Jifeng He at Oxford and Beyond:

An Appreciation

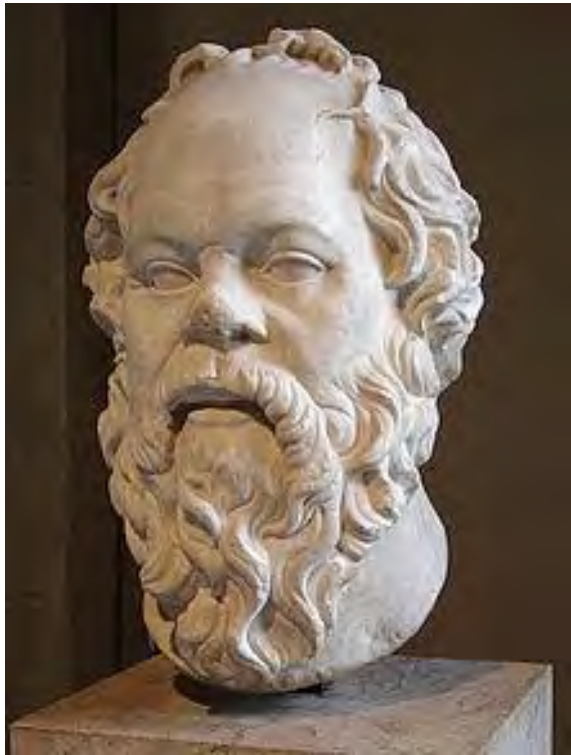
EST 1892 LSBU

Museophile

EAST CHINA NORMAL UNIVERSITY

華東師範大學

Socrates – Σωκράτης



未经审视的生活对于一个人来说是不值得过的

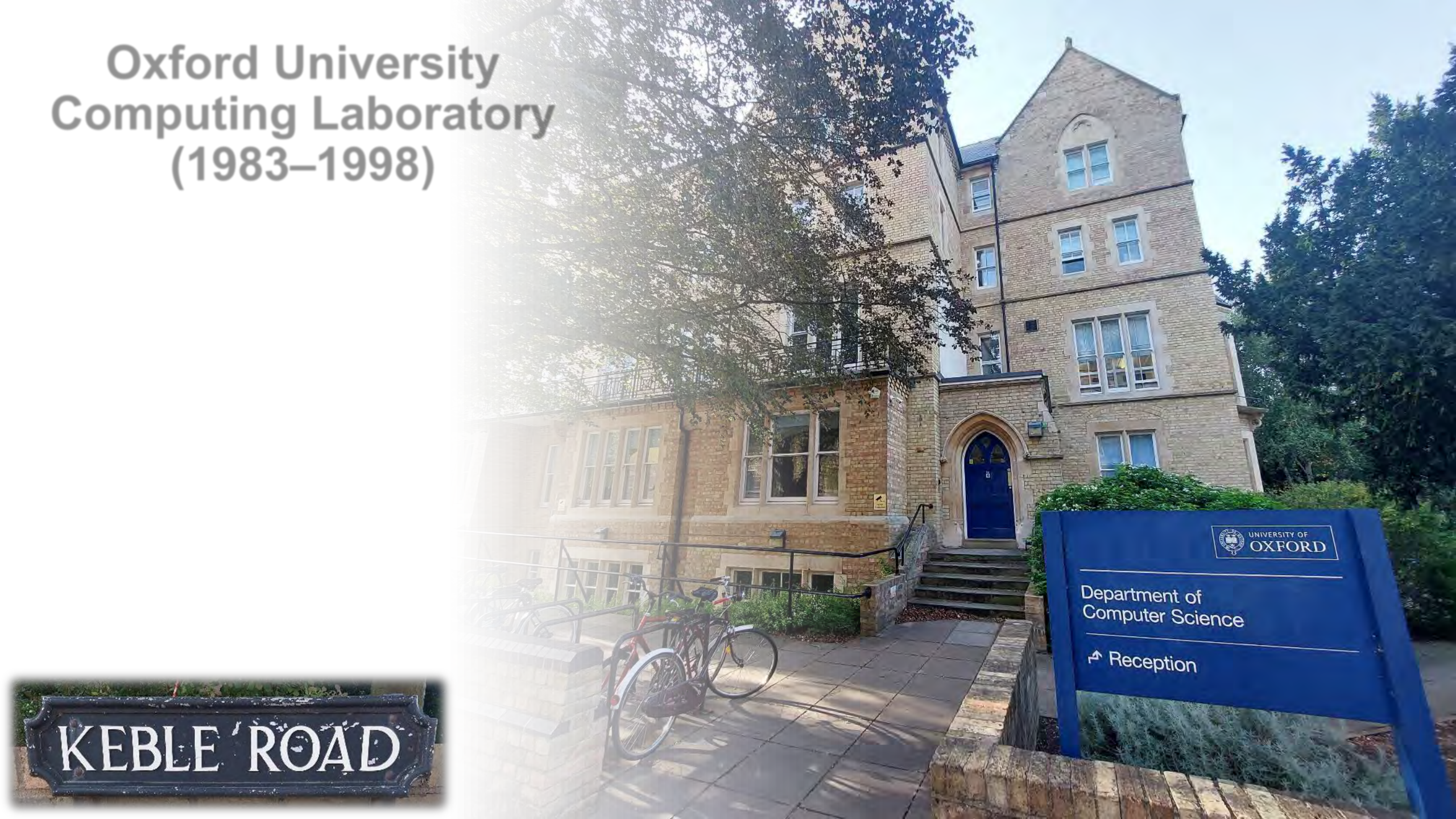
Jifeng He

Fudan University

East China Normal University

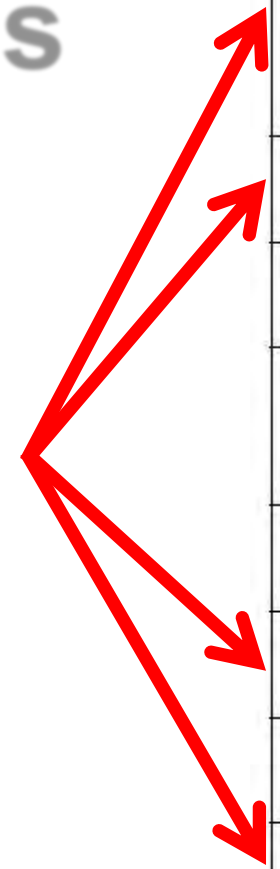
East China Normal University

Oxford University Computing Laboratory (1983–1998)

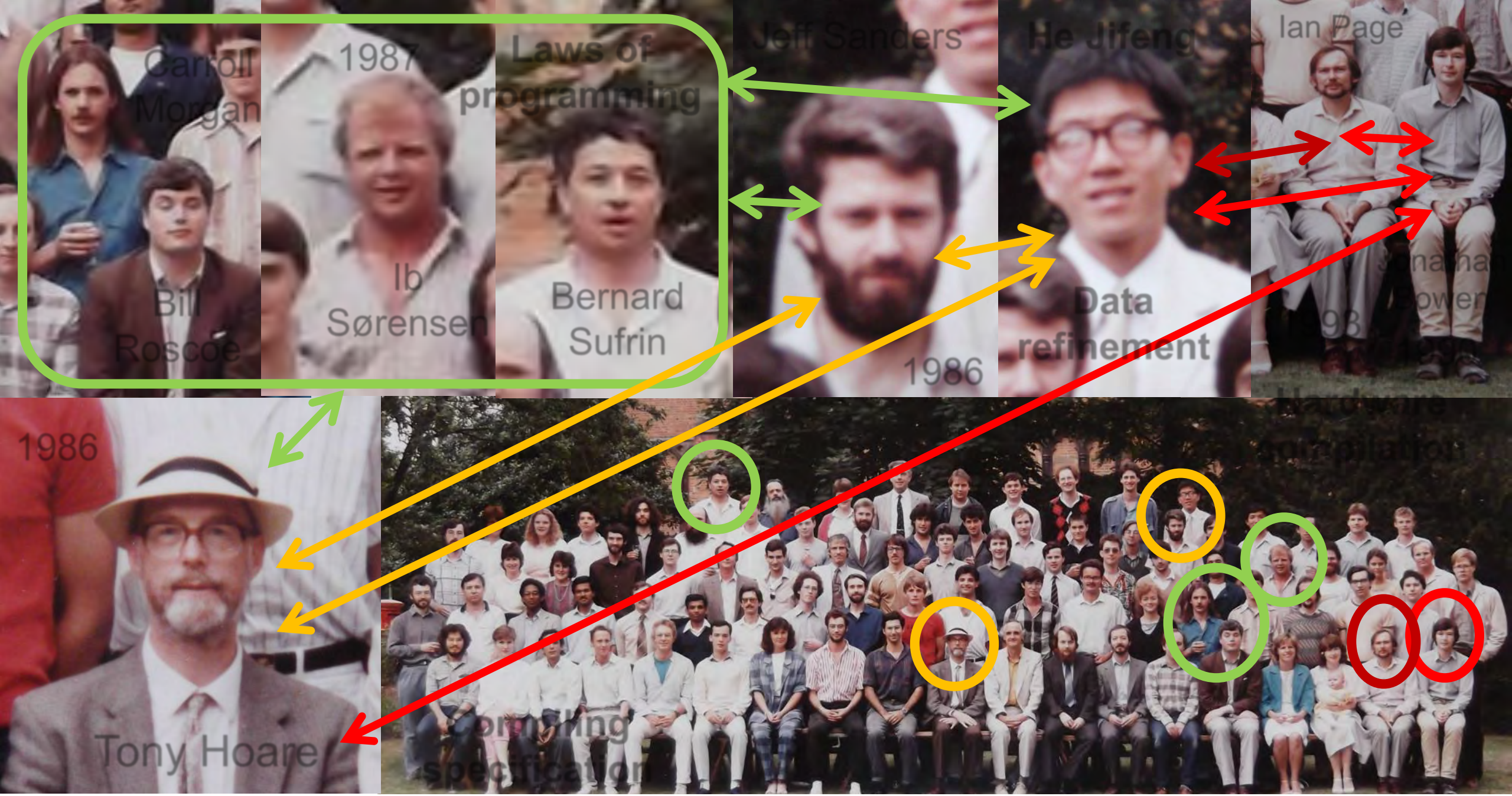



Bernard Sufrin
Jim Woodcock
He Jifeng
Bill Roscoe
Ian Page
Jeff
Sanders
Tony Hoare

Top ten papers on Scopus



No.	Title	Authors	Year
1.	Laws of programming	Tony Hoare, Ian Hayes, Jifeng He, Carroll Morgan, Bill Roscoe, Jeff Sanders, Ib Holm Sørensen, Michael Spivey, and Bernard Sufrin	1987
2.	Data refinement refined – Resume	Jifeng He, Tony Hoare, and Jeff Sanders	1986
3.	Probabilistic models for the Guarded Command Language	Jifeng He, Karen Seidel, and Annabelle McIver	1997
4.	The rely-guarantee method for verifying shared variable concurrent programs	Qiwen Xu, Willem-Paul de Roever, and Jifeng He	1997
5.	rCOS: A refinement calculus of object systems	Jifeng He, Xiaoshan Li, and Zhiming Liu	2006
6.	Prespecification in data refinement	Tony Hoare, Jifeng He, and Jeff Sanders	1987
7.	A formal semantics of UML sequence diagram	Xiaoshan Li, Zhiming Liu, and Jifeng He	2004
8.	The weakest prespecification	Tony Hoare and Jifeng He	1987
9.	Normal form approach to compiler design	Tony Hoare, Jifeng He, and Augusto Sampaio	1993
10.	A process algebraic framework for specification and validation of real-time systems	Adnan Sherif, Ana Cavalcanti, Jifeng He, and Augusto Sampaio	2010


Carroll Morgan
1987
Laws of programming
Jeff Sanders
He Jifeng
Ian Page
Bill Roscoe
Ib Sørensen
Bernard Sufrin
1986
Data refinement
1986
Tony Hoare

AN ALGEBRAIC APPROACH TO VERIFIABLE COMPILING SPECIFICATION AND PROTOTYPING OF THE PROCOS LEVEL 0 PROGRAMMING LANGUAGE

C.A.R. Hoare He Jifeng Jonathan Bowen Paritosh Pandya

Oxford University Computing Laboratory
Programming Research Group

Oxford OX1 3QD
England

Phone: +44-865-273838 Fax: +44-865-273839
E-mail: procos@prg.oxford.ac.uk

SUMMARY. A compiler is specified by a description of how each construct of the source language is translated into a sequence of object code instructions. The meaning of the object code can be defined by an interpreter written in the source language itself. A proof that the compiler is correct must show that interpretation of the object code is at least good (for any relevant purpose) as the corresponding source program. The proof is conducted using standard techniques of data refinement. All the calculations are based on algebraic laws governing the source language. The theorems are expressed in a form close to a logic program, which may used as a compiler prototype, or a check on the results of a particular compilation. A subset of the occam programming language and the transputer instruction set are used to illustrate the approach. An advantage of the method is that it is possible to add new programming constructs without affecting existing development work.

1. Introduction

Compilation is specified as a relation between a source program p and the corresponding object code c. Further details of compilation are given by a symbol table Ψ, mapping the global identifiers of p to storage locations of the target machine. This compilation relation will be abbreviated as a predicate

$$\mathcal{C}\, p\, c\, \Psi \quad .$$

The internal structure of p, c and Ψ will be elaborated as the need arises.

Improvement is a relation between a product q and a product p that holds whenever for any purpose the observable behaviour of q is as good as or better than that of p. More precisely, if q satisfies every specification satisfied by p, and maybe more. For example, if

Time Interval Semantics and Implementation of a Real-Time Programming Language

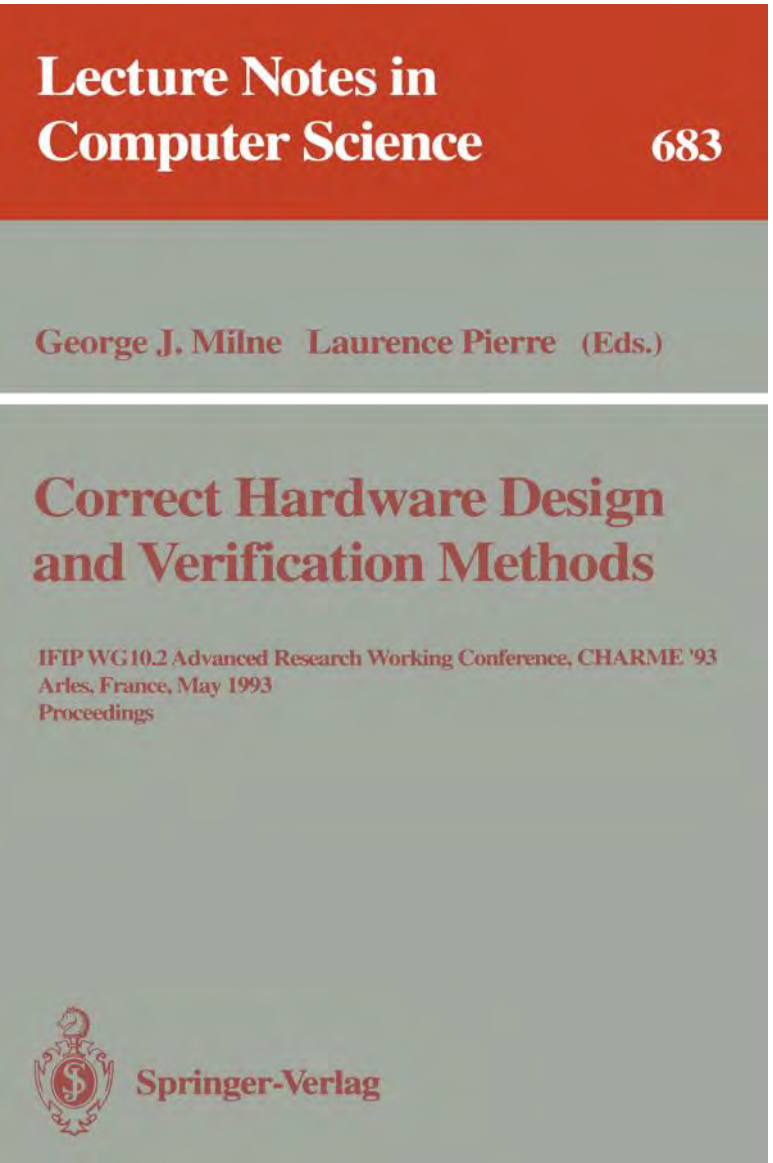
He Jifeng
Jonathan Bowen

Oxford University Computing Laboratory
Programming Research Group
11 Keble Road, Oxford OX1 3QD, UK

Jifeng.He@comlab.ox.ac.uk
Jonathan.Bowen@comlab.ox.ac.uk

Thanks to ESPRIT BRA 3104 **ProCoS** project and UK IED Safemos project (IED3/1/1036).

Hardware compilation



Towards a Provably Correct Hardware Implementation of Occam

He Jifeng*, Ian Page and Jonathan Bowen**

Oxford University Computing Laboratory, Programming Research Group
11 Keble Road, Oxford OX1 3QD, England
Email: {Jifeng.He,Ian.Page,Jonathan.Bowen}@comlab.ox.ac.uk

Abstract. This paper shows how to compile a program written in a subset of occam into a *normal form* suitable for further processing into a netlist of components which may be loaded into a *Field-Programmable Gate Array* (FPGA). A simple state-machine model is adopted for specifying the behaviour of a synchronous circuit where the observable includes the state of the control path and the data path of the circuit. We identify the behaviour of a circuit with a program consisting of a very restricted subset of occam. Algebraic laws are used to facilitate the transformation from a program into a normal form. The compiling specification is presented as a set of theorems that must be proved correct with respect to these laws. A rapid prototype compiler in the form of a logic program may be implemented from these theorems.

PROVABLY CORRECT SYSTEMS

Modelling of Communication Languages and Design of Optimized Compilers

HE JIFENG

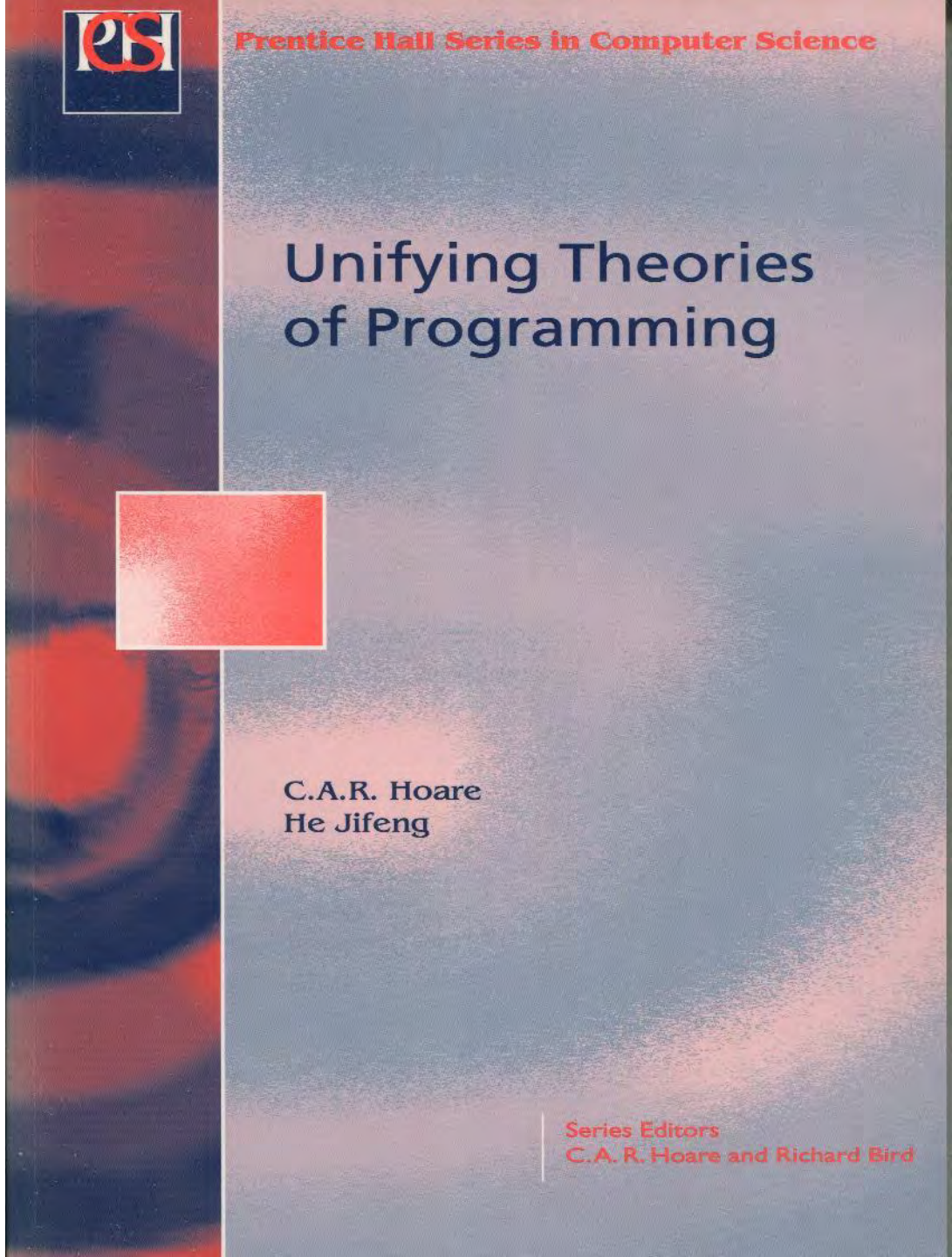
THE McGRAW-HILL INTERNATIONAL SERIES IN SOFTWARE ENGINEERING

ProCoS-WG
ESPRIT Working Group 8694

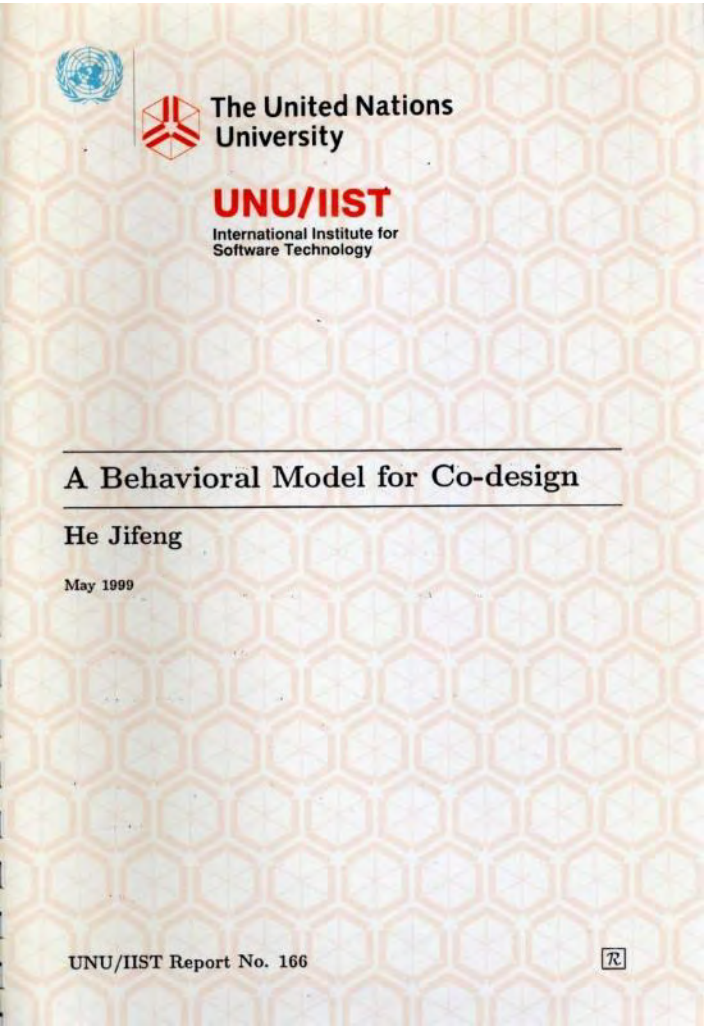
ProCoS Working Group

Unifying Theories of Programming (1998)

Tony Hoare: *"I must emphasise that all the effective research was conducted by Jifeng, who formalized the definitions, postulated the axioms, and proved the theorems. I enjoyed discussing the goal of research with him, and I wrote much of the English prose. But all of the new results were due to him."*



UNU/IIST, Macau (1998–2005)


The United Nations University

UNU/IIST

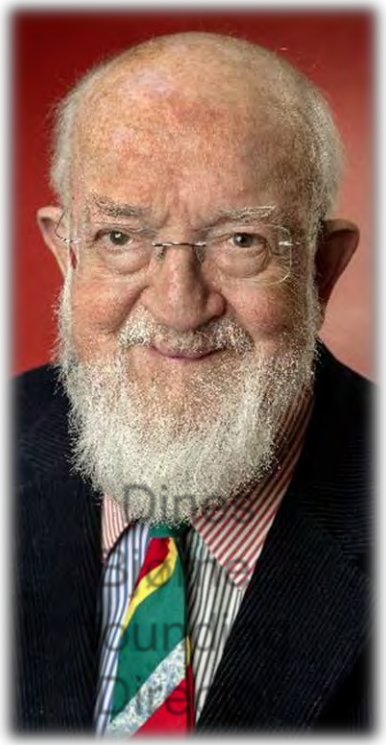
International Institute for Software Technology

A Behavioral Model for Co-design

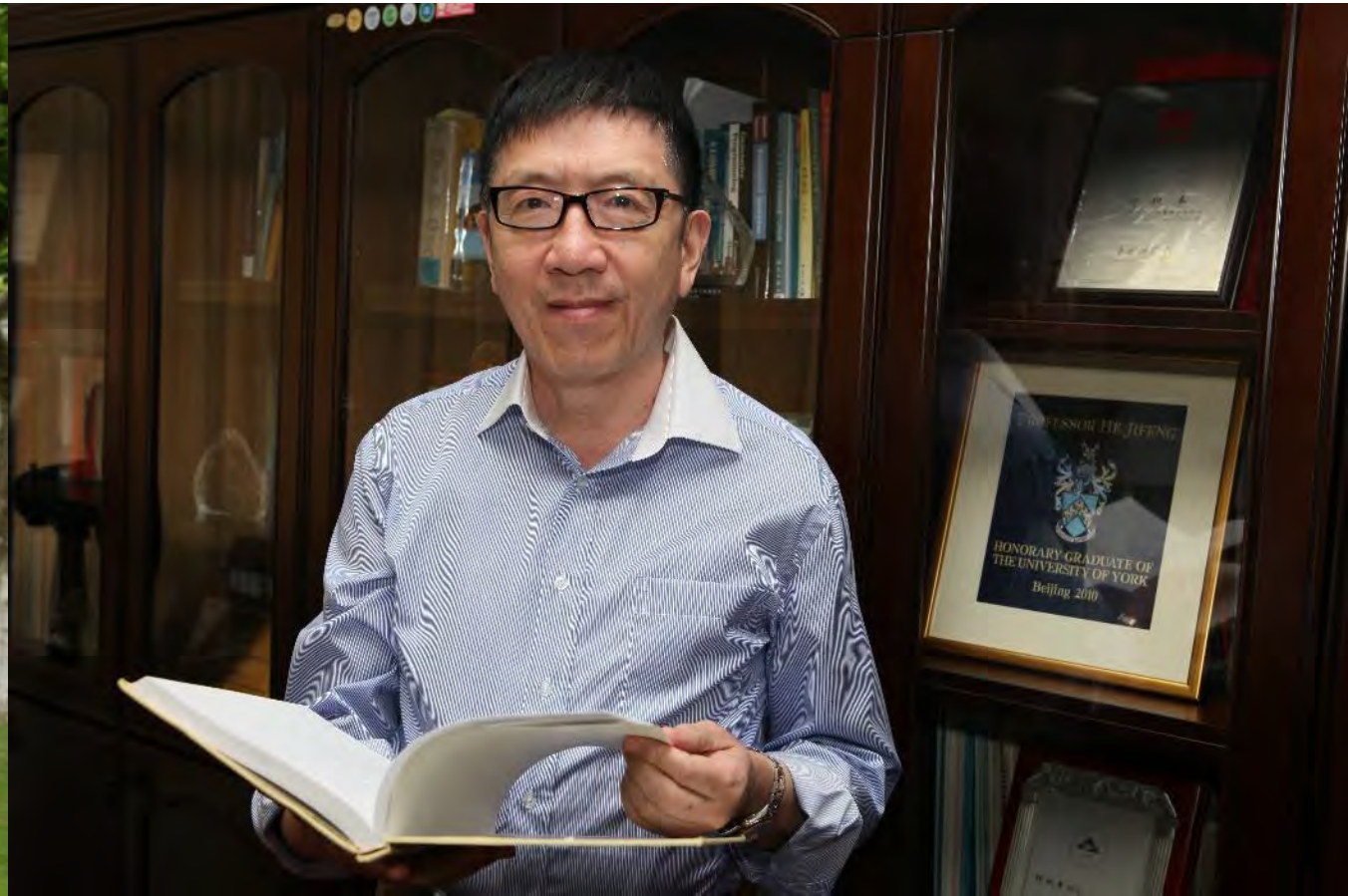
He Jifeng

May 1999

UNU/IIST Report No. 166



Jifeng He

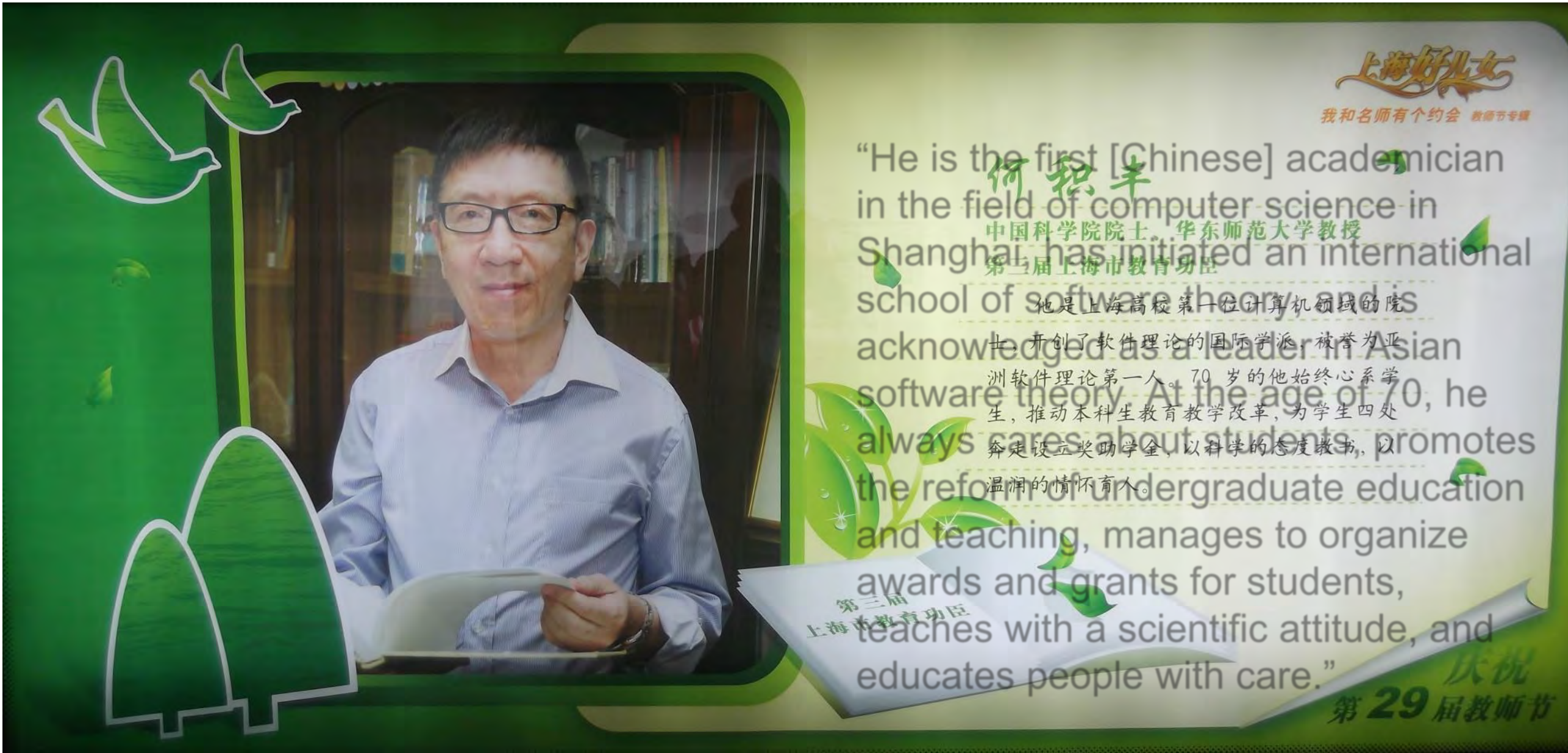


Jifeng He

上海市教育功臣

Shanghai, 2013

Poster in a public walkway



70th birthday Festschrift, 2013

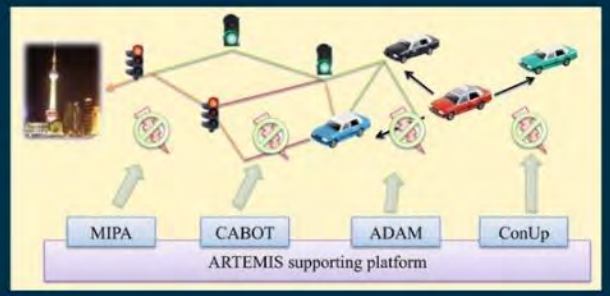
Festschrift

LNCS 8051

Zhiming Liu
Jim Woodcock
Huibiao Zhu (Eds.)

Theories of Programming and Formal Methods

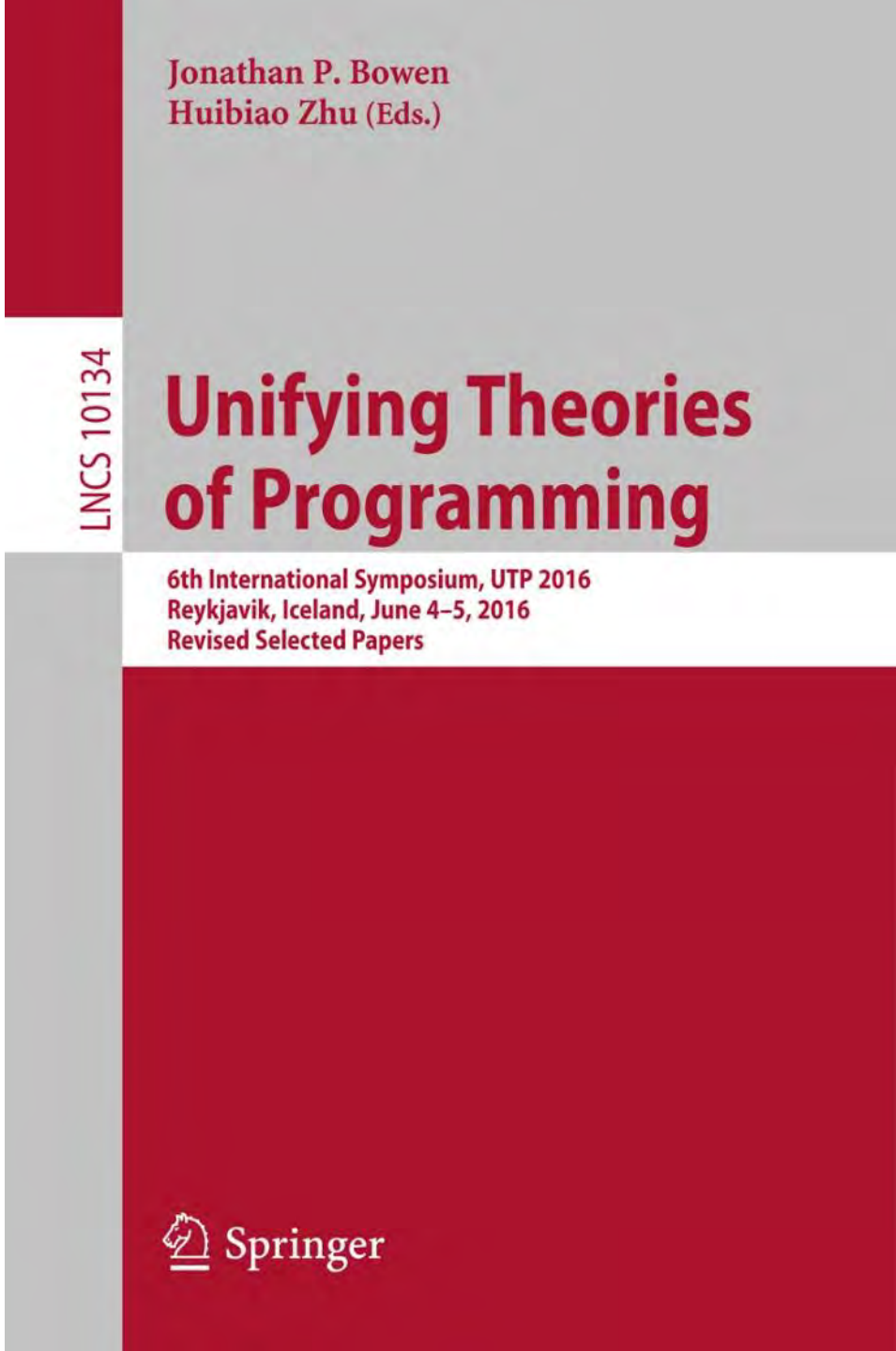
Essays Dedicated to Jifeng He on the Occasion of His 70th Birthday



Springer

Unifying Theories of Programming symposia

He, Jifeng (2016), *A New Roadmap for Linking Theories of Programming*. In UTP 2016, Springer, LNCS 10134, pp. 26–43.



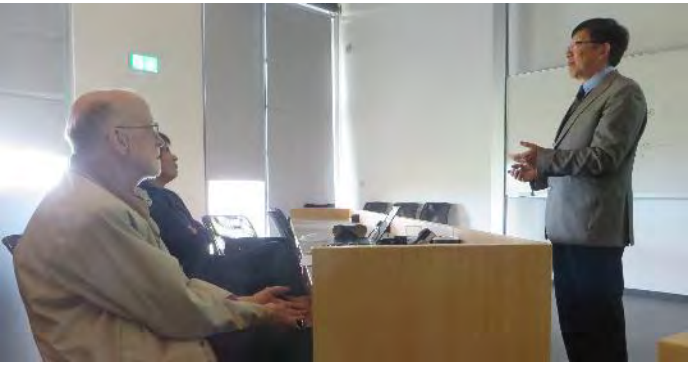
UTP 2016, Reykjavik – photographic memories

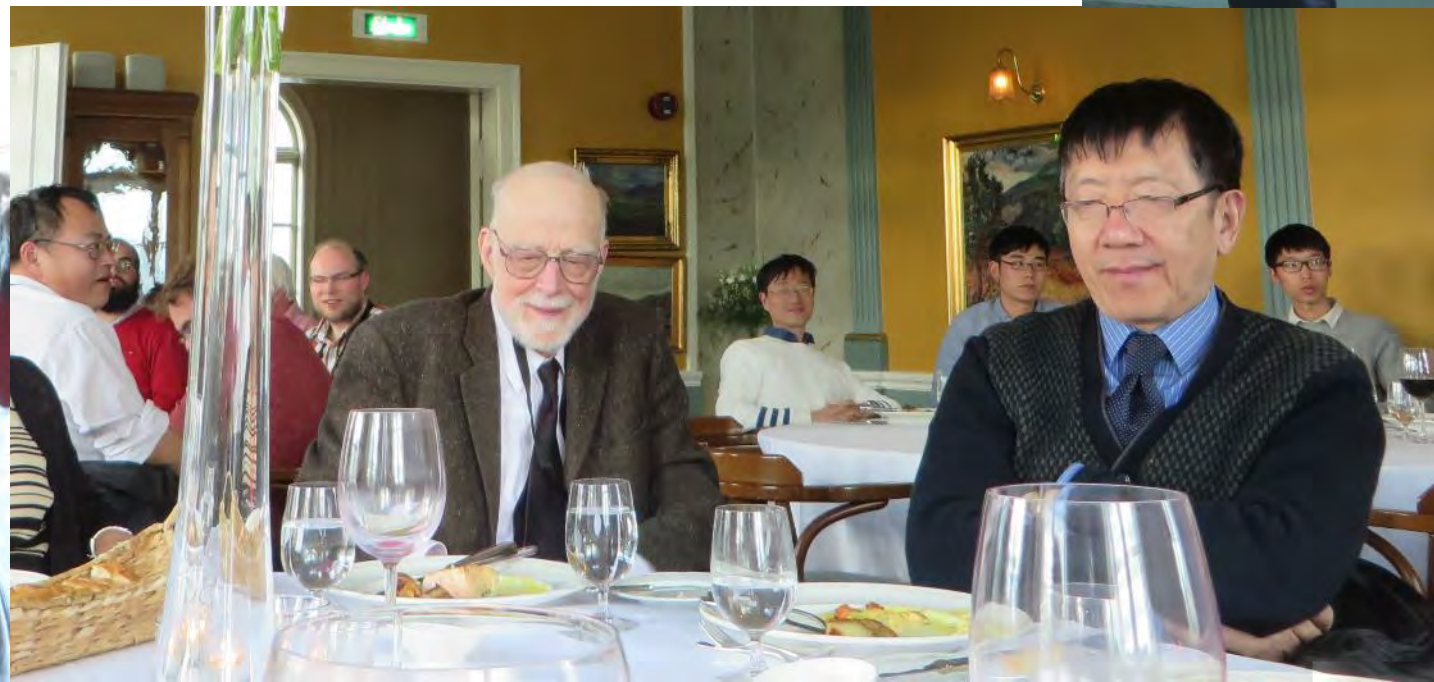
Jifeng's keynote

Tony's keynote

UTP 2016, Reykjavik – photographic memories

UTP 2016, Reykjavik – photographic memories





BCS-FACS, London, 2018

Invited talk by Jifeng He

Commentary by Tony Hoare

Artificial Intelligence

Speaking at the World AI Conference (WAIC) held in Shanghai, Sept. 2022. (Photograph supplied by Kelly He.)

Jifeng He

Jifeng He

国家自然科学奖

证 书

为表彰国家自然科学奖获得者，特颁发此证书。

项目名称：设计严格安全软件的完备演算系统

奖励等级：二等

获 奖 者：何积丰（华东师范大学）

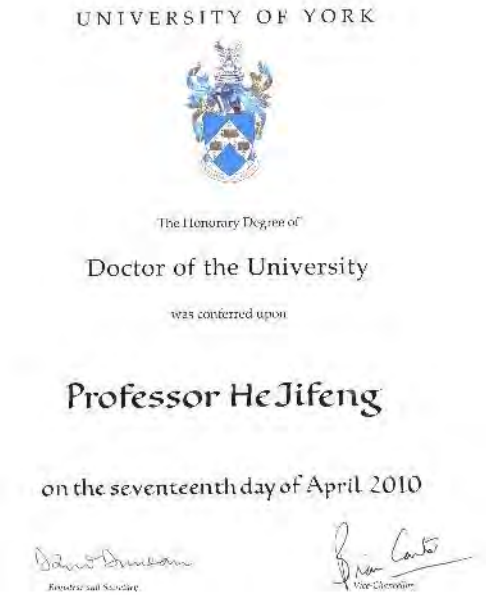
中华人民共和国国务院

2003年[illegible]月[illegible]日

证书号：2002-Z-107-2-03-01

何积丰 ====== 胡锦涛

Jifeng He


UNIVERSITY OF YORK

The Honorary Degree of

Doctor of the University

was conferred upon

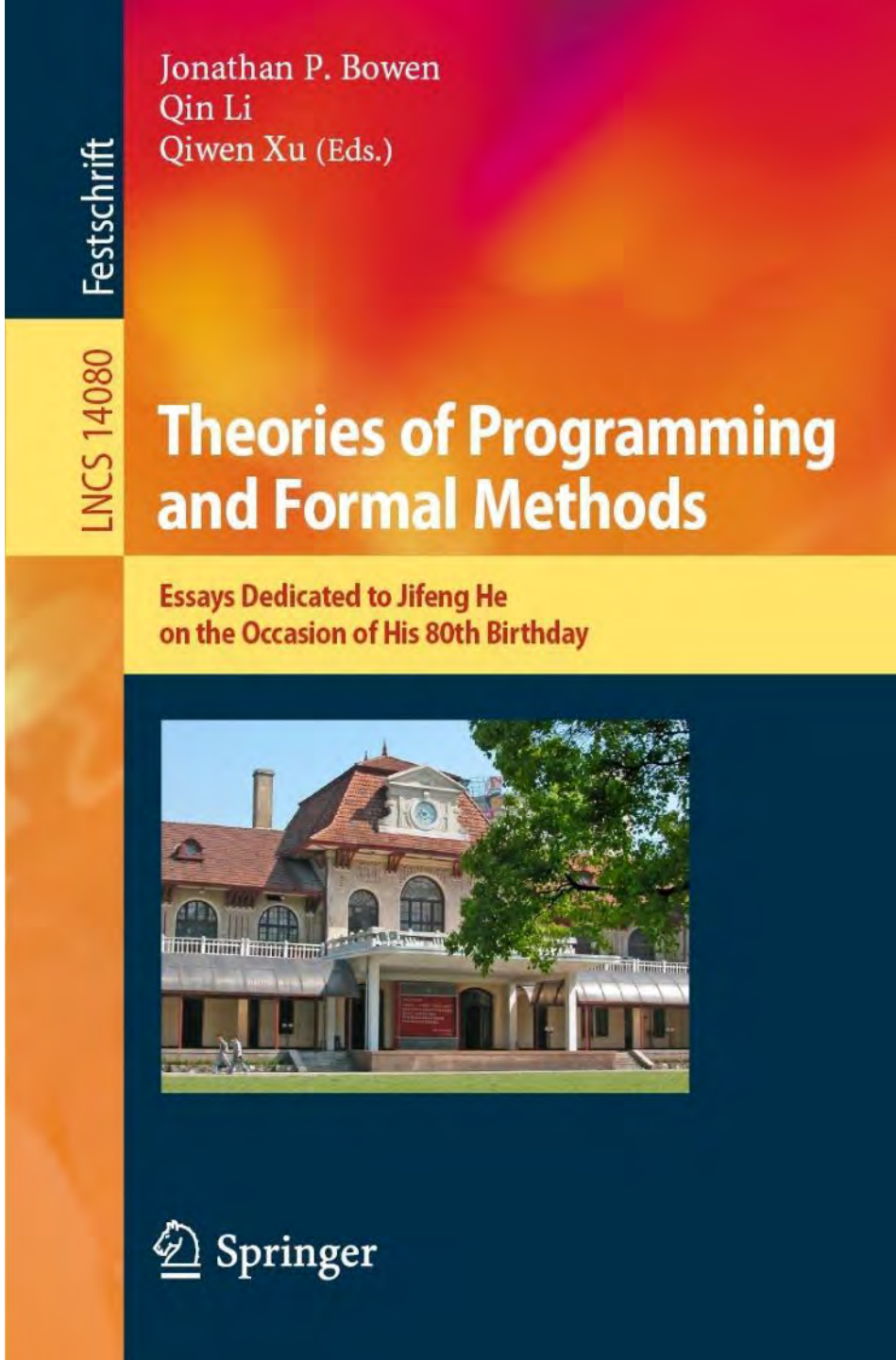
Professor He Jifeng

on the seventeenth day of April 2010

Jifeng He

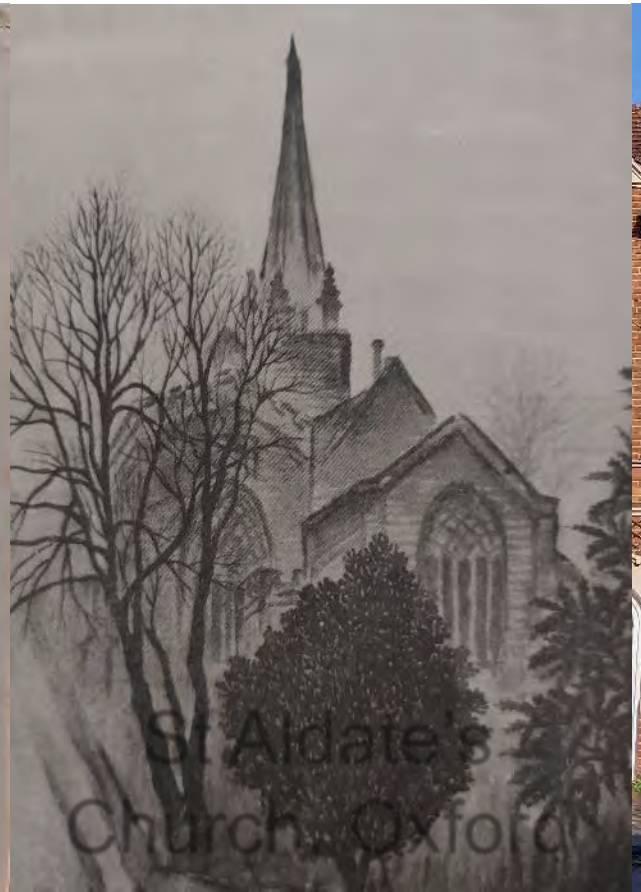
Jifeng He

80th birthday Festscrift, 2023



Commemorative Blue Plaques (Oxford)

蒋彝



3 Keble Road, Oxford

Oxford University Computing Laboratory (11 Keble Road)

Oxford University Computing Laboratory (11 Keble Road)

Jifeng He (何积丰)

Happy 80th birthday!

Jifeng He at Oxford and Beyond:
An Appreciation

EST 1892 LSBU

Museophile

EAST CHINA NORMAL UNIVERSITY

華東師範大學