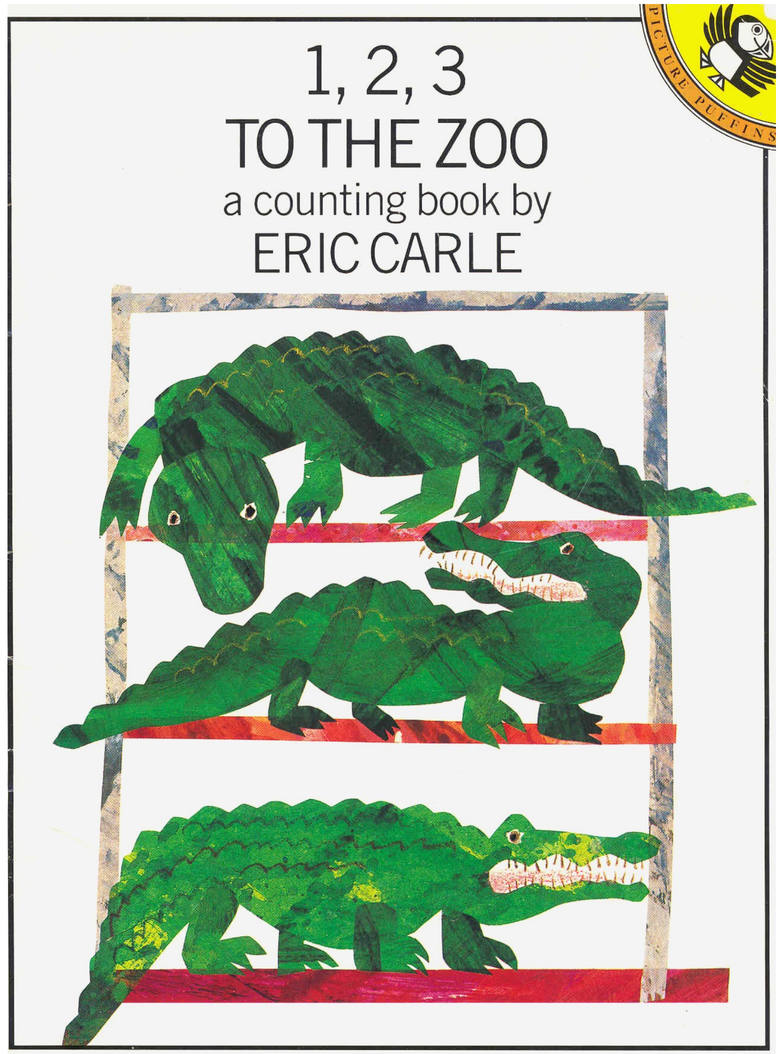
As Natural as 0, 1, 2

Philip Wadler
University of Edinburgh
wadler@inf.ed.ac.uk

Part 0

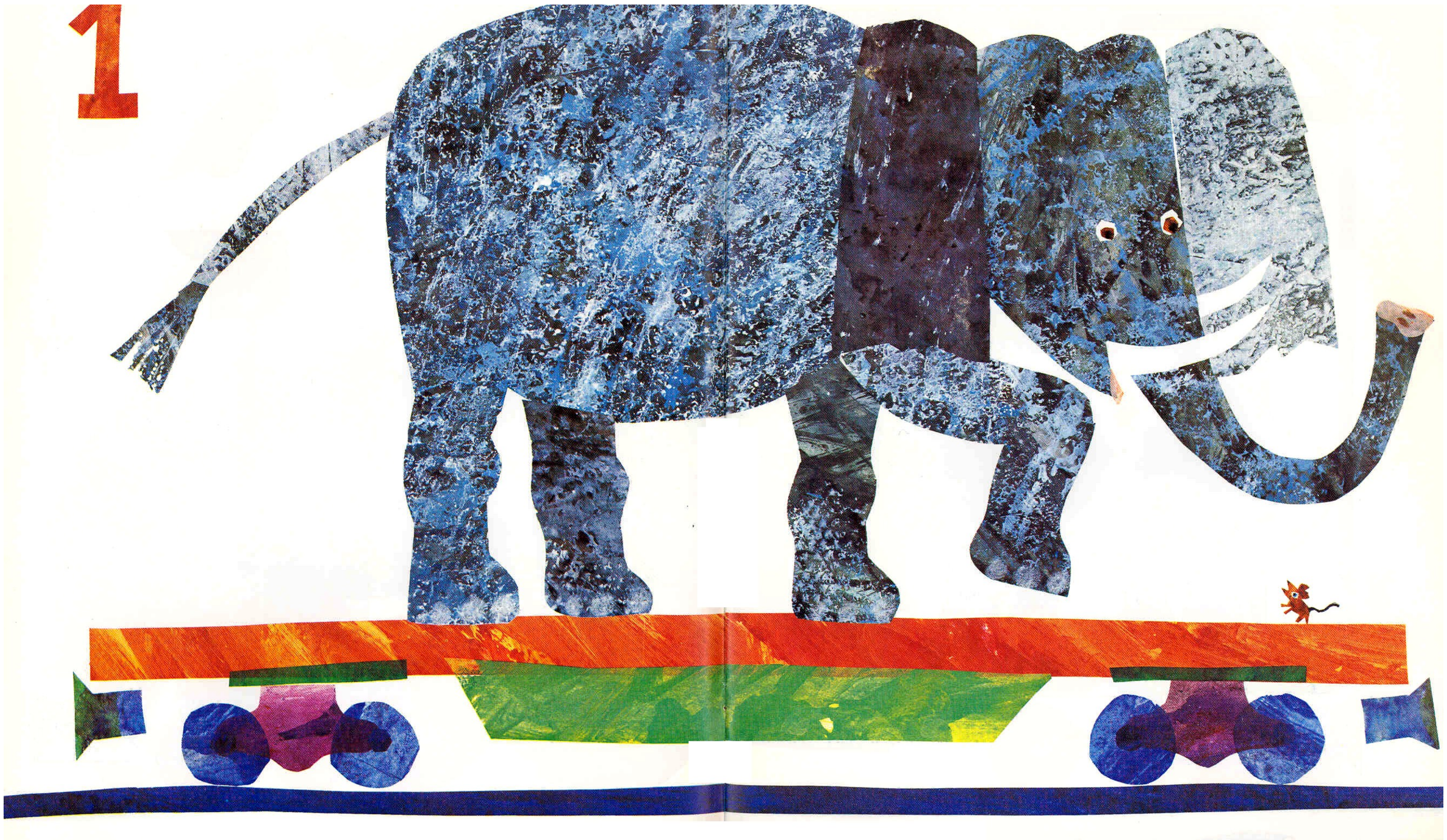
Counting starts at zero

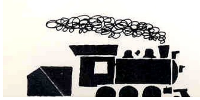
Carle



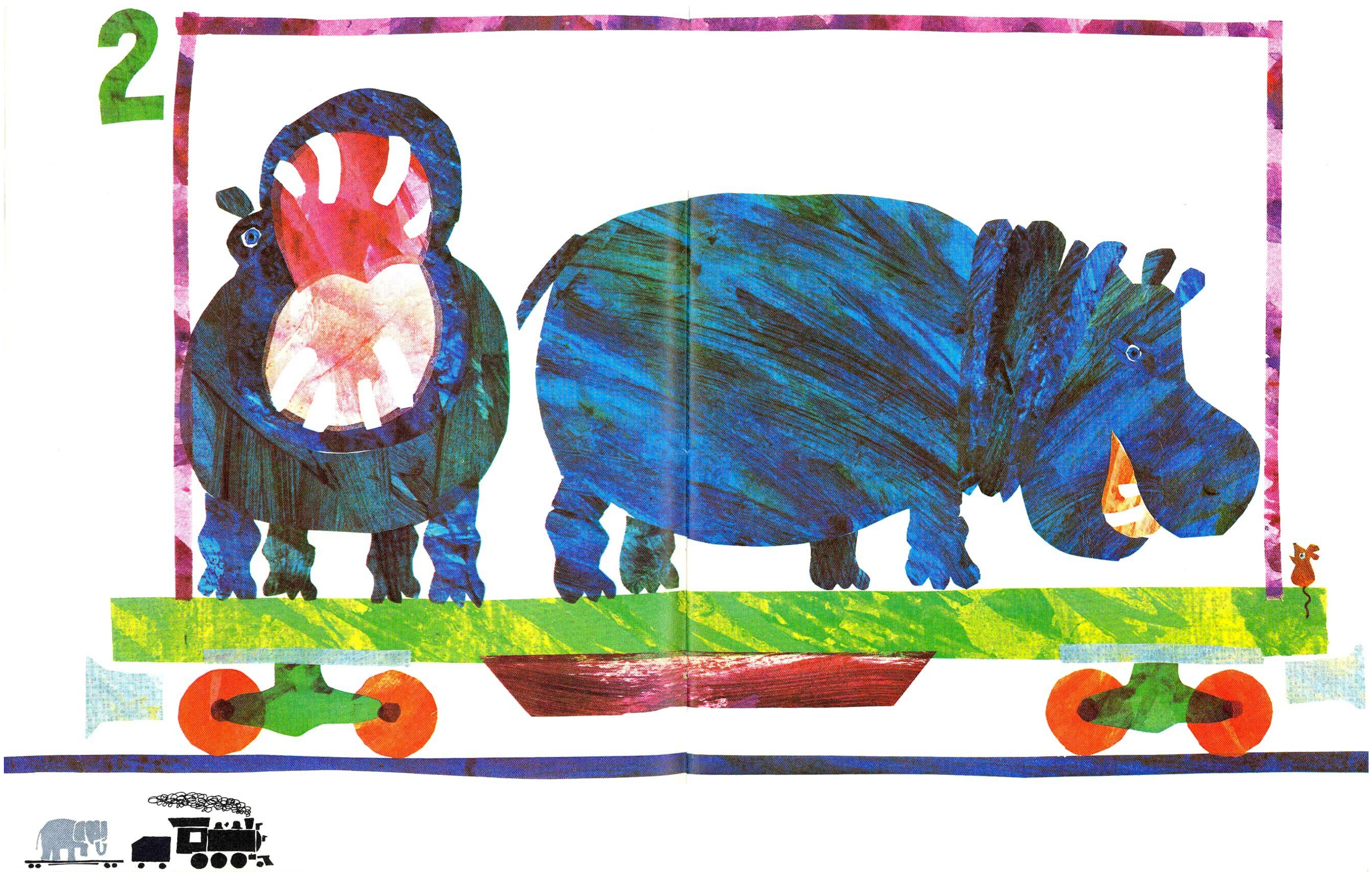
Carle

Carle

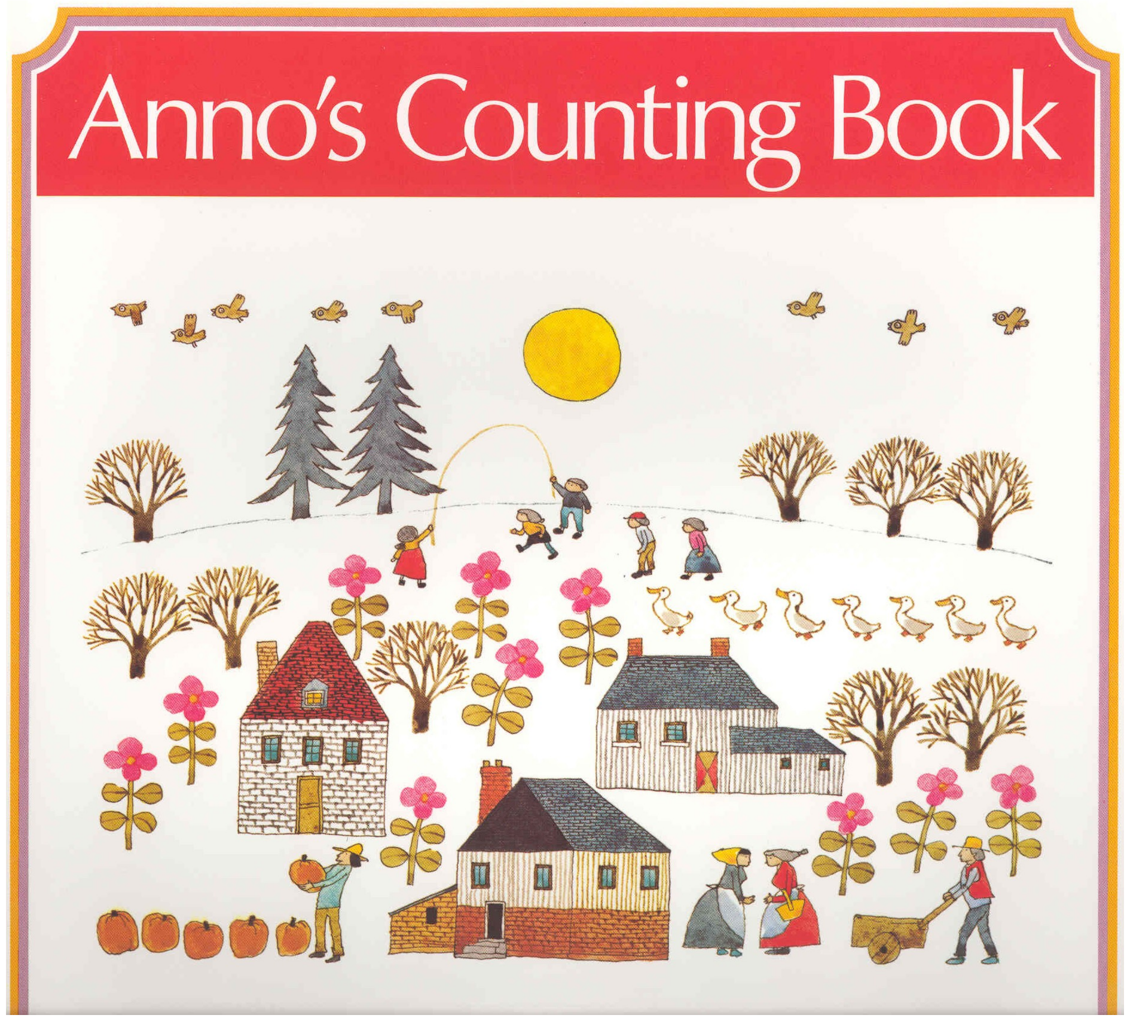




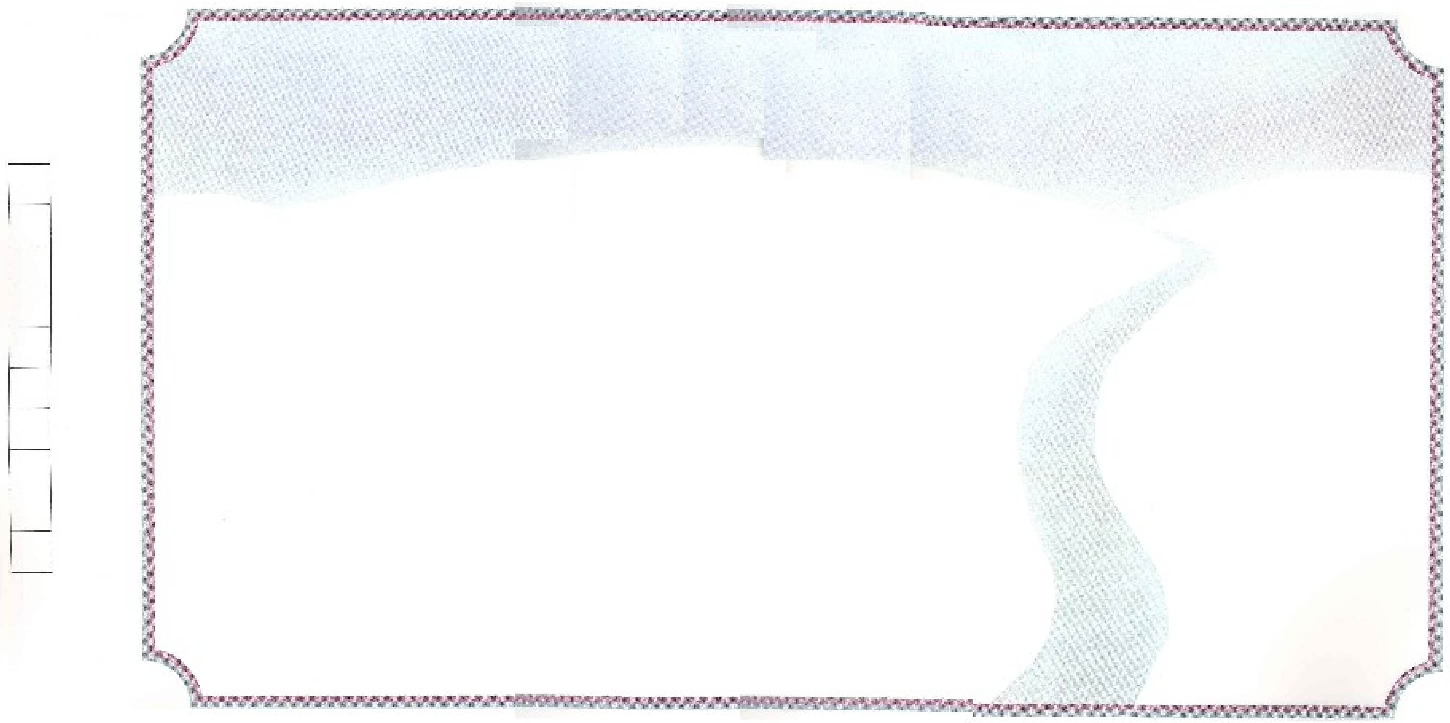
Carle



Anno



Anno



0

Anno

1

Anno

2

Zero appears in India, 7th century CE

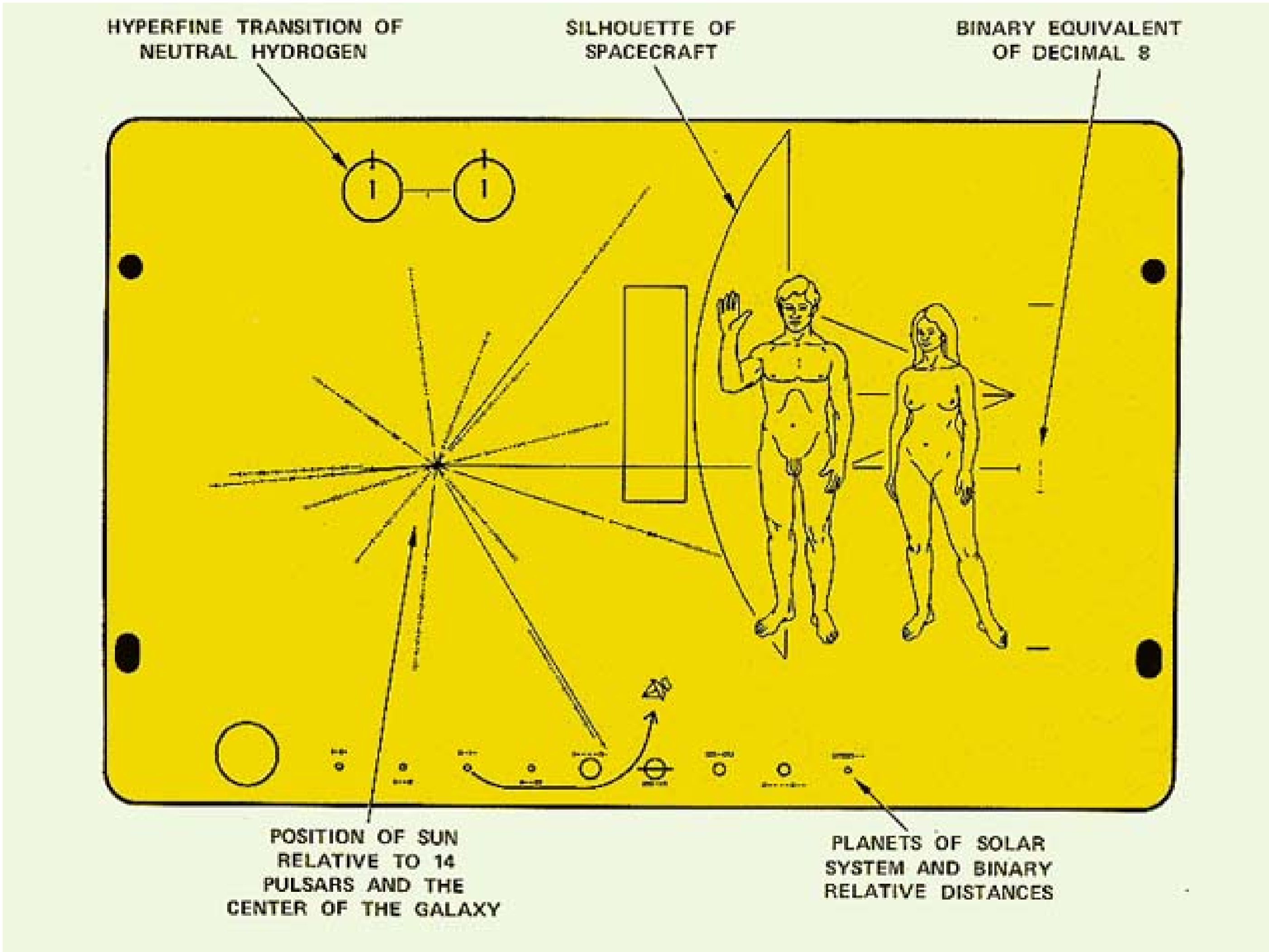
Counting in Japanese

0	leā	零	零
1	īchī	一	一
2	nī	二	二
3	san	三	三
4	shī	四	四
5	gō	五	五

Part 1

Aliens

How to talk to aliens



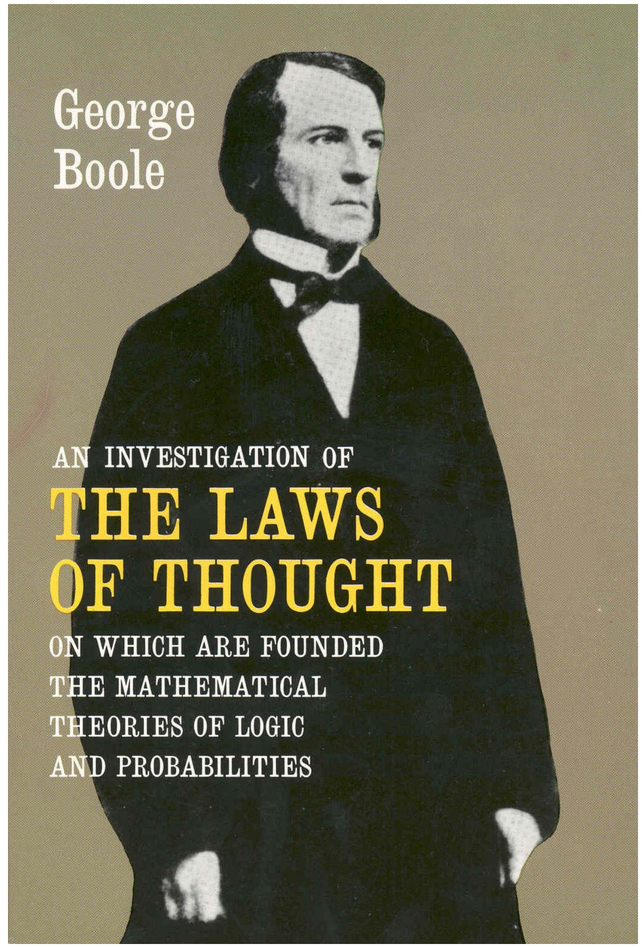
Independence Day

A universal programming language?

Part 2

Boolean algebra

George Boole (1815–1864)



Boole 1847: Mathematical analysis of logic

The primary canonical forms already determined for the expression of Propositions, are

All Xs are Ys,	$x(1-y)=0,$	....A.
No Xs are Ys,	$xy=0,$	....E.
Some Xs are Ys,	$v=xy,$	....I.
Some Xs are not Ys,	$v=x(1-y)$	....O.

On examining these, we perceive that E and I are symmetrical with respect to x and y, so that x being changed into y, and y into x, the equations remain unchanged. Hence E and I may be interpreted into

No Ys are Xs,
Some Ys are Xs,

respectively. Thus we have the known rule of the Logicians, that particular
26|27 affirmative and universal negative Propositions admit of simple conversion. |

Boole 1854: Laws of Thought

PROPOSITION IV.

That axiom of metaphysicians which is termed the principle of contradiction, and which affirms that it is impossible for any being to possess a quality, and at the same time not to possess it, is a consequence of the fundamental law of thought, whose expression is $x^2 = x$.

Let us write this equation in the form

$$x - x^2 = 0,$$

whence we have

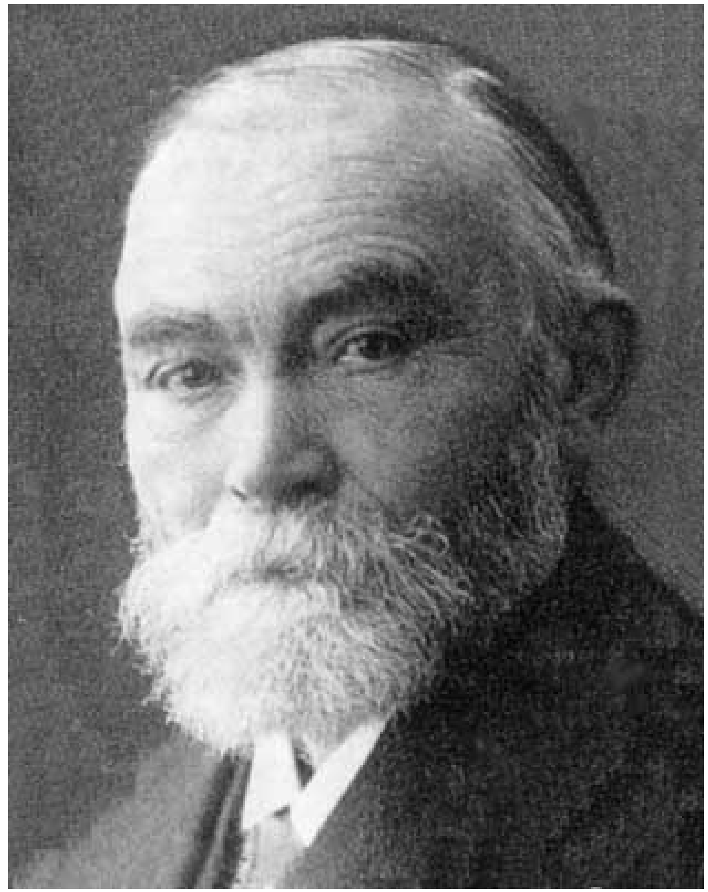
$$x(1 - x) = 0; \tag{1}$$

both these transformations being justified by the axiomatic laws of combination and transposition (II. 13). Let us, for simplicity

Part 3

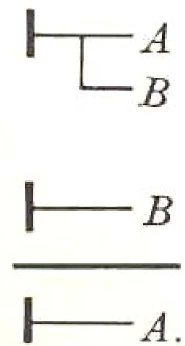
Frege's *Begriffsschrift*

Gotlob Frege (1848–1925)



Frege 1879 — *modus ponens*

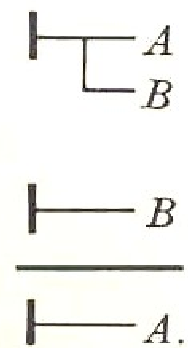
We could write this inference perhaps as follows:



This would become awkward if long expressions were to take the places of A and B, since each of them would have to be written twice. That is why I use the following

Frege 1879 — *modus ponens*

We could write this inference perhaps as follows:

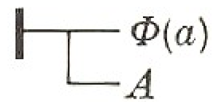


This would become awkward if long expressions were to take the places of A and B, since each of them would have to be written twice. That is why I use the following

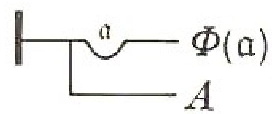
$$\frac{\vdash B \supset A \qquad \vdash B}{\vdash A}$$

Frege 1879 — quantification

It is clear also that from



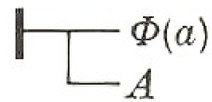
we can derive



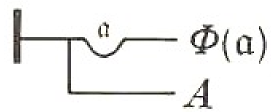
if A is an expression in which a does not occur and if a stands only in the argument places of $\Phi(a)$.[14] If $—\overset{\mathfrak{a}}{\smile}— \Phi(\mathfrak{a})$ is denied, we must be able to specify a meaning for a such that $\Phi(a)$ will be denied. If, therefore, $—\overset{\mathfrak{a}}{\smile}— \Phi(\mathfrak{a})$ were to be denied and

Frege 1879 — quantification

It is clear also that from



we can derive

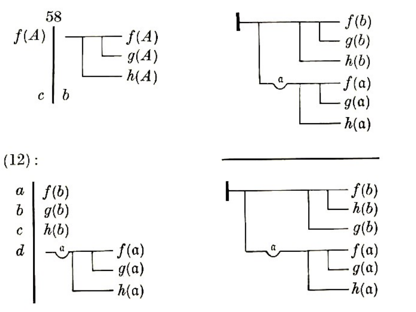


if A is an expression in which a does not occur and if a stands only in the argument places of $\Phi(a)$.[14] If $\overset{\mathfrak{a}}{\smile}\,\Phi(\mathfrak{a})$ is denied, we must be able to specify a meaning for a such that $\Phi(a)$ will be denied. If, therefore, $\overset{\mathfrak{a}}{\smile}\,\Phi(\mathfrak{a})$ were to be denied and

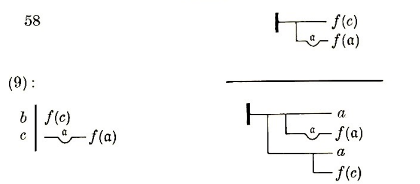
$$\frac{\vdash A \supset \Phi(a)}{\vdash A \supset \forall a.\, \Phi(x)}$$

Frege 1879

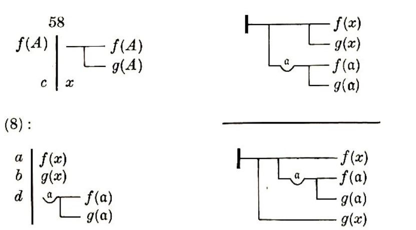
We see how this judgment replaces one mode of inference, namely, Felapton or Fesapo, between which we do not distinguish here since no subject has been singled out.



(60).

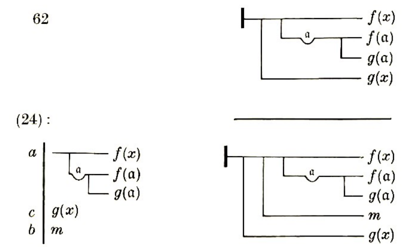


(61).

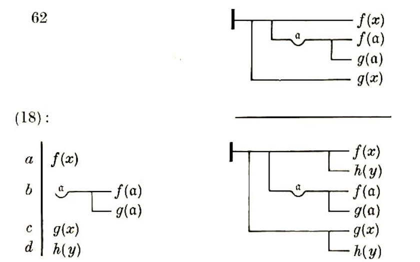


(62).

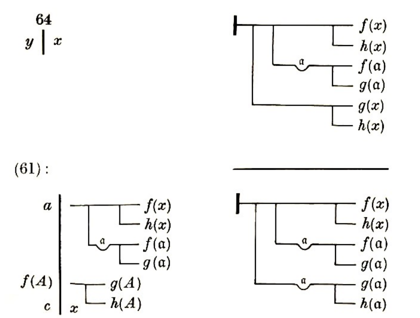
This judgment replaces the mode of inference Barbara when the minor premiss, $g(x)$, has a particular content.



(63).



(64).



(65).

Part 4

Gentzen's Natural Deduction

Gerhard Gentzen (1909–1945)

Gentzen 1934: Natural Deduction

&–*I*

$$\frac{\mathfrak{A} \qquad \mathfrak{B}}{\mathfrak{A} \,\&\, \mathfrak{B}}$$

&–*E*

$$\frac{\mathfrak{A} \,\&\, \mathfrak{B}}{\mathfrak{A}} \qquad \frac{\mathfrak{A} \,\&\, \mathfrak{B}}{\mathfrak{B}}$$

∨–*I*

$$\frac{\mathfrak{A}}{\mathfrak{A} \vee \mathfrak{B}} \qquad \frac{\mathfrak{B}}{\mathfrak{A} \vee \mathfrak{B}}$$

∨–*E*

$$\frac{\mathfrak{A} \vee \mathfrak{B} \qquad \begin{array}{c}[\mathfrak{A}] \\ \mathfrak{C}\end{array} \qquad \begin{array}{c}[\mathfrak{B}] \\ \mathfrak{C}\end{array}}{\mathfrak{C}}$$

∀–*I*

$$\frac{\mathfrak{F}a}{\forall \mathfrak{x}\, \mathfrak{F}\mathfrak{x}}$$

∀–*E*

$$\frac{\forall \mathfrak{x}\, \mathfrak{F}\mathfrak{x}}{\mathfrak{F}a}$$

∃–*I*

$$\frac{\mathfrak{F}a}{\exists \mathfrak{x}\, \mathfrak{F}\mathfrak{x}}$$

∃–*E*

$$\frac{\exists \mathfrak{x}\, \mathfrak{F}\mathfrak{x} \qquad \begin{array}{c}[\mathfrak{F}a] \\ \mathfrak{C}\end{array}}{\mathfrak{C}}$$

⊃–*I*

$$\frac{\begin{array}{c}[\mathfrak{A}] \\ \mathfrak{B}\end{array}}{\mathfrak{A} \supset \mathfrak{B}}$$

⊃–*E*

$$\frac{\mathfrak{A} \qquad \mathfrak{A} \supset \mathfrak{B}}{\mathfrak{B}}$$

¬–*I*

$$\frac{\begin{array}{c}[\mathfrak{A}] \\ \curlywedge\end{array}}{\neg \mathfrak{A}}$$

¬–*E*

$$\frac{\mathfrak{A} \quad \neg \mathfrak{A}}{\curlywedge} \qquad \frac{\curlywedge}{\mathfrak{D}} \; .$$

Gentzen 1934: Natural Deduction

$$\dfrac{\begin{matrix}[A]^x \\ \vdots \\ B\end{matrix}}{A \supset B}\ \supset\text{-I}^x \qquad \dfrac{A \supset B \qquad A}{B}\ \supset\text{-E}$$

$$\dfrac{A \qquad B}{A \mathbin{\&} B}\ \&\text{-I} \qquad \dfrac{A \mathbin{\&} B}{A}\ \&\text{-E}_0 \qquad \dfrac{A \mathbin{\&} B}{B}\ \&\text{-E}_1$$

A proof

$$\dfrac{\dfrac{[B \mathbin{\&} A]^z}{A}\,\&\text{-E}_1 \qquad \dfrac{[B \mathbin{\&} A]^z}{B}\,\&\text{-E}_0}{\dfrac{A \mathbin{\&} B}{(B \mathbin{\&} A) \supset (A \mathbin{\&} B)}\,\supset\text{-I}^z}\,\&\text{-I}$$

Simplifying proofs

$$\dfrac{\dfrac{\begin{matrix}[A]^x\\ \vdots\\ B\end{matrix}}{A \supset B}\,\supset\text{-I}^x \qquad \begin{matrix}\vdots\\ A\end{matrix}}{B}\,\supset\text{-E} \qquad\Rightarrow\qquad \begin{matrix}\vdots\\ A\\ \vdots\\ B\end{matrix}$$

$$\dfrac{\dfrac{\begin{matrix}\vdots\\ A\end{matrix} \qquad \begin{matrix}\vdots\\ B\end{matrix}}{A \,\&\, B}\,\&\text{-I}}{A}\,\&\text{-E}_0 \qquad\Rightarrow\qquad \begin{matrix}\vdots\\ A\end{matrix}$$

Simplifying a proof

$$
\dfrac{\dfrac{\dfrac{\dfrac{[B \mathbin{\&} A]^z}{A}\,\&\text{-E}_1 \qquad \dfrac{[B \mathbin{\&} A]^z}{B}\,\&\text{-E}_0}{A \mathbin{\&} B}\,\&\text{-I}}{(B \mathbin{\&} A) \supset (A \mathbin{\&} B)}\,\supset\text{-I}^z \qquad \dfrac{[B]^y \quad [A]^x}{B \mathbin{\&} A}\,\&\text{-I}}{A \mathbin{\&} B}\,\supset\text{-E}
$$

Simplifying a proof

$$\dfrac{\dfrac{\dfrac{\dfrac{[B \mathbin{\&} A]^z}{A}\,\&\text{-E}_1 \qquad \dfrac{[B \mathbin{\&} A]^z}{B}\,\&\text{-E}_0}{A \mathbin{\&} B}\,\&\text{-I}}{(B \mathbin{\&} A) \supset (A \mathbin{\&} B)}\,\supset\text{-I}^z \qquad \dfrac{[B]^y \quad [A]^x}{B \mathbin{\&} A}\,\&\text{-I}}{A \mathbin{\&} B}\,\supset\text{-E}$$

$$\Downarrow$$

$$\dfrac{\dfrac{\dfrac{[B]^y \quad [A]^x}{B \mathbin{\&} A}\,\&\text{-I}}{A}\,\&\text{-E}_1 \qquad \dfrac{\dfrac{[B]^y \quad [A]^x}{B \mathbin{\&} A}\,\&\text{-I}}{B}\,\&\text{-E}_0}{A \mathbin{\&} B}\,\&\text{-I}$$

Simplifying a proof

$$\dfrac{\dfrac{\dfrac{\dfrac{[B \,\&\, A]^z}{A}\,\&\text{-E}_1 \qquad \dfrac{[B \,\&\, A]^z}{B}\,\&\text{-E}_0}{A \,\&\, B}\,\&\text{-I}}{(B \,\&\, A) \supset (A \,\&\, B)}\supset\text{-I}^z \qquad \dfrac{[B]^y \quad [A]^x}{B \,\&\, A}\,\&\text{-I}}{A \,\&\, B}\supset\text{-E}$$

$$\Downarrow$$

$$\dfrac{\dfrac{\dfrac{[B]^y \quad [A]^x}{B \,\&\, A}\,\&\text{-I}}{A}\,\&\text{-E}_1 \qquad \dfrac{\dfrac{[B]^y \quad [A]^x}{B \,\&\, A}\,\&\text{-I}}{B}\,\&\text{-E}_0}{A \,\&\, B}\,\&\text{-I}$$

$$\Downarrow$$

$$\dfrac{[A]^x \quad [B]^y}{A \,\&\, B}\,\&\text{-I}$$

Part 5

Church's Lambda Calculus

Alonzo Church (1903–1995)

Church 1932: Lambda Calculus

An occurrence of a variable **x** in a given formula is called an occurrence of **x** as a *bound variable* in the given formula if it is an occurrence of **x** in a part of the formula of the form λ**x**[**M**]; that is, if there is a formula **M** such that λ**x**[**M**] occurs in the given formula and the occurrence of **x** in question is an occurrence in λ**x**[**M**]. All other occurrences of a variable in a formula are called occurrences as a *free variable*.

A formula is said to be *well-formed* if it is a variable, or if it is one

Lambda

Reduction rules

$$
\begin{aligned}
(\lambda x.\, u)\, t &\quad\Rightarrow\quad u[t/x] \\
\langle t, u \rangle_0 &\quad\Rightarrow\quad t \\
\langle t, u \rangle_1 &\quad\Rightarrow\quad u
\end{aligned}
$$

Simplifying a term

$$(\lambda z.\, \langle z_1, z_0 \rangle)\, \langle y, x \rangle$$

Simplifying a term

$$(\lambda z.\, \langle z_1, z_0 \rangle)\, \langle y, x \rangle$$

$$\Downarrow$$

$$\langle \langle y, x \rangle_1, \langle y, x \rangle_0 \rangle$$

Simplifying a term

$$(\lambda z.\, \langle z_1, z_0 \rangle)\, \langle y, x \rangle$$

$$\Downarrow$$

$$\langle \langle y, x \rangle_1, \langle y, x \rangle_0 \rangle$$

$$\Downarrow$$

$$\langle x, y \rangle$$

Church 1940: Typed Lambda Calculus

$$\dfrac{\begin{array}{c}[x : A]^x \\ \vdots \\ u : B\end{array}}{\lambda x.\, u : A \supset B}\ \supset\text{-I}^x \qquad \dfrac{s : A \supset B \qquad t : A}{s\, t : B}\ \supset\text{-E}$$

$$\dfrac{t : A \qquad u : B}{\langle t, u\rangle : A \,\&\, B}\ \&\text{-I} \qquad \dfrac{s : A \,\&\, B}{s_0 : A}\ \&\text{-E}_0 \qquad \dfrac{s : A \,\&\, B}{s_1 : B}\ \&\text{-E}_1$$

A program

$$
\dfrac{\dfrac{\dfrac{[z : B \mathbin{\&} A]^z}{z_1 : A}\,\&\text{-E}_1 \qquad \dfrac{[z : B \mathbin{\&} A]^z}{z_0 : B}\,\&\text{-E}_0}{\langle z_1, z_0 \rangle : A \mathbin{\&} B}\,\&\text{-I}}{\lambda z.\, \langle z_1, z_0 \rangle : (B \mathbin{\&} A) \supset (A \mathbin{\&} B)}\,\supset\text{-I}^z
$$

Simplifying programs

$$
\dfrac{\dfrac{\begin{matrix}[x : A]^x \\ \vdots \\ u : B\end{matrix}}{\lambda x.\, u : A \supset B}\ \supset\text{-I}^x \qquad \begin{matrix}\vdots \\ t : A\end{matrix}}{(\lambda x.\, u)\, t : B}\ \supset\text{-E} \qquad \Rightarrow \qquad \begin{matrix}\vdots \\ t : A \\ \vdots \\ u[t/x] : B\end{matrix}
$$

$$
\dfrac{\dfrac{\begin{matrix}\vdots \\ t : A\end{matrix} \qquad \begin{matrix}\vdots \\ u : B\end{matrix}}{\langle t, u\rangle : A \,\&\, B}\ \&\text{-I}}{\langle t, u\rangle_0 : A}\ \&\text{-E}_0 \qquad \Rightarrow \qquad \begin{matrix}\vdots \\ t : A\end{matrix}
$$

Simplifying a program

$$\dfrac{\dfrac{\dfrac{\dfrac{[z : B \mathbin{\&} A]^z}{z_1 : A}\,\&\text{-E}_1 \qquad \dfrac{[z : B \mathbin{\&} A]^z}{z_0 : B}\,\&\text{-E}_0}{\langle z_1, z_0\rangle : A \mathbin{\&} B}\,\&\text{-I}}{\lambda z.\,\langle z_1, z_0\rangle : (B \mathbin{\&} A) \supset (A \mathbin{\&} B)}\,\supset\text{-I}^z \qquad \dfrac{[y : B]^y \quad [x : A]^x}{\langle y, x\rangle : B \mathbin{\&} A}\,\&\text{-I}}{(\lambda z.\,\langle z_1, z_0\rangle)\,\langle y, x\rangle : A \mathbin{\&} B}\,\supset\text{-E}$$

Simplifying a program

$$
\dfrac{\dfrac{\dfrac{[z : B \mathbin{\&} A]^z}{z_1 : A}\,\&\text{-E}_1 \qquad \dfrac{[z : B \mathbin{\&} A]^z}{z_0 : B}\,\&\text{-E}_0}{\langle z_1, z_0 \rangle : A \mathbin{\&} B}\,\&\text{-I}}{\lambda z.\, \langle z_1, z_0 \rangle : (B \mathbin{\&} A) \supset (A \mathbin{\&} B)}\,\supset\text{-I}^z \qquad \dfrac{[y : B]^y \quad [x : A]^x}{\langle y, x \rangle : B \mathbin{\&} A}\,\&\text{-I}
$$

$$
\dfrac{\qquad\qquad}{(\lambda z.\, \langle z_1, z_0 \rangle)\, \langle y, x \rangle : A \mathbin{\&} B}\,\supset\text{-E}
$$

$$\Downarrow$$

$$
\dfrac{\dfrac{\dfrac{[y : B]^y \quad [x : A]^x}{\langle y, x \rangle : B \mathbin{\&} A}\,\&\text{-I}}{\langle y, x \rangle_1 : A}\,\&\text{-E}_1 \qquad \dfrac{\dfrac{[y : B]^y \quad [x : A]^x}{\langle y, x \rangle : B \mathbin{\&} A}\,\&\text{-I}}{\langle y, x \rangle_0 : B}\,\&\text{-E}_0}{\langle \langle y, x \rangle_1, \langle y, x \rangle_0 \rangle : A \mathbin{\&} B}\,\&\text{-I}
$$

Simplifying a program

$$\dfrac{\dfrac{\dfrac{\dfrac{[z : B \,\&\, A]^z}{z_1 : A}\,\&\text{-E}_1 \qquad \dfrac{[z : B \,\&\, A]^z}{z_0 : B}\,\&\text{-E}_0}{\langle z_1, z_0\rangle : A \,\&\, B}\,\&\text{-I}}{\lambda z.\,\langle z_1, z_0\rangle : (B \,\&\, A) \supset (A \,\&\, B)}\supset\text{-I}^z \qquad \dfrac{[y : B]^y \quad [x : A]^x}{\langle y, x\rangle : B \,\&\, A}\,\&\text{-I}}{(\lambda z.\,\langle z_1, z_0\rangle)\,\langle y, x\rangle : A \,\&\, B}\supset\text{-E}$$

$$\Downarrow$$

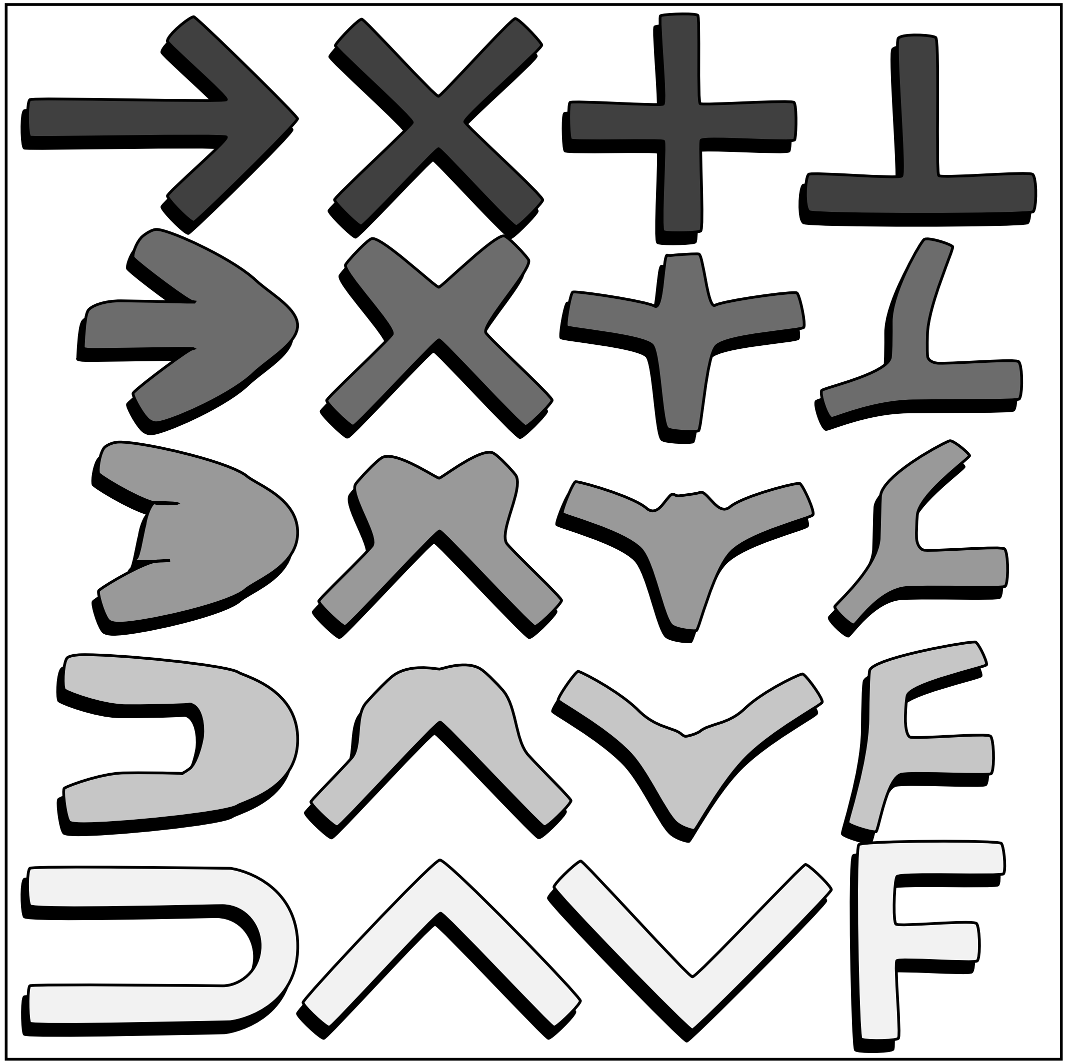
$$\dfrac{\dfrac{\dfrac{[y : B]^y \quad [x : A]^x}{\langle y, x\rangle : B \,\&\, A}\,\&\text{-I}}{\langle y, x\rangle_1 : A}\,\&\text{-E}_1 \qquad \dfrac{\dfrac{[y : B]^y \quad [x : A]^x}{\langle y, x\rangle : B \,\&\, A}\,\&\text{-I}}{\langle y, x\rangle_0 : B}\,\&\text{-E}_0}{\langle\langle y, x\rangle_1, \langle y, x\rangle_0\rangle : A \,\&\, B}\,\&\text{-I}$$

$$\Downarrow$$

$$\dfrac{[x : A]^x \quad [y : B]^y}{\langle x, y\rangle : A \,\&\, B}\,\&\text{-I}$$

Part 6

The Curry-Howard Isomorphism



LC'90 *The Curry-Howard homeomorphism*

Haskell Curry (1900–1982) / William Howard

Howard 1980

THE FORMULAE-AS-TYPES NOTION OF CONSTRUCTION

W. A. Howard

Department of Mathematics, University of Illinois at Chicago Circle, Chicago, Illinois 60680, U.S.A.

Dedicated to H. B. Curry on the occasion of his 80th birthday.

The following consists of notes which were privately circulated in 1969. Since they have been referred to a few times in the literature, it seems worth while to publish them. They have been rearranged for easier reading, and some inessential corrections have been made.

Howard 1980

1. Formulation of the sequent calculus

Let $P(\supset)$ denote positive implicational propositional logic. The prime formulae of $P(\supset)$ are propositional variables. If α and β are formulae, so is $\alpha \supset \beta$. A *sequent* has the form $\Gamma \to \beta$, where Γ is a (possibly empty) finite sequence of formulae and β is a formula. The axioms and rules of inference of $P(\supset)$ are as follows.

(1.1) Axioms: all sequents of the form
$$\alpha \to \alpha$$

(1.2)
$$\frac{\Gamma, \alpha \to \beta}{\Gamma \to \alpha \supset \beta}$$

(1.3)
$$\frac{\Gamma \to \alpha \qquad \Delta \to \alpha \supset \beta}{\Gamma, \Delta \to \beta}$$

(1.4) Thinning, permutation and contraction rules

Howard 1980

2. *Type symbols, terms and constructions*

By a type symbol is meant a formula of $P(\supset)$. We will consider a λ-formalism in which each term has a type symbol α as a superscript (which we may not always write); the term is said to be of type α. The rules of term formation are as follows.

(2.1) Variables X^{α}, Y^{β},... are terms

(2.2) λ-abstraction: from F^{β} get $(\lambda X^{\alpha}.F^{\beta})^{\alpha \supset \beta}$.

(2.3) Application: from $G^{\alpha \supset \beta}$ and H^{α} get $(G^{\alpha \supset \beta} H^{\alpha})^{\beta}$.

Part 7

Programs and Proofs

Programs

- Lisp (McCarthy, 1960)
- Iswim (Landin, 1966)
- Scheme (Steele and Sussman, 1975)
- ML (Milner, Gordon, Wadsworth, 1979)
- Hope (Burstall, MacQueen, Sannella, 1980)
- Miranda (Turner, 1985)
- Haskell (Hudak, Peyton Jones, and Wadler, 1987)
- O'Caml (Leroy, 1996)
- Links (Wadler et al, 2005)

Proofs

- Automath (de Bruijn, 1970)
- Type Theory (Martin Löf, 1975)
- ML/LCF (Milner, Gordon, and Wadsworth, 1979)
- HOL (Gordon and Melham, 1988)
- CoQ (Huet and Coquand, 1988)
- Isabelle (Paulson, 1993)

Proofs/Programs

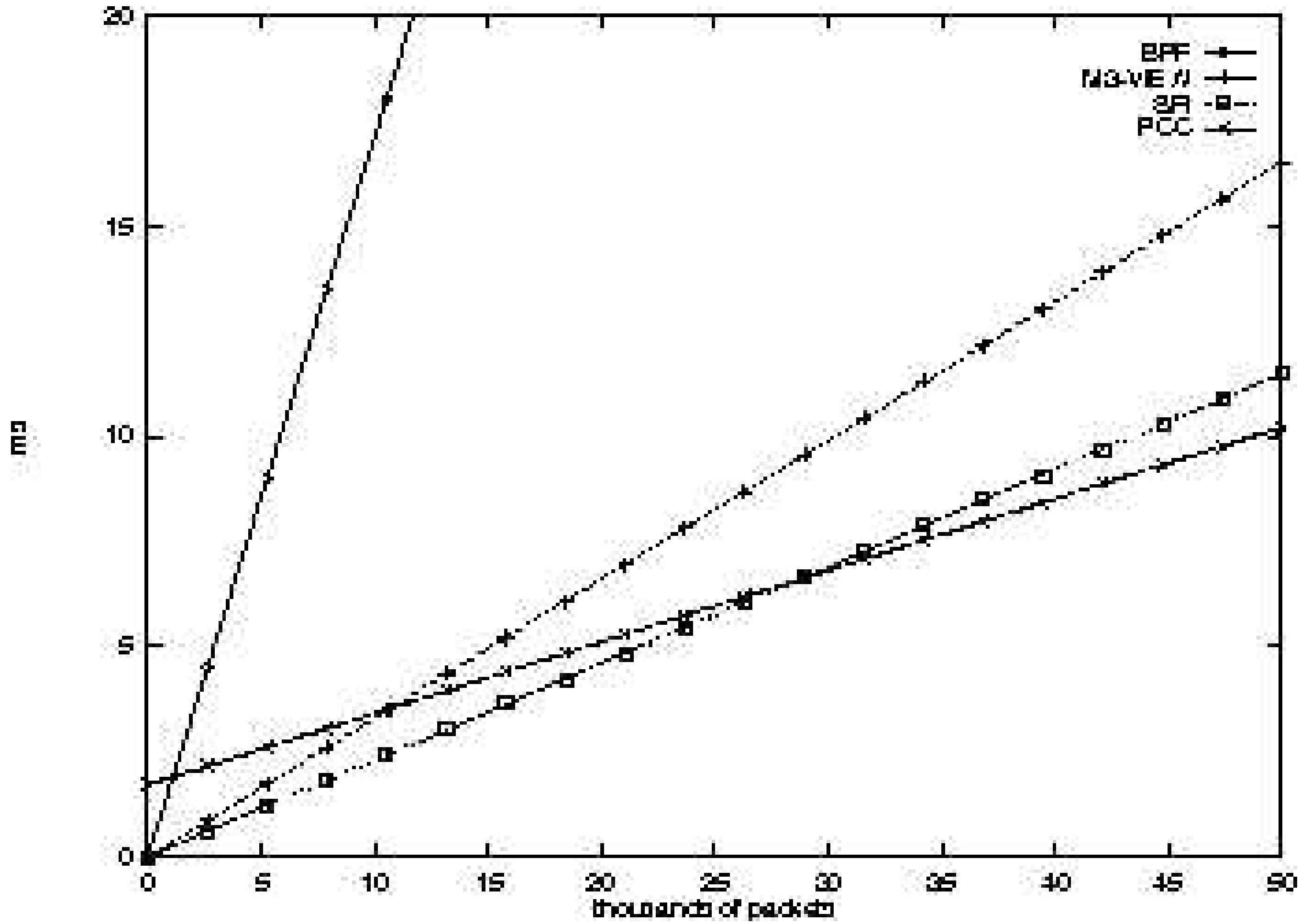
- Hindley/Milner (1969/1975)
- Girard/Reynolds (1972/1975)
- Linear Logic/Syntactic Control of Interference (1987/1978)
- Classical Logic/Continuation-Passing Style (1990)
- And dual to Or/Call-by-value dual to Call-by-name (2000)

Part 8

Programs and Proofs on the Web

Java (Gosling, Joy, and Steele, 1996)

Proof-Carrying Code (Necula and Lee, 1996)



Typed Assembly Language (Morrisett et al 1998)

Typed Assembly Language (Morrisett et al 1998)

What do you want to type check today?

XQuery

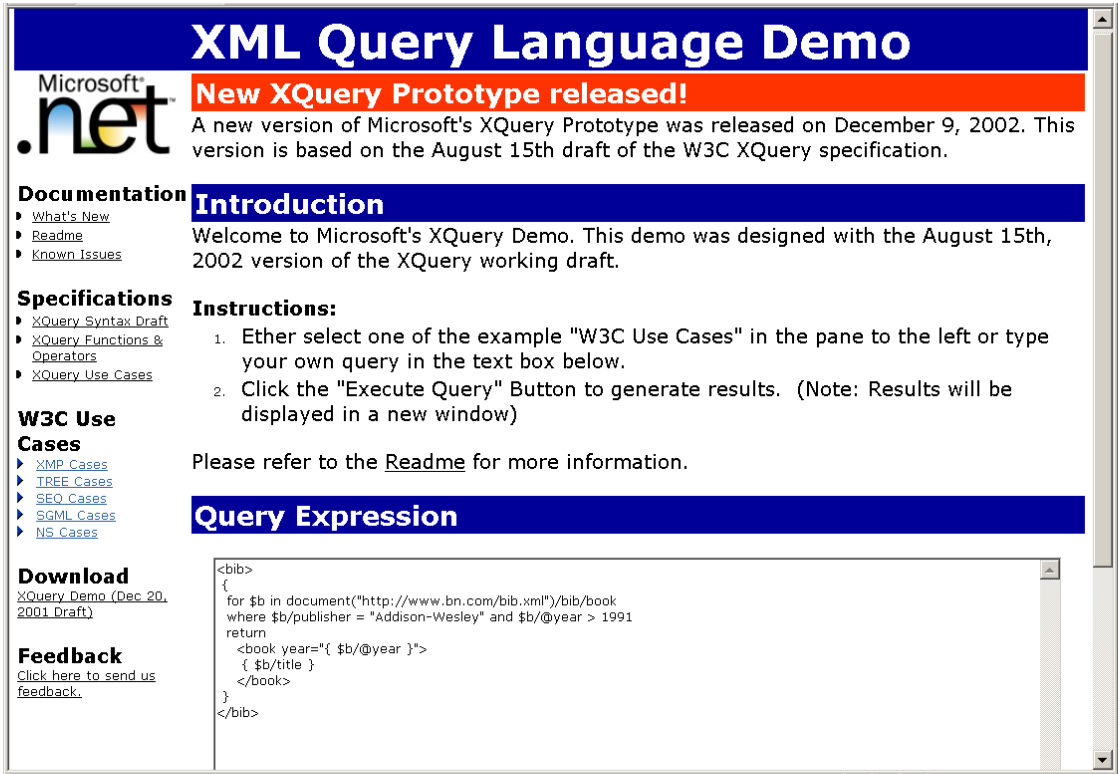
XQuery 1.0: An XML Query Language

W3C Working Draft 16 August 2002

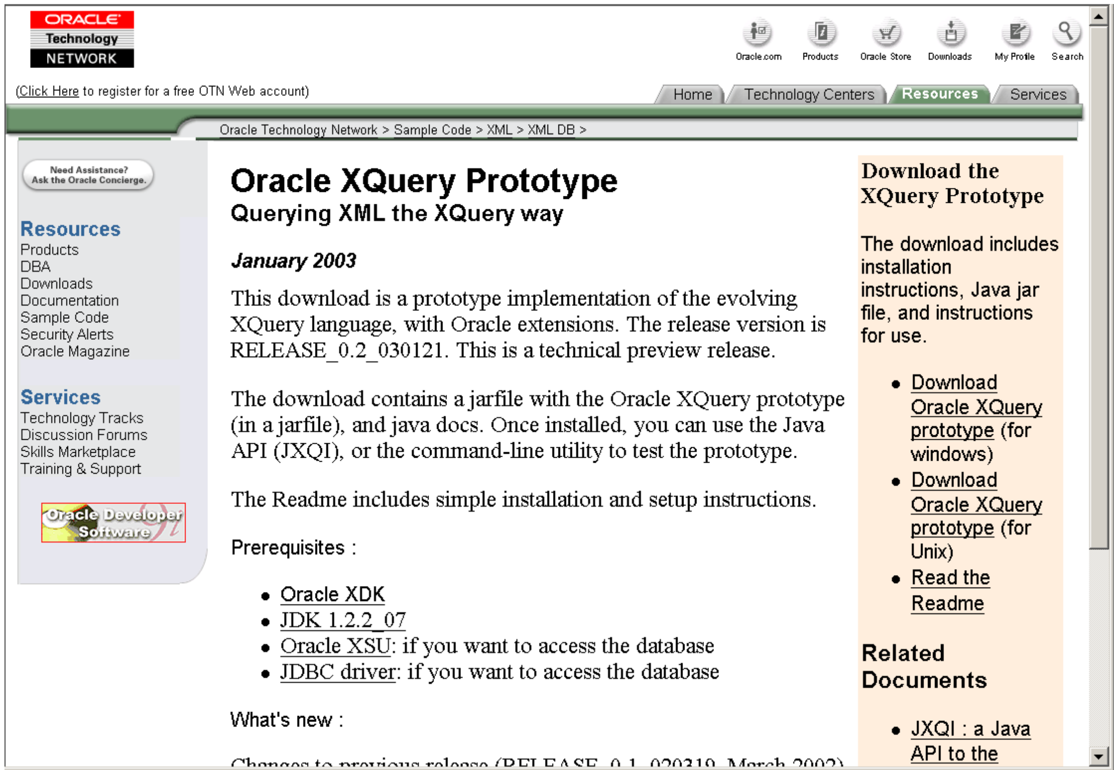
This version:
http://www.w3.org/TR/2002/WD-xquery-20020816/
Latest version:
http://www.w3.org/TR/xquery/
Previous versions:
http://www.w3.org/TR/2002/WD-xquery-20020430/
http://www.w3.org/TR/2001/WD-xquery-20011220/
http://www.w3.org/TR/2001/WD-xquery-20010607/

Editors:
Scott Boag (XSL WG), IBM Research <scott_boag@us.ibm.com>
Don Chamberlin (XML Query WG), IBM Almaden Research Center <chamberlin@almaden.ibm.com>
Mary F. Fernandez (XML Query WG), AT&T Labs <mff@research.att.com>
Daniela Florescu (XML Query WG), XQRL <dana@xqrl.com>
Jonathan Robie (XML Query WG), DataDirect Technologies <jonathan.robie@datadirect-technologies.com>
Jérôme Siméon (XML Query WG), Bell Labs, Lucent Technologies <simeon@research.bell-labs.com>

XQuery - Microsoft



XQuery - Oracle



Part 9

Conclusion

Frege

Undone by Russell's Paradox

Church and Curry

Attended 1982 Conference on
Lisp and Functional Programming

Gentzen

"He once confided in me that he was really quite contented since now he had at last time to think about a consistency proof for analysis."

Frege

Undone by Russell's Paradox

Church and Curry

Attended 1982 Conference on
Lisp and Functional Programming

Gentzen

"He once confided in me that he was really
quite contented since now he had at last time to think
about a consistency proof for analysis."
Died in prison, 4 August 1945

Special thanks to

Martina Sharp, Avaya Labs

for scanning all the pictures

Adam and Leora Wadler

for loaning me their books