

Knowledge engineering

This article is about methods for developing expert systems. For application of knowledge based technology to the domain of manufacturing and CAD, see [Knowledge based engineering](#).

Knowledge engineering (KE) refers to all technical, scientific and social aspects involved in building, maintaining and using [knowledge-based systems](#).

1 Background

1.1 Expert systems

One of the first examples of an [expert system](#) was [MYCIN](#), an application to perform medical diagnosis. In the MYCIN example, the domain experts were medical doctors and the knowledge represented was their expertise in diagnosis.

Expert systems were first developed in artificial intelligence laboratories as an attempt to understand complex human decision making. Based on positive results from these initial prototypes, the technology was adopted by the US business community (and later worldwide) in the 1980s. The Stanford heuristic programming projects led by [Edward Feigenbaum](#) was one of the leaders in defining and developing the first expert systems.

2 History

In the earliest “cowboy” days of expert systems there was little or no formal process for the creation of the software. Researchers just sat down with domain experts and started programming, often developing the required tools (e.g. [inference engines](#)) at the same time as the applications themselves. As expert systems moved from academic prototypes to deployed business systems it was realized that a methodology was required to bring predictability and control to the process of building the software. There were essentially two approaches that were attempted:

1. Use conventional software development methodologies
2. Develop special methodologies tuned to the requirements of building expert systems

Many of the early expert systems were developed by large consulting and system integration firms such as [Andersen Consulting](#). These firms already had well tested conventional [waterfall methodologies](#) (e.g. Method/1 for Andersen) that they trained all their staff in and that were virtually always used to develop software for their clients. One trend in early expert systems development was to simply apply these waterfall methods to expert systems development.

Another issue with using conventional methods to develop expert systems was that due to the unprecedented nature of expert systems they were one of the first applications to adopt [rapid application development](#) methods that feature iteration and prototyping as well as or instead of detailed analysis and design. In the 1980s few conventional software methods supported this type of approach.

The final issue with using conventional methods to develop expert systems was the need for [knowledge acquisition](#). *Knowledge acquisition* refers to the process of gathering expert knowledge and capturing it in the form of rules and ontologies. Knowledge acquisition has special requirements beyond the conventional specification process used to capture most business requirements.

These issues led to the second approach to knowledge engineering: development of custom methodologies specifically designed to build expert systems.^[1] One of the first and most popular of such methodologies custom designed for expert systems was the [Knowledge Acquisition and Documentation Structuring \(KADS\)](#) methodology developed in Europe. KADS had great success in Europe and was also used in the United States.^[2]

3 See also

- [Knowledge level modeling](#)
- [Knowledge management](#)
- [Knowledge representation](#)
- [Knowledge retrieval](#)
- [Knowledge tagging](#)
- [Method engineering](#)

4 References

- [1] Feigenbaum, Edward; McCorduk, Pamela (1983). *The Fifth Generation* (1st ed.). Reading, MA: Addison-Wesley. ISBN 978-0-201-11519-2. OCLC 9324691.
- [2] Schreiber, August Th.; Akkermans, Hans; Anjewierden, Anjo; Dehoog, Robert; Shadbolt, Nigel; Vandeveld, Walter; Wielinga, Bob (2000), *Knowledge engineering and management: the CommonKADS methodology* (1st ed.), Cambridge, MA: The MIT Press, ISBN 978-0-262-19300-9

5 External links

- [Data & Knowledge Engineering – Elsevier Journal](#)
- [Knowledge Engineering Review, Cambridge Journal](#)
- [The International Journal of Software Engineering and Knowledge Engineering – World Scientific](#)
- [IEEE Transactions on Knowledge and Data Engineering](#)
- [Expert Systems: The Journal of Knowledge Engineering – Wiley-Blackwell](#)

6 Text and image sources, contributors, and licenses

6.1 Text

- **Knowledge engineering** *Source:* https://en.wikipedia.org/wiki/Knowledge_engineering?oldid=768265712 *Contributors:* Dominus, Delirium, Scott, Universimmedia, Wile E. Heresiarch, Jonathan.mark.lingard, Khalid hassani, Erich gasboy, Beland, Zuidervled, Andreas Kaufmann, Till Ulen, Remuel, Vdeboer, Mdd, Alai, Ceyockey, Smoothhenry, Ruud Koot, Waldir, Ketiltrout, Rjwilmsi, Isotope23, Jacob1044, Planetneutral, YurikBot, Kafziel, Sclin, Supten, MaxVeers, Arthur Rubin, Aeosynth, SmackBot, Yamaguchi[?]?, TimBentley, SlimJim, Master of Puppets, Uthbrian, Xyzzyplugh, Sina2, RichMorin, Dicklyon, Nabeth, Tawkerbot2, CmdrObot, Cydebot, Ag2324, Biblbroks, Thijs!bot, Dsp13, Ay6112, Dxthom, R'n'B, Overix, VolkovBot, Kilmer-san, CharlesGillingham, Mario Žamić, The Thing That Should Not Be, Brews ohare, Vincent graymore, Spot6224, Addbot, Cerber, DOI bot, Swinberg, Fgnievinski, Yobot, Yzeng, AnomieBOT, LogoX, Jon g hall, Zouxiaohui, Murdomck, Citation bot 1, DrilBot, Vmkern, Dz8113, Primefac, Dcirovic, ChuispastonBot, Logicalgregory, Helpful Pixie Bot, Mark Marathon, Gborgonovo, Mogism, Mohiuddin6644, Me, Myself, and I are Here, Phamnhatkhanh, Biogeographist, Wiki55445, MadScientistX11, Questioner123 and Anonymous: 38

6.2 Images

6.3 Content license

- Creative Commons Attribution-Share Alike 3.0