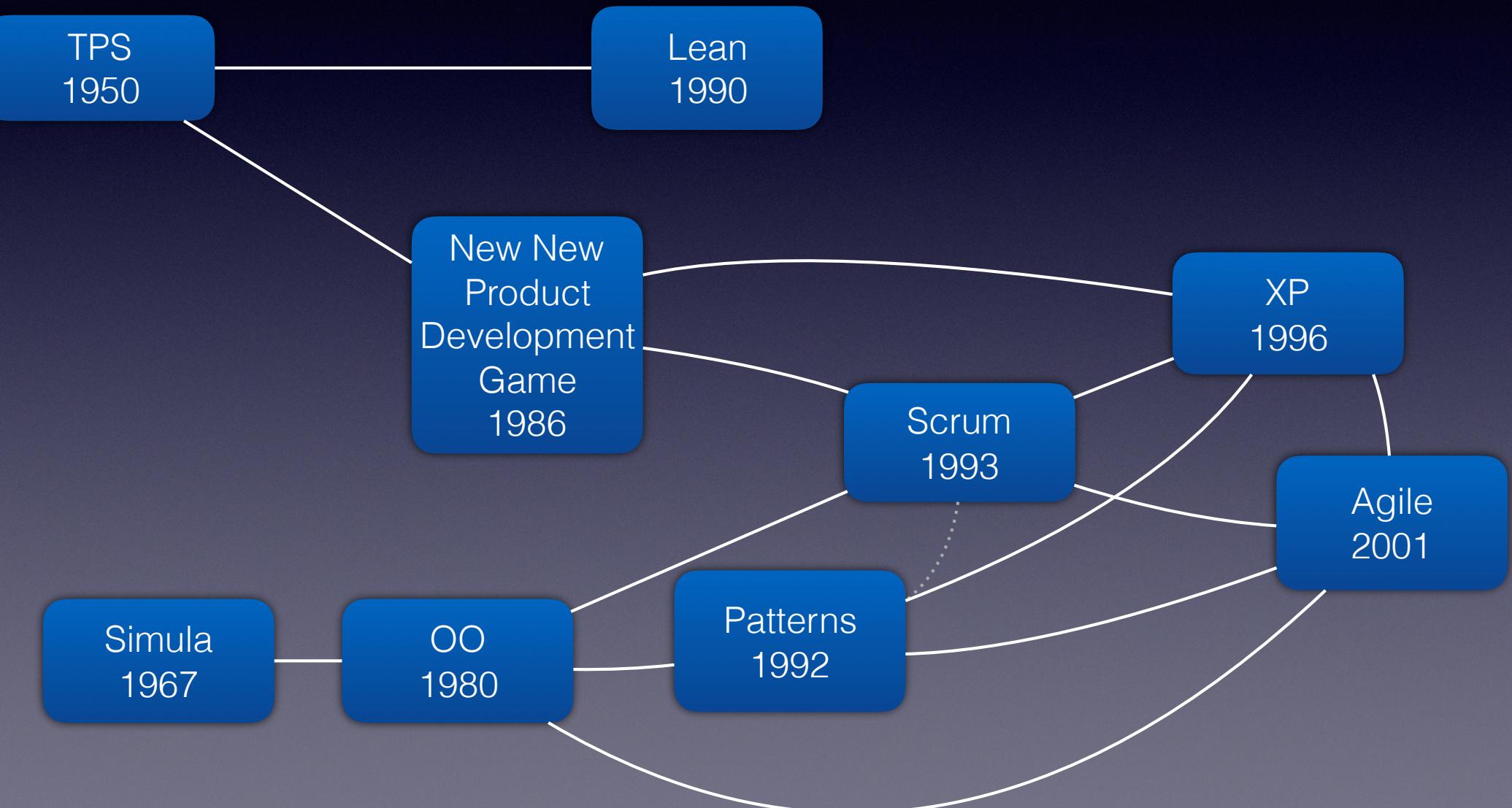


# The Dehumanisation of Agile and Objects

James O Coplien

# A Family Tree



# A Short History of XP

From: Kent Beck  
To: Jeff Sutherland <jsutherland>  
Reply: [70761.1216@compuserve.com](mailto:70761.1216@compuserve.com)  
Date: Mon, 15 May 1995 18:01:15 -0400 (EDT)  
Subj: HBR paper

---

Is there a good place to get reprints of the SCRUM paper from HBR? I've written patterns for something very similar and I want to make sure I steal as many ideas as possible.

Kent

# Another short History of XP

diction? How might it be possible to move from Test-Driven Development to Driven Development?

Kent Beck is best defined in terms of relationships. Following work by Jim Coplien and Ward Cunningham on software development process, with Ron Jeffries and the C3 team at Chrysler he invented and named Extreme Programming, resulting in the Jolt Productivity Award-winning *Extreme Programming Explained: Embrace Change*. He is the co-author of *Planning Extreme Programming* with Martin Fowler, with whom he also collaborated on *Refactoring: Improving the Design of Existing Systems*. With Ward Cunningham he wrote HotDraw, a widely copied drawing editor framework, pioneered patterns for software development, and popularized CRC cards. He channeled the Ancient Smalltalk Masters to produce *The Smalltalk Best Practice Patterns*, and is currently reviving a decades-old technique in *Test-Driven Development By Example* (Addison Wesley). He lives on a southern Oregon farm with a dwindling but still-impressive gaggle of children, his lovely wife Cindee, four dogs, and a bunch of chickens.

M. Marchesi and G. Succi (Eds.): XP 2003, LNCS 2675, pp. 459–462, 2003.  
© Springer-Verlag Berlin Heidelberg 2003

# The Agile Manifesto

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Kent Beck

- Mike Beedle
- Arie van Bennekum
- Alistair Cockburn
- Ward Cunningham
- Martin Fowler

James Grenning

- Jim Highsmith
- Andrew Hunt
- Ron Jeffries
- Jon Kern
- Brian Marick

Robert Cecil Martin

- Steve Mellor
- Ken Schwaber
- Jeff Sutherland
- Dave Thomas

© 2001, the above authors  
this declaration may be freely copied in any form,  
but only in its entirety through this notice.

# Nonaka's Text

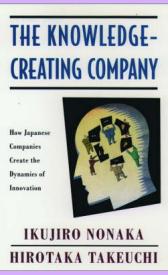
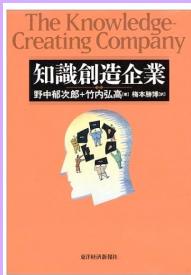
# Agile/Scrum (Software)

“The New New Product Development Game”

“Scrum”

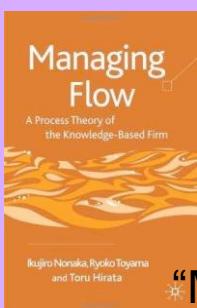
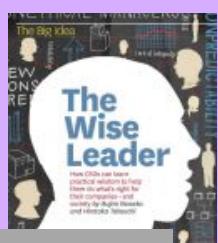
“The Knowledge Creating Company”(HBR)

SECI-model



Fractal Organization

アメリカ海兵隊(U.S. Marine)



Phronetic Leadership



“Managing Flow”  
“Wise Leadership”(HBR)

1986

1991

1993

1994/1

1994/2

1995

2001

2001

2008

2010

2012

2013

Org. Patterns(by Jim Coplien) (at PLoP)

First Sprint of Scrum by Jeff Sutherland

Second Sprint of Scrum (with Cope's Ideas)

Scrum Master

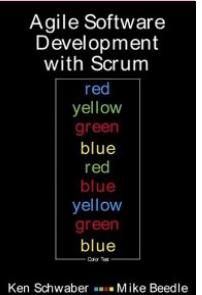
Daily Scrum

“Agile Software Development with Scrum”  
(by Ken Schwaber, Mike Beedle)

“The Agile Manifesto”



“Software in 30 days”

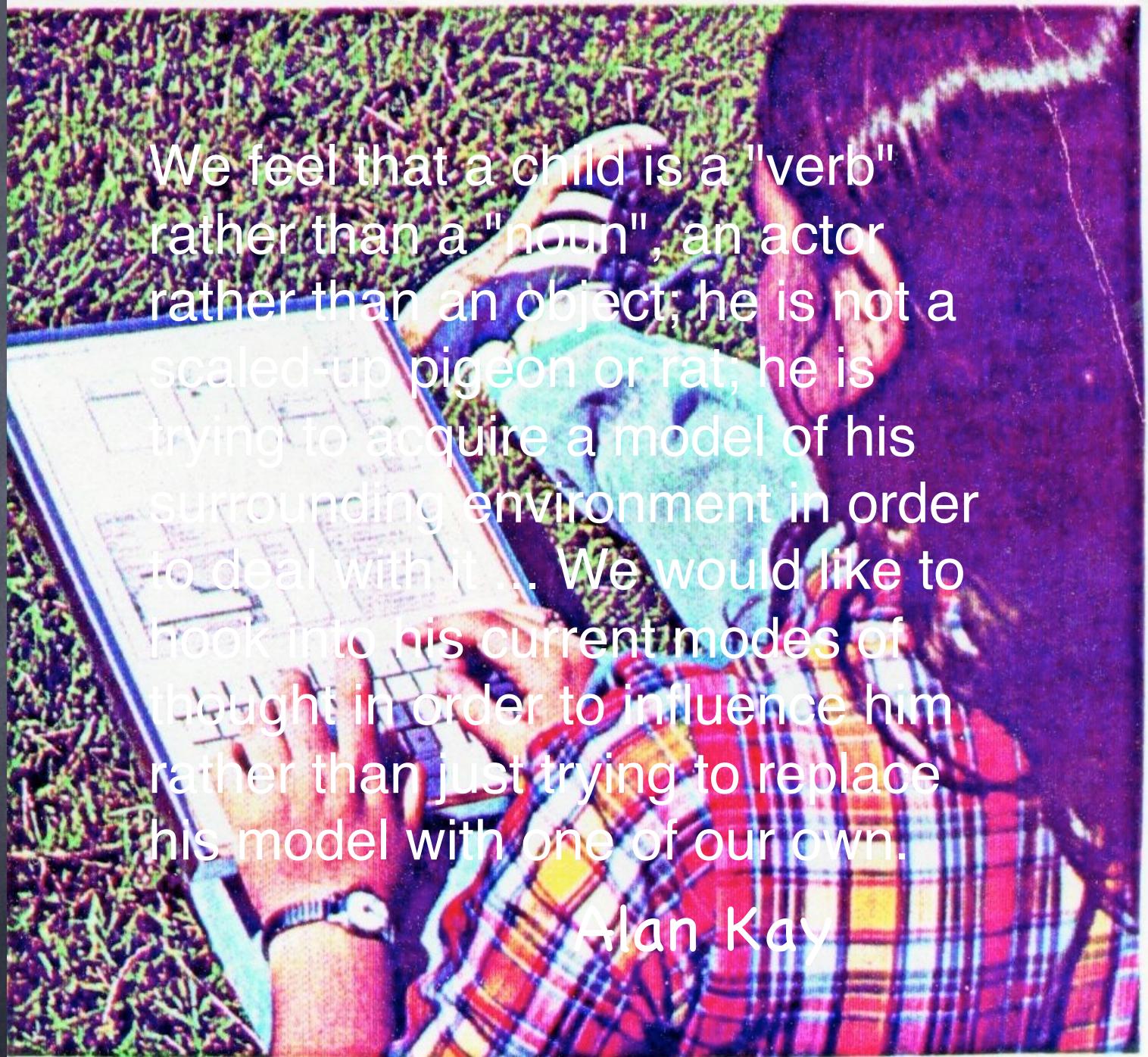


アジャイル開発とスクラム-顧客・技術・経営をつなぐ協調的ソフトウェア開発

Collaborative Software Development That Connects Customers, Engineers, and Management

# The Same Focus

	Agile / TPS	OO
About People	Individuals and Interactions	Anthropomorphic UC design, CRC cards
Ongoing Repair	Responding to Change	Programming by difference, new modularity
Outward Focus	Customer Collaboration	Engaging end users through their mental models
Quality	Working Software	TDD, unit testing,



We feel that a child is a "verb" rather than a "noun", an actor rather than an object; he is not a scaled-up pigeon or rat, he is trying to acquire a model of his surrounding environment in order to deal with it ... We would like to hook into his current modes of thought in order to influence him rather than just trying to replace his model with one of our own.

Alan Kay

Alan C. Kay  
Xerox Palo Alto Research Center

Abstract

This note speculates about the emergence of personal, portable information manipulators and their effects when used by both children and adults. Although it should be read as science fiction, current trends in miniaturization and price reduction almost guarantee that many of the not nea

teacher? Maybe. But first, it must decide that it is a necessary and desirable goal to do so.

What we would like to do in this brief note is to discuss some aspects of the learning process which we feel can be augmented through technological media. Most of the notions have at their root a number of theories about the child

Piaget's and others' work on the bases and forms of children's thought is a fairly convincing argument for believing that computers are an almost ideal medium for the expression of a child's epistemology. What is an "operational model" if not an algorithm, a procedure for accomplishing a goal? Algorithms are fairly informal and not necessarily logically consistent (as anyone who has ever spent a few hours debugging a program well knows). This fits in well with the child's viewpoint which is global and interested in structure rather than strict implication of "truths". On the other hand, the

model of themselves, what it is that they are trying to communicate, and what the child's current model of the situation is), to those well-intentioned people who would like to teach (but lack talent), to those who take it as a job, or worse, drifted into it because "ed" was the easiest way through college and now begrudge their fate through their young charges.

model of themselves, what it is that they are trying to communicate, and what the child's current model of the situation is), to those well-intentioned people who would like to teach (but lack talent), to those who take it as a job, or worse, drifted into it because "ed" was the easiest way through college and now begrudge their fate through their young charges.

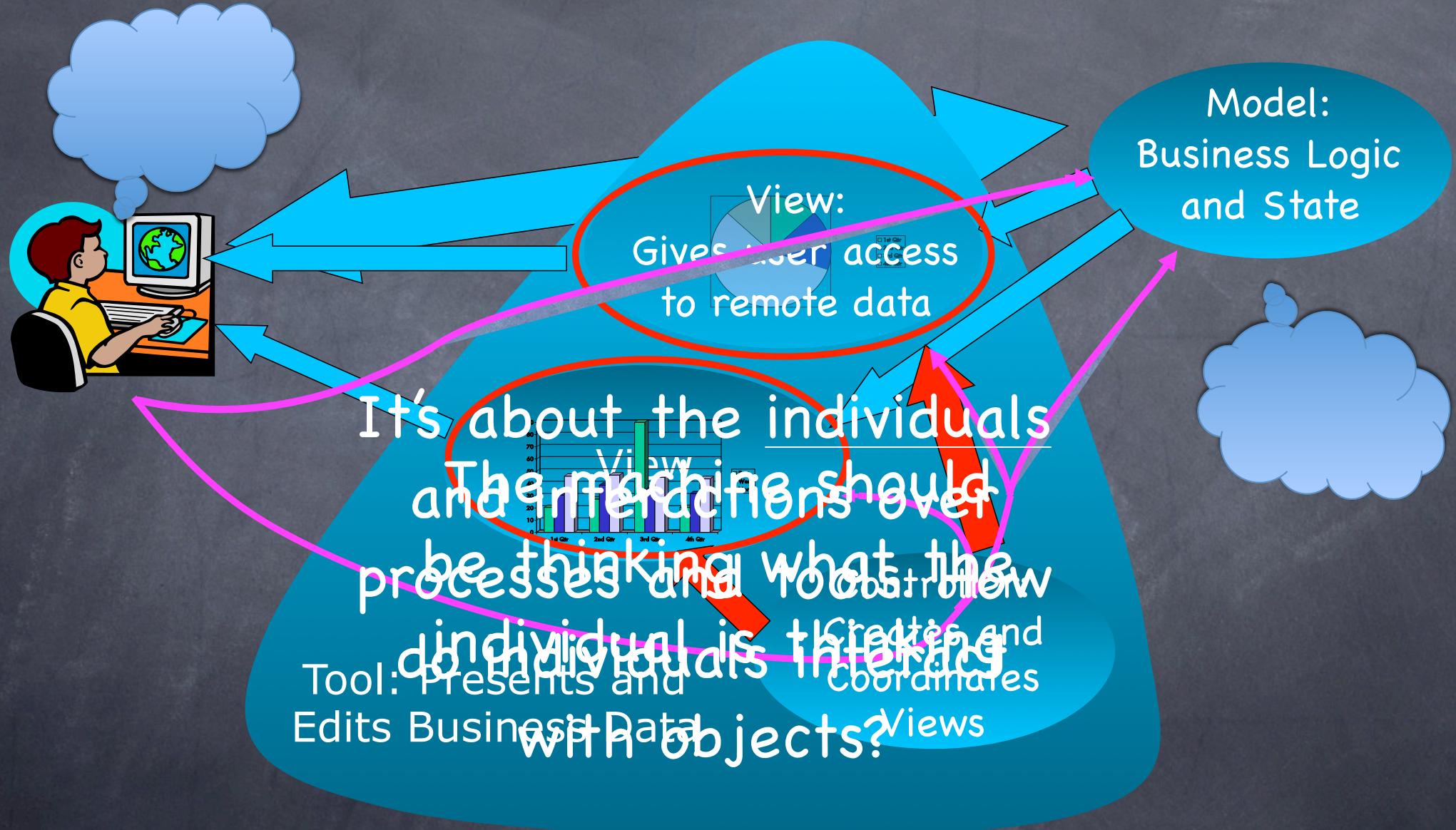
book did, however, allow centuries of human knowledge to be encapsulated and transmitted to everybody; perhaps an active medium can also convey some of the excitement of thought and creation!

Technologists point out that at least the bottom categories would be eliminated through a teaching machine. What they seldom understand is that what is delivered is a box which at best squarely fits the middle category: well intentioned but lacking talent! Can technology ever deliver a box with the middle category?

# Genetic Epistemology

- “... knowledge, particularly in the young child, is retained as a series of operational models, each of which is somewhat ad hoc and need not be logically consistent with the others. (They are essentially algorithms and strategies rather than logical axioms, predicates and theorems [operational rather than predicative]). It is much later in development that logic is used and even then through extralogical strategies.”

# The Ghost in the Machine



# Where have objects gone wrong?

- Most programmers program in classes
- Classes are static; we sell use cases
- We no longer can understand what our programs do!
- The GUI comes later or is separated in a client/server architecture — so users don't understand what our programs do, either
- ... And use cases aren't designed, so users can't understand them from the interface, either (nor can the coders understand them from the code)

# OO has lost its roots

- There is little reasoning and few methods that focus on “networks of cooperating objects” (DCI is an exception)
- Microservices could be kind of right except everyone is looking at the wrong end
- Focus on unit testing and object integrity has replaced “operational models” and the human focus
- Patterns, which could have opened this up, have become pre-packaged technical solutions

# ... and the good parts are little understood

- What part of MVC handles input events?
- What is the role of the Controller?
- How many of you draw object diagrams or do use cases?
- Languages do not support multiple classifications according to behaviour (roles) — only structure (multiple inheritance) or impoverished APIs (interfaces)

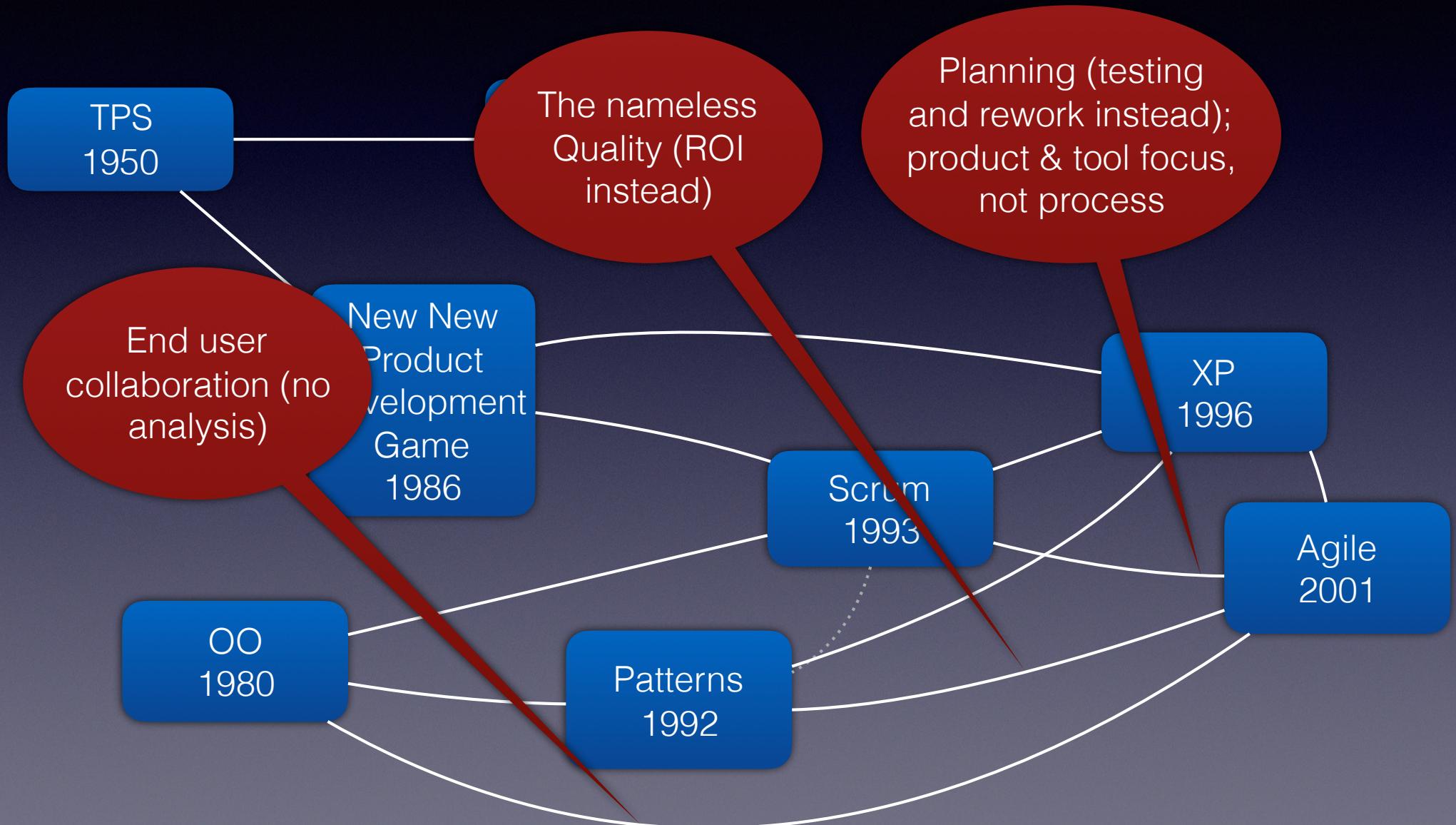
# ... and it matters

- “We observed correctness, time consumption, and locality of reference during reading comprehension tasks. We present a method which relies on the Eigenvector Centrality metric from Social Network Analysis to study the locality of reference in programmers by inspecting their sequencing of reading language element declarations and their permanence time in the code. Results indicate that DCI code in **trygve** supports more comprehensible code regarding correctness and improves the locality of reference, reducing context switching during the software discovery process.”
- Research by Héctor Valdecantos et al.

# Where has Agile gone wrong?

- People think it means “fast”
- Social design with CRC cards has been replaced by individual or pairwise testing
- Very little about the user mental model
- Processes and tools (e.g., Jira, testing) over individuals and interactions: it’s about programming
- Certification over customer collaboration (how many UX people here?)

# Agile has lost its roots



# ... and the good parts are little understood

- What's the purpose of the Daily Scrum?
- What percentage of its Sprints should a great Scrum team fail?
- What is the best bug tracking strategy?
- How do platform Scrum teams interface with other Scrum teams?

# Where we are today

	Agile / TPS	OO
About People	Individuals and Interactions	Anthropomorphic UC design, CRC cards
Ongoing Repair	Deferring to the last responsible Change moment	Programming by Refactor & difference, new test quality in modularity
Outward Focus	Customer-driven Collaboration	Engaging end users through their Design Patterns mental models
Quality	Working Software Certification	User-Centred Testing TDD, Unit testing

# The Common Enemies

- Tools over people
- Commercialisation over Doing the Right Thing
- Commoditisation
- Over-simplification
- Ignorance and Desperate Hope



# Toyota is becoming more efficient by replacing robots with humans

By Max Nisen | @MaxNisen | April 7, 2014



Let's get a guy with a hammer instead. *Reuters/Mick Tsikas*

Car makers have embraced automation and replaced humans with robots for years. But Toyota is deliberately taking a step backward and replacing automated machines in some factories in Japan and creating heavily manual production lines staffed with humans, according to [Bloomberg](#).

# Restoring Focus

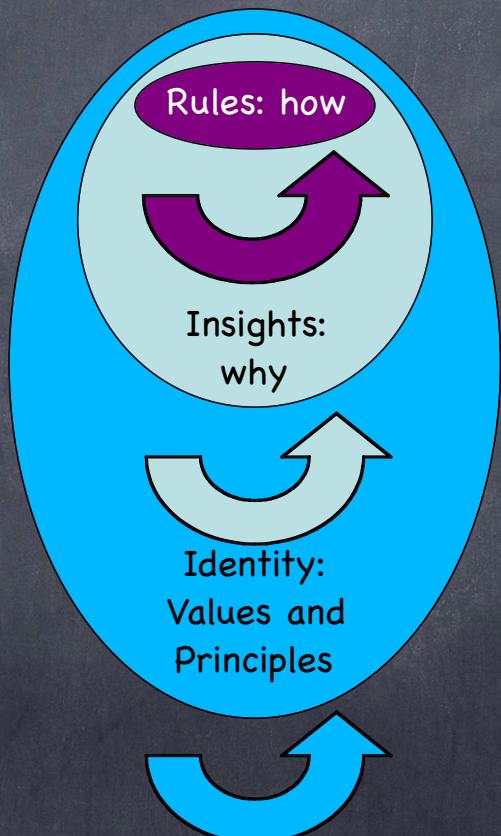
	Agile / TPS	OO
About People	Individual Interaction	Mob Programming
Ongoing Repair	Responsive Change	Organizational Learning: beyond Process to Structure
Outward Focus	Customer Collaboration	UX Knowledge
Quality	Working	Balance of Process and Product Focus

# Elusive lands over the horizon

- End-user programmability
- The Design Movement

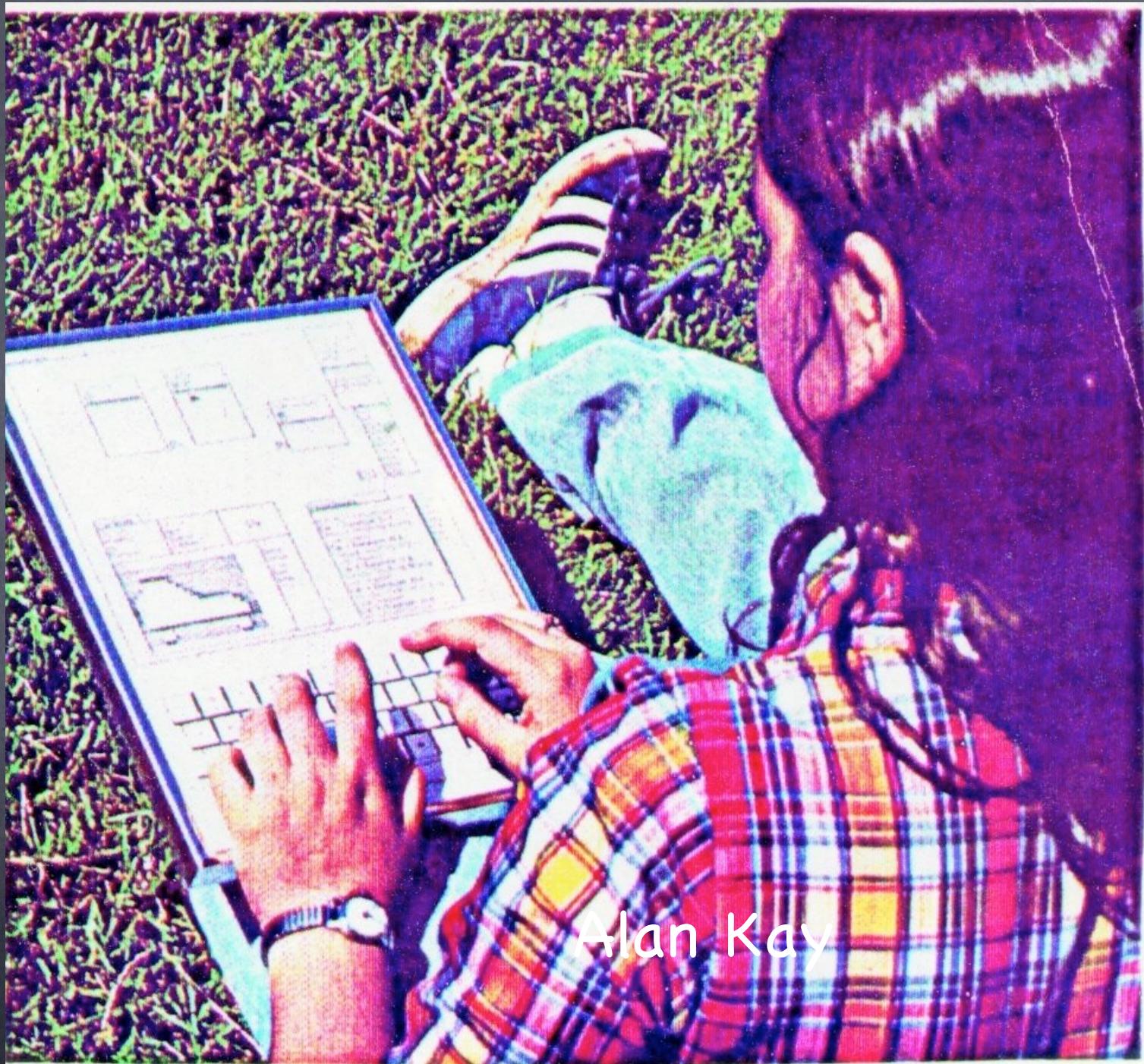
# Organizational Learning Models

- ⦿ Single-loop learning
  - ⦿ The rules change
  - ⦿ Strategy, structure, culture untouched
  - ⦿ Change the process
- ⦿ Double-loop learning
  - ⦿ Knowledge and Understanding
  - ⦿ Change the structure
- ⦿ Triple-loop learning
  - ⦿ Values and Principles
  - ⦿ Raison-d'être of an organization
  - ⦿ Change slowly



# Failure to Dream

- We tend to be risk-averse
- We tend to work within areas we can control
- Both networks of objects and teams of individuals take us beyond the scope of individual control
- Dream to do great things



Alan Kay

This book is dedicated to Nobel Laureate Muhammad Yunus and the Grameen Bank for originating microenterprise development and the Accion International President's Advisory Board, responsible for much of microenterprise development in the western hemisphere.

The strategy for bootstrapping the poor out of poverty has been a model for freeing hundreds of thousands of software developers from developer abuse caused by poor management practices.

# What can you do?

- At this conference, for each talk, ask: Was there compelling evidence for the claims that this is better than anything else, or that it even works?
- At this conference, for each talk, ask: How does this improve the quality of life, and for how many people?
- Take home at least five ideas that will help your team increase the human value of your product or service