

HISTORY · LIVING DOCUMENT

Knowledge Engineering: Lineage, Principles, and Evidence

LARRY TODD WILSON

REVISED OCTOBER 18, 2026

CONTENTS

01	How to read this
02	My story: learning how to learn (1983–1989)
03	The story in one arc
04	Purpose
05	Where knowledge engineering came from
06	How knowledge engineering was learned
07	Map to principles
08	Principle 1: Experts know more than they can tell
09	Principle 2: Acquisition is the work, not the obstacle
10	Principle 3: Knowledge belongs at the point of performance
11	Principle 4: Knowledge needs a shared structure, grounded in memory and language
12	Principle 5: Govern the system from inside
13	Principle 6: The model is the substrate, not the authority
14	What I added
15	Evidence ledger
16	References

My work rests on the foundation of knowledge engineering, Feigenbaum's knowledge principle that knowledge is the source of power, and joins six ideas built on it into one system. Experts know more than they can tell, so knowledge is elicited during the work, delivered at the point of performance, and typed by the language it is spoken in. The system is governed from inside, and the language model is its substrate, not its authority. This document sets each of the six principles beside the research it stands on, names what I added, and states the evidence for it.

HOW TO READ THIS

Each principle has five parts: the claim, the mechanism, the lineage it draws on, what I added, and the evidence. Labels mark the support for each claim: every principle's evidence, every timeline entry and every ledger row carries one of three labels, and the story and the purpose each carry one label for the whole section.

LABEL	MEANING
Established	Published research supports it. A linked reference is given.
Practitioner record	My own delivered work supports it. The record is my account and is not yet published.
Proposed	My claim, designed but not yet tested.

Client particulars are omitted throughout. The method is mine by agreement; client content remains confidential.

MY STORY: LEARNING HOW TO LEARN (1983–1989)

In 1983, I bought every issue of *Byte* magazine I could find. Each one was thick, and I studied the covers, trying to anticipate what each new technology was and what it might be able to do. I kept hoping for some kind of magic that would let me think like an electrical engineer. It never came. I could understand what the machines were for, but not how the hardware worked.

What caught my eye was who was building them. IBM had built its PC largely from off-the-shelf parts, and small companies were already building compatible machines from the same kinds of parts: Columbia Data Products in 1982, then Eagle and Compaq. I did not understand how to build a

computer, but I could infer the power of computing that so many builders were putting within reach, and its potential to augment human intellect. I did not yet have the word for that potential.

I could do something else. While still a student, I took a job selling microcomputers from HP, Compaq, Kaypro and IBM. Then I installed the business school's first computer lab at Auburn University, in Tichenor Hall. Its IBM XT's ran DOS 2.0, and I taught the business professors to use WordPerfect, Lotus 1-2-3 and Peachtree Accounting on them.

I graduated from Auburn in 1984 with an undergraduate degree in marketing. In practice, it was a degree in marketing research, which meant multivariate analysis, which meant statistics. I learned a great deal about regression analysis, and I knew that I did not want a career in marketing research and statistics.

As graduation came, I sat in a moment of real panic. I was about to go out into the job market without knowing my occupation or how to begin a career. In that moment, I had an epiphany, an immediate and visceral certainty: if I learned how to learn, I would not face that kind of stress again.

Learning how to learn was the genesis of what followed. It gave new meaning to two courses I had taken, one in epistemology and one in metaphysics, and I began looking for where they met the emerging world of microcomputers. That search drew me to the decision sciences, and then to symbolic AI and expert systems.

From 1984 to 1989, I studied everything I could find. My new heroes were the pioneers of symbolic AI, and each left me a kernel of an idea. Newell and Simon treated a problem as a space of states, operators and goals, searched with rules of thumb. They studied reasoning through verbal protocols, in which people think aloud as they work, a method Ericsson and Simon later set out as protocol analysis. They also represented knowledge as production systems, rules that pair a condition with an action. Simon added bounded rationality, the recognition that people decide within limits of information, time and mental capacity. With Chase, he showed that chess masters perceive a position as a few meaningful chunks, patterns recognized from long experience. Feigenbaum set out the knowledge principle, that a system's power lies in its domain knowledge rather than in general reasoning, and he identified the acquisition of that knowledge from experts as the field's bottleneck. Minsky proposed frames, structures of expectation with slots and default values. Schank and Abelson proposed scripts, the stereotyped sequences of events by which people understand familiar situations. Shortliffe, Buchanan and their Stanford colleagues built MYCIN, an expert system that could explain its own reasoning, answering why it asked a question and how it reached a conclusion. These ideas later shaped my designs.

Chomsky mattered in another way. His generative grammar treated a language as a system of rules that produces its sentences, and it reshaped psycholinguistics, the study of how people produce and understand language. I later realized that his work had shaped how I listened for patterns in language. Those patterns became the ground of the Eight Ways of Knowing, several of which also fit with neural substrates (Principle 4).

In 1989, while writing my master's thesis at Auburn, I read Engelbart's 1962 report, *Augmenting Human Intellect: A Conceptual Framework*, and what I had sensed in 1983 and 1984 found its name: augmentation. The epiphany of 1984 had given me a purpose in life, and the thesis, *Establish, Organize, and Represent: The Foundation for Occupational Applications*, was where I first put that purpose on paper and produced something from it: software that would augment a knowledge worker during the work itself. The epiphany and the thesis anchor the timeline that follows.

This account is practitioner record.

THE STORY IN ONE ARC

I have practiced one idea for more than thirty years: expert knowledge can be elicited, structured, and delivered to people at the moment they do the work. After 1987, much of the AI field moved away from eliciting knowledge from experts. Language models have now made it central again, and my instruments carry it forward.

YEAR	WHAT HAPPENED	LABEL
1962	Engelbart's <i>Augmenting Human Intellect</i> frames computers as instruments that raise human capability. Engelbart is my inspiration and becomes my mentor.	Established (the report); practitioner record (the mentorship)
1984–1989	I study knowledge engineering and begin the thinking that leads to my 1989 thesis.	Practitioner record
1989	My thesis, <i>Establish, Organize, and Represent: The Foundation for Occupational Applications</i> , sets out a framework for knowledge workers using software applications to support performance. It is a Master of Science thesis at Auburn University (158 pages), directed by Charles A. Snyder in the Department of Management and listed in the December 8, 1989 commencement program. Engelbart reviewed it.	Established (the degree, title, director and date); practitioner record (Engelbart's review)
1992	I found LearnerFirst, which packages commercial electronic performance support systems, used by more than 4,000 organizations. Its titles include Benchmarking, Process Management, Implementing ISO 9000, Software Engineering Risk Management and Work Profiling. Its slogan is "Learn it, do it, document it, all at the same time."	Practitioner record
1990s	I work with Gloria Gery, who introduced electronic performance support in the late 1980s and defined it in her 1991 book. Our joint projects showed that knowledge at the point of performance can be delivered.	Practitioner record
1995	I deliver custom learning systems and applications to large organizations.	Practitioner record
1996	I begin documenting, extensively, how to make implicit knowledge explicit, and the method takes form as Knowledge Harvesting, built on five core types of knowledge: declarative, procedural, conditional, social and systemic. Over the following decade, each type is deepened and improved.	Practitioner record
1997	The Knowledge Harvesting® mark is registered in 2008 (U.S. Trademark Reg. No. 3,500,735), and the registration records first use in commerce on October 28, 1997. It is renewed in 2017.	Established

YEAR	WHAT HAPPENED	LABEL
1996–2025	Knowledge Harvesting serves hundreds of expert projects in energy, chemicals, pharmaceuticals, consumer goods, standards bodies, education and government, under nondisclosure agreements that reserve the method to me. I mentor knowledge management professionals from 1998, deploy the method globally through consultants in 2000, package it as Knowledge Harvesting Kits in 2002, build it into enterprise knowledge programs in 2004, and reintroduce Work Profiling as "Knowledge Harvesting Lite" in 2009.	Practitioner record
2001–2002	I take part in the National Science Foundation workshop on Converging Technologies for Improving Human Performance (December 3–4, 2001), on its panel on expanding human cognition and communication, and write "Accelerating Convergence of Biotechnology, Nanotechnology, & Information Technology" for its report. On that panel, Robert Horn proposes the Human Cognome Project. For about two years afterward, Horn and I work together on systemic thinking and on what he called thinking bigger thoughts.	Established (the workshop, panel and paper); practitioner record (the collaboration)
2008	Carl Frappaolo publishes "Implicit knowledge" in <i>Knowledge Management Research & Practice</i> . It introduces implicit knowledge to knowledge management as a third category beside tacit and explicit: knowledge that can be articulated but has not yet been. Frappaolo and I developed the term together.	Established (the publication); practitioner record (the co-development)
2022–2024	I move the method onto generative AI at knowledgeharvesting.ai, building AI assistants for elicitation and taking part in AI hackathons through 2023 and 2024.	Practitioner record
2025–2026	I codify my knowledge engineering practice as the Knowledge Harvesting Prompts: instruments that deliver real-time performance support.	Practitioner record

The seven-ideas map, below, traces the lineage by concept.

PURPOSE

My purpose is augmentation, not artificial general intelligence: governed knowledge as a route to human capability, with AI as substrate and partner.

- **Singulars.** I call my clients Singulars: the only people in their organizations who could figure it out. Their knowledge is what the instruments exist to honor and carry forward, especially the knowledge of elders and retirees.
- **Three layers of knowledge.** Doctrine (policy), practices (procedures) and states of consciousness (performance) can each be encoded. A community can then use its own structure as a shared language, with each other and with AI.
- **Lifting human conscious states.** This is a stated purpose of the instruments, not a side benefit. Conscious states are the province of people, and they evolve alongside AI. The aim is present in my 1989 thesis, which patterned its Tutor on mondo, the Zen dialogue between teacher and pupil, and described forward thinking as attention that runs from meditation to athletics. The instruments carry that aim forward.

- **Working principles.** In 2008 I wrote down the principles I work by: seek transparency and greater disclosure; validate by going upstream, back to the irreducible ideas; stay open and keep questioning; acknowledge and appreciate; be patient; get recursive; and apply the scientific method.
- **Benefit that can be shown.** Altruism is a design commitment to be evaluated, not a label. Each project should say who benefits, keep experts and affected people as partners rather than data sources, secure consent and attribution, trace every consequential claim to evidence with its limits visible, ask whose expertise is absent, name who can challenge and correct the model, and state what outcomes would count as improvement.
- **Scale without a successor.** The mission is to reinstall knowledge engineering as a prominent AI discipline that many people practice without me. Passing it to a single successor would count as failure.

These are my stated aims, not tested claims.

WHERE KNOWLEDGE ENGINEERING CAME FROM

Knowledge engineering grew out of symbolic AI, split into three branches after 1987, and is reconverging with the neural line now. I entered at the fork and carried elicitation through all three branches. Rows marked Wilson are practitioner record unless a reference is given; the 1989 thesis row and the 1997 trademark row are established by references 69 and 65. The rest are established, and most are listed in the references.

YEAR	WHAT HAPPENED	PEOPLE	LINE
1943	Neurons modeled as logical units, the start of the copy-the-brain line.	McCulloch, Pitts	Neural
1946–1953	The Macy conferences study feedback, control and communication across machines and people.	Macy conference participants	Cybernetics
1955–1956	The Dartmouth proposal and workshop name artificial intelligence: intelligence as symbols, rules and programs a person can read.	McCarthy, Minsky, Rochester, Shannon	Symbolic
1956	Human problem solving modeled as programs, beginning with the Logic Theorist.	Newell, Simon	Symbolic
1958	The perceptron, the first trainable network.	Rosenblatt	Neural
1962	<i>Augmenting Human Intellect</i> frames computers as instruments that raise human capability. Engelbart becomes Wilson's inspiration and later his mentor.	Engelbart	Augmentation
1965	DENDRAL infers molecular structure from chemists' rules. Domain knowledge, not general reasoning, drives performance: the knowledge principle.	Feigenbaum, Lederberg, Buchanan, Djerassi	Knowledge engineering

YEAR	WHAT HAPPENED	PEOPLE	LINE
1966	<i>The Tacit Dimension</i> : we can know more than we can tell.	Polanyi	Tacit knowledge
1969	<i>Perceptrons</i> argues the limits of single-layer networks. Neural funding falls sharply, though research continues.	Minsky, Papert	Neural
1970s	MYCIN encodes one expert domain, infection diagnosis, as rules.	Shortliffe, Buchanan	Knowledge engineering
1974	Backpropagation first described.	Werbos	Neural
1974	<i>Cybernetics of Cybernetics</i> places the observer inside the system observed.	von Foerster	Cybernetics
1976	TEIRESIAS lets experts transfer and debug knowledge directly.	Davis	Knowledge acquisition
1977	Knowledge engineering is named. Extraction from experts is named the field's bottleneck.	Feigenbaum	Knowledge engineering
1980	R1/XCON goes into use at DEC, configuring computer orders and saving millions a year.	McDermott	Knowledge engineering
1980	The Neocognitron, an early convolutional network.	Fukushima	Neural
Early 1980s	Feigenbaum co-founds Teknowledge and IntelliCorp. Corporations form AI groups.	Feigenbaum	Knowledge engineering
1982	Hopfield networks revive interest in neural computation.	Hopfield	Neural
1983	<i>The Fifth Generation</i> presents knowledge engineering as a national priority.	Feigenbaum, McCorduck	Knowledge engineering
1984	Cyc begins hand-encoding common sense, a project of decades.	Lenat	Knowledge engineering
1984–1985	Repertory-grid tools at Boeing draw out the distinctions experts use.	Boose	Knowledge acquisition
1974–1979	Edinburgh begins undergraduate AI teaching with AI2 (1974–75) and AI1 (1978–79). Stanford and Carnegie Mellon teach symbolic computation and problem-solving methods through exercises, mastery tests and integrated programming projects.	McCarthy, Newell, Bundy and colleagues	Teaching
1984–1989	Wilson studies knowledge engineering and begins the thinking that leads to his 1989 thesis.	Wilson	Wilson
1985	In an introductory expert-systems course at the University of Arizona, about half of students' project time goes to gathering knowledge from human experts and translating it into rules.	Bahill, Ferrell	Teaching
1986	The Banff Knowledge Acquisition Workshops gather a community around the bottleneck.	Gaines, Shaw, Boose	Knowledge acquisition
1986	Multi-layer networks become trainable through backpropagation.	Rumelhart, Hinton, Williams	Neural
1987	The specialized AI hardware market collapses and funding falls. Rule engines continue as business rules systems; ILOG is founded the same year.	—	Knowledge engineering

YEAR	WHAT HAPPENED	PEOPLE	LINE
1987	Methods of knowledge acquisition are surveyed, from interviews and observation to protocol analysis.	Olson, Rueter	Knowledge acquisition
1988	The knowledge engineer is described as a student, who watches and questions an expert and tests a model on selected cases.	Clancey	Knowledge acquisition
1988	In student expert-system projects, developing and debugging the knowledge base takes about 60% of the work and interviewing about 10%.	Bahill, Harris, Senn	Teaching
1989	The Critical Decision Method elicits judgment from people deciding under pressure.	Klein, Calderwood, MacGregor	Knowledge acquisition
1989	Wilson's thesis, <i>Establish, Organize, and Represent: The Foundation for Occupational Applications</i> , sets out software that guides knowledge workers through a task while they do the work, built on Engelbart's augmentation. It is a Master of Science thesis at Auburn University directed by Charles A. Snyder. Engelbart's review of it is practitioner record.	Wilson	Wilson
1991	<i>Electronic Performance Support Systems</i> delivers expert knowledge to workers at the moment of need.	Gery	Performance support
1992	LearnerFirst packages commercial electronic performance support systems, used by more than 4,000 organizations. Advertisements in <i>Quality Progress</i> present them to quality professionals as software for doing the work, not as training.	Wilson	Wilson
1993	Ontology defined as a shared, machine-readable specification.	Gruber	Ontology
1995	Tacit-to-explicit conversion becomes management practice.	Nonaka, Takeuchi	Knowledge management
1995	Wilson delivers custom learning systems and applications to large organizations.	Wilson	Wilson
1996	Knowledge Harvesting takes form as a method for making implicit knowledge explicit, working with five types of knowledge: declarative, procedural, conditional, social and systemic.	Wilson	Wilson
1997	The Knowledge Harvesting® mark is registered in 2008; the registration records first use in commerce on October 28, 1997.	Wilson	Wilson
1998–2004	Wilson mentors knowledge management professionals, deploys the method globally through consultants, packages it as Knowledge Harvesting Kits and builds it into enterprise knowledge programs.	Wilson	Wilson
1990s–2010s	Elicitation survives mainly in cognitive task analysis and consulting practice.	Clark, Feldon and others	Knowledge acquisition
2000	KADS becomes CommonKADS, a methodology for knowledge engineering and management.	Schreiber and colleagues	Knowledge engineering
2001–2002	At the National Science Foundation workshop on converging technologies, Horn proposes the Human Cognome Project; Wilson serves on the same panel and writes a paper for its report.	Horn, Wilson	Wilson

YEAR	WHAT HAPPENED	PEOPLE	LINE
2003	Wilson embeds the method in more than 100 online communities and captures guidance on effective online collaboration.	Wilson	Wilson
2008	Implicit knowledge introduced to knowledge management as a third category beside tacit and explicit.	Frappaolo, with Wilson	Knowledge management
2009	Work Profiling returns as "Knowledge Harvesting Lite," a scalable form of the method.	Wilson	Wilson
2012	The Knowledge Graph brings ontology work to everyday search.	Google (Singhal)	Ontology
2012	A neural network wins ImageNet, nearly halving the error rate.	Krizhevsky, Sutskever, Hinton	Neural
2016	Knowledge Harvesting Live demonstrates knowledge capture in real time.	Wilson	Wilson
2019	AIable.org offers a practical method for judging whether a work task is susceptible to machine learning.	Wilson	Wilson
2022	Knowledge Harvesting moves onto generative AI at knowledgeharvesting.ai.	Wilson	Wilson
2023	Hinton leaves Google to warn that no one knows how to control these systems.	Hinton	Neural
2023	ISO/IEC 42001 sets requirements for an AI management system, including monitoring, measurement and evaluation.	ISO	Governance
2024	GraphRAG grounds language models in knowledge graphs.	Edge and colleagues	Reconvergence
2025–2026	The Knowledge Harvesting Prompts deliver knowledge engineering as instruments.	Wilson	Wilson
2026	Wilson reconstructs the sequence of experiences by which knowledge engineering was learned.	Wilson	Wilson
2026	ISO 9001:2026, the sixth edition, published in September 2026, requires in clause 7.1.6 that an organization determine the knowledge necessary to operate its processes and achieve the intended results of its quality management system. It states: "This knowledge shall be retained, applied and shared to the extent necessary."	ISO	Governance

The fork after 1987. The line split three ways: performance support (Gery), knowledge management (Nonaka and Takeuchi) and ontology (Gruber). I carried elicitation through all three: performance support at LearnerFirst, knowledge management in Knowledge Harvesting, and ontology in the Eight Ways of Knowing.

The pioneers compared.

FIGURE	THEIR FOCUS	MY RELATION
McCarthy, Minsky	Intelligence as formal logic and symbol manipulation	Distant: I never aimed at general intelligence.

FIGURE	THEIR FOCUS	MY RELATION
Newell, Simon	Human problem solving implemented as programs	Partial: I model expert reasoning, but to transfer it to people, not as cognitive theory.
Feigenbaum	Named knowledge engineering; power lies in domain knowledge	Closest: I practice his central claim.
Shortliffe, Buchanan	One expert domain encoded as rules (MYCIN)	The same unit of work: one expert, one domain.
Davis	Tools for experts to transfer knowledge (TEIRESIAS)	Direct overlap with elicitation.
Lenat	Common sense hand-encoded at scale (Cyc)	Shared persistence after the field moved on.
Klein	Elicitation anchored on critical incidents	Parallel line; see Principle 2.
Gery	Electronic performance support	Collaborator on unpublished projects. I reached the same idea independently, through Engelbart's augmentation, in my 1989 thesis.
Engelbart	Augmenting human intellect	Inspiration, mentor, and reviewer of my thesis, which is built on his augmentation.
Frappaolo	Implicit knowledge	Co-developer of the term with me.
Horn	Information Mapping; visual language; the Human Cognome Project	Fellow panelist at the 2001 workshop and collaborator afterward on ways of thinking bigger thoughts.

Two differences define my place. Much of the field shifted its focus after 1987, while communities such as CommonKADS and cognitive task analysis kept elicitation central; I built one practice that carried elicitation across performance support, knowledge management and ontology. The pioneers published and were cited; my work was held under nondisclosure agreements, which is why most of my record is not yet citable.

HOW KNOWLEDGE ENGINEERING WAS LEARNED

In 2026 I set out to recover the reading and programming assignments that the pioneers of AI gave their students between about 1965 and 1995. The evidence answered a better question: by what sequence of experiences did a person become a knowledge engineer? No single curriculum was shared by Stanford, Carnegie Mellon, Edinburgh and the other centers. The course and project facts below are established where a reference is given. The sequence itself is my reconstruction from converging sources, and it is labeled Proposed.

Formation began before elicitation. Learners first became fluent in symbolic computation, as in John McCarthy's Stanford course Computing with Symbolic Expressions. They then applied problem-solving methods to concrete problems, as in the self-paced course that Allen Newell, John McDermott and Charles Forgy taught at Carnegie Mellon, where students graded their own mastery tests, discussed their errors and tested again. Edinburgh began undergraduate AI teaching with AI2 in 1974–75 and AI1 in 1978–79.

Expertise entered with the expert. In the University of Arizona's 1985 expert-systems course, about half of students' project time went to gathering knowledge from human experts and translating it into rules. The instructors concluded that students who acted as their own experts, or who worked only from books, missed a valuable part of the experience (Bahill and Ferrell (1986)). A later report on student projects put the average project at about 100 hours: 15% learning the subject, 10% interviewing the expert, 60% developing and debugging the knowledge base, and 15% testing (Bahill, Harris and Senn, 1988). The interview was a small part of the work. In these projects, most of the work was finding and repairing what the knowledge base did not yet know. Established.

The knowledge engineer as student. Clancey (1988) described the knowledge engineer as an active learner who watches and questions an expert and tests a model, incrementally, on selected cases. The decisive skill is to detect a gap and turn it into a good question. Established.

Thirteen formative experiences. Proposed.

STAGE	EXPERIENCE	CAPABILITY FORMED
1	Manipulate symbolic structures computationally	Treat knowledge as representable and executable
2	Apply different problem-solving methods to concrete tasks	Build a repertoire of reasoning methods
3	Represent the same situation in several formalisms	See that representation is consequential
4	Attempt, inspect, discuss, correct and retry	Treat error as evidence
5	Integrate several methods in one sustained project	Exercise architectural judgment
6	Study and reconstruct working intelligent systems	Recognize reusable reasoning architectures
7	Place each technique within the wider field	Gain conceptual orientation
8	Enter a real domain and form a provisional model	Become capable of informed inquiry
9	Work directly with a human expert	Meet expertise as situated human judgment
10	Translate expertise into an executable representation	Make understanding explicit enough to test
11	Debug the knowledge base	Find what the represented understanding lacks
12	Test with cases and expose the model to expert criticism	Turn performance into evidence about knowledge
13	Diagnose a knowledge gap and form the next question	Become a self-directing knowledge engineer

Before stage 8, the learner is mainly becoming competent in AI methods. Once the learner builds, tests and revises a model of another person's expertise, the learner is becoming a knowledge engineer.

The core cycle. Within the thirteen experiences runs one recurring cycle: form a provisional model, expose it to the expert, represent it so that it can run, test it on cases, find where it fails, determine what is not yet understood, return to the expert with a better question, and revise. Three sources of resistance drive it. The domain resists the provisional model, the expert resists the representation, and the working system resists both. Proposed.

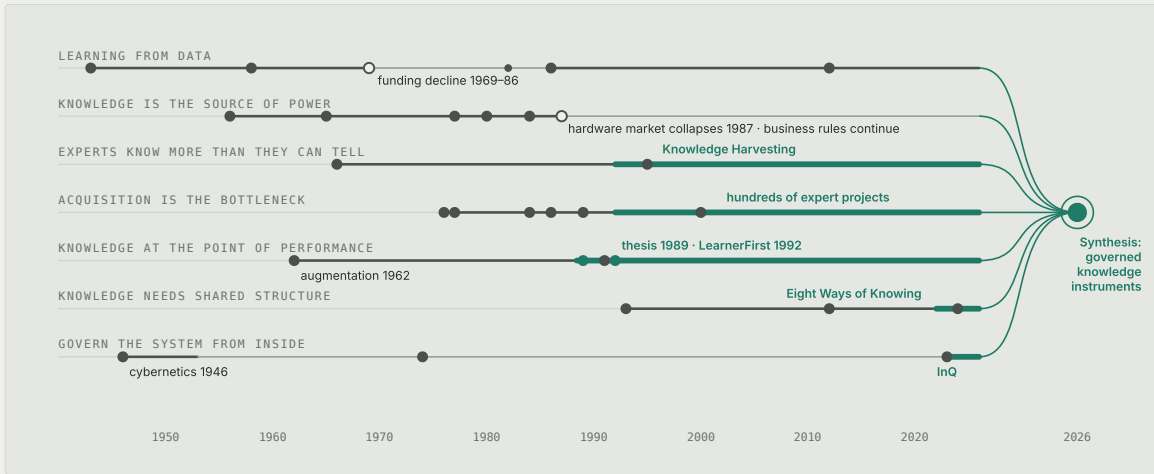
Three modes of inquiry. A literature cross-check I made in January 2026 proposes three modes for the next question. Episode-oriented inquiry asks what happened in one case and what the expert noticed. Process-oriented inquiry reconstructs the steps actually taken and where they departed from the usual path. System-oriented inquiry asks which conditions interact to produce an outcome. The mode is chosen by the gap in the model, not by the expert's seniority. The three modes match three of the Eight Ways of Knowing: Episodic, Procedural and Systemic. Proposed; the research literature does not establish this triad as a standard.

A working definition. Knowledge engineering is the disciplined development of usable, testable representations of expertise through domain study, appropriately chosen inquiry, implementation, case-based criticism, diagnosis of gaps, and revision. Its social value depends on the purposes served, the people involved, and the evidence of its actual effects. Proposed.

MAP TO PRINCIPLES

The seven-ideas map has seven lanes; this document has six principles. The first lane is the foundation the other six build on, so it appears as lineage rather than as a principle of its own.

Seven ideas, *and where they meet*



Larry Todd Wilson · Itwilson.com

— Knowledge Harvesting practice — reduced activity

Each lane is one idea. A thick green line marks where Knowledge Harvesting practice carried the idea, and a thin grey line marks reduced activity. All seven lanes meet in one synthesis: governed knowledge instruments.

MAP LANE	WHERE IT APPEARS HERE
Knowledge is the source of power	Lineage: DENDRAL, Feigenbaum, XCON and Cyc, above
Experts know more than they can tell	Principle 1
Acquisition is the bottleneck	Principle 2
Knowledge at the point of performance	Principle 3
Knowledge needs shared structure	Principle 4
Govern the system from inside	Principle 5
Learning from data	Principle 6: the model that learning from data produced

Principle 1: Experts know more than they can tell

Claim. Experts omit most of the decisions they make when they describe complex work. Cognitive task analysis research puts the figure at about 70% of decisions. Established.

Mechanism. Practiced skill becomes automatic, so it runs without conscious access. People also report on their own thinking less accurately than they believe. Established.

Lineage.

- Polanyi (1966) named the tacit dimension: we can know more than we can tell. Knowledge management later read his work as a two-way split between tacit and explicit knowledge.
- Frappaolo (2008) refines that split with a third category, implicit knowledge: knowledge that can be articulated but has not yet been. Carl Frappaolo and I developed the term together; the published paper is Frappaolo's.
- Nisbett and Wilson (1977) showed that people often cannot report the causes of their own judgments.
- Anderson (1982) explained how knowledge becomes proceduralized and leaves conscious report.
- Yates, Feldon and Clark (2011) and Feldon and Clark (2006) report that experts omit about 70% of decisions.

What I added. I treat the omission as the work itself and design elicitation to recover it in every session. This is what I call the 70% problem. With Carl Frappaolo, I introduced implicit knowledge as a third category beside tacit and explicit, and Knowledge Harvesting targets both tacit and implicit knowledge.

Evidence. The 70% figure is established: it was first reported in a conference workshop (Yates, Feldon and Clark, 2011) and is supported by a peer-reviewed study of surgical teaching (Sullivan and colleagues, 2014). My practitioner record is that structured elicitation recovers decisions an ordinary interview misses.

Principle 2: Acquisition is the work, not the obstacle

Claim. Elicitation tied to one real activity, done while the work is being done and confirmed by the holder, recovers more usable expertise than an abstract interview.

Mechanism. Verbal reports are most accurate when given during the task. Recall anchored on a real incident reaches judgment that general questions miss. Patterns the holder confirms in their own words are more trustworthy than patterns an analyst infers.

Lineage.

- [Feigenbaum \(1977\)](#) and his group named knowledge acquisition the field's bottleneck.
- [Ericsson and Simon](#) showed that concurrent verbal reports are valid data when they report what the person is attending to.
- [Klein, Calderwood and MacGregor \(1989\)](#) anchored elicitation on critical incidents in the Critical Decision Method.
- [Hoffman, Shadbolt, Burton and Klein \(1995\)](#) compared elicitation methods empirically.
- [Tofel-Grehl and Feldon \(2013\)](#) found a large effect for training built on cognitive task analysis (Hedges' $g = 0.871$).
- [Olson and Rueter \(1987\)](#) surveyed direct and indirect methods of knowledge acquisition.
- [Bahill and Ferrell \(1986\)](#) found that about half of students' project time went to working with human experts; [Bahill, Harris and Senn \(1988\)](#) found that developing and debugging the knowledge base took about 60% of project time, and interviewing about 10%.
- [Clancey \(1988\)](#) described the knowledge engineer as a student who learns by testing a model on cases and asking the next good question.
- [Birt and colleagues \(2016\)](#) examined member checking, the qualitative-research practice of returning findings to participants.

What I added.

- One session holds one critical knowledge area, named as an activity.
- Eight modules carry the work: focus, find, elicit, organize, package, apply, evaluate, adapt. Focus yields the critical knowledge areas.
- Activity, Task and Step decomposition with seven atomic verbs gives the work a grammar. The verbs are proprietary and are not listed here.
- A PDSA loop runs inside the session: learn it, do it, check it, document it, at the same time. It descends from LearnerFirst's 1992 slogan, "Learn it, do it, document it, all at the same time."

LearnerFirst's advertisements in *Quality Progress*, the flagship magazine of the American Society for Quality Control, used it to show quality professionals that the products were not training products but software for accomplishing the work.

- A pattern returns to the holder only as a tentative observation in the holder's own words. Nothing is built on it until the holder confirms it. This is member checking built into the instrument.
- One elicitation engine, expressed four times: Tutor (1989; reference 69), Builder-Player (documented as patentable subject matter, not patented; built first in Borland Delphi and later in JavaScript), the Knowledge Harvesting steps, and the 96 cells. The 1989 Tutor led the user through a task with questions and cues, in a manner the thesis compared to the Socratic method.

Evidence. The research basis is established. My practitioner record is hundreds of expert projects. A proposed test, the Expert Judgment Experiment, applies Mitchell's evaluation principles by freezing a competence model (cue, condition, judgment) and testing whether it predicts the expert's decisions before they are made, checked by a second expert who never attended the sessions.

Principle 3: Knowledge belongs at the point of performance

Claim. Knowledge engineering reaches people at scale when its skill is carried inside instruments they run while they work, not taught as a practice they must first learn.

Mechanism. A practice needs trained practitioners, which recreates the bottleneck. An instrument carries the ontology, the questions, the criteria and the governance, so any knowledge worker gets them at the moment of need. Learning, doing, checking and documenting happen as one act.

Lineage.

- Engelbart (1962) framed computers as instruments that augment human intellect. Engelbart was my inspiration, became my mentor, and reviewed my 1989 thesis.
- Gery (1991) defined electronic performance support, a term she introduced in the late 1980s: expert knowledge delivered to the worker at the moment of need.

What I added. This is the lane where my record is longest.

- **1989.** My thesis, *Establish, Organize, and Represent: The Foundation for Occupational Applications*, set out software that guides a knowledge worker through a task during the work itself, with a Tutor whose support the user can adjust from full to none. The software was to support the user's own capabilities, not replace them. I reached this idea through Engelbart's augmentation, independently of and at the same time as Gery's early work on electronic performance support.
- **1992.** LearnerFirst delivered commercial electronic performance support systems, used by more than 4,000 organizations.
- **1990s.** I worked with Gloria Gery. Our joint projects found that knowledge at the point of performance can be delivered. That work is unpublished.
- **2025–2026.** The Knowledge Harvesting Prompts codify my knowledge engineering experience as instruments that deliver real-time performance support. They are today's form of the electronic performance support system: the skill of a knowledge engineer, carried inside an instrument that any knowledge worker can run with a commercial language model.

Evidence. Practitioner record, and strong: a design documented in my 1989 thesis, followed by thirty-four years of delivered performance support, from LearnerFirst to the Knowledge Harvesting Prompts. Published evidence for electronic performance support in general is mostly case studies. To make my record citable, assemble the LearnerFirst documentation (products, customer counts, dates), record the work with Gery, and publish one case of an instrument used without me present.

Principle 4: Knowledge needs a shared structure, grounded in memory and language

Claim. Expert knowledge comes in the Eight Ways of Knowing. Each has a distinct pattern in language, and several correspond to established memory systems. A fixed set of eight, crossed with twelve support information types, gives people and AI one structure to share.

Mechanism. People turn what they know into speech through the forms their language offers. Dan Slobin called this thinking for speaking. Each way of knowing surfaces in its own question and its own grammar: what is so, how it is done, when it applies, who is involved, what the body registers, what the knower notices about their own knowing, why it holds, and what happened once. Asking in the matching form brings out knowledge a single kind of question misses.

Lineage. Two lines meet here.

- Memory science. [Squire \(2004\)](#) sets out declarative memory (semantic and episodic) and nondeclarative memory (including procedural). Tulving (1972) separated episodic from semantic memory.
- Language and learning. Psycholinguistics, shaped by Chomsky's generative grammar (1957), studies how people produce and understand language; Slobin's thinking for speaking belongs to it. Paris, Lipson and Wixson (1983) named conditional knowledge: knowing when and why. Flavell (1979) named metacognition. The revised Bloom taxonomy ([Kathwohl, 2002](#)) types knowledge as factual, conceptual, procedural and metacognitive. CommonKADS and Gruber's ontology work gave knowledge engineering its shared structures.

WAY	QUESTION	MEMORY GROUNDING	LANGUAGE GROUNDING	LABEL
Declarative	What	Semantic memory (Squire, Tulving)	Definitions, predication	Established
Procedural	How	Procedural memory (Squire, Anderson)	Imperatives, sequences	Established
Conditional	When	Condition-action rules (Anderson); named by Paris et al.	Conditional clauses: if, when, unless	Established in learning research
Social	Who	Mentalizing network; not a memory system	Person reference, speech acts	Proposed
Embodied	Body	Interoception; findings mixed	Embodied metaphor (Lakoff and Johnson)	Proposed
Metacognitive	Awareness	Metacognition (Flavell)	Epistemic modality: I think, I'd check	Established

WAY	QUESTION	MEMORY GROUNDING	LANGUAGE GROUNDING	LABEL
Systemic	Why	No distinct substrate identified	Causal and purpose connectives	Proposed
Episodic	Story	Episodic memory (Tulving, Squire)	Narrative structure (Labov)	Established

The twelve support information types are linguistic. They are functional categories, and each matches a known relation in discourse. Rhetorical Structure Theory (Mann and Thompson, 1988) names relations such as purpose, cause, condition, evidence and elaboration. Robert Horn's Information Mapping (Horn, 1976) showed that information can be organized by type, independent of its subject. Horn and I served on the same National Science Foundation panel in 2001 and continued afterward to explore ways of thinking bigger thoughts; I am grateful for those conversations. My twelve support information types are proprietary and are not listed here.

What I added. The Eight Ways of Knowing are the types of knowledge I have taught for decades, under a new name. For many years there were five core types: declarative, procedural, conditional, social and systemic. They were part of the Knowledge Harvesting methodology licensed to organizations, and dozens of knowledge management professionals learned the stages of Knowledge Harvesting with them. A history I wrote in 2008 records how the work on the types deepened: procedural knowledge was found to work at several levels in the mid-1990s; declarative knowledge was extended in the late 1990s, so that learners could approach the same knowledge from several angles; contextual knowledge, for interpreting signals, was added by 2000 and later renamed integral knowledge; and more was learned about social knowledge in 2006. My work with Robert Horn after 2002 deepened systemic knowledge. That history also adds levels and associations as dimensions beyond type. Asking by type of question goes back to my 1989 thesis. Its Tutor started from language and grouped the user's answers by question: how and why for structure and cause, what for the task and its parts, where for space, when for time, and which and how much for attributes. Information is tangible because it has been verbalized; it can be measured and enacted. Knowledge is always inferred. The Eight Ways of Knowing are a convenient way to group what information reveals, not something that can be measured directly. The patterns of language that make elicitation work are not limited by subject, profession, culture or geography, which is why the method's verbs are domain-neutral. The Eight Ways of Knowing are joined to the twelve support information types in one grid of 96 cells, and the grid is used to ask, not only to file. My position is that the ways not yet grounded in neuroscience lack evidence because no one has looked for them, not because they have been disproved. Stated that way, it is an open research question.

Evidence. Five of the Eight Ways of Knowing have established grounding in memory or learning research. All eight have a linguistic signature. A proposed agreement test would have independent coders assign the same ways and support information types to the same transcripts (Cohen's kappa).

Principle 5: Govern the system from inside

Claim. A knowledge system is governed by observing every use, not by periodic certification alone. Measurement shows what was disclosed; people decide whether it is right.

Mechanism. Two functions are kept apart. InQ (Information Quotient) measures the information density of each exchange and compares it over time, so gaps and gains show as they happen. InQ does not validate the information. Validation belongs to people: the holder confirms, and experts approve changes before people or AI use them.

Lineage.

- Cybernetics, from the Macy conferences (1946–1953), studied feedback and control across machines and people. Von Foerster's *Cybernetics of Cybernetics* (1974) placed the observer inside the system observed.
- AI took two models from its founding. The neural line follows the Macy brain model; governance still follows Dartmouth's model of control: stand outside the machine, test it, certify it. Hinton left Google in 2023 to warn that no one knows how to control these systems. Governing from inside, by observing every use, is the alternative this principle takes.
- Information theory gives density a measure. Shannon (1948) defined information. In psycholinguistics, the uniform information density account (Jaeger, 2010) shows that speakers regulate how much information each stretch of speech carries.
- Management standards now require governance during operation. ISO 9001:2026, clause 7.1.6, requires an organization to retain, apply and share the knowledge its processes need, to the extent necessary. The clause's note counts experience-based knowledge held by people among the forms that knowledge takes. ISO/IEC 42001 (2023) sets requirements for an AI management system, including monitoring, measurement and evaluation.

What I added. InQ applies an information-density measure to elicitation, computed on every use and typed by the Eight Ways of Knowing. The idea is older than the measure: a note I wrote in 2009 records that particular natural-language phrases go with particular types of information, so that types could be identified quantitatively. Human authority holds throughout: AI agreement is not evidence, and every change keeps its provenance.

Evidence. The standards and the density research are established. InQ is proposed: its scoring method is still under construction. Once built, it should show that its readings track something observable, such as gaps the holder later fills.

Principle 6: The model is the substrate, not the authority

Claim. In my design, the language model elicits, structures and retrieves. It is never the authority on what an expert knows.

Mechanism. A model lacks the tacit knowledge it is asked to capture, and its agreement proves nothing. Treating a model as a durable institutional knowledge system, without grounding, governance, evaluation and maintenance, is the present danger. People keep four responsibilities: confirming knowledge, judging validity, deciding at gates, and pursuing the purpose of lifting human conscious states.

Lineage.

- The neural and symbolic lines reconverge after 2023. Enterprises ground models in knowledge graphs; GraphRAG (Edge et al., 2024) is the clearest case. It shows what structure gives a model: a graph divided into nested communities, each summarized in advance, lets a model answer questions about a whole collection rather than isolated passages. That is the purpose my 2008 history gave to levels and associations, which were added so that a person could see the whole of an expert's knowledge. GraphRAG's graph, however, is extracted by the model and confirmed by no one, so it is only as good as its extraction. In my design, elicitation keeps drawing human knowledge in, each element is typed by the 96 cells, and the holder confirms it before anything is built on it.
- Panickssery, Bowman and Feng (2024) found that models judging text favor their own output, which supports the rule that AI agreement is not evidence.
- Hutchins (1995), in *Cognition in the Wild*, showed that cognition can be distributed across people and artifacts.

What I added. The elicited, human-confirmed record sits above the model as the layer through which it is governed, in the line of Engelbart's augmentation rather than general intelligence. Cognition is distributed across five roles: the knowledge holder; the knowledge engineer, who supplies the ontology, prompts, criteria and governance; the AI; the knowledge model or graph; and the user who validates. The stance is in my 1989 thesis, which required that the software support the user's own intellectual capabilities rather than replace them, and which held that the system cannot judge whether an answer is correct.

Evidence. The research basis is established; the design itself is practitioner record. A proposed test would compare the model's unaided answers with the expert-confirmed record to show what the elicitation added.

WHAT I ADDED

I know of no predecessor that joined these six ideas. My addition is the combination, delivered as instruments: elicitation during the work, typed by the language knowledge is spoken in, confirmed by the holder, carried to the point of performance, measured on every use, and governing the model rather than governed by it. The specific combination has not been tested: the Eight Ways of Knowing crossed with the twelve support information types, giving 96 ways to read and tag information and to ask good questions of people or AI, governed from inside and joined to a language model. That is the frame I am going forward with. Proposed.

From winter to spring. My thesis is that 1987 was knowledge engineering's winter and 2027 could be its spring. The spring would recover explicit knowledge, provenance, domain models and accountable reasoning, while modern AI overcomes the knowledge-acquisition bottleneck. Today's danger is treating a language model as a durable institutional knowledge system without grounding, governance, evaluation and maintenance. This is my stated thesis, not a tested claim.

EVIDENCE LEDGER

My record is long and real, but most of it is not yet citable. The next steps below turn practitioner record into published evidence and run the four proposed tests.

EVIDENCE	SUPPORTS	LABEL	NEXT STEP
LearnerFirst electronic performance support systems, 1992, used by more than 4,000 organizations	Principle 3	Practitioner record	Assemble product documentation, customer counts and dates.
Joint projects with Gloria Gery	Principle 3	Practitioner record, unpublished	Write a dated account of the projects and their findings.
Engelbart as mentor and reviewer of the 1989 thesis	Story, Principle 3	Practitioner record	The thesis closes by inviting his comments, which documents that I sought his review. Locate his reply, correspondence or review notes.
1989 thesis, <i>Establish, Organize, and Represent: The Foundation for Occupational Applications</i>	Story, Principles 2, 3, 4 and 6	Established (the degree, title, director and date); practitioner record (Engelbart's review)	The commencement program and the reading edition are cited in reference 69. Engelbart's review is tracked in its own row.
Builder-Player, documented as patentable subject matter	Principle 2	Practitioner record	Locate the dated documentation.
LearnerFirst advertisements in <i>Quality Progress</i> ("Learn it, do it, document it, all at the same time")	Principle 2, Principle 3	Practitioner record	Locate the issues in the <i>Quality Progress</i> archive and cite them.
Knowledge Harvesting® trademark; the registration records first use in commerce on October 28, 1997	Story	Established (reference 65)	None.
Five types of knowledge taught to dozens of knowledge management professionals	Principle 4	Practitioner record	Locate dated workshop and mentoring materials, such as the Knowledge Harvesting workshop binder.
2008 history of how the types of knowledge grew	Principle 4	Practitioner record, dated internal document	Publish the document or an excerpt with its date.
Implicit knowledge, developed with Carl Frappaolo	Principle 1	Established (the 2008 paper); practitioner record (the co-development)	Locate any dated record of the joint work.
National Science Foundation panel, 2001, and collaboration with Robert Horn afterward	Story, Principle 4	Established (the panel and paper, reference 70); practitioner record (the collaboration)	Cite the dated reports and correspondence.

EVIDENCE	SUPPORTS	LABEL	NEXT STEP
Knowledge Harvesting Prompts, delivering real-time performance support	Principles 3 and 6	Practitioner record	Publish one case of an instrument used without me present.
Hundreds of expert projects under nondisclosure	Principles 1 and 2	Practitioner record	Ask former clients for outcome-only statements, with written consent.
Thirteen formative experiences of knowledge engineering (research report, September 24, 2026)	How knowledge engineering was learned	Proposed	Have an independent reader check the report against its sources, then publish it.
Episode, Process and System modes of inquiry (literature cross-check, January 3, 2026)	How knowledge engineering was learned, Principle 4	Proposed	Test whether choosing the mode by the gap yields better follow-up questions.
Expert Judgment Experiment: forward test of a frozen competence model	Principle 2	Proposed	Run Stage 1 with a second, independent expert.
Coder agreement on the Eight Ways of Knowing and twelve support information types	Principle 4	Proposed	Have two coders tag the same transcripts and report kappa.
InQ (Information Quotient) density readings across sessions	Principle 5	Proposed	Finish the scoring method, then compare readings over time.
Model's unaided answers against the confirmed record	Principle 6	Proposed	Run on one completed record.

REFERENCES

1. Anderson, J. R. (1982). Acquisition of cognitive skill. *Psychological Review*, 89(4), 369–406.
2. Bahill, A. T., & Ferrell, W. R. (1986). Teaching an introductory course in expert systems. *IEEE Expert*, 1(4), 59–63.
3. Bahill, A. T., Harris, P. N., & Senn, E. (1988). Lessons learned building expert systems. *AI Expert*, 3(9), 36–45.
4. Birt, L., Scott, S., Cavers, D., Campbell, C., & Walter, F. (2016). Member checking: a tool to enhance trustworthiness or merely a nod to validation? *Qualitative Health Research*, 26(13), 1802–1811.
5. Boose, J. H. (1985). A knowledge acquisition program for expert systems based on personal construct psychology. *International Journal of Man-Machine Studies*, 23(5), 495–525.
6. Buchanan, B. G., & Shortliffe, E. H. (Eds.) (1984). *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley.
7. Chase, W. G., & Simon, H. A. (1973). Perception in chess. *Cognitive Psychology*, 4(1), 55–81.
8. Chomsky, N. (1957). *Syntactic Structures*. Mouton.
9. Clancey, W. J. (1988). The knowledge engineer as student: metacognitive bases for asking good questions. In H. Mandl & A. Lesgold (Eds.), *Learning Issues for Intelligent Tutoring Systems* (pp. 80–113). Springer.
10. Davis, R. (1976). *Applications of Meta Level Knowledge to the Construction, Maintenance and Use of Large Knowledge Bases*. PhD thesis, Stanford University (Stanford Artificial Intelligence Laboratory Memo AIM-283).
11. Edge, D., et al. (2024). From local to global: a Graph RAG approach to query-focused summarization. arXiv:2404.16130.
12. Engelbart, D. C. (1962). Augmenting Human Intellect: A Conceptual Framework. Stanford Research Institute.
13. Ericsson, K. A., & Simon, H. A. (1993). *Protocol Analysis: Verbal Reports as Data* (rev. ed.). MIT Press.
14. Feigenbaum, E. A. (1977). The art of artificial intelligence: themes and case studies of knowledge engineering. *Proceedings of IJCAI-77*.
15. Feigenbaum, E. A., & McCorduck, P. (1983). *The Fifth Generation*. Addison-Wesley.
16. Feldon, D. F., & Clark, R. E. (2006). Instructional implications of cognitive task analysis as a method for improving the accuracy of experts' self-report. In G. Clarebout & J. Elen (Eds.), *Avoiding simplicity, confronting complexity: Advances in studying and designing (computer-based) powerful learning environments* (pp. 109–116). Sense Publishers.
17. Flavell, J. H. (1979). Metacognition and cognitive monitoring: A new area of cognitive–developmental inquiry. *American Psychologist*, 34(10), 906–911.
18. Frappaolo, C. (2008). Implicit knowledge. *Knowledge Management Research & Practice*, 6(1), 23–25.
<https://doi.org/10.1057/palgrave.kmrp.8500168>
19. Fukushima, K. (1980). Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4), 193–202.
20. Gaines, B. R., & Boose, J. H. (Eds.) (1988). *Knowledge Acquisition for Knowledge-Based Systems*. Academic Press.
21. Gery, G. J. (1991). Electronic Performance Support Systems. Weingarten Publications.
22. Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2), 199–220.
23. Hoffman, R. R., Shadbolt, N. R., Burton, A. M., & Klein, G. (1995). Eliciting knowledge from experts: a methodological analysis. *Organizational Behavior and Human Decision Processes*, 62(2), 129–158.
24. Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *PNAS*, 79(8), 2554–2558.
25. Horn, R. E. (1976). *How to Write Information Mapping*. Information Resources.
26. Hutchins, E. (1995). *Cognition in the Wild*. MIT Press.
27. ISO 9001:2026. *Quality management systems — Requirements* (6th ed.), clause 7.1.6, Organizational knowledge. International Organization for Standardization. See ISO's announcement of the sixth edition.

28. ISO/IEC 42001:2023. *Information technology — Artificial intelligence — Management system*. International Organization for Standardization and International Electrotechnical Commission.
29. Jaeger, T. F. (2010). Redundancy and reduction: speakers manage syntactic information density. *Cognitive Psychology*, 61(1), 23–62.
30. Klein, G. A., Calderwood, R., & MacGregor, D. (1989). Critical decision method for eliciting knowledge. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(3), 462–472.
31. Krathwohl, D. R. (2002). A revision of Bloom's taxonomy: an overview. *Theory Into Practice*, 41(4), 212–218.
32. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems* 25.
33. Labov, W., & Waletzky, J. (1967). Narrative analysis: oral versions of personal experience. In J. Helm (Ed.), *Essays on the Verbal and Visual Arts* (pp. 12–44). University of Washington Press.
34. Lakoff, G., & Johnson, M. (1980). *Metaphors We Live By*. University of Chicago Press.
35. Lenat, D. B. (1995). CYC: a large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11), 33–38.
36. Lindsay, R. K., Buchanan, B. G., Feigenbaum, E. A., & Lederberg, J. (1980). *Applications of Artificial Intelligence for Organic Chemistry: The DENDRAL Project*. McGraw-Hill.
37. Mann, W. C., & Thompson, S. A. (1988). Rhetorical structure theory: toward a functional theory of text organization. *Text*, 8(3), 243–281.
38. McCarthy, J., Minsky, M., Rochester, N., & Shannon, C. (1955). *A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence*.
39. McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5, 115–133.
40. McDermott, J. (1982). R1: a rule-based configurer of computer systems. *Artificial Intelligence*, 19(1), 39–88.
41. Minsky, M. (1974). A framework for representing knowledge. MIT Artificial Intelligence Laboratory Memo 306.
42. Minsky, M., & Papert, S. (1969). *Perceptrons*. MIT Press.
43. Mitchell, M. (2026). Six principles for evaluating cognitive capabilities in AI models. *AI Magazine*, 47(2), e70061.
44. Newell, A., & Simon, H. A. (1956). The logic theory machine: a complex information processing system. *IRE Transactions on Information Theory*, 2(3), 61–79.
45. Newell, A., & Simon, H. A. (1972). *Human Problem Solving*. Prentice-Hall.
46. Nisbett, R. E., & Wilson, T. D. (1977). Telling more than we can know: verbal reports on mental processes. *Psychological Review*, 84(3), 231–259.
47. Nonaka, I., & Takeuchi, H. (1995). *The Knowledge-Creating Company*. Oxford University Press.
48. Olson, J. R., & Rueter, H. H. (1987). Extracting expertise from experts: methods for knowledge acquisition. *Expert Systems*, 4(3), 152–168.
49. Panickssery, A., Bowman, S. R., & Feng, S. (2024). LLM evaluators recognize and favor their own generations. NeurIPS 2024.
50. Paris, S. G., Lipson, M. Y., & Wixson, K. K. (1983). Becoming a strategic reader. *Contemporary Educational Psychology*, 8(3), 293–316.
51. Polanyi, M. (1966). *The Tacit Dimension*. Doubleday. Reissued by the University of Chicago Press, 2009.
52. Roco, M. C., & Bainbridge, W. S. (Eds.) (2002). *Converging Technologies for Improving Human Performance: Nanotechnology, Biotechnology, Information Technology and Cognitive Science*. National Science Foundation.
53. Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408.
54. Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323, 533–536.

55. Schank, R. C., & Abelson, R. P. (1977). *Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures*. Lawrence Erlbaum.
56. Schreiber, G., et al. (2000). *Knowledge Engineering and Management: The CommonKADS Methodology*. MIT Press.
57. Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27(3), 379–423, and 27(4), 623–656.
58. Simon, H. A. (1955). A behavioral model of rational choice. *Quarterly Journal of Economics*, 69(1), 99–118.
59. Singhal, A. (2012, May 16). Introducing the Knowledge Graph: things, not strings. *Google Official Blog*.
60. Slobin, D. I. (1996). From "thought and language" to "thinking for speaking." In J. J. Gumperz & S. C. Levinson (Eds.), *Rethinking Linguistic Relativity* (pp. 70–96). Cambridge University Press.
61. Squire, L. R. (2004). Memory systems of the brain: a brief history and current perspective. *Neurobiology of Learning and Memory*, 82(3), 171–177.
62. Sullivan, M. E., Yates, K. A., Inaba, K., Lam, L., & Clark, R. E. (2014). The use of cognitive task analysis to reveal the instructional limitations of experts in teaching procedural skills. *Academic Medicine*, 89(5), 811–816.
63. Tofel-Grehl, C., & Feldon, D. F. (2013). Cognitive task analysis-based training: a meta-analysis of studies. *Journal of Cognitive Engineering and Decision Making*, 7(3), 293–304.
64. Tulving, E. (1972). Episodic and semantic memory. In E. Tulving & W. Donaldson (Eds.), *Organization of Memory* (pp. 381–403). Academic Press.
65. U.S. Patent and Trademark Office. *KNOWLEDGE HARVESTING*, Registration No. 3,500,735 (registered September 16, 2008; the registration records first use in commerce on October 28, 1997). See [Justia Trademarks record](#).
66. University of Edinburgh. History of AI teaching at Edinburgh. History of Artificial Intelligence, Computer Science and Cognitive Science at Edinburgh.
67. Von Foerster, H. (Ed.) (1974). *Cybernetics of Cybernetics*. Biological Computer Laboratory, University of Illinois.
68. Werbos, P. J. (1974). *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. PhD thesis, Harvard University.
69. Wilson, L. T. (1989). *Establish, organize, and represent: The foundation for occupational applications* [Master's thesis, Auburn University]. Directed by C. A. Snyder, Department of Management. Listed in Auburn University, *Commencement Exercises, Fall Quarter, 1989* (December 8, 1989), p. 15, as "EOR—Establish, Organize and Represent the Foundation for Occupational Applications." A web reading edition is at ltwilson.com/eor.
70. Wilson, L. T. (2002). Accelerating convergence of biotechnology, nanotechnology, & information technology. In M. C. Roco & W. S. Bainbridge (Eds.), *Converging technologies for improving human performance: Nanotechnology, biotechnology, information technology and cognitive science*. National Science Foundation.
71. Wilson, L. T. (2026). *Knowledge Harvesting modes and expertise: literature cross-check* (January 3, 2026). Unpublished.
72. Wilson, L. T. (2026). *Reconstructing the sequence of experiences by which knowledge engineering was learned* (research findings through September 24, 2026). Unpublished.
73. Yates, K. A., Feldon, D. F., & Clark, R. E. (2011). Using cognitive task analysis to capture expert knowledge and skills for research and instructional design. Professional development workshop, Annual Meeting of the American Educational Research Association, New Orleans, April 2011.



Knowledge Engineering: Lineage, Principles, and Evidence · Larry Todd Wilson · Revised October 18, 2026
 The current version is at <https://ltwilson.com/history/> · © 2026 Larry Todd Wilson. All rights reserved.