

为软件工程的未来建立基础

——第十二届国际软件工程会议参加记

徐家福
(南京大学)

杨冬青
(北京大学)

我俩受国家教委派遣参加了1990年3月26日至30日在法国尼斯举行的第十二届国际软件工程会议。今汇报如下：

一、基本情况

1. 会议主题是“为未来建立基础” 整个会议内容（如研究成果、经验报告、回顾、全会报告、分组报告及讨论等）集中反映了对未来有用的基础。

2. 参加此次会议的有来自世界二十多个国家的代表约一千余人，既有学术界人士，又有工业界人士；有老兵，也有新秀。此次我国仅二人参加，人数明显少于过去几届。

3. 会议录用文章甚严。从投寄的二百

公司。一些私人公司在支持与他们的产品相关的其它学科的基础研究方面十分开明，但在支持他们将享有的AI基础研究方面却相当保守。

KO: 这是否尤其指电器和计算机公司？

JMcC: 是的，也包括对IBM的批评。IBM和Bell实验室都支持出色的物理学研究和相当基础的物理学研究，但就能否给AI的基础研究拨相当的经费的问题却犹豫不决。MCC作了一些基础性工作，但也可能是无意的。在我批评的对象中，还包括美国国防

多篇文章中仅选用二十四篇。一般说来，质量较高。

4. 会议安排了三个全会报告、十四个分组、四个专题讨论、四个专门小组，并配有展览。形式多样，内容丰富，有些问题讨论相当热烈。

二、全会报告

1. 会议共安排了三个全会报告。一是欧洲共同体报导了所属成员国在软件工程未来基础方面所进行的工作；二是美国麻省理工学院(M. I. T.) Barbara Liskov教授所做的题为“分布式程序结构”的报告；三是美国国防高级研究项目机构Stephen L. Squires博士所做的“高性能计算”的报告。第一、

部，甚至美国自然科学基金会有时也采取过份实用主义的方针。

KO: 这种状况将导致应用研究在某些阶段的停滞，对吗？

JMcC: 是的。那些支持物理学基础研究的机构仍然对那些物理学家感激不尽——发明原子弹的基础研究早在1930年就展开了。在AI的历史上，我们还没有这种情况，因此可以说支持AI研究的人是相当保守的。

〔王献图译自 Expert Systems, 1989, vol.6, No4 奚建清、王怀民校〕

第三两个报告侧重报导有关项目的情况，今着重汇报第二个报告。

2. **Barbara Liskov** 的报告题为“分布式程序结构”。基本内容有四点，即对象，事务元，分布式程序的结构，以及方法学。

(1) 分布式程序由一组相互通信的对象组成。每个对象提供了一组可由其它对象调用的运算，顾客调用，服务者进行处理。单个对象的结构是，具有一个状态，若干个“线脉”，用以处理调用和独立工作，状态是封装的，但可以包含别的对象。对象驻留在单个结点上。分布式程序中的对象和一般对象类似，但特点有五：一、大，二、生命周期长，三、可以主动，四、通信要求消息传递，而且多数参数是按值传递的，五、并发性。对并发与失败要留心，例如，A进行交付，B进行清算。问题是，第一、由于失败，交付未完成；第二、交付和清算并发进行。解决方法是利用事务元。

(2) 事务元相对并发与失败而言是原子性的（不可分的），其特性有二：一是可顺序性，即使各个事务元并行运行，其效果却和它们按某种顺序依次运行时相同；二是整体性，即一个事务元要么全部完成，要么无作用。原子性使我们容易理解各个对象所进行的工作，使我们在描述行为时无须担心失败与并发，顾客程序人员也无须担心，但却须处理中途停顿。服务者程序人员可能要求更为老练。有两级原子性，一级是原子事务元，可多次调用运算，但整个计算要么进行，要么停顿；另一级是原子运算。

总之，对象间通过调用运算进行通信；运算调用是原子性的，或者是单一的，或者作为包含事务元的一部分；每一调用必须保持对象状态的一致性。

(3) 分布式程序的结构。分布式程序由两类对象构成，一类是单一对象，另一类是复合对象。单一对象的特点是，第一，封装状态通过运算调用取接；第二，各个“线脉”运行调用和背景任务；第三，驻留在单个结

点上；第四，用模块实现。复合对象则由一组单一对象组成，其中所有的单一对象都属于同一个抽象类型，并且都用相同的模块来实现。从顾客和指明者的观点看，复合对象仍然是一个对象，它是具有一个状态的实体，该状态通过调用运算可被改变与注视。有两类复合对象，一类是各个单一对象均驻留在各个顾客结点上，顾客和单一对象的通信在结点上进行；第二类复合对象是，对象驻留在若干个服务者结点上，但不在顾客结点上。第一类的效率较高，第二类易于管理。

(4) 方法学。一是创建数据抽象，二是考察数据抽象的实现。

结论是，在组织分布式系统中有两个关键性概念，即对象与原子性，方法学不过是基于识别有用数据抽象的自顶向下分解的直接应用而已。

三、分组报告

会议安排了过程模型1、形式验证等十四个分组，报告要点如下：

1. 过程模型1

一篇是美国TRW Walker Royce的“用于增殖开发大型软件系统的Ada过程模型”，作者集中报导了在大型项目CCPDS-R中使用这一模型的情况。实际上，这是一个增殖开发与演示相结合的模型，以Ada作为设计语言是该模型的一大特点，经过各方人员的审慎考究，这一模型已受到广泛欢迎。

另一篇是日本SRA公司Atara T. Nakagawa等人的“软件过程与代数：OBJ为OBJ”。作者认为，把软件工程看成产品与变动的产物要比直接看成一组活动更好，这一途径可使纯技术方面和其它方面（如管理方面）分开，易于定义各种活动的作用，使我们能分析与控制更新的“涟漪作用”。该文提供了将这一途径用于基于OBJ的代数方法学的实例。文中所述的软件过程模型也可看成利用OBJ

来开发软件的环境。称作St. OBJ 的环境也是用OBJ本身来描述的。

2. 形式验证

一篇是美国马里兰大学Sergio Cardenas等人的“功能规格说明的评估准则”。作者认为，“功能正确性”是一种推导程序并证明该程序适应其规格说明的技术。把程序及其规格说明都看作函数，利用基于符号执行及指称语义的技术，发展了一种证明方法学，作者推广了这一功能规格说明理论，以便于按一种前后一致的方式刻画生命期中的各种方法的模型，对一个给定的规格说明给出若干可能的实现，便可开发出对一种实现（相对另一种实现而言）的评估技术。

另一篇是比利时 Louvain 天主教大学 Christine Lafonfaine等人的“利用VDM案例研究中的B定理证明程序进行形式软件开发的实验”。作者认为，形式软件开发需要工具支持，以便确保整个设计链的严格首尾一致，从而有助于软件工程师的工作。该文对这样一种工具（即B定理证明程序）进行了评估，在简介该工具之后，报导了所做的实验性评估，描述了将B用于控制VDM开发，利用一案例研究中的选段加以阐明，从而给出了VDM逻辑框架的形式描述，并开辟了软件开发的重用前景。

3. 实时与反应系统

一篇是美国Thomson-CSF公司 Armen Gabrielian等人的“实时软件的多级规格说明与验证”。作者提出了一种多重抽象级上实时软件规格说明的状态途径，其中每级规格说明都借助层次多重状态机(HMS)来执行；高层定义需求、低层表示各种实现需求的方法，从而大大简化了规格说明过程，并可导致复用规格说明。它和精化过程根本不同。作者还提供了一种验证多级规格说明一致性的方法。

另一篇是芬兰 Tampere 技术大学的H. M. Järvinen等人的“反应系统的对象式规格说明”。作者提出了一种并发系统操作规

格说明的新方法与一种对象式规格说明语言。和对象式程序设计语言相比，对象结构化层次式的状态转换系统，单一的对象方法取代为合作的多重对象动作的作用，从而避免了进程通信显式机构，引进了一种简单的非确定型执行模型。它不要求对各个动作的显式动用，该方法具有形式化基础，强调规格说明的结构化推导，在继承性的两种变形中反映了自顶向下与自底向上的方法学。

4. 环境

一篇是美国麻省大学和AT&T贝尔实验室Jack C. Wileden等人的“规格说明级的相互操作能力”。作者认为，开发结合多种语言书写成份并在各种机器上运行的系统的需求不断增加，其成功与否大部分取决于各个成份间的相互操作能力，亦即，在不同的背景下，各个成份间的通信与协同工作的能力。虽然很少现有的相互操作方法提供了表示一级上的支持，作者却研究了一种提供规格说明一级上支持的方法，发展了这种支持的模型，它由如下四种成份组成，(1)一种统一的类型模型，它是一种描述各相互操作程序所共享的实体的记法；(2)语言约束，它把语言的各种类型模型和统一的类型模型联系起来；(3)基础实现，用以理解各种不同的相互操作程序所使用的类型；(4)自动协助，用以简化将各种成份组合成一种可相互操作的整体的工作，文中讨论了表示级和规格说明级的方法，描述了规格说明级方法当前的原型实现等。

另一篇是法国CGE合作研究中心Jean-Louis Giavitto等人的“增殖型Adage构架的设计抉择”。Adage是一个增殖型的软件开发支撑环境，其系统结构的基本思想是，围绕一公共数据模型的广谱服务概念。该模型推广了实体关系模型，提供了对支撑类属性与增殖性有用的概念，如继承性与自反射等，服务可对最终用户与管理者的要求进行响应。

5. 度量学与可靠性

一篇是日本东芝公司Masayaki Hiraya-

ma等人的“软件生命周期的质量模型与度量实践”。作者提出了一种软件质量的量化方法。传统方法(如McCabe, Halstead等)着重源码的定量估算,文中提出的方法既着重设计又着重源码的估算。

另一篇是美国AT&T贝尔实验室的Wills Ehrlich等人的“软件可靠性模型在产品测量与测试过程中的应用”。作者将大型工业系统测试中所采集的数据的软件可靠性模型用来测量从顾客眼光看的软件质量。特别是,软件质量是通过系统软件失败率(表示成每小时系统运算的失败次数)来度量的。

6. 经验报告1

一篇是美国大西洋系统协会Tom DeMarco等人的“程序成档中录象的使用”。利用VHS录象技术作为程序成档的媒体具有一些迷人的好处,例如减轻苦活、方便成档者的诱导、降低价格,以及将产品迎合目标观众的心理等等。

另一篇是美国Lawrence Livermore国家实验室Lin Zuccoui等人的“利用对象式开发支持原型制作”。作者报导了利用半形式软件开发方法与原型制作以开发实时监控计算机系统的工作。原型用以定义并澄清系统需求,半形式方法用来捕捉需求,并加以成档。系统的结果描述用来把原型迁移到第一版成品系统。这种两者相结合的方法有助于克服单独使用原型制作方法中面临的一些问题。

7. 形式验证工具

一篇是巴西EMBRAPA信息学部M. C. Costa等人的“逻辑兴奋”。当形式需求规格说明提供为一种逻辑理论时,理论的符号兴奋可能有助于确认,兴奋者显示形式规格说明的逻辑模型。文章示出,如何从一个证明系统推导出极小的模型。

另一篇是西德Dortmund大学Harald Ganzinger等人的“模块化分序Horn子句规格说明的系统支持”。作者对CEC带条件等式的分序规格说明重写法则实验工作进行了简

介。CEC和诸如OBJ-S等系统的区别在于,为了借助条件项重写来正确执行规格说明,它可以检验语义的先决条件。CEC可按两种不同的方式来支撑规格说明的模块化结构。

8. 原型制作

一篇是瑞士研究与开发社团H. Oswald等人的“指明并执行层次Petri网的环境”。作者描述了一个构造并执行嵌入系统需求规格说明的环境。利用高阶Petri网表示规格说明,环境由一个网编辑程序、一个网模拟程序、一个网browser、以及一个模拟引擎组成。利用基于网编辑程序的作图可以构造层次式结构化的规格说明。

另一篇是法国ORSAY南巴黎大学C. Choppy等人的“拼接抽象与具体模块:规格说明,开发与原型制作”。作者提供了一种用以开发大型模块化软件的方法学,其中模块可在各个不同的开发阶段存在并交互作用,每个模块或者是完全抽象的,亦即,是一个代数规格说明,或者是完全具体的。在后一种情形,计算可按目标码直接运行,或者是居于抽象与具体之间的任一中间位置。

9. 经验报告2

一篇是法国GEC ALSTHOM Gérard Guiho等人的“SACEM软件确认”。作者报导了SACEM软件的确认过程。SACEM是一个部分嵌入系统(硬、软结合),它连续控制巴黎RER A线上所有火车的速度,这是一个罕见的商用系统,其中将一些现代化技术(如形式规格说明,断言,形式证明等)用于软件确认。

另一篇是美国Reston Mark Downson的“利用Graphite元工具的经验”。Graphite是一种元工具,它用来从图结构规格说明生成图结构处理Ada源程序,在该实验中,它用来生成简单层次数据管理工具的部分程序。实验表明,利用Graphite提高了生产率、程序质量、灵活性,并能分离若干重要的考虑因素。

10. 设计与系统结构

一篇是意大利Studi大学Francesco Parisi Presicce的“模块化系统设计的基于规则的途径”。文章表明,设计模块化软件系统(它根据一组预定义的抽象数据类型实现给定的规格说明)的问题利用产生式并把推导序列(如果存在的话)翻译成代数模块规格说明互连,便可归结为推导目标规格说明的问题。该方法以存在有可复用的模块库为基础,其中每个模块均表示成产生式形式的可见接口界面。

另一篇是美国WEST高级技术公司Marthy Ganti等人的“对象式软件应用结构”。该结构基于分离应用领域知识(包括语义、限制、功能等)与应用代码。应用领域知识嵌入在一个领域模型中,后者可由多个应用共享,通过利用对象式技术,使之成为可能。

11. 人工智能用于软件工程

一篇是美国南加州大学信息科学研究所W. Lewis Johnson等人的“演化转换库的构作”。作者开发了一些基于知识的工具,以支持规格说明的演化开发,借助一组演化转换来完成演化,这些转换是用于形式规格说明的、更改意义的转换。已经对规格说明语言Gist开发了一个演化转换库。作者评价了先前所做的工作,并介绍了当前构作的多用演化库的工作。

另一篇是美国AT&T贝尔实验室Premkumar Devanbu等人的“LaSSIE,一个基于知识的软件信息系统”。作者讨论了开发大型软件系统任务中所固有的“不可见性”问题。这一问题并无直接解案,但却有几种系统(如关系码分析程序、复用库管理程序、项目管理数据库等等)可以看成涉及不可见性问题的若干方面。作者认为,这些系统对于处理不可见性问题的一些重要方面(如语义衍生,多重视图,以及智能定标等)未见合适。作者构作了一个称作LaSSIE的系统,利用知识表示和推理技术直接讨论上述三方

面的问题,从而有助于研究不可见性。文中还对该系统进行了评估,讨论了一些悬而未决的问题以及正在进行的工作。

12. 过程模型2

一篇是美国Oregon大学William N. Robinson的“需求规格说明中的协商行为”。作者认为,协商是规格说明的组成部分,在获取规格说明过程中,用户间进行协商,用户与分析人员进行协商。在设计规格说明中,设计者们自行协商,设计者与项目负责人协商,在对一个系统进行规格说明的全部过程中,人们交流要求与限制,以便对项目或他们自身有益。研究这一行为对于合同协商及办公室管理等都是有价值的。然而,软件工程却未讨论协商行为。该文表示出,协商如何用于规格说明,提供了一些自动工具,以促进规格说明中的集成行为。

另一篇是加拿大McGill大学Nazim H. Madharzi等人的“Prism=方法学+面向进程的环境”。Prism是一个实验性的面向进程的环境,支持软件进程模型的方法开发、初启与执行。该文描述了Prism模型以及捕捉这一模型的各个成份的系统结构,该结构设计成保留产品软件进程描述,其生命期是由一个高级(或元)进程描述的显示表示所支持的。作者还描述了一个9步Prism方法,以构作并剪裁进程模型,还给出了支持这一描述的若干要领。Prism中进程模型是利用一个混合进程模型语言来构作的,该语言是以一个高阶Petri网形式体系和法则为基础的。

13. 技术转让

一篇是美国奥斯汀MCC James D. Babcock等人的“MCC的软件技术计划中技术转让的演化:从说教到论理”,MCC的软件技术计划(STP)具有双重使命:创建工具和方法,以帮助大型、复杂、分布式软件系统的开发组,以及确保在STP股东组织内部这些技术的广泛传播。STP已经发展了丰富的传统技术转让方法,虽然这些方法使我们能成功地把研究成果交流给股东,但他们却很

少能实际使用所交付的技术。最近我们重新估价了我们的技术转让方法，发现如何才能更好地付诸实施。

另一篇是美国 Contel 技术中心 Rubén Prieto Diag 的“软件复用分类的实现”。该文报导了复用库技术的开发实现的经验，着重谈到利用有刻划的分类组织复用软件库的经验。

14. 配置管理

一篇是美国 Vantage 分析系统公司 David B. Bernstein 等人的“基于申述性描述的程序设计任务的自动维护”。作者认为，由于中、大型软件项目往往会有很多相关的重复代码块，其本质是，理解和书写都比较简单，但实现与维护既烦琐又易出错，作者新近在设计与实现大型系统 CAE 的中心数据结构时遇到了这一问题，构作了一个“应用特定”的应用生成程序，数据结构是用一种申述性记法描述的，编写了一个专用翻译程序来处理这一描述。文中报导了使用这一风范的经验，并和其它途径进行了比较。

另一篇是西德 Pussau 大学 Thomas Rose 等人的“基于决择的配置进程模型”。为了澄清大型、小型、以及多型程序设计之间的关系，配置管理连接有一个软件进程数据模型，这一模型容许把概念版本、基于文件的版本、以及配置管理分开。除了描述该模型外，作者还报导了一个原型实现。

四、专题讨论

会议安排了四个专题讨论，即，软件再工程，现实生命安全临界软件，工业中使用形式方法的经验，以及技术转让中使用已定义的过程的经验。今着重汇报软件再工程这一专题。

组织人是 Andersen Consulting 的 G. M. E. Lafue。

软件再工程方面的工作包括，不管有关软件的状况如何，对现有软件的移植、修

改、建立文档等。典型说来，要进行再工程的软件实现的功能很重要。然而它们却是很久以前实现的，使用的是过时的技术，含有死码、冗余、不一致性等等。由于多年的修补，结构已不清楚，文档不可靠，难以维护。

再工程由两个阶段组成：1. 逆向工程阶段，分析现有软件，揭示其设计并抽象成高层次的表示；2. 正向工程阶段，从还原成的高层次表示产生出新的目标软件。

由于对软件再工程工具蓬勃发展的市场需求，由于缺乏满足实质需要的商品化的再工程工具，也由于再工程研究工作的复杂性，软件再工程引起了相当重视。参加本专题讨论的六位专家各自从他们认为关键的方面发表了意见。

组织人 Lafue 首先提出了软件再工程今后的研究和实践所要解决的问题，例如，商品化的工具对软件理解支持到什么程度，对程序设计知识的识别和表示能做多深，应用领域知识表示的重要性，形式方法的重要性，软件工程在再工程中的作用等。

Kozaczynski 强调了问题领域知识的重要性。他认为可以从四个层次上来理解系统：程序设计语言层、控制结构层、通用算法层、问题领域层。现在我们有能力构作理解前三层的工具，今后的进展取决于对问题领域的表示和推理的能力。未来的再工程不仅要有软件工程的知识，更重要的是要有特定问题领域的知识，亦即，要在独立于领域的工具基础上构作出面向银行、保险、调度等特定领域的再工程系统。

Biggerstaff 强调了再工程中人的因素的重要性。他认为在可预见的将来，完全自动的再工程系统是不可行的。人的理解，人的设计活动，人的决策是再工程过程中的必要元素。他介绍了美国 MCC 采用非形式和半形式的信息模式进行概念抽象、抽取领域模型（包括问题、程序、应用领域）的方法。他指出半形式的、面向人的、领域专用的抽象

在正向和逆向工程中都起重要作用。

Chikofsky讨论了CASE与再工程的关系。他认为CASE的将来发展越来越多地依赖于用户从现有系统获取信息的能力。现在所谓逆向工程的程序分析工具主要是获取程序级的信息，而不是去识别系统中程序间的相互作用。将来的关键是系统层的，这也是CASE最能支持逆向工程的所在。CASE对再工程过程的主要支持是分析。好的CASE环境能帮助用户分析存于CASE字典中的系统描述的关键方面，如数据冗余、相似的处理模式、数据规范化、不完全的说明、识别可重用成份等等，这些分析可用于单个子程序、程序、子系统以及整个系统。CASE、逆向工程、再工程一起为软件组织形成一个策略的紧密集合，去把握和再生他们现有的软件财富。CASE、逆向工程、再工程互相支持和依靠。

Bush和Maicocchi也对为自动进行现有软件的逆向工程、工业界应完成的任务，以及对于理想的再工程环境的要求等问题进行了讨论。

五、专门组

会议还安排了四个专门组，即，度量学进展，从数据库到对象库，研讨会报告，以及系统工程。今着重汇报第二个专门组的情况。

主题是“从数据库到对象库”，主讲人是GIP Altair的F. Bancilhon。这是对计算机数据管理和数据处理的发展和现状的回顾和总结。

1. 数据管理现状

在数据库市场上，关系系统成为占统治地位的系统。SQL事实上已成为关系数据语言的标准。然而，也有许多公司甚至还没有使用数据库，其中不少公司正在艰难地向关系系统前进。另一方面，关系技术对于一些新的应用领域很不适用，例如CAD/CAM，

CASE，OA等，因而，需要有新的技术。

2. 数据管理的特性

Bancilhon列举了数据管理的各种特性，如持久性、磁盘管理、数据共享、数据可靠性、数据安全性、随机查询、第一范式和第一范式的关系、复杂对象、对象标识、封装、类型、继承、超载和延迟联编、易扩充性、计算完全性、版本、分布等。对于各种特性，特别是，复杂对象、对象标识、封装、易扩充性等，讨论有关要求、性能等问题，以及与之相应的实现技术。

3. 各类数据管理和数据处理系统

Bancilhon列举了各类数据管理和数据处理系统：文件系统、数据库系统（层次、网状、关系）、非第一范式数据库系统、复杂对象数据库系统、语义数据库系统、对象式数据库系统、可扩充的数据库系统、程序设计语言、数据库程序设计语言、持久的程序设计语言、工程数据库等等。他介绍了各类系统的主要特征，讨论了各类系统对于数据管理的各种特性的观点及实现程度，其中重点讨论了各种数据库系统：对象式系统，可扩充的系统，程序设计语言等关于复杂对象、封装、类型、易扩充性等的观点和实现。对于一些体现新技术的系统，还介绍了典型产品，或原型系统，或重点研究结构。

六、认识与体会

1. 重要性

1968年由于解决当时出现的软件危机，出现了软件工程，二十多年来，其发展颇为迅速，原因在于需求推动，由于开发与维护各类软件的需求日益增长，软件开发势必走工程化道路。否则，不仅研制周期长，质量欠佳等问题无法解决，而且不少软件开发任务也无法实现。当然，可喜的是，国内这方面的认识近年来已有显著提高。

2. 艰巨性

软件开发与维护走工程化道路决非易

符号与代数计算的新阶段

——“符号计算系统的设计与实现”

(首次)国际研讨会参加记

徐家福 (南京大学)

我受国家教委派遣参加了1990年4月10日至4月12日在意大利卡布里召开的“符号计算系统的设计与实现”国际研讨会,今汇报如下。

一、一般情况

1. 这是符号计算领域内的首次国际会议。其目的是,提供这一领域的最新面貌,促进学术界、工业界、以及用户团体之间的学术交流。主题围绕符号计算系统的设计与实现的创新的方法与技术,例如,自动推理、几何模型构造与计算、自动程序设计等等。

2. 会议报告并讨论了二十五篇论文和十篇短文。内容涉及以下几个方面。(1)符号与代数计算:系统设计;(2)符号与代数

事。因为工程化不是孤立的,它涉及到形式化,自动化等问题。国际上虽然发展迅速,但二十多年来却未见重大突破,目前仍然处于渐变阶段。形式化问题颇多,远未普及;自动化出现了一些辅助工具,高层次的自动化涉及到智能化与自然化等问题,国内外均仍处于实验、探索阶段。

3. 坚持远近结合,理论联系实际

发展我国软件工程,一方面,国外经验可资借鉴;另一方面,还必须根据我国国情,进行自主研究与开发,相信国内的力量,我们无理由妄自尊大,也不应该妄自菲

计算;实现方法与技术;(3)理论;(4)自动推理;(5)软件环境与语言;(6)软件环境与用户接口界面。

3. 此次会议国内只有我一人参加。会议主席意大利“La Sapienza”罗马大学信息与系统学系Alfonso Miola教授和其它国家的一些代表态度都比较友好,和他们交流了情况,讨论了一些技术问题,特别是,和自动程序设计有关的问题。这是一个小型的研讨会,来自十多个国家约一百名代表参加了这次会议,规模不大,但文章质量高,讨论比较深入。今后有选择地参加一些这样的会议,“性能价格比”较高。

二、符号与代数计算:系统设计

1. 英国Bath大学J.H. Davenport的

薄。从我们多次进行国际学术交往来看,一般说来,国外学术界与工业界人士是相互尊重、相互学习、取长补短的。正由于如此,人家发展才比较快。重要的是,学术界也好,工业界也好,既要有长远考虑,又要千里之行始于足下,要远近结合,注意理论联系实际。不重视实际工作当然不对,但对理论工作不予以足够的重视,将可能铸成大错,吃大亏,何况好的实际工作总是有理论基础的,好的理论工作也是有其实际背景的。

限于水平,欠妥之处,请批评指正。