

Ken Kahn

ToonTalk@gmail.com

Educational Background

Massachusetts Institute of Technology, Sept. 1973 to Feb. 1979

Ph.D. in Electrical Engineering and Computer Science, Jan. 1979

Thesis: *Creation of Computer Animation from Story Descriptions*

M.S. in Electrical Engineering, Aug. 1975

Thesis: *Mechanization of Temporal Knowledge*

University of Pennsylvania, Sept. 1969 to June 1973; University of Stockholm, Junior Year Abroad 1971

B.A. in Economics *magna cum laude* with distinction, June 1973

Thesis: *Bankruptcy, Information Costs, and Equilibrium*

Professional Experience

Instructor Yale-NUS, January to May 2023

Visiting Researcher, University of Hong Kong, September to December 2022

Senior Researcher, Department of Education, University of Oxford. Jan 2017 to April 2021 (retired May 2021)

Visiting Associate Professor, Division of Science, Yale-NUS, Jan. to May 2019

Senior Researcher, Research Support IT Services, University of Oxford. May 2006 to Dec. 2016

Senior Visiting Fellow, University Scholars Programme, National University of Singapore, Jan. to May 2015; Jan to May 2016, Jan to May 2017

Consultant, researcher, and developer at the London Knowledge Lab, Institute of Education. June 2003 to January 2014

Founder and President, Animated Programs, September 1992 to September 2009

Visiting Research Scientist, Royal Institute of Technology, Sweden. Sept. 2002 to Sept. 2003; July 2004 to September 2005

Leverhulme Trust Fellowship at the Institute of Education, London, Sept. 2003 through June 2004

Consultant to the University of Stockholm, Sweden, November 1999 to August 2001

Consultant to the Institute of Education, University of London, November 1998 to August 2001

Consultant to Uppsala University, Sweden, November 1998 to November 1999

Visiting Scholar, CSLI, Stanford University, September 1992 to September 1996

Independent consultant to IBM and Swedish Institute of Computer Science, 1995

Member of Research Staff, Xerox Palo Alto Research Center (PARC), August 1984 to August 1992

Guest Researcher, ICOT (Institute for New Generation Computer Technology), Tokyo, November 1983 and June 1992

Consultant for the Rand Corporation, Atari Research, Syracuse University, and CitiCorp, 1982 to 1983

Associate Professor, Computing Science Department, Uppsala University, July 1981 to August 1984

Research Associate, UPMail, Computing Science Dept. Uppsala University, July 1980 to August 1984

Visiting Professor and Researcher, University of Stockholm, January 1980 to June 1980

MIT Research Scientist and Lecturer, January 1979 to December 1979

Research Assistant, Research Staff, and IBM Graduate Student Fellowship Recipient, Artificial Intelligence Laboratory, Project MAC (Laboratory for Computer Science), MIT, September 1973 to January 1979

Programmer of statistical and graphical study of census data, Philadelphia Social History Project, Sept. 1972 to July 1973

Applications programmer, consultant and liaison to the humanities departments, University of Pennsylvania Computer Center, Jan. 1970 to July 1971

Research Experience

As part of my master's thesis at MIT, I implemented two versions of a time specialist, a computer program capable of accepting a wide range of temporal statements, checking their consistency, and making inferences to answer questions [52, 2]. Following that I joined the LOGO group [120] and taught elementary school children to do natural language programming [51] and computer animation [67] using several small systems I built for this purpose [1]. I explored the idea of program interpolation which led to three publications about interpolating LOGO turtle programs [81, 75, 108]. In 1983, while consulting for Atari, I resumed research on natural language tools for children, this time based upon Prolog [5].

From 1976 to 1980 I was the designer and implementer of the object-oriented computer language "Director" [23, 56, 60, 54]. Director was used for programming computer animation, knowledge-based systems, and experiments in programming by children. Simultaneously, I was working on my doctoral thesis, building a system capable of automatically creating simple computer-animated films on the basis of vague, incomplete story descriptions [53, 24, 26, 27, 55]. During this time, I took courses in animation and filmmaking (two at Harvard University) and made several films which were shown at film festivals, local theaters, and cable TV [67, 68, 70, 69]. In 1988 and 1989, as a refresher, I took three animation courses at De Anza College and made several short animated films.

After a year as a Lecturer and Research Scientist at MIT in 1979, I went to Sweden, initially to the University of Stockholm, and then to Uppsala University, where I began exploring multi-paradigm programming. After learning about Prolog, I used it to implement an actor language, "Intermission" [4]. Next came "Uniform", a language based on extended unification [8, 3]. The language was an attempt to combine the important features of Lisp, actor languages, and Prolog into a simple coherent framework. My research on combining the best of Lisp and Prolog led to my design and, with a colleague, implementation of LM-Prolog [57, 7, 58, 30, 31], an extended Prolog system on Lisp Machines that was sold by Lisp Machines, Incorporated.

While in Sweden I became excited about the potential for partial evaluation to win back performance sacrificed in the quest for simple generic programs. As a realistic example, I worked on automatically generating a compiler from LM-Prolog to Lisp by doing partial evaluation of the LM-Prolog interpreter written in Lisp [28, 29, 6]. The partial evaluator was written in LM-Prolog and generated efficient specializations of Lisp programs. It was during this period that my close interactions with the Japanese Fifth Generation Project began [14].

From Sweden I went to Xerox Palo Alto Research Center (PARC) to continue my research on multi-paradigm systems with the "LOOPS" group. During 1985, I was one of the two designers of CommonLoops [9], the basis of the Common Lisp Object System (CLOS) standard. At this time I was also chair of the Common Lisp object-oriented programming subcommittee. (Despite having been instrumental in bringing CLOS to the world, I am not particularly proud of it.) During this period, I also collaborated on other projects with several researchers at PARC. These ranged from CoLab (pioneering software support for collaboration) [12], virtual copies (a new approach to creating computational objects) [33], and integrating access-oriented programming with other programming paradigms [10].

In 1986 I initiated and led the Vulcan project, whose purpose was building high level programming abstractions within a concurrent logic programming framework [32, 59, 11, 34, 60, 62, 61]. For nearly two years I managed a project involving six researchers. Much of the research focused upon programming language support for building distributed systems [13]. When funding was cut, similar efforts were continued, though on a smaller scale [40, 64, 37, 35]. I focused upon connections between concurrent logic (and more generally constraint) programming and concurrent object-oriented frameworks or actors [63, 38, 36]. The theme underlying Vulcan and subsequent research was "clean and small but real".

From 1989 to 1992, I combined my interest in animation with my interests in language design and concurrency by leading a project to visualize concurrent programs and animate their executions in a coherent and general manner [39, 71]. My efforts resulted in a picture parser and an animator capable of interpreting PostScript drawings of concurrent constraint programs and automatically producing animations of program executions [41]. I explored applications of this system to visualizing object-oriented designs [66], real-time control, and non-programming uses (e.g. workflow and animated proofs [65]).

In September 1992 I left Xerox PARC to found Animated Programs, a company whose mission was to bring the power and fun of programming to children and adults. I am the sole designer and developer of ToonTalk, a concurrent animated programming language [42, 43, 15, 16, 44, 17, 45, 46, 18, 47, 19, 21, 50, 93]. I have presented ToonTalk at a dozen conferences on game development and educational software and have advised doctoral students [20, 49, 72, 80] whose research builds upon ToonTalk. From 2014 to 2015 I built a new ToonTalk for the web [97, 100, 103].

I have participated in two EU-funded educational research projects building upon ToonTalk [48, 22, 73]: Playground, which supported very young children in making their own computer games; and WebLabs, where children build scientific models, explore mathematics computationally, and publish reports with runnable models on the Web. In both projects my role included supporting and enhancing the technologies underlying the research, contributing to project design and planning, building tools and components for use by both children and researchers, helping children use these tools to build games or to explore science and mathematics, and testing the tools and learning material with children.

I taught elementary school children to program graphics, animation, and natural language while a member of the Logo Group of the MIT AI Laboratory in 1975 [114]. Beginning in 1995, I taught ToonTalk to children. I taught fourth grade students at the Encinal School in Menlo Park, California for several hours a week for three years. I taught a diverse group of children at Plugged In Greenhouse, a creative arts and technology studio for low income children between 6 to 12 years old, in East Palo Alto, California. I taught ToonTalk workshops in museums in San Jose, California and Tokyo, Japan. I have worked with children at the Theydon Bois Primary School in Essex and the Cherwell School in Oxford on exploring infinite sets in ToonTalk as part of the WebLabs Project [90].

From September 2005 to March 2006 I was the primary software designer and developer in the London Knowledge Lab/MIT Media Lab project, funded by the BBC Digital Curriculum Project (renamed to Jam). Within this project I designed and built two game construction kits for learning mechanics and an associated metagame for assisting in learning mechanics and game construction [77]. In May 2006 I began leading the Constructing2Learn Project at Oxford University [78, 79], where we developed tools and learning designs to enable non-programmers to build serious computer models. From August 2007 to December 2016 I led the Oxford University Modelling4All Project which developed Web 2.0 and web-based services for constructing and running computer models [78, 79, 85, 86, 89, 92, 99, 104]. We exhibited the Epidemic Game Maker at the Royal Society Summer Science Exhibition [94]. I collaborated with the European Explaining Religion Project in building an agent-based model of modes of religiosity [93]. I built the prototype ABM for modeling cancer tumors that underlies the microC project [112]. From May 2006 to October 2008 I became a member of the London Knowledge Lab's participation in the European ReMath Project where we have developed a construction kit called MoPiX based upon algebra [98]. In October 2007, I joined the London Knowledge Lab MiGen Project part-time [83, 84, 87, 88, 91, 101] followed by the European Metafora Project [95]. In 2011 I led a project introducing OLPC laptops to West Papua, Indonesia [96]. I wrote a book chapter about a personal historical perspective on the use of computers in education [102]. As a partner in the European eCraft2Learn project [109], I developed AI programming tools for children [105, 106, 107, 110, 111, 115, 116, 118, 119, 122].

In 2019 I worked on providing intelligent support to the LIFE game, which trains health workers to deal with pneumonia cases and neonatal problems. I also developed the Onyx web-based deep neural network for detecting disease indications in fingernails and toenails. In 2020, I developed a deep learning

model to classify text messages between community health workers [121]. I also developed a web-based neural network that responds verbally to questions from the CDC Africa Covid FAQ.

Since retiring from Oxford in 2021 I have largely focussed on AI programming by children, especially with the aid of chatbots [123, 124, 126, 127]. Based on this work in 2025 I published a book, *The Learner's Apprentice: AI and the Amplification of Human Creativity* [125] aimed at school teachers and a general audience.

University Teaching Experience

- Taught "AI Projects and Case Studies" at Yale-NUS, January to May 2023
- Taught "AI Projects and Case Studies" at the National University of Singapore, August to December 2021
- Taught "Agent-based modelling" at Singapore University of Technology and Design, 9-10 Jan 2020
- Taught "Agent-based modelling" at Yale-NUS College, Jan to May 2019.
- Taught "Computational Thinking and Modelling" at the National University of Singapore, Jan to May 2015; Jan to May 2016, Jan to May 2017
- Taught on average six non-credit courses on agent-based modelling yearly at the University of Oxford from 2009 to 2016
- Visiting Lecturer for a graduate course on Visual Programming at the Royal Institute of Technology, Sweden. Spring 2000.
- Lecturer for undergraduate and graduate courses in Artificial Intelligence, Description Languages, Lisp Machines, and Logic Programming, Uppsala University, Sweden, Sept. 1980 to August 1984.
- Lecturer for two courses in Artificial Intelligence, University of Stockholm, Sweden, Jan. to June 1980.
- Lecturer for "Structure and Interpretation of Computer Languages", Department of Electrical Engineering and Computer Science, MIT, Feb. 1979 to June 1979.
- Teaching Assistant for "Problem-Solving Paradigms", Department of Electrical Engineering and Computer Science, MIT, January 1976 to May 1976.

Student Supervision

I co-supervised six students who completed their Ph.D. in computer science (University of Oxford, Stanford University, Linköping University, Stockholm University, and Universidade de Trás-os-Montes e Alto Douro, Portugal). I have advised eight master's students (MIT, University of Washington, and Stockholm University). I have been the host/advisor for five undergraduate summer students, two graduate students, and three postdoctoral fellows. I have supported over one hundred undergraduate research projects.

Editorial and Academic Activities

Member of program committee: Second International Logic Programming Conference, Uppsala, Sweden, 1984; Third International Conference on Logic Programming, London, 1986; North American Conference on Logic Programming, Cleveland, Ohio, 1989; Joint Conference on Object-Oriented Programming and European Conference on Object-Oriented Programming, Ottawa, Canada, 1990; North American Conference on Logic Programming, Austin, Texas, 1990; Symposium on Partial Evaluation and Semantics-Based Program Manipulation, New Haven, Connecticut, 1991; International Logic Programming Symposium, San Diego, California, 1991. IEEE Conference on Visual Programming. Symposium on End User and Domain Specific Programming, 2002 and 2003.

General chair: International Logic Programming Symposium, San Diego, California, 1991.

Area editor: Knowledge Representation, Reasoning, and Expert Systems, *The Journal of Logic Programming*, 1986-1991.

Member of the Senior Program Committee for AAAI 2026.

Editorial board: *The Journal of Logic Programming*, *Higher-Order and Symbolic Computing*, *New Generation Computing*, *IEEE Parallel and Distributed Technology: Systems and Applications*.

Patents

United States Patent Number 5,517,663. Animated User Interface for Computer Program Creation, Control, and Execution. May 14, 1996.

Research Projects

- AI and Medicine Project, MIT, 1973 to 1975.
- LOGO Project, MIT, 1975 to 1977, 1979.
- Actors and Open Systems Project, MIT, 1975 to 1979.
- Programming Methodology and Artificial Intelligence, Uppsala University, 1980 to 1984.
- LOOPS and CommonLoops Multi-paradigm Programming Project, Xerox PARC, 1984 to 1987.
- Vulcan Distributed Programming Project, Xerox PARC, 1986 to 1990.
- Janus Distributed Constraint Programming Project, Xerox PARC, 1989 to 1992.
- Visual Janus Project, Xerox PARC 1990 to 1992.
- Playground Project, EU, led by the Institute of Education, 1998 to 2001.
- WebLabs Project, EU, led by the Institute of Education, 2002 to 2005.
- BBC Digital Curriculum Project, London Knowledge Lab and MIT Media Lab, 2005 to 2006.
- Constructing2Learn Project, JISC, Oxford University Computing Services, May 2006 to November 2007.
- ReMath Project, EU, London Knowledge Lab, May 2006 to October 2008.
- MiGen, TLRP, London Knowledge Lab, October 2007 to December 2011.
- Modelling4All, Eduserv Foundation, Oxford University Computing Services, August 2007 to 2016. Internally funded since August 2010.
- Metafora, EU, London Knowledge Lab, September 2010 to September 2013.
- eCraft2Learn, EU, University of Oxford, January 2017 to December 2018.
- Intelligent support for Life-saving Instruction for Emergencies Game, January 2019 to March 2019.
- Onyx: Development of an AI application & training module to identify systemic disease based on the visual appearance of nails, March to August 2019.

Invited lectures and seminars since 2001

February 2026, National University of Singapore ([slides](#))

January 2025, National University of Singapore ([slides](#))

January 2024, Chinese University of Hong Kong ([slides](#), [video](#))

January 2024, Education University of Hong Kong ([slides](#))

December 2023, Hong Kong University ([video](#))

October 2023, MIT CSAIL ([video](#) - first 10-15 minutes missing)

May 2023, Stanford University ([video](#) presented remotely)

September 2022, University of Hong Kong

September 2022, AI&K12 Conference, Laguna Hills, CA (presented remotely)

July 2022, Royal Institute of Technology, Sweden
September 2020, AI in K12 conference, Irvine, CA (presented remotely)
May 2019, University College London
April 2019, Beijing Normal University, China
April 2019, Nanyang Technical University, Singapore
January 2019, National University of Singapore
October 2018, MIT CSAIL
October 2018, Columbia Teachers College
October 2018, Goldman Sachs
October 2018, George Mason University
October 2018, AAAI Symposium on AI for K-12
March 2018, Politeknik Negeri Bandung, Indonesia
February 2018, Universitas Pendidikan Indonesia (2 invited talks)
January 2018, Yale-NUS, Singapore
January 2018, Nanyang Technical University, Singapore
November 2017, University of Oslo, Norway
November 2017, University of East Finland
September 2017, Mälardalen University, Västerås, Sweden
July 2016, invited talk Brazilian Computer Society Conference, Porto Alegre, Brazil
July 2016, University of Campinas, Brazil
November 2015, invited talk at Masaryk University, Czech Republic
July 2015, Waseda University, Tokyo (2 talks)
March 2015, National Institute of Education, Singapore
March 2015, National University of Singapore
April 2014, American University of Paris
May 2013, University of York
April 2013, National University of Singapore
March 2013, State University of Papua
March 2013, Udayana University, Bali
September 2012, National University of Singapore
September 2012, National University of Australia
January 2012, Open University
September 2011, National University of Singapore
August 2011, State University of Papua
April 2011, University of Warwick
December 2009, University of Bath
November 2009, University College London
October 2009, invited talk at the European Simulation and Modelling Conference, Leicester, UK
September 2009, University of Edinburgh, UK
September 2009, seminar at European Social Simulation Association Conference
July 2009, University of Cambridge, UK
May 2009, Oxford e-Science Research Centre
February 2009, National Academy of Science Workshop on The Scope and Nature of Computational Thinking
August 2008, National University of Singapore
July 2008, University of Sydney, Australia
January 2008, OLPC, Cambridge, Massachusetts
January 2008, FriamGroup, Santa Fe, New Mexico
January 2008, Northwestern University, Chicago, Illinois
October 2007, keynote at the WWW/Internet International Conference, Vila Real, Portugal
July 2007, Santa Fe Institute, New Mexico
July 2007, Microsoft Research, Redmond Washington
May 2007, Swedish Institute of Computer Science, Kista, Sweden
February 2007, Centre for Applied Research in Educational Technologies at Cambridge University

January 2007, University of Surrey, Sociology Department, UK.
 November 2006, Oxford University Complex Systems Seminar, UK.
 November 2006, Plenary, ISSEP 2006, Vilnius, Lithuania.
 October 2006, Oxford University Zoology Department, UK.
 April 2006, London Knowledge Lab, University of London, UK.
 March 2006, Oxford Brookes University, Department of Computing, Oxford, UK.
 Jan. 2006, Oxford University E-Learning seminar, Department of Educational Studies, UK.
 Jan. 2006, University of Edinburgh, UK.
 Nov. 2005, University of Stockholm Department of Education, Sweden.
 Oct. 2005, University of Bratislava, Slovakia.
 Oct. 2005, Knowledge Media Institute, Open University.
 September 2005, University of Bristol.
 June 2005, Psychology of Programming Interest Group 17th Workshop, invited keynote, Brighton, UK.
 June 2005, Northwestern University, Evanston, USA.
 June 2005, University of Warwick, UK.
 May 2005, Microsoft Research Cambridge, UK.
 April 2005, London Knowledge Lab, University of London, UK.
 March 2005, University of Pennsylvania, Philadelphia, USA.
 March 2005, SRI International, Menlo Park, CA, USA.
 Feb. 2005, Oxford University E-Learning seminar, Department of Educational Studies, Oxford, UK.
 Nov. 2004, Swedish Institute of Computer Science, Stockholm, Sweden.
 Oct. 2004, Oxford University Computing Laboratory, Oxford, UK.
 September 2004, MIT Media Lab Europe, Dublin, Ireland.
 July 2004, University of Canterbury Computer Science Department, Christchurch, New Zealand.
 July 2004, Human Interface Technology Laboratory, Christchurch, New Zealand.
 July 2004, Te Manawa Museum, Palmerston North, New Zealand.
 June 2004, New Zealand Game Developers Conference, Invited speaker, Dunedin, New Zealand.
 April 2004, Cambridge University Computing Laboratory, Cambridge, UK.
 April 2004, IBM Research Center, Yorktown Heights, NY, USA.
 June 2003, Linköping University Computer Science Department, Linköping, Sweden.
 May 2003, Swedish HMI workshop, Söderköpings Brunn, Sweden.
 May 2003, MIT Media Lab, Cambridge, MA, USA.
 May 2003, MIT Laboratory for Computer Science, Cambridge, MA, USA.
 July 2002, Waseda University Department of Computer Science, Tokyo, Japan.
 June 2002, Keio University Department of Computer Science, Tokyo, Japan.
 June 2002, Visual Programming Environment Symposium, Invited speaker, Tokyo, Japan.
 March 2002, University of California at Berkeley Computer Science Department, Berkeley, CA, USA.
 Aug. 2001, EuroLogo Conference, Keynote speaker, Linz, Austria.

Books

[125] Ken Kahn. *The Learner's Apprentice: AI and the Amplification of Human Creativity*, Constructing Modern Knowledge Press, 2025.

Journal articles and book chapters

[122] Ming Gao, Jingjing Zhang; Yu Lu, Ken Kahn and Niall Winters. Grit and learning: Understanding the sequential learning behaviours in Snap!. *Journal of Computer Assisted Learning*. 2022. DOI: 10.1111/jcal.12766.

[121] O'Donovan, J., Kahn, K., MacRae, M. et al. Analysing 3429 digital supervisory interactions between Community Health Workers in Uganda and Kenya: the development, testing and validation of an open access predictive machine learning web app. *Hum Resour Health* 20(6). 2022. <https://doi.org/10.1186/s12960-021-00699-5>.

[120] Ken Kahn. Radical Ideas: Joy and Empowerment. In *20 Things to Do with a Computer Forward 50*, edited by Gary Stager, Constructing Modern Knowledge Press. 2021.

[117] Ken Kahn and Niall Winters. Learning by Enhancing Half-Baked AI Projects. KI - Künstliche Intelligenz, 1-5, June 2021. <https://doi.org/10.1007/s13218-021-00732-8>.

[115] Ken Kahn and Niall Winters. Constructionism and AI: A history and possible futures, *British Journal of Educational Technology*, May 2021. <https://doi.org/10.1111/bjet.13088>. Earlier version published at the Constructionism Conference 2020, Dublin, Ireland, 2020.

[114] Cynthia Solomon, Brian Harvey, Ken Kahn, Henry Lieberman, Mark L. Miller, Margaret Minsky, Artemis Papert, and Brian Silverman. 2020. History of Logo. *Proc. ACM Program. Lang.* 4, HOPL, Article 79, June 2020, 66 pages. <https://doi.org/10.1145/3386329>.

[113] Niall Winters, Rebecca Eynon, Anne Geniets, James Robson and Ken Kahn. Can we avoid digital structural violence in future learning systems?, *Learning, Media and Technology*. 2019. DOI: 10.1080/17439884.2020.1708099.

[112] Dimitrios Voukantsis, Kenneth Kahn, Martin Hadley, Rowan Wilson, and Francesca M Buffa. Modelling genotypes in their microenvironment to predict single- and multi-cellular behaviour. *GigaScience*, giz010, 2019. <https://doi.org/10.1093/gigascience/giz010>.

[102] Ken Kahn. A Half Century Perspective On The Role Of Computers In Learning And Teaching. Volume 6: Music Learning with Massive Open Online Courses, IOS Press, 2015. DOI:10.3233/978-1-61499-593-7-213.

[99] Ken Kahn. An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo. Book review, *Physics Today* 68, 8, 55. 2015.

[93] Harvey Whitehouse, Ken Kahn, Michael E. Hochberg and Joanna J. Bryson. The role for simulations in theory construction for the social sciences: case studies concerning Divergent Modes of Religiosity. *Religion, Brain & Behavior*, Volume 2, Issue 3. September 2012.

[91] Richard Noss, Alexandra Poulouvassilis, Eirini Geraniou, Sergio Gutierrez-Santos, Celia Hoyles, Ken Kahn, George D. Magoulas, and Manolis Mavrikis. The design of a system to support exploratory learning of algebraic generalisation. *Computers & Education*, 29 September 2011, ISSN 0360-1315 DOI: 10.1016/j.compedu.2011.09.021.

[90] Ken Kahn, Evgenia Sendova, Ana Isabel Sacristan, and Richard Noss. Young Students Exploring Cardinality by Constructing Infinite Processes. *Technology, Knowledge and Learning Journal*, May 2011.

[81] Ken Kahn. A Program to Interpolate (And Extrapolate) Between Turtle Programs. *International Journal of Computers for Mathematical Learning*, Volume 12, Number 3, December 2007 ([75] Revised).

[80] Leonel Morgado and Ken Kahn. Towards a specification of the ToonTalk language. *Journal of Visual Languages and Computing*, ISSN 1045-926X, August 2008.

[93] Ken Kahn. Should LOGO Keep Going FORWARD 1?, *Informatics in Education* 6(2):307-320. Also in

Proceedings of the 2007 EuroLogo Conference, Bratislava, August 2007.

[77] Ken Kahn, Richard Noss, Celia Hoyles, and Duncan Jones. Designing digital technologies for layered learning. *Informatics Education – The Bridge between Using and Understanding Computers*. Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2006.

[74] Y. Mor, R. Noss, C. Hoyles, K. Kahn, and G. Simpson. Designing to See and Share Structure in Number Sequences. *The International Journal for Technology in Mathematics Education*, Vol. 13, No. 2, 2006.

[22] Y. Mor, C. Hoyles, K. Kahn, R. Noss, and G. Simpson. Thinking in Process. *Micromath 20/2*, Association of Teachers of Mathematics, Summer 2004.

[21] Ken Kahn. ToonTalk – Steps Towards Ideal Computer-Based Learning Environments. In Mario Tokoro and Luc Steels, editors, *A Learning Zone of One's Own: Sharing Representations and Flow in Collaborative Learning Environments*, IOS Press Inc, June 2004.

[20] Leonel Morgado, Maria Gabriel Cruz, and Ken Kahn. ToonTalk in Kindergartens: Field Notes. In Antonio Mendez-Vilas, José António Mesa González, Inés Solo de Zaldívar Maldonado, editors, "Information Society and Education - Proceedings of the International Conference on Information and Communication Technologies in Education (ICTE2002)", *Journal of Digital Contents* 1(1): Formatex, Badajoz, Spain, 2003. ISSN: 1696-313X, ISBN: 84-607-8369-3

[19] Chris DiGiano, Ken Kahn, Allen Cypher, and David Canfield Smith. Integrating Learning Supports into the Design of Visual Programming Systems. *Journal of Visual Languages and Computing*. 12, 501-524. 2001.

[18] Ken Kahn. Generalizing by Removing Detail. *Communications of the ACM*, 43(3), March 2000. Extended version in *Your Wish Is My Command: Programming By Example*, Henry Lieberman, editor, Morgan Kaufmann, 2001.

[17] Ken Kahn. Helping Children to Learn Hard Things: Computer Programming with Familiar Objects and Actions. In *The Design of Children's Technology*, A. Druin, editor, Morgan Kaufmann, 1998.

[16] Ken Kahn. Drawings on Napkins, Video Game Animation, and Other Ways of Programming Computers. *Communications of the ACM*, August 1996.

[15] Ken Kahn. ToonTalk -- An Animated Programming Environment for Children. *Proceedings of the National Educational Computing Conference*, Baltimore, Maryland, June 1995. Extended version in the *Journal of Visual Languages and Computing* 7(2), 197–217, June 1996.

[14] Ken Kahn. A decade of progress in concurrent logic programming: A braid of research threads from ICOT, Xerox PARC, and Weizmann Institute. Launching the new era, *Communications of the ACM*, 36(3), March 1993.

[13] Ken Kahn and Mark S. Miller. Language design and open systems. In *The Ecology of Computation*, B.A. Huberman, editor. North Holland, 1988.

[12] Mark Stefik, Gregg Foster, Daniel Bobrow, Ken Kahn, Stan Lanning, and Lucy Suchman. Beyond the chalkboard: Using computers to support collaboration and problem-solving meetings. *Communications of the ACM*, January 1987.

[11] K. Kahn, E. Tribble, M. Miller, and D. Bobrow. Vulcan: Logical concurrent objects. In *Research Directions in Object-Oriented Programming*, B. Shriver and P. Wegner, editors, 75--112. The MIT Press,

1987. Also in *Concurrent Prolog*, MIT Press, Ehud Shapiro, editor.

[10] Mark Stefik, Daniel Bobrow, and Ken Kahn. Integrating access-oriented programming into a multi-paradigm environment. *IEEE Software*, 3, 1986. Also in Proceedings of the International Conference on System Science, Hawaii, 1986; *AI Tools and Techniques*, M. Richer, editor, Ablex Publishing, 1989; and *Tutorial: Object-Oriented Computing*, Vol. 2, G. Peterson, editor, IEEE Press, 1987.

[9] Daniel Bobrow, Ken Kahn, Gregor Kiczales, Larry Masinter, Mark Stefik, and Frank Zdybel. CommonLoops: Merging Common Lisp and object-oriented programming. In *Proceedings of the ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications*, September 1986. Also in *Research Foundations in Object-Oriented and Semantic Database Systems*, A. Cardenas and Dennis McLeod, editors. Prentice Hall, 1990 and *Tutorial: Object-Oriented Computing*, Vol. 1, G. Peterson, editor, IEEE Press, 1987.

[8] Ken Kahn. The implementation of Uniform, a knowledge-representation programming language based upon equivalence of descriptions. In *Progress in Artificial Intelligence*. L. Steels and J. Campbell, editors, Ellis Horwood Limited Publishers, 1985. Earlier version in Proceedings of ECAI-82, Orsay, France, July 1982.

[7] Ken Kahn and Mats Carlsson. How to implement Prolog on a Lisp machine. In *Implementations of Prolog*. J. Campbell, editor, Ellis Horwood Ltd., 1984.

[6] Ken Kahn. Partial evaluation as an example of the relationships between programming methodology and artificial intelligence. *AI Magazine*, 5, Spring 1984. Also in *Artificial Intelligence Programming Environments*, R. Hawley, editor, Ellis Horwood, 1987 and in *Readings from the AI Magazine*, Volumes 1-5, R. Englemore, editor, AAAI, 1988.

[5] Ken Kahn. A grammar kit in Prolog. *New Horizons in Educational Computing*, M. Yazdani, editor. Ellis Horwood Ltd., 1984. Also In *Instructional Science* and *Proceedings of the AISB Easter Conference on AI and Education*, Exeter, England, April 1983.

[4] Ken Kahn. Intermission -- actors in Prolog. In *Logic Programming*, S. Tarnlund and K. Clark, editors, Academic Press, 1982. An earlier version was presented at the First Logic Programming Workshop, Hungary, 1980.

[3] Ken Kahn. Uniform -- a language based upon unification which unifies (much of) Lisp, Prolog, and Act 1. In *International Joint Conference on Artificial Intelligence*, August 1981. Also in *Logic Programming: Functions, Relations, and Equations*, DeGroot, D. and Lindstrom, G., editors, Prentice Hall, 1986.

[2] Ken Kahn and Tony Gorrry. Mechanizing temporal knowledge. *Artificial Intelligence*, 9:87--108, August 1977.

[1] Ken Kahn. [Three interactions between AI and education](#). In *Machine Intelligence 8: Machine Representations of Knowledge*, E. Elcock and D. Michie, editors, Ellis Horwood Ltd. and John Wiley & Sons, 1977.

Conference Proceedings

[126] Ken Kahn and Sylvia Libow Martinez, [The Constructionist Role of Programming in the AI Era](#). Revisiting Seymour Papert's Teaching Children Thinking from 1970, Proceedings of the Constructionism Conference 2025, Zurich, 2025.

[123] Ken Kahn. Programming as a Dialogue with GPT-4: Is this the future of programming?,

Constructionism/FabLearn Conference 2023, New York City, 2023.

[119] Ken Kahn, Ramana Prasad, and Gayathri Veera. AI Snap! Blocks for Speech Input and Output, Computer Vision, Word Embeddings, and Neural Net Creation, Training, and Use, Proceedings of the EAAI-2022 Workshop at AAAI-2022, Vancouver, February 2022. [Full length version here](#).

[118] Xi Zhao, Jingjing Zhang, Wenshuo Li, Ken Kahn, Yu Lu, and Niall Winters, Learners' non-cognitive skills and behavioral patterns of programming: A sequential analysis, Proceedings of 21th IEEE International Conference on Advanced Learning Technologies, July 2021.

[116] Ken Kahn, Yu Lu, Jingjing Zhang, Niall Winters, Ming Gao, Deep Learning Programming by All. Constructionism Conference 2020, Dublin, Ireland, 2020.

[111] Dimitrios Loukatos, Ken Kahn, and Dimitris Alimisis. Flexible Techniques for Fast Developing and Remotely Controlling DIY Robots, with AI flavor. Edurobotics 2018 Conference, Rome, Italy, October 2018. In *Educational robotics in the context of the maker movement*, Moro, M., Alimisis, D. and Iocchi, L., editors, Springer, 2019.

[110] Ken Kahn, Rani Megasari, Erna Piantari and Enjun Junaeti. AI Programming by Children using Snap! Block Programming in a Developing Country, EC-TEL Conference, Leeds, UK, September 2018.

[109] Ken Kahn, Calkin Suero Montero, and Christian Voigt. STEAM Learning in formal and informal settings via craft and maker projects. Proceedings of the Interaction Design and Children Conference, Trondheim, Norway, June 2018.

[108] Ken Kahn, Interpolating (and extrapolating) 3D turtle programs in Beetle Blocks, Constructionism Conference, Vilnius, Lithuania, August 2018.

[107] Ken Kahn and Niall Winters, AI Programming by Children, Constructionism Conference, Vilnius, Lithuania, August 2018.

[106] Ken Kahn, Challenges in introducing AI programming to children. Poster at London Computing Education Research Symposium, London, June 2018.

[105] Ken Kahn and Niall Winters, Child-friendly Programming Interfaces to AI Cloud Services. Proceedings of the EC-TEL 2017 Conference, Tallinn, Estonia, September 2017.

[104] Ken Kahn and Howard Noble, Learning agent-based modelling via exploration, tutorial-guided construction, and reflection. Proceedings of the Social Simulation Conference, Rome, September 2016.

[103] Ken Kahn, A Demonstration of ToonTalk where Children Build Programs by Demonstration in a Game World. Demo, Proceedings of the Interaction Design and Children Conference 2016, Salford, UK.

[101] Sergio Gutierrez-Santos, Stefano Capuzzi, Ken Kahn, Sokratis Karkalas, and Alexandra Poulouvassilis. Scalable Monitoring of Student Interaction Indicators in Exploratory Learning Environments. In Proceedings of the 25th International Conference Companion on the World Wide Web (WWW '16 Companion). Geneva, Switzerland, 2016.

[100] Ken Kahn, Integrating programming languages with web browsers. Constructionism 2016, Bangkok, February 2016.

[98] Ken Kahn and Niall Winters, Algebraic Equations as a Creative Medium. Demo, Constructionism 2014, Vienna, August 2014.

- [97] Ken Kahn. ToonTalk reborn: Re-implementing and re-conceptualising ToonTalk for the Web. Constructionism 2014, Vienna, August 2014.
- [96] Ken Kahn, Yanty Rumengan, and Julius Naibaho. Peer-to-peer Learning in West Papua: A One Laptop per Child Experience. Constructionism 2014, Vienna, August 2014.
- [95] M. Mavrikis, K. Kahn, and T. Dragon. Constructionist Discussions With and Around Microworld Referable Objects. Proceedings of Constructionism 2012, Athens
- [94] Ken Kahn, Howard Noble, Arthur Hjorth, and Fábio Ferrentini Sampaio. Three-minute Constructionist Experiences. Constructionism 2012, Athens, August 2012.
- [92] Ken Kahn and Howard Noble. The Modelling4All Project -- A web-based modelling tool embedded in Web 2.0. Revised and extended version of [85] presented at Constructionism 2010, Paris.
- [89] Ken Kahn and Howard Noble. Modelling4All - Supporting a web-based community making, learning, sharing, and exploring computer models, CAL Conference, Brighton, March 2009.
- [88] E. Geraniou, M. Mavrikis, K. Kahn, C. Hoyles and R. Noss. [Developing a Microworld to Support Mathematical Generalisation](#). *PME 33: International Group for the Psychology of Mathematics Education*, 19-24 July 2009, Thessaloniki, Vol.3, 49-56. 2009.
- [87] D. Pearce, E. Geraniou, S. Gutierrez-Santos, M. Mavrikis, M. Cocea, C. Hoyles, K. Kahn, G. Magoulas, R. Noss, A. Poulouvassilis, and N. Winters. [An Eco-Friendly Intelligent Environment for Exploring Mathematical Generalisation](#). *Proceedings of AIED 2009 (Interactive Event)*, UK, 6-10 July 2009.
- [85] Ken Kahn and Howard Noble, The Modelling4All Project -- A web-based modelling tool embedded in Web 2.0. Simutools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques, ICST, Brussels, Belgium.
- [84] D. Pearce, E. Geraniou, M. Mavrikis, S. Gutierrez-Santos, and K. Kahn. [Using Pattern Construction and Analysis in an Exploratory Learning Environment for Understanding Mathematical Generalisation: The Potential for Intelligent Support](#). *Proceedings of the 1st International Workshop on Intelligent Support for Exploratory Environments* held in conjunction with the EC-TEL'08 Conference. 2008.
- [83] D. Pearce, M. Mavrikis, E. Geraniou, S. Gutiérrez, and K. Kahn. [An Environment for Expressing Mathematical Generalisation using Pattern Construction and Analysis](#). Workshop for HCI for Technology Enhanced Learning, HCI2008 Culture, Creativity, Interaction. 2008.
- [82] Jere Confrey, Celia Hoyles, Duncan Jones, Ken Kahn, Alan P. Maloney, Kenny H. Nguyen, Richard Noss and Dave Pratt. Designing Software for Mathematical Engagement Through Modelling. *Proceedings of the Seventeenth ICMI Study Conference*, Hanoi University of Technology, 3-8 December 2006.
- [79] Ken Kahn. Building Computer Models from Small Pieces. *Summer Computer Simulation Conference*, San Diego, CA, July 2007.
- [78] Ken Kahn. Comparing Multi-Agent Models Composed from Micro-Behaviours. *Third International Model-to-Model Workshop*, Marseilles, France, March 2007.
- [76] Ken Kahn. Time Travelling Animated Program Executions. *Proceedings of the Software Visualisation Conference*, Brighton, UK. 2006.
- [75] Ken Kahn. A Program to Interpolate (and Extrapolate) Between Turtle Programs. *Proceedings of the Bridges Conference: Mathematical Connections in Art, Music, and Science*. August 2006.

[73] Ken Kahn, Evgenia Sendova, Ana Isabel Sacristan, and Richard Noss. Making Infinity Concrete by Programming Never-ending Processes. *Proceedings of the 7th International Conference on Technology and Mathematics Teaching*, Bristol, UK. July 2005. Journal version: [90].

[72] Morgado, Leonel, Rosa Morgado, Maria Gabriel Cruz, and Ken Kahn. Embedding Computer Activities into the Context of Preschools. In *Proceedings of Challenges 2005*, Paulo Dias & Cândido Varela de Freitas, editors, 471-478. Centro de Competências Nónio Séc. XXI, Universidade do Minho, Braga, Portugal. 2005. ISBN 972-8746-13-05.

[50] Ken Kahn. The Child-Engineering of Arithmetic in ToonTalk. *Proceedings of the Interaction Design and Children Conference*, College Park, Maryland, June 2004.

[49] Leonel Morgado, Maria Gabriel Cruz, and Ken Kahn. Taking Programming into Kindergartens. *EuroLogo Proceedings*, Portugal. August 2003.

[48] J. Tholander, K. Kahn, and C.-G. Jansson. Real Programming of an Adventure Game by an 8-year-old. *Proceedings of the International Conference of the Learning Sciences*, Seattle, WA. 2002.

[47] Ken Kahn. ToonTalk and Logo - Is ToonTalk a colleague, competitor, successor, sibling, or child of Logo? *Proceedings of the EuroLogo Conference*, August 2001.

[46] Ken Kahn. From Prolog and Zelda to ToonTalk, *Proceedings of the International Conference on Logic Programming*, Danny De Schreye, editor, MIT Press, 1999.

[45] Ken Kahn. A Computer Game to Teach Programming, in *Proceedings of the National Educational Computing Conference*, June 1999.

[44] Ken Kahn. Seeing Systolic Computations in a Video Game World, *Proceedings of the IEEE Conference on Visual Languages*, Boulder, Colorado, September 1996.

[43] Ken Kahn. Metaphor Design -- Case Study of an Animated Programming Environment. *Proceedings of the Computer Game Developers Conference*, April 1995.

[42] Ken Kahn. Programming as a Video Game. *Proceedings of the Computer Game Developers Conference*, April 1994.

[41] Kenneth M. Kahn. Concurrent constraint programs to parse and animate pictures of concurrent constraint programs. In *Proceedings of the International Conference on Fifth Generation Computer Systems*, ICOT, June 1992.

[40] Vijay A. Saraswat, Kenneth Kahn, and Jacob Levy. Janus--A step towards distributed constraint programming. In *Proceedings of the North American Logic Programming Conference*. MIT Press, October 1990.

[39] Kenneth M. Kahn and Vijay A. Saraswat. Complete visualizations of concurrent programs and their executions. In *Proceedings of the IEEE Visual Language Workshop*, October 1990.

[38] Kenneth Kahn and Vijay Saraswat. Actors as a special case of concurrent constraint programming. In *Proceedings of the Joint Conference on Object-Oriented Programming: Systems, Languages, and Applications and the European Conference on Object-Oriented Programming*. ACM Press, October 1990.

[37] K. Kahn and W. Kornfeld. Money as a concurrent logic program. In *Proceedings of the North American Conference on Logic Programming*. The MIT Press, 1989.

- [36] Kenneth Kahn. Objects -- a fresh look. In Stephen Cook, editor, *Proceedings of the Third European Conference on Object-Oriented Programming*, pages 207--224. Cambridge University Press, July 1989.
- [35] Vijay A. Saraswat, David Weinbaum, Ken Kahn, and Ehud Shapiro. Detecting stable properties of networks in concurrent logic programming languages. In *Proceedings of the Seventh Annual ACM Symposium on Principles of Distributed Computing (PODC 88)*, pages 210--222, August 1988.
- [34] Eric Dean Tribble, Mark Miller, Ken Kahn, Daniel Bobrow, Curtis Abbott, and Ehud Shapiro. Channels: A generalization of streams. In *Proceedings of the Fourth International Conference on Logic Programming*, May 1987.
- [33] Sanjay Mittal, Ken Kahn, and Daniel Bobrow. Virtual copies: At the boundary between classes and instances. In *Proceedings of the ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications*, September 1986.
- [32] Ken Kahn, Eric Dean Tribble, Mark Miller, and Daniel Bobrow. Objects in concurrent logic programming languages. In *Proceedings of the ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications*, September 1986.
- [31] Lars-Henrik Eriksson, Ken Kahn, and Manny Rayner. Incorporating mutable arrays into logic programming. Technical report, University of Uppsala, UPMAIL, July 1984. Also in *Proceedings of the Second International Logic Programming Conference*.
- [30] Ken Kahn. A primitive for the control of logic programs. In *Proceedings of the IEEE International Symposium on Logic Programming*, February 1984.
- [29] Ken Kahn. The compilation of Prolog programs without the use of a Prolog compiler. In *Proceedings of the Fifth Generation Computer Systems Conference*, 1984.
- [28] Ken Kahn. A partial evaluator of Lisp written in Prolog. In *First International Logic Programming Conference*, September 1982.
- [27] Ken Kahn. How to program a society. In *Artificial Intelligence and Simulation of Behavior Conference*, July 1980.
- [26] Ken Kahn. Making aesthetic choices. In the International *Joint Conference on Artificial Intelligence*, August 1979.
- [25] Ken Kahn and Carl Hewitt. Dynamic graphics using quasi-parallelism. In *Proceedings of the ACM/SIGGRAPH Conference*, August 1978. Also AI Memo 480, MIT, June 1978.
- [24] Ken Kahn. Ani: An example of computational creativity. In *Proceedings of AISB/GI Conference*, July 1978.
- [23] Ken Kahn. An actor-based computer animation language. In S. Treu, editor, *Proceedings of the SIGGRAPH/ACM Workshop on User-Oriented Design of Interactive Graphics Systems*, pages 37--43, October 1976. Revision of: LOGO Working Paper 48, AI Working Paper 120, MIT, February 1976. Also published in: *Creative Computing*, Vol. 6, No. 11, Nov. 1980, pp. 75-84.

Technical reports and other publications

- [66] Ken Kahn and Ramana Rao. A circus of objects: The multi-faceted visualization of object-oriented designs. Technical report, Xerox PARC, 1992.
- [65] Kenneth M. Kahn. Animation of Horn Clause proofs. Technical report, SSL, Xerox PARC, 1992.
- [64] Vijay A. Saraswat, Kenneth M. Kahn, and Jacob Levy. Distributed constraint programming--the DC framework and Janus. Technical report, Xerox PARC, August 1989.
- [63] K. Kahn and E. Shapiro. Logic programs with implicit state. Technical report, Weizmann Institute of Science, Rehovot, Israel, 1988.
- [62] Ken Kahn and Mark Miller. Objects with state in Prologs with freeze. Technical report, ISL, Xerox PARC, 1987.
- [61] Ken Kahn, Curtis Abbott, Daniel Bobrow, Mark Miller, and Eric Dean Tribble. Concurrent logic programming abstractions. Technical report, ISL, Xerox PARC, 1987.
- [60] Susan Hirsh, Ken Kahn, and Mark Miller. Interming: Unifying positional and keyword notations. Technical report, Xerox PARC, 1987.
- [59] Cara Holman, Alan Borning, Ken Kahn, and Mark Miller. Constraints and logic programming. Technical report, University of Washington, Dept. of Computer Science, December 1986.
- [58] Mats Carlsson and Ken Kahn. LM-Prolog user manual. Technical Report 24, UPMAIL, Uppsala University, November 1983.
- [57] Ken Kahn. Unique features of LM-Prolog. Technical report, Uppsala University UPMAIL, February 1983.
- [56] Ken Kahn. Director guide. Technical Report 482B, MIT AI Lab, December 1979.
- [55] Ken Kahn. Creation of computer animation from story descriptions. Technical Report 540, MIT AI Lab, August 1979.
- [54] Ken Kahn and Henry Lieberman. Computer animation: Snow White's dream machine. *Technology Review*, 80:34--46, October/November 1977.
- [53] Ken Kahn. Technical report, MIT AI Lab, February 1976. LOGO Working Paper 47, AI Working Paper 119.
- [52] Ken Kahn. Mechanization of temporal knowledge. Technical Report 155, MIT Project MAC, September 1975.
- [51] Ken Kahn. A Logo natural language system. Technical report, MIT AI Lab, December 1975. LOGO Working Paper 46.
- [124] Ken Kahn. [Exploring the Future: Programming with ChatGPT](#), Hello World Magazine, Issue 22, October 2023.
- [127] Ken Kahn. [Is learning to program obsolete?](#), District Administration, October 2025.

Programming languages

Director*, Uniform*, Intermission*, LM-Prolog*, CommonLoops, CLOS, Vulcan, Lucy, Janus, Pictorial Janus*, ToonTalk*, MoPiX*, ToonTalk Reborn*. * indicates sole or primary designer and implementer.

Films and plays

[71] Kenneth M. Kahn. Towards visual concurrent constraint programming. Technical report, SSL, Xerox PARC, 1992. Contains a play written and directed by Ken Kahn, performed during San Francisco Bay Area ACM SIGCHI Seminar.

[70] Ken Kahn. [Two nights on a computer](#). MIT AI Lab, March 1978. Computer animation, 4.5 minutes.

[69] Ken Kahn. [Cinderella](#). MIT AI Lab, October 1978. Computer animation, 3 minutes.

[68] Ken Kahn. [Mulberry bush](#). Carpenter Center for the Visual Arts, Harvard University, 1976. Hand-drawn animation, 3 minutes, honorable mention, First International Student Film Festival.

[67] Ken Kahn. [Cons](#). Carpenter Center for the Visual Arts, Harvard University and MIT AI Lab, 1975. Computer and hand-drawn animation, 2 minutes.