FGCS 计划之回顾系列报告(之三)　　计算机科学 1994Vol. 21№. 4

并行推理机及其基本软件综述

张　静　刘海燕　编译　　王献昌　审校

摘要　本文简要介绍 FGCS 计划的总体框架、PIM 机及其基本软件。FGCS 原型系统的核心是并行推理系统，包括并行推理机 PIM 和它的操作系统 PIMOS。知识库管理系统(KBMS)建立在并行推理系统上，并与 PIMOS 共同构成 FGCS 原型系统的基本软件。在此基础上，开发出约束逻辑程序设计语言的 LP(Language Processor)，并行定理证明器、自然语言处理系统等高级知识程序设计软件，以支持有力的推理和知识处理。为评估 PIM 和探索知识处理的新领域，研制了几个实验性应用系统。总之，有关 PIM 的成果和基本软件远远超过了 FGCS 计划的初始研究目标。

1982 年 6 月 FGCS 计划实施开始，历时十年，最终达到了预期目的。该计划以"逻辑"为未来知识处理之理论基础，逻辑程序设计为第五代计算机系统之核心，符号计算的高度并行处理为实现知识信息处理系统之必备。因此，FGCS 计划旨在研究开发新计算机技术，把知识处理和使用逻辑程序设计的并行处理结合起来(图 1)。

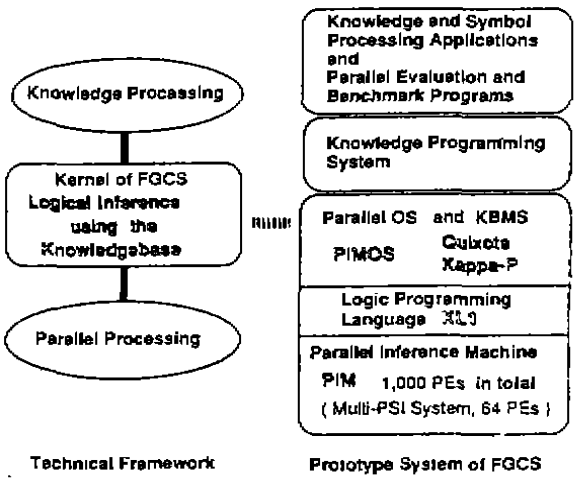

图 1　FGCS 计划的框架

1. 研究目标和计划

1.1　研究开发的范围

FGCS 计划的总体目标是：发展知识信息处理的计算机新技术。以"数理逻辑"为理论基础，研究课题建立在知识和符号处理软硬件技术之上，并分以下三类：

1. 并行推理系统(PIS)。并行推理系统是 FGCS 计划的核心，包括 PIM 机、KL1 语言处理程序(LP)、PIMOS。FGCS 原型系统是并行推理系统，它的建立是通过完善许多围绕逻辑程序设计而开发的实验性软件和硬件部件，其目标是：拥有约 1000 个 PE(element processor)、执行速度超过 100MLIPS(Logical Inference Per Second)、研制一并行操作系统 PIMOS。因此，FGCS 原型系统可视为一个有关符号和知识处理的巨型计算机(图 2)。

2. 知识库管理系统(KBMS)和知识程序设计软

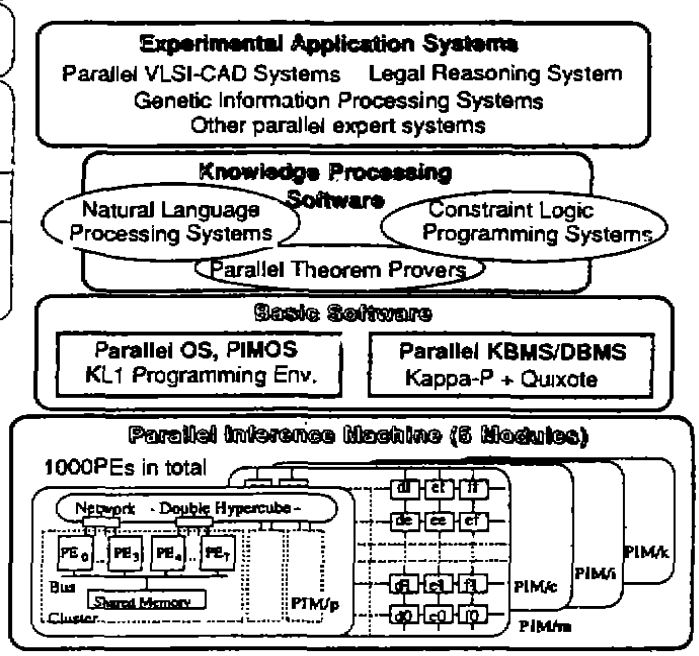

图 2　原型系统的组织

件。课题包括：·知识表示和知识库管理；·高级问题求解和推理软件；·自然语言处理软件。

这些课题的目标是在数理逻辑基础上为知识处理建立新理论、开发新技术。用于描述各种知识框架。知识框架产生于人类社会系统，是"自然"知识库的一部分。ICOT 试图把知识框架作为"人工"知识库的成分存贮于计算机系统，建立起各种智能系统。为描述知识框架需提供一种高级知识表示语言、它通过一个比并行推理系统更"聪明"的复杂推理机制来执行。

自然语言处理的研究目标是：方便使用，人机交互，覆盖知识表示方法和推理机制的所有研究。

ICOT 已在顺序推理机上实现了许多实验性软件。为这些课题的研究助一臂之力。

3. 基准和评估系统。课题包括：·并行推理系统的基准软件；·实验性并行应用软件。

在计算机科学中、为开展单元技术的研究、为建立知识处理的一般方法和技术，应建立一个典型问题的实验性软件系统。用以评估研究过程中出现的理论和方法。典型问题来源于工程系统（如机器设计、机器故障诊断），也来源于社会系统（如医学、工厂管理、政府服务）。典型问题的知识框架应处理为规则和事实的集合。

目前。探索知识处理的计算机技术远远落后于科学计算。近来涌现的专家系统和机器翻译系统属高级知识处理系统。但是，系统知识库规模太小（规则、事实数量平均只有几百个）。无力评估有 1000 个 PE 的并行推理系统的最大能力。因此，开拓新的应用领域、开发大规模应用系统势在必行。

1.2 总体研究开发计划

ICOT 经三年的研究讨论。确定了主要的研究领域和目标、制定了研究开发计划（1981 年末）。

当时、对逻辑程序设计的研究处于初级阶段。虽然欧洲已开始使用逻辑程序设计语言、主要用于自然语言处理、但计算机科学家还没有发掘逻辑语言的可行性和潜力。出现了一些令人关注的问题：一是语言级别太高、不能描述操作系统；二是逻辑程序执行开销太大、不能实用。

与高级语言相关的并行体系结构的研究也不成熟。虽然一些数据流体系结构被认为对于知识和符号处理很有潜力、但是无法有效地评估其并行应用的可行性。

此外、对于建立并行推理系统的核心尚没有健全的非常必要的单元技术。

鉴于上述技术背景、ICOT 详细制定了研究计划、把十年周期分为三个阶段并订出每阶段的研究任务：

·初始阶段（3 年）：潜在单元技术的研究；研究工具的开发。

·中间阶段（4 年）：为达到最终目标的主单元技术的第一次选择；中规模系统的实验性建立。

·最后阶段：为达到最终目标的主单元技术的第二次选择；最终全规模系统的实验性建立。

2. 初始阶段的推理系统

2.1 个人顺序推理机（PSI-I）

在初始阶段、为评估逻辑语言的描述能力和执行速度、推出个人顺序机 PSI，它是一个逻辑程序设计工作站，试图达到 DEC 10 Prolog 在 DEC 20 系统上的执行速度。而 DEC 10 是当时世界上最快的逻辑程序设计系统。PSI 为软件开发提供了一个公共的研究工具。

首先、ICOT 为 PSI 设计了基于 Prolog 的机器语言 KL0、并为 KL0 研制了采用特征体系结构（tag architecture）的硬件系统。接着，又设计出系统描述语言 ESP（一种逻辑语言）、具有类和继承机制，程序可以有效地模块化。ESP 不仅用来写 PSI 的操作系统 SIMPOS，还为知识处理写出许多实验性软件系统。

PSI 和 SIMPOS 的研究开发是成功的、特别是逻辑语言的软件生产率颇高。PSI 的执行速度约 35KLIPS，超过其预期目标。ICOT 已研制推广了近 100 台 PSI 机（PSI-Ⅰ）作为公共研究工具。而且意识到可通过有效地使用编译器的优化能力来完善体系结构。

同时，ICOT 开展了关于并行逻辑语言的研究。如 Prolog 和并发 Prolog，从中获益匪浅。在初始阶段即将结束时，还诞生了一种更为简单的新的并行逻辑语言 GHC，是一种适合于并行推理机的机器语言。

2.2 PSI 对研究计划的影响

开发 PSI-Ⅰ和 SIMPOS 的丰富经验对中间阶段的研究计划有如下直接影响：

①程序生产效率。SIMPOS 回答了与逻辑语言相关的两个重要问题：一是写操作系统（详细描述控制机制）的可行性；二是写一个大规模程序的可应用性。SIMPOS 有一个多窗口用户界面。有 100,000 多个 ESP 程序行，由 20 位软件研究员和工程师用近两年时间完成。当时、大多数软件工程师对逻辑语言并不熟悉。

②执行性能。PSI-Ⅰ的硬件和固件的速度(约35KLIPS)充分满足了大多数知识处理应用。PSI主存容量80MB,可谓当时主流计算机中之相当大的内存。PSI-Ⅰ有11块印刷电路板,若采用VLSI芯片,可为并行机形成PE;而KL0 LP,是以固件形式实现的。

ICOT估计,编译器对目标代码的优化在很大程度上可以提高执行速度。后来,是通过引入"WAM"码实现优化的。从PSI-Ⅰ和SIMPOS已证明,逻辑语言对于复杂的知识处理应用是一种切实可行的高效生产工具。

3. 中间阶段的推理系统

3.1 并行推理系统

中间阶段的主要目标是:KL1 LP的并行实现和并行操作系统PIMOS的开发。

全版本GHC的机器实现很复杂,于是出现了简化版本的FGHC,最终设计出基于FGHC的并行逻辑语言KL1。KL1属AND-并行逻辑程序设计语言,其LP包括自动内存管理机制和数据流进程同步机制,是编写和编译大型并行程序必不可少的,但存在两个问题:一是能否有效地实现这些机制;二是什么样的硬件和固件支持是有效且可行的。

PIMOS不同于传统操作系统,其设计也带来一些重要问题。它不需主进程调度和内存管理(由LP完成),但仍保留有关主存和PE的资源管理,控制用户程序的执行。PIMOS还要增设一个更为困难的功能,即必须允许用户把工作划分为并行执行的进程且分配给许多PE,同时负责平衡处理机负载以达到最佳执行性能,但在知识和符号处理应用中,程序的动态结构不规则,很难把握动态程序结构。

为解决设计LP和PIMOS时出现的新问题,可采用合适的并行硬件做平台,通过试错法展开实际的软件实验。

PSI-Ⅱ是一个更小、性能更高的PSI机,对PSI-Ⅰ做了多方面改进,如内存容量、机箱型号、磁盘容量及网络连结,旨在提供一个更好的作为公用工具的工作站,并为并行硬件形成PE。该并行硬件,即Multi-PSI系统,是PIM的小规模实验性机型,可视为开发并行软件的平台。

PSI-Ⅱ的CPU选用VLSI门阵列芯片,机箱大小约为PSI-Ⅰ的六分之一;执行速度是330KLIPS,约10倍于PSI-Ⅰ;主存容量扩展至320KB,以保证迅速进行大型应用原型系统的开发。

Multi-PSI系统于1988年春季完成,有64个PE,以8×8的网状网络互连,每个PE有三个印刷电路板,配以80MB主存,一个机箱有8个PE,因此一个Multi-PSI系统共有5GB存贮空间。该系统硬件非常稳定,目前ICOT已生产出6个Multi-PSI系统并分布到主要的研究现场。

KL1分布式LP由各种复杂功能模块(如松耦合内存的分布式拉圾收集器)有机结合而成。基于数据流模型的自动进程同步机制在分布式PE上实现起来非常困难,有些机制的实现还必须涉及一些PIMOS功能,如目标程序代码的请调式加载程序。与LP相关的一些支持功能,如系统调试,系统诊断和系统维护,也很重要。

KL1 LP以固件实现,对于执行一个KL1程序,PE或PSI-Ⅱ的速度约为150KLIPS,而对于执行一个顺序ESP程序,PSI-Ⅱ的速度达到300KLIPS。可见,自动进程同步引起的KL1开销使执行速度减半,但可用有效的并行处理来补救。

KL1 LP首次用PSI-Ⅱ的固件实现,用作KL1的伪并行模拟程序和PIMOS的开发工具,最后被扩展移植到Multi-PSI系统。

PIMOS第一个局部模块是在UNIX环境下用C语言完成的,是KL1 LP和PIMOS的一个子集,称之为PIMOS开发支持系统(PDSS)。目前已被推广应用于教育领域。PIMOS的第一个版本(带KL1的固件LP)产生于PSI-Ⅱ,称之为伪Multi-PSI系统,应用于KL1的个人程序设计环境。

可重入局部模块的建立和评估是KL1 LP和PIMOS的核心,采用KL1、ESP、C语言实现。所应考虑的是LP功能和PIMOS功能之间的合理平衡。

3.2 并行推理系统的总体设计

FGCS计划伊始,对并行处理的研究集中在硬件方面,主要兴趣是图像处理或大规模科学计算的SIMD或MIMD机。用Fortran或C语言编制应用程序,通过内部程序或子程序库实现程序的并行执行控制。SIMD或MIMD机排斥了包括不规则计算和要求通用并行程序设计语言及环境的大多数应用程序,但某些数据流机则有潜力具备函数式语言和通用并行程序设计环境。

通用并行程序设计环境对于开发大规模符号和知识处理应用并行程序必不可少,而且应提供以下功能:

1)分布式存贮器的自动存贮管理机制(并行垃圾收集器)。

2)基于数据流模式的自动进程同步机制。

3)为达到最好的工作划分和负载平衡而采取的各种支持机制。

前两种功能包含在 LP 中，并行操作系统提供第三种功能。它们解决了如何编制并行程序并映射到并行机上运行的问题。

在通用并行程序设计环境中上述映射机制分以下三层实现：

1)一个由 PE 和互连网络组成的并行硬件系统(PIM 硬件)。

2)一个由运行时刻例行程序、内部函数、编译器组成的并行 LP(KL1 LP)。

3)一个包括程序设计环境的并行操作系统(PIMOS)。

硬件系统的作用是提供一个稳定的平台。ICOT 认为使用 VLSI 技术太冒险，最终决定不把 PE 做得太复杂，但是为了提高执行速度，尽力为 PE 增加支持 KL1 的合算硬件。特征体系结构可支持自动存贮管理机制和 KL1 程序的快速执行。自动同步机制、部分工作划分和负载平衡(作为 KL1 原语)都以固件实现。

实现 KL1 LP 和 PIMOS 并非易事，由于无前例可援，无法对这些软件系统的开销给出可靠的质量估计。PIMOS 和 KL1 对用户隐藏了大多数有关 PE 和 PIM 硬件的网络系统的结构细节，程序员得完成两步工作：首先，根据应用问题和算法的并行模型，进行算法设计，划分程序模块。其次，对一个由细粒度进程组成的 KL1 程序，程序员可决定细粒度进程的分组，并指出怎样把它们调度给 PE。

4. 最后阶段的研究与开发

最后阶段的目的是开发新技术和方法。详细计划是：①从 Multi-PSI 系统的软硬件技术跳跃至有几百个 PE 的 PIM 的软硬件技术。②用 KL1 和 PIM 对大型、复杂的知识处理应用问题的并行处理提出挑战。这些目标指望为逻辑程序设计建立更好的并行程序设计方法，而大型、复杂应用程序的开发可创造新方法以建立更加智能的系统，并作为并行推理系统的实际基准程序。最后阶段的计划是：

1. *有效的并行软件技术*。(a)并行模拟和程序设计技术；并行程序设计范型；并行算法。(b)并行进程到并行处理机的有效映射技术：动态负载平衡技术；性能调试支持。

2. *使用并行推理系统的能力来建立智能系统的新方法*。(a)一个较高级推理机和较高级程序设计语言的开发。(b)知识表示和知识库管理方法(知识程序设计方法)。

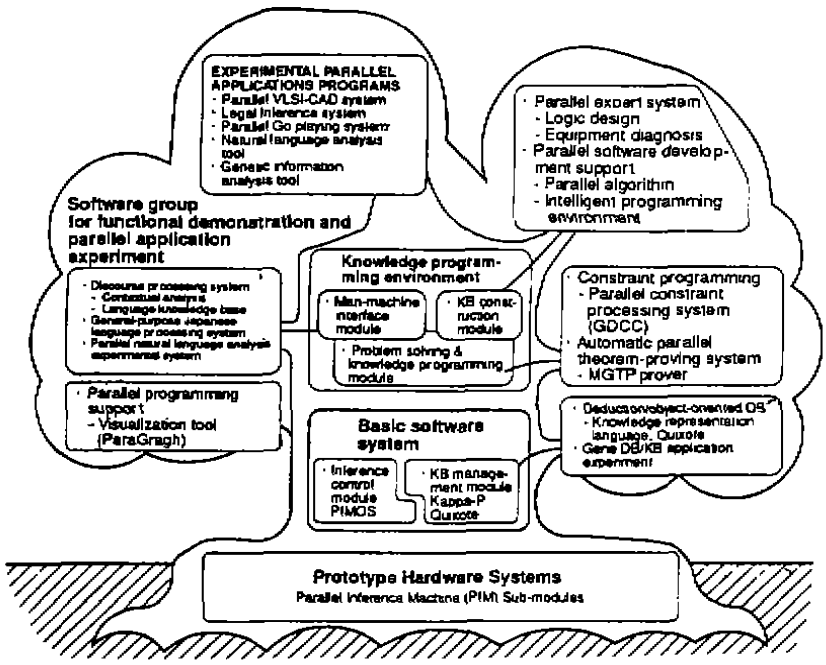

图 3　最后阶段的研究课题

最后阶段的研究开发课题和结果是：

1. PIM *硬件开发*。目标是建立几个不同体系结构的模型，所有 PIM 模块的 PE 数达 1000 个。PIM 硬件的作用是：①为软件研究员提供一个开发大规模知识处理软件的高级平台；②从 PE、网络系统的体系结构和硬件结构中获取评估数据，分析大规模并行程序的性能。PE 的体系结构有两种选择：CISC 指令集(固件实现)或 RISC 指令集；PE 的互连网络有多种选择：二维网状网络(象 Multi-PSI)，纵横制交换网、公共总线或相干高速缓存。

表 1 PIM 模块的特征

Item	PIM/p	PIM/c	PIM/m	PIM/i	PIM/k
Machine instructions	RISC-Type+ macro instructions	Horizontal microinstructions	Horizontal microinstructions	RISC-Type	RISC-Type
Target cycle time	60 nsec	65 nsec	50 nsec	100 nsec	100 nsec
LSI devices	Standard cell	Gate array	Cell base	Standard cell	Custom
Process Technology (line width)	0.96μm	0.8um	0.8μm	1.2μm	1.2μm
Machine configuration	Multicluster connections (8 PEs linked to a shared memory) in a hypercube network	Multicluster connections (8PEs + CC linked to a shared memory) in a crossbar network	Two-dimensional mesh network connections	Shared memory connections through a parallel cache	Two-level parallel cache connections
Number of PEs connected	512PEs	256PEs	256PEs	16PEs	16PEs

表 1 列出了 PIM 五个模块的主要特征。每个模块根据其主要目的和功能来决定所需的 PE 数。大模块有 256 到 512 个 PE，用于软件实验，小模块有 16 或 20 个 PE，用于体系结构的实验和评估。

2. PIM 模块的 KL1 LP。由于 PIM/m 和 Multi-PSI 系统有相似的体系结构，因此可通过简单修改和扩展 Multi-PSI 系统的 LP 来得到 PIM/m 的 KL1 LP。由于 PIM/p、PIM/c、PIM/i、PIM/k 采用簇(cluster)结构，因而它们的 KL1 LP 需重新开发。为避免重复开发 KL1 LP，用 KL1-C 语言写 PIMOS 和应用程序并被编译生成 KL1-B 语言(类似于图 5 所示的"WAM")。在 KL1-B 和实际的机器指令之间定义附加层：虚拟硬件层，它有一个虚拟机器指令集"PSL"。用 PSL 描述 KL1-B 解释器的规格说明(即虚拟 PIM 处理程序为四个 PIM 模块公用)，并被半自动地转换为实际的解释程序和运行路径。

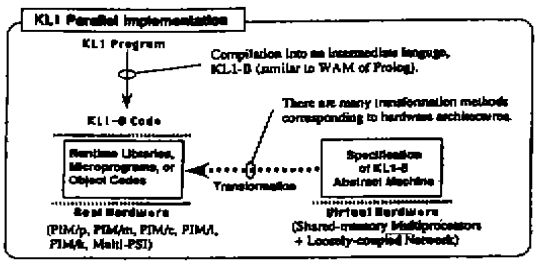

图 4 KL1 语言处理器和 VPIM

3. PIMOS 的提高和扩展。目标是开发一种面向对象的语言 AYA，它是一个并行文件系统，扩展性能调试工具作为 AYA 的程序设计环境。PIMOS 的提高目标是成为符号和知识处理的大规模并行机的标准并行操作系统，用 KL1 写成。它是独立的，自包含的操作系统并带有 KL1 程序设计环境。它的资源管理和用户程序执行控制等功能独立于 PIM 硬件体系结构，并基于非集中式管理模式，以 KL1 的原语："meta-call"实现。PIMOS 支持多用户，通过网络存取。

4. 并行 DBMS 和 KBMS。知识库管理系统(KBMS/PIMOS，用 KL1 写成)有两层。低层是一个并行 DBMS：Kappa-P，它是 Kappa-Ⅱ(PSI 机上用 ESP 实现的嵌套关系模型的 DBMS)的并行版本，拥有几个和 PIM PE 相连的磁盘驱动，以求得高吞吐量和高信息检索速度。它能灵活处理不规则大小和结构的数据，管理社会中积累起来的"自然数据库"，如自然语言词典，也适合于 DNA 数据库、规则数据库，如法律数据、合同条件。高层是一个基于演绎的、面向对象的 KBMS，它提供一个知识表示语言 Quixote，它的 LP 建立在 Kappa-P 上，目前正在开发之中。采用 Quixote，可从一个简单的数据库构造知识库，开始要积累被动事实数据，逐渐增加活动规则数据，最终形成一个完全的知识库。

5. 知识程序设计软件。知识程序设计软件由各种不同的实验性程序和工具(理论研究中建立并发展到知识处理的单元技术)组成，大多数用 KL1 实现，可视为并行推理系统的应用程序，知识程序设计软件主要包括：

(1)约束逻辑程序设计系统

最后阶段开发的 GDCC 是一个高级并行约束逻辑程序设计语言。GDCC LP(KL1 实现)包含一个约束求解程序，采用并行处理以提高速度。可通过一个简单的机器人管理程序评估 GDCC。

(2)定理证明和程序转换

定理证明程序 MGTP(KL1 实现)是法律推理系统的一个基于规则的推理器。它使系统采用一阶逻辑的知识表示，负载平衡优化成功，并行处理能力同 PE 数成正比。

(3)自然语言处理

开发软件工具和语言数据库是为了实现自然语言接口。语言工具箱(LTB)包括自然语言分析器，句子生成器，语言数据库和句法规则词典。

6. 基准的和实验性的并行应用系统。为评估并行推理系统、各种工具和方法，为发展并行程序设计方法、知识表示技术和较高级推理机制，ICOT 努力开拓了知识处理的新应用领域；如 VLSI CAD 系统，遗传信息处理和法律专家系统，博奕系统等。

5. 并行推理系统的评估

1. 对 KL1 的评估　事实证明：工作划分和负载平衡方法是成功的，KL1 语言的规格说明是切实可行的，由于采用自动存储管理机制和自动数据流同步机制，所以 KL1 的并行软件生产率比传统语言的软件生产率高得多。

2. 对 PIMOS 的评估　PIMOS 的功能(其中一些作为 KL1 的函数)对硬件上运行和调试用户程序非常有效。某些特殊工作上的资源管理和执行机制也正如人们的期望，如允许程序员使用 4000 个进程优先级，方便编制各种搜索算法和纯理论计算。

3. 对硬件支持的评估　比较 PIM PE 和一般微处理机的执行速度，得出 1LIPS 近似等价 100IPS，即 500KLIPS 的 PE 相当于 50MIPS 的微处理机。但是，KL1 程序执行特点不同于一般微处理机的基准程序执行特点。ICOT 认为，未来处理机设计应把特征体系结构做为一般用途微处理机标准功能的一部分。

4. 对高级程序设计开销的评估　尽管 KL1 生产率高，但与传统语言程序执行速度相比，KL1 开销不小。有两种直接补偿开销的办法：一是采用简单的子程序调用机制，把 C 语言程序连接到 KL1 程序上。二是改进编译器的优化技术。方法二比方法一更优美。

6. 结论

EGCS 计划是日本的第一个国家计划，其研究开发目标(特别是并行推理系统)已经达到。ICOT 不断公布、推广研究成果，传播 KL1 LP 和 PIMOS，希望它们能被移植到 MIMD 机上，为未来知识处理技术提供一个研究平台。FGCS 计划亦得到世界同仁的宝贵建议和帮助，因此，FGCS 取得的成就是世界范围内从事逻辑程序设计并行处理和相关领域的科研人员辛苦协作的结果。

主要参考文献：Shunichi Uchida，Summary of the Parallel Inference Machine and its Basic Software，Proceedings of the International Conference on Fifth Generation Computer Systems 1992，ICOT

(上接第 64 页)

使用 ZOOM 方法还有利于构造快速原型，通常有以下方法：(1)完成系统模型步工作后，最简单地实现驱动模块，使系统模型可以运行起来。我们可通过支撑工具观察其运行，抽查对象的内部数据，并测试系统模型是否正确地反映了现实世界。(2)对每一系统功能构造原型，使得系统功能在形式地说明后可立即得到演示和测试。(3)最快、最经济地实现系统需求说明，而不必考虑实现的效率问题。

主要参考文献

[1]S. Bailin，An Object-Oriented Requirements Specification Method，CACM.，1989，5，32(5)，pp. 608-623

[2]G. Booch，Object-Oriented Development，IEEE Tran. Softw. Eng.，1986，2，SE12(2)，pp. 211-221

[3]M. Jackson，System Development，Prentice Hall，1982

[4]J. Knudsen，Teaching Object-Oriented Programming is More Than Teaching Object-Oriented Programming Languages，ECOOP'88，LNCS 322，1988，pp. 21-40

[5]B. Meyer，Object-Oriented Software Construction，Prentice Hall，1988

[6]B.Sanden，An Entity-Life Modelling Approach to the Design of Concurrent Software，CACM.，1989，3，32(3)，pp. 330-343

[7]S. Shlaer，An Object-Oriented Approach to Domain Analysis，Softw. Eng. Notes.，1989，7，14(5)，pp. 66-77

[8]仲萃豪等，"应用软件的开发方法"，《计算机科学》，1991，1，pp. 5-15