A Timeline for Logic, λ-Calculus, and Programming Language Theory

Dana S. Scott

University Professor Emeritus
Carnegie Mellon University

Visiting Scholar
University of California, Berkeley

dana.scott@cs.cmu.edu

Taken from talks prepared for:

TURING CENTENNIAL CELEBRATION
Princeton University, May 10-12, 2012

ACM TURING CENTENARY CELEBRATION
San Francisco, June 15-16, 2012

A Quick Look Back to Beginnings

1870s

Begriffsschrift	Frege (1879)

1880s

What are numbers?	Dedekind (1888)
Number-theoretic axioms	Peano (1889)

1890s

Vorlesungen über die Algebra der Logik	Schröder (1890–1905)
Grundgesetze der Arithmetik	Frege (1893-1903)
Formulario Mathematico	Peano (1895-1901)
Grundlagen der Geometrie	Hilbert (1899)

1900s

Diophantine problem	Hilbert (1900)
Russell's Paradox	Russell (1901)
Principles of Mathematics	Russell (1903)
Richard's Paradox	Richard (1905)
Theory of Types	Russell (1908)

1910s

Principia Mathematica	Whitehead-Russell (1910-12-13)
Calculus of relatives	Löwenheim (1915)

WW I

1920s

Löwenheim-Skolem Theorem	Skolem (1920)
Propositional calculus completeness	Post (1921)
Monadic predicate calculus decidable	Behmann (1922)
Abstract proof rules	Hertz (1922)
Primitive recursive arithmetic	Skolem (1923)
Combinators	Schönfinkel (1924)
Function-based set theory	von Neumann (1925)
"Conceptual" undecidability	Finsler (1926)
Epsilon operator	Hilbert-Bernays (1927)
Combinators (again)	Curry (1927)
Ackermann function	Ackermann (1928)
Entscheidungsproblem	Hilbert-Ackermann (1928)
Abriss der Logistik & simple type theory	Carnap (1929)

It was very reasonable for Hilbert and Ackermann to emphasize the Decision Problem, as special cases had been solved.

A Very Busy Decade

1930s

Combinatory logic	Curry (1930-32)
Herbrand's Theorem	Herbrand (1930)
Completeness proof	Gödel (1930)
Partial consistency proof	Herbrand (1931)
Incompleteness	Gödel (1931)
Untyped λ-calculus	Church (1932-33-41)
Studies of primitive recursion	Péter (1932-36)
Non-standard models	Skolem (1933)
Functionality in Combinatory Logic	Curry (1934)
Grundlagen der Mathematik	Hilbert-Bernays (1934-39)
Natural deduction	Gentzen (1934)
Number-theoretic consistency & ε_0-induction	Gentzen (1934)
Inconsistency of Church's System	Kleene-Rosser (1936)
Confluence theorem	Church-Rosser (1936)
Finite combinatory processes	Post (1936)
Turing machines	Turing (1936-37)
Recursive undecidability	Church-Turing (1936)
General recursive functions	Kleene (1936)
Further completeness proofs	Maltsev (1936)
Improving incompleteness theorems	Rosser (1936)
Fixed-point combinator	Turing (1937)
Computability and λ-definability	Turing (1937)

Starting out with Gödel and ending up with Turing,
it would take a long time to comprehend
and apply all the developments in this period.

What's Happened Since the 1930s?

The 1940s

Simple type theory & λ-calculus	Church (1940)
Primitive recursive functionals	Gödel (1941-58)

WW II

Recursive hierarchies	Kleene (1943)
Theory of categories	Eilenberg-Mac Lane (1945)
New completeness proofs	Henkin (1949-50)

The 1950s

Computing and Intelligence	Turing (1950)
Rethinking combinators	Rosenbloom (1950)
IAS Computer (MANIAC)	von Neumann (1951)
Introduction to Metamathematics	Kleene (1952)
IBM 701	Thomas Watson, Jr. (1952)
Arithmetical predicates	Kleene (1955)
FORTRAN	Backus et al. (1956-57)
ALGOL 58	Bauer et al. (1958)
LISP	McCarthy (1958)
Combinatory Logic. Volume I.	Curry-Feys-Craig (1958)
Adjoint functors	Kan (1958)
Recursive functionals & quantifiers, I.&II.	Kleene (1959-63)
Countable functionals	Kleene-Kreisel (1959)

The 1960s

Recursive procedures	Dijkstra (1960)
ALGOL 60	Backus et al. (1960)
Elementary formal systems	Smullyan (1961)
Grothendieck topologies	M.Artin (1962)
Higher-type λ-definability	Kleene (1962)
Grothendieck topoi	Grothendieck et al. SGA 4 (1963-64-72)
CPL	Strachey, et al. (1963)
Functorial semantics	Lawvere (1963)
Continuations (1)	van Wijngaarden (1964)
Adjoint functors & triples	Eilenberg-Moore (1965)
•Cartesian closed categories•	Eilenberg-Kelly (1966)
ISWIM *& SECD machine*	Landin (1966)
CUCH *& combinator programming*	Böhm (1966)
New foundations of recursion theory	Platek (1966)
Normalization Theorem	Tait (1967)
AUTOMATH *& dependent types*	de Bruijn (1967)
Finite-type computable functionals	Gandy (1967)
ALGOL 68	van Wijngaarden (1968)
Normal-form discrimination	Böhm (1968)
Category of sets	Lawvere (1969)
Typed domain logic	Scott (1969-93)
Domain-theoretic λ-models	Scott (1969)
Formulae-as-types	Howard (1969 -1980)
Adjointness in foundations	Lawvere (1969)

Theorem. The category of $\mathbf{T_0}$-topological spaces and continuous functions is ***not*** cartesian closed.

Theorem. The category of $\mathbf{T_0}$-topological spaces ***with*** an equivalence relation and continuous functions ***respecting*** equivalences ***is*** cartesian closed.

Cartesian closed categories give us the algebraic version of typed λ-calculus.

The 1970s

Continuations (2)	Mazurkiewicz (1970)
Continuations (3)	F. Lockwood Morris (1970)
Continuations (4)	Wadsworth (1970)
Categorical logic	Joyal (1970+)
Elementary topoi	Lawvere-Tierney (1970)
Denotational semantics	Scott-Strachey (1970)
Coherence in closed categories	Kelly (1971)
Quantifiers and sheaves	Lawvere (1971)
Martin-Löf type theory	Martin-Löf (1971)
System F, Fω	Girard (1971)
Logic for Computable Functions	Milner (1972)
From sheaves to logic	Reyes (1974)
Polymorphic λ-calculus	Reynolds (1974)
Call-by-name, call-by-value	Plotkin (1975)
Modeling Processes	Milner (1975)
SASL	Turner (1975)
Scheme	Sussman-Steele (1975-80)
Functional programming & FP	Backus (1977)
First-order categorical logic	Makkai-Reyes (1977)
Edinburgh LCF	Milner et al. (1978)
Let-polymorphic type inference	Milner (1978)
Intersection types	Coppo-Dezani (1978)
ML	Milner et al. (1979)
**-Autonomous categories*	Barr (1979)
Sheaves and logic	Fourman-Scott (1979)

This decade saw the importance of constructive logic, the applications to language design and semantics, and the connections to category theory become much clearer.

The 1980s

Frege structures	Aczel (1980)
HOPE	Burstall et al. (1980)
The Lambda Calculus Book	Barendregt (1981-84)
Structural Operational Semantics	Plotkin (1981)
Effective Topos	Hyland (1982)
Dependent types & modularity	Burstall-Lampson (1984)
Locally CCC & type theory	Seely (1984)
Calculus of Constructions	Coquand-Huet (1985)
Bounded quantification	Cardelli-Wegner (1985)
NUPRL	Constable et al. (1986)
Higher-order categorical logic	Lambek-P.J.Scott (1986)
Cambridge LCF	Paulson (1987)
Linear logic	Girard et al. (1987-89)
HOL	Gordon (1988)
FORSYTHE	Reynolds (1988)
Proofs and Types	Girard et al. (1989)
Integrating logical & categorical types	Gray (1989)
Computational λ-calculus & monads	Moggi (1989)

Type theory, resource logic, and computer-assisted theorem proving finally became practical during these years.

The 1990s

HASKELL	Hudak-Hughes-Peyton Jones-Wadler (1990)
Higher-type recursion theory	Sacks (1990)
STANDARD ML	Milner, et al. (1990-97)
Lazy λ-calculus	Abramsky (1990)
Higher-order subtyping	Cardelli-Longo (1991)
Categories, Types and Structure	Asperti-Longo (1991)
STANDARD ML of NJ	MacQueen-Appel (1991-98)
QUEST	Cardelli (1991)
Edinburgh LF	Harper, et al. (1992)
Pi-Calculus	Milner-Parrow-Walker (1992)
Categorical combinators	Curien (1993)
Translucent types & modular	Harper-Lillibridge (1994)
Full abstraction for PCF	Hyland-Ong/Abramsky, et al. (1995)
Algebraic set theory	Joyal-Moerdijk (1995)
Object Calculus	Abadi-Cardelli (1996)
Typed intermediate languages	Tarditi, Morrisett, et al. (1996)
Proof-carrying code	Necula-Lee (1996)
Computability and totality in domains	Berger (1997)
Typed assembly language	Morrisett, et al. (1998)
Type theory via exact categories	Birkedal, et al. (1998)
Categorification	Baez (1998)

Abstract ideas now found many applications in language implementation and in compiling.

The New Millennium

Predicative topos	Moerdijk-Palmgren (2000)
Sketches of an Elephant	Johnstone (2002+)
Differential λ-calculus	Ehrhard/Regnier (2003)
Modular Structural Operational Semantics	Mosses (2004)
A λ-calculus for real analysis	Taylor (2005+)
Homotopy type theory	Awodey-Warren (2006)
Univalence axiom	Voevodsky (2006+)
The safe λ-calculus	Ong, et al. (2007)
Higher topos theory	Lurie (2009)
Functional Reactive Programming	Hudak, et al. (2010)

Univalent Foundations Program @ IAS & HoTT Book
Voevodsky, et al. (2012-13)

In the natural world, convergent evolution can give creatures analogous structures — even though they cannot mate. But, in the intellectual world, analogous structures can be taken advantage of through interfertilization of areas and in finding new applications.

And that we have seen happen with the λ-calculus many, many times over the years.