **interacts** with others, and the largest service system comprises the global economy. Cities, city departments, businesses, business departments, nations, and government agencies are all service systems.

Every service system is b**oth a provider and client of service** that is connected by **value propositions** in value chains, value networks, or value-creating system ….
(Maglio & Spohrer, 2008, p. 18)

A high- or low-ability client may prefer an offering as an *input*, or *output*, of coproduction

Offering as input	**Self-service logic** (independence and convenience maximization)
	Partnership logic (value co-development)

Self-service logic
(independence and
convenience maximization)

Partnership logic
(value co-development)

Offering as
input

Offering as
output

Industrial logic
(production cost
reduction)

Service logic
(customer satisfaction)

Customer value
through transactions

Customer value
through relationship

Scope
Service
content

Scope
People
content

Scope
Physical content

Rafael Ramirez and Johan Wallin. *Prime Movers: Define Your Business or Have Someone Define It Against You*, 2000, p. 141.

Affordances are relational in an ecological perception

service system

Affordances for A

service beneficiary A
(high ability)

Offering config A as input

Affordances for B

service beneficiary B
(low ability)

Offering config B as output
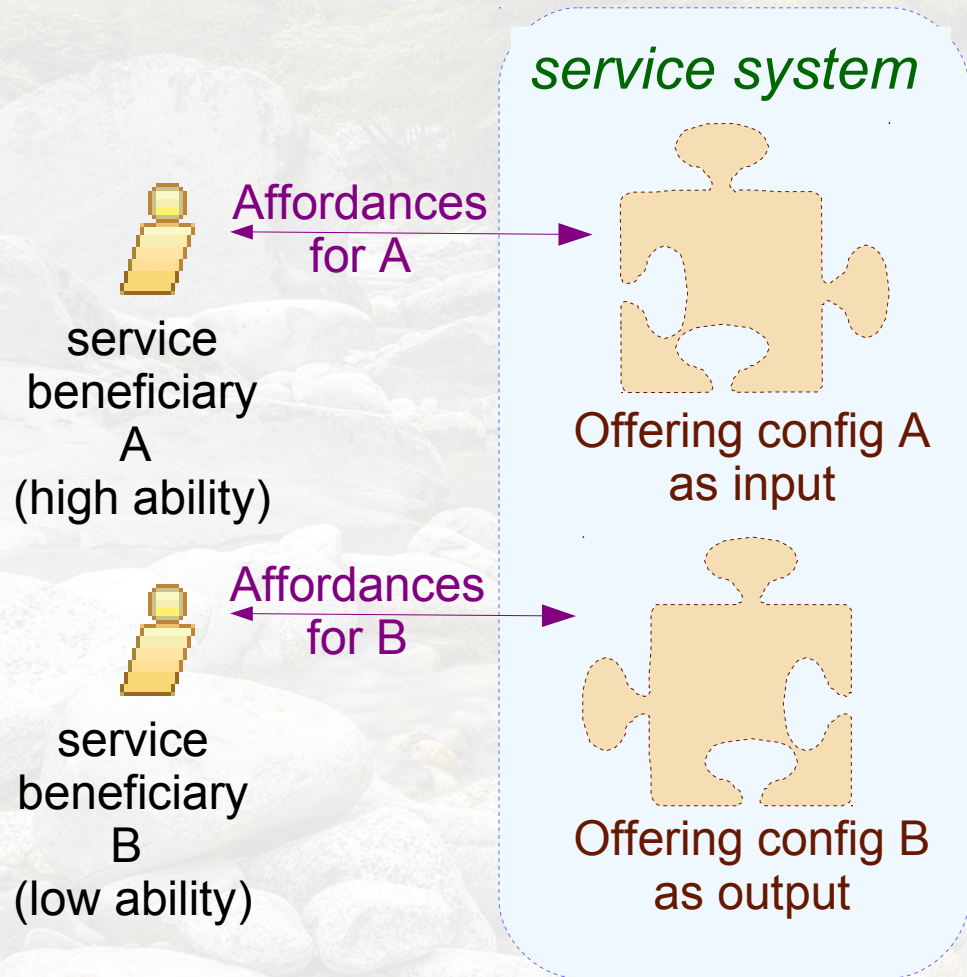
The term *affordance* refers to whatever it is about the environment that contributes to the kind of interaction that occurs. [….]

An affordance relates attributes of something in the environment to an interactive activity by an agent who has some ability, and an ability relates attributes of an agent to an interactive activity with something in the environment that has some affordance.

The relativity of affordances and abilities is fundamental. Neither an affordance nor an ability is specifiable in the absence of specifying the other.

James G. Greeno 1994. "Gibson's Affordances."
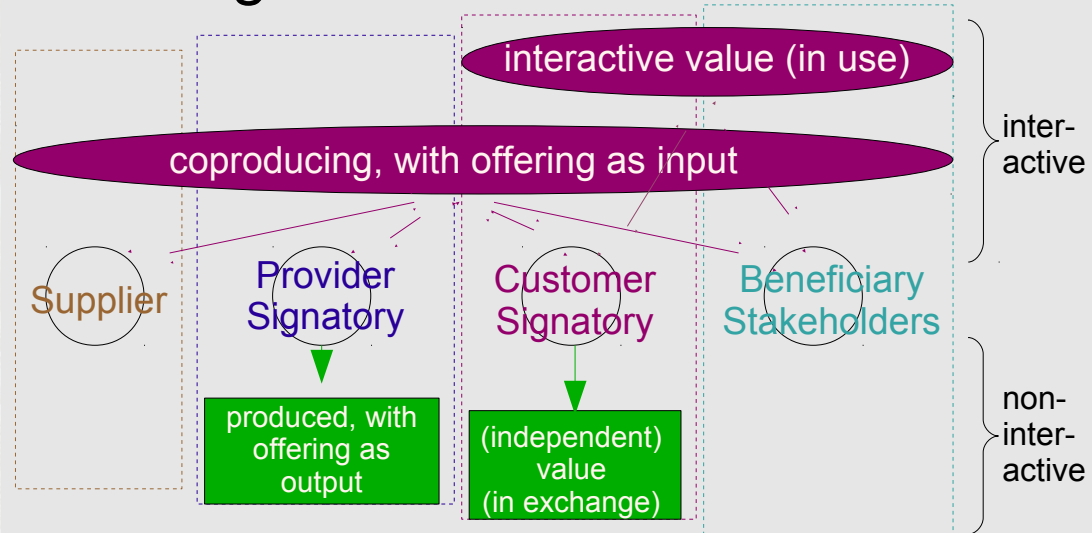*Psychological Review* 101 (2): 336–342.

The theory of firms adding value cost has given way to mobilizing customers towards creating their own value

Adding ~~value~~ cost

Added ~~value~~ cost

Added ~~value~~ cost

Added ~~value~~ cost

Suppliers

Service Provider

Customer
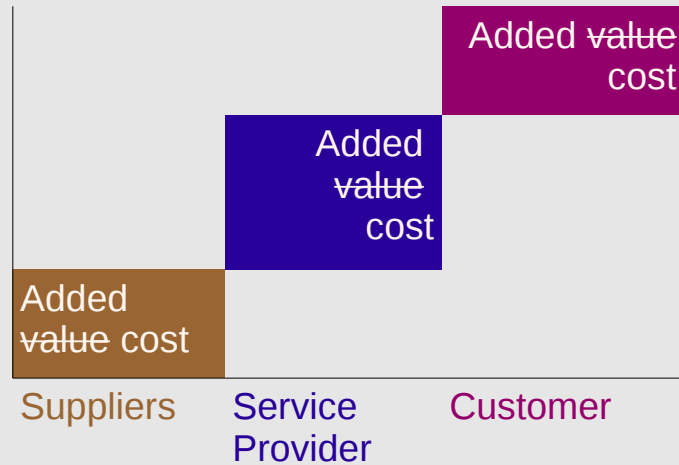
Our traditional about value … [says] every company occupies a position on the value chain. Upstream, suppliers provide inputs. The company then adds values to these inputs, before passing them downstream to then next actor in the chain [whether another business or the final consumer].

Enabling interactive value creation

interactive value (in use)

coproducing, with offering as input

inter-active

Supplier

Provider Signatory

Customer Signatory

Beneficiary Stakeholders

produced, with offering as output

(independent) value (in exchange)

non-inter-active

… IKEA's strategic intent [is] to understand how customers can create their own value and create a business system that allows them to do it better. IKEA's goal is not to *relieve* customers of doing certain things but to *mobilize* them to do easily certain things they have never done before. Put another way, IKEA invents value by enabling customers' own value-creating activities. … Wealth is [the ability] to realize your own ideas.

Agenda

2. Functions in things ~ Affordances alongside interactions
   Synthesis (of form) ~ Cocreating value

Wicked problems, affordances, service systems   November 2017   David Ing, 2017

Agenda

1. Solution to a problem in context ~ Wicked problems

2. Functions in things ~ Affordances alongside interactions

3. Production systems ~ Service systems

Wicked problems, affordances, service systems   November 2017   David Ing, 2017

Wicked problems, affordances, service systems   November 2017

By 2008, a smarter world was envisioned

Pre-digital physical infrastructure	Converging physical and digital infrastructure	
World as invisible or unobserved	Our world is becoming INSTRUMENTED	
Analog / synchronous connections, person-to-person and machine-to-machine	Our world is becoming INTERCONNECTED	
Things as dumb or unresponsive to interaction	Virtually all things, processes and ways of working are becoming INTELLIGENT	

Wicked problems, affordances, service systems   November 2017

Computing, cognition and the future of knowing

How humans and machines are forging a new age of understanding

IBM

Dr. John E. Kelly III
Senior Vice President,
IBM Research and Solutions Portfolio

Cognitive computing refers to systems that learn at scale, reason with purpose and interact with humans naturally. Rather than being explicitly programmed, they learn and reason from their interactions with us and from their experiences with their environment.
[...]
Those systems have been deterministic; cognitive systems are probabilistic. They generate not just answers to numerical problems, but hypotheses, reasoned arguments and recommendations about more complex — and meaningful — bodies of data.

The Future of Cognitive Computing

Computing, cognition and the future of knowing

How humans and machines are forging a new age of understanding

IBM

Dr. John E. Kelly III
Senior Vice President,
IBM Research and Solutions Portfol

The Cognitive Era (2011—)

The potential for something beyond programmable systems was foreseen as far back as 1960, when computing pioneer J.C.R. Licklider wrote his seminal paper "Man-Computer Symbiosis." Much of modern computing is based on Licklider's research and insights:

"Man-computer symbiosis is an expected development in cooperative interaction between men and electronic computers. It will involve very close coupling between the human and the electronic members of the partnership. "

Image 1

The Tabulating Era
(1900s – 1940s)

The Programming Era
(1950s – present)

The Cognitive Era
(2011 –)

"The Systems Approach and Its Enemies", (Churchman, 1979)

Common to all these enemies is that none of them accepts the reality of the "whole system": we do not exist in such a system. Furthermore, in the case of morality, religion, and aesthetics, at least a part of our reality as human is not "in" any system, and yet it plays a central role in our lives.

To me these enemies provide a powerful way of learning about the systems approach, precisely because they enable the rational mind to step outside itself and to observe itself (from the vantage point of the enemies). [....]

We must face the reality that the enemies offer: what's really happening in the human world is politics, or morality, or religion, or aesthetics. This confrontation with reality is totally different from the rational approach, because the reality of the enemies cannot be conceptualized, approximated, or measured (Churchman, 1979, pp. 24–53).

Agenda

3. Production systems ~ Service systems
    Programming era ~ Cognitive era

Wicked problems, affordances, service systems   November 2017   David Ing, 2017

Agenda

1. Solution to a problem in context ~ Wicked problems
   Problem seeking ~ Problem solving

2. Functions in things ~ Affordances alongside interactions
   Synthesis of form ~ Cocreating value

3. Production systems ~ Service systems
   Programming era ~ Cognitive era

Wicked problems, affordances, service systems   November 2017   David Ing, 2017

If they can get you asking the wrong questions, they don't have to worry about answers (Thomas Pynchon)

Type **1** error	**False positive**: finding a (statistical) relation that isn't real
Type **2** error	**False negative**: missing a (statistical) relation that is real
Type **3** error	**Tricking ourselves**: Unintentional error of solving wrong problems precisely (through ignorance, faulty education or unreflective practice)
Type **4** error	**Tricking others**: Intentional error of solving wrong problems (through malice, ideology, overzealousness, self-righteousness, wrongdoing)

Ian I. Mitroff and Abraham Silvers. 2010. *Dirty Rotten Strategies: How We Trick Ourselves and Others into Solving the Wrong Problems Precisely.* Stanford University Press.

Organizations embark on *open innovation* initiatives to sweep in external knowledge, practices and resources in cooperation with partners. This contrasts to the mainstream *private innovation* approach of in-house research and development sponsored solely by an incorporated funder, with intellectual property protected by copyright. Few organizations simultaneously engage in both approaches, within and across the levels of programs, projects and individuals. How does *learning* occur in such an organization -- and the communities of members within the organization -- in both cumulative and distributed ways?

The *open innovation learning* exhibited by IBM in the decade of 2001-2011 provides a foundation for building both descriptive theories and normative theories. Legal protocols for *open source* licensing began in 1998, and "open innovation" became popular in the business press from 2003. At the beginning of the 2001-2011 period, a behaviour of *open sourcing* by commercial enterprises departed from a tradition of *private sourcing* that presumes trade secrets for competitive advantage that maintaining economic viability. After a decade, *Open Sourcing while Private Sourcing* (*OSwPS*) had been demonstrated as a successful way of doing business at IBM, and had also become adopted by other companies and institutions.

The primary method employed to appreciate the phenomenon of OSwPS is multiparadigm inquiry. Theories are developed inductively from seven case studies, characterized in five containing contexts over the period, in a process orientation observing events, activities and choices ordered over time. Three descriptive theories have been built in parallel perspectives based on contrasting philosophies. Pursuit of a normative theory subsequently led to the proposal of additional hypotheses.

Emerging theories of open innovation learning challenge a presumption that commercial and non-commercial interests are incompatible. Open sourcing *while* private sourcing is a demonstrable way of conducting a viable business.

David Ing is a cofounder of the *Trito Innovation Colab*, centered in Toronto, Canada. An alumnus of IBM after 28 years, this research was conducted during doctoral studies of the *Aalto University School of Science* in Finland. He received a master's degree from the *Kellogg School of Management* at *Northwestern University*, and a bachelor's degree from *Trinity College* at the *University of Toronto*. He has served as president of the *International Society for the Systems Sciences*, and is an ambassador for the *International Society of Service Innovation Professionals*.

Jim Spohrer is a Director of the IBM Cognitive Opentech Group at *IBM Research Almaden*, in San Jose, California. Previously, he was Director of IBM Global University Programs, a cofounder of the Almaden Services Research group, and the founding CTO of IBM's Venture Capital Relations Group in Silicon Valley. He has a Ph.D. in artificial intelligence from *Yale University*, and a bachelor's degree in Physics from *MIT*.

Open Innovation Learning

Theory building on open sourcing while private sourcing

Ing

Open Innovation Learning

Theory building on open sourcing while private sourcing

David Ing

Foreword by Jim Spohrer