2002 年 9 月
第 4 卷第 9 期

中国工程科学
Engineering Science

Sep. 2002
Vol.4 No.9

学术论文

高端计算软件平台的研发

陈左宁
（国家并行计算机工程技术研究中心，北京 100080）

[摘要] 高端计算软件平台是整机系统实现友善性、可用性和高效性的基本保障。随着计算处理规模愈来愈大，以及 Peer-to-peer 和 Grid 这类新型计算模式的出现，平台软件的设计面临新的挑战。首先，对高端计算软件平台发展状况以及面临的设计难题进行论述和分析；然后，介绍一个高端计算平台软件系统设计情况，并进行讨论。

[关键词] 高端计算；大规模并行处理；操作系统；并行开发环境；并行应用环境
[中图分类号] TP311 [文献标识码] A [文章编号] 1009－1742（2002）09－0045－05

1 概述

高端计算机的研制对国民经济发展和国防建设起着重要的推动作用，系统软件平台是整机系统实现友善性、可用性和高效性的基本保障。目前，国际上高端计算软件平台的核心技术掌握在少数几个大公司手中，并且一般是非自由软件。我国高端计算系统软件研发力量较弱，尤其是核心软件的开发，这在相当程度上影响了我国高端计算机的发展。

高端计算软件平台设计目前所遇到的最大挑战，一方面来自于超级计算机愈来愈大的规模所带来的系统的复杂性，另一方面来自于网络分布式环境下的新型计算模型。平台的好用性、可扩展性和可用性一直是研发的难点和热点。

2 发展状况及面临的挑战

2.1 发展现状

高端计算系统软件平台广义上包括各类硬件基础之上的支持高端计算的软件环境和支撑集中式的高端计算机的软件平台，或者网络分布式计算环境的软件平台。

图 1 显示了高端计算基础硬件平台（环境）的发展情况。集中式的高端计算机已经从单节点或少量节点结构发展到大规模多节点、多层次存储结构，称为 cluster of clusters 或 constellation，网络分布式环境也发展到 Grid 计算。在这个基础平台之上，软件平台自底向上可分为三个层次，操作系统环境、通用的分布/并行支持环境以及面向行业或应用领域的支撑环境。针对不同的硬件平台或硬件环境，包括的软件以及考虑的问题会有所不同。

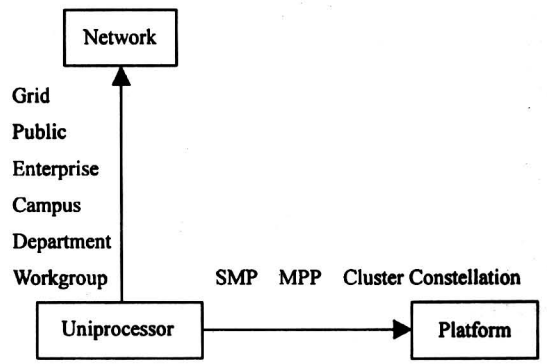

图 1 高端计算基础硬件平台（环境）的发展情况

Fig.1 The development of high end computing fundamental hardware platform (environment)

[收稿日期] 2002－06－21；修回日期 2002－07－05
[作者简介] 陈左宁（1957－），女，北京市人，中国工程院院士，国家并行计算机工程研究中心研究员

高端计算的低端是支持中小规模并行的超级服务器，其特点是注重与其他商品化硬软部件的兼容性、系统的易管理性和可用性等。目前这类系统的基础软件平台（不包括应用支撑环境）技术上比较成熟，产品化程度也比较高。操作系统环境遵循工业标准，以类 Unix 为主，近年来 Linux 和视窗软件也占有越来越多的比例，多采用单层（目前单一核心操作系统技术可支持到几十个处理器规模）或双层结构。通用的分布/并行支持环境往往包含数据库、丰富的中间件以及服务管理软件。由于并行编程语言以及工具多样化，编程模型趋向以 MPI 支持的消息编程模型和以 Open MP 支持的共享编程模型为主。这类平台软件基本上是在商品化操作系统之上研发机群管理系统，国内近年来以曙光、联想和浪潮为代表，推出具有品牌效应的机群管理系统。

高端计算的另一端是超大规模并行/协同计算或数据处理。目前对高端计算软件平台设计的最大挑战来自于数千乃至上万处理器的超大规模计算，以及 Peer-to-peer 和 Grid 这类新型计算模式。从发展趋势看，有限的不断增长的计算需求正向着按需提供计算能力发展，今后的超级计算机中心将转变为超级数据和超级应用中心[1]，Grid 技术将突破规模限制，通过资源的最大共享实现无限可扩展的高端计算（在一定权限的约束下）[2]。在这里传统的应用界面、编程模型、资源管理和使用方式受到了冲击，以往由于理论上和方法上难以取得重大突破，而不得不交给用户处理的问题，如并行程序自动生成、任务或数据的自动平衡分配、非线性或非规则问题的自动并行识别与优化等，面对超大规模以及更加复杂的硬件环境，平台设计者必需考虑其解决方法。

2.2 面临的问题

针对超大规模并行/协同计算，操作系统环境要解决的关键问题，包括大规模复杂资源模型的定义与表示，有效安全的资源管理、调度和使用模型及其实现技术，单一系统映像技术。大规模、复杂资源模块组成的计算机系统的突出特点是时间上的异步性、空间上的伸缩性、结构上的异构性、局部异常的多发性、资源实体的动态迁移性。传统的操作系统资源模型很难表示清楚这些特点。多年来，即使在中等规模环境下（如局域网的分布式环境，百十个节点的 Cluster 系统），整体的单一系统映像技术难以达到完全实用。关键原因之一是操作系统无论自身结构上还是管理算法上没有实质性突破，仅仅是单一集中操作系统的简单扩展。

随着系统规模的扩大，传统的开发环境很难支撑成千上万并发或协同任务的开发和运行。目前，实用的开发工具仅能支持百十个并发实体，并且往往是同构的。怎样描述大规模并发或协同任务的动态行为，怎样对其执行过程进行监测、分析和动态调试，适应大规模复杂环境的程序并行成分自动识别，任务或数据自动划分与分布，自动任务负载平衡以及其他并行优化编译技术，这些都是通用的分布/并行支持环境遇到的难题。要使开发出的软件能够充分发挥基础平台潜在性能，需要适合于体系结构特点的开发软件及工具。

多年来，串行程序的设计经过结构化、过程化、面向对象技术、组件技术和可视化编程技术等发展阶段，设计环境也已相当丰富，开发效率已显著提高。对比之下，并行程序设计比串行程序设计困难得多，开发方法落后。面向应用的支撑环境要解决的关键问题，包括并行或协同计算应用模型[3]、大规模并行软件设计方法及实现技术、可扩展并行算法或解法器设计、并行应用软件的构架与构件技术、大规模数据场可视化技术等。仅仅对串行程序的开发方法进行简单的扩充改造，实践证明无法适应超大规模复杂环境下的并行程序开发。

除此之外，安全性与高可用性问题也是这类软件平台在总体设计时必须考虑的问题。

3 超大规模并行计算机平台软件设计

高端计算机按照提供的功能特点划分，可分为两大类，Capacity 和 Capability。属于第一类的机器具有大量作业的高吞吐能力以及对大量服务请求的及时响应能力；属于第二类的机器目标定位在支撑挑战性大型应用课题的解算。超大规模并行计算机通常属于后一类，其应用范围往往是针对某一领域，如美国能源部的 ASCI 计划，生物工程 Bluegene 计划，日本的大气科学地球模拟器计划等，这类系统追求的目标是最高性能，因而其软件平台往往是量身定做，具有很高的性能。同时，人机和谐的桌边超级计算的需求，要求软件平台提供容易理解、使用方便安全、与主流平台软件兼容的界面，提供并行程序的互操作、代码重用以及新型开发方法等。可用性问题也对平台软件的设计提出

了很高的要求，典型的例子是美国 DOE 的 ASCI-Red 巨型机，它由 9 000 片 Pentium Pro 和 263 GB 的主存构成，仅这部分就包含 490×10^9 个晶体管，假如每个 CPU 的 MTBF 大于 10 年，整机的 MTFB 仅有 10 h，而这仅仅计算了主机部分的永久性故障[4]。正因为面临这样一些挑战性问题，这类软件平台的设计带有更多的研究探索成分。

4　一个超大规模并行计算机平台软件的实现

目前银河系列、曙光系列和神威系列高端计算平台软件代表了国内在该领域的最高水平，除行业应用平台外，研究工作与国外基本同步。多年来，结合神威系列并行计算机特别是超级计算机的研制，研究人员在平台软件设计方面的研究探索取得了一批成果。先后在国产并行机上推出了并行 UNIX 系统以及 ADA、C*、HPF 和 Open MP 编译系统。目前，正在研制具有自主版权的核心系统软件，希望逐渐形成国产化的可移植的全套高端计算平台软件。研究工作围绕着平台软件的可扩展性、好用性、可用性及平台软件的互操作性展开。下面简介神威系列并行机软件平台（SWSE）的研究情况。

4.1　可扩展性（Scalability）

计算平台的可扩展性长期以来都是一个富有挑战性的课题，微软的 Jim Gray 将其列入信息技术今后应该解决的 12 个方向性研究课题之一 。高端计算平台的可扩展性是体系结构、编程模型、语言和开发运行环境以及操作系统环境综合可扩展性的体现。对操作系统而言，可扩展性反映在硬件规模扩大时资源的管理和调度以及系统服务能力是否能够成比例增加。语言/开发运行环境的静态可扩展性反映在算法和数据结构并发成分的描述能力上；而其动态可扩展性反映在编译器或运行环境对数据分布、任务负载均衡的处理能力和对大规模并行任务的调试监控能力方面。

针对操作系统可扩展问题，SWSE 研究人员提出从操作系统结构设计入手，设计多级分层结构，按照亲缘性分配策略对操作系统的数据以及功能进行合理的布局，解决单一核心操作系统集中管理产生的热点，以及简单多核心操作系统消息开销对可扩展性带来的负面影响，加上多种有效的对超大规模资源调度管理的策略，希望解决超大规模资源管理的高效性与可扩展性难题（图 2）。

针对语言/开发运行环境的可扩展性问题，设计了以延迟分布（图 3）、反馈式编译（图 4）为代表的综合编译优化方法，尝试解决超大规模并行时优化问题。共享编程模型对串行程序改成并行程序非常方便，但是如何在大规模并行环境下实现共享编程模型是公认的难题。如何通过分析用户对数据的使用模式、减少远程访问、通信的向量化以及远程数据预取等是编译器研究的重点。其中，延迟分布是 SWSE 的研究人员创设的一个编译优化方法，它利用 First-touch 策略在程序执行时确定变量的分布类型，提高了数据分布的针对性，减少了数据重分布次数，有效解决了大规模分布式共享的效率问题。反馈式优化通过动态采集并行程序的运行轨迹及访存轨迹，反馈给运行系统或目标优化器，动态调整数据空间或程序空间的局域性，有效地解决了部分不规则问题在分布式系统上的效率问题。

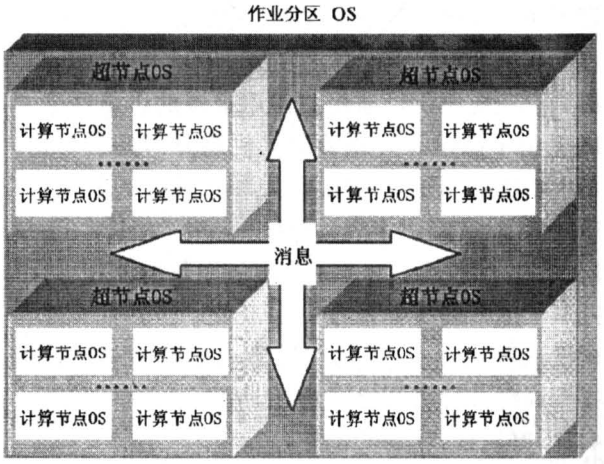

图 2　操作系统图

Fig.2　The graph of OS

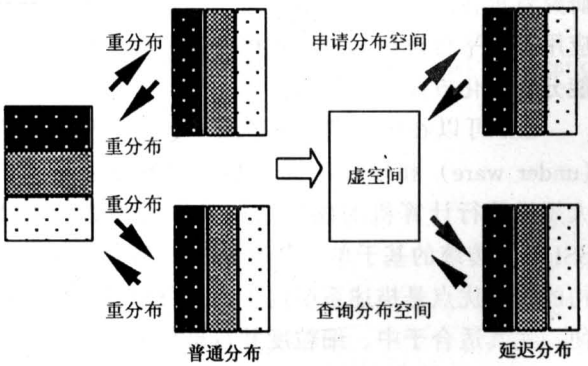

图 3　延迟分布图

Fig.3　The graph of delay distribution

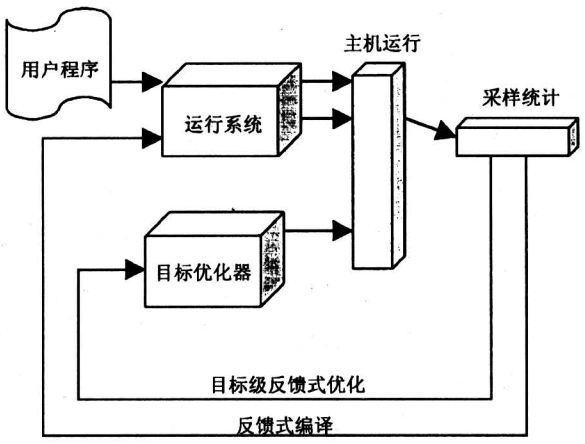

图4 反馈式优化图

Fig.4 The graph of feedback optimization

如何隐藏多层次的复杂的平台结构、提供统一的并行编程模型是目前国际上研究的热点之一。20世纪80年代以来，并行语言研究领域围绕着与机器无关的高级并行语言这一主题，取得了一些阶段性的研究成果，但总体上没能在超大规模并行机上得到广泛应用，目前用户使用最多的仍是消息库。HPF由于描述能力和编译优化能力不足正在走向消亡，Open MP由于在非共享的并行机上难以取得高效而限制了其通用性。在吸收HPF、Open MP经验的基础上，下一步将研究数据分布与数据共享相结合的新型编程模型，着重解决页面迁移、软件预取、Cache优化、反馈式编译等关键技术，推动高级并行语言的研究与应用。

4.2 好用性（Usability）

好用性或者易用性问题一直是困扰超级计算机用户的一个难题。美国能源部的ASCI计划将解题环境作为一个重要的研究目标（PSE）。SWSE的研究着重在操作环境的单一系统映像（SSI）、并行应用软件平台设计以及大规模并行程序的调试和性能分析优化方面。

SSI可以在中间件（middle ware）、基础软件（under ware）和硬件不同层次上实现。目前，超大规模并行计算机的硬件结构是不可能直接提供SSI的。传统的基于单一共享核心的对称式OS结构的最大优点是描述系统内各类并行事件自然、方便，尤其适合于中、细粒度并行事件的处理，易于实现SSI，但仅适用于中小规模系统。基于多核心的分布式OS由于协同管理、一致性处理等复杂问题，难以实现实用的单一映像系统。SWSE设计的桌面超级计算环境，将大量的硬软件资源包装在超级计算门户下，实现了单一系统映像。

SWSE包含了1个面向行业的大型软件系统，将数据并行高层建模技术、并行算法与并行识别技术、软件设计自动化及人机交互界面技术集成在一起。目前，已经初步构建了一个系统，它支持在算法层次上描述应用问题的求解过程，支持接近数学形式的形式化描述，通过并行性自动识别、分析和优化进而自动生成通用并行程序。

降低超级计算机的使用门槛，需要在开发环境上隐藏细节，对于那些难于理解的、琐碎的、机器相关的工作，尽量提供自动或半自动的工具帮助用户完成。SWSE的研究人员重点研究了基于Web环境、具有平台无关特性、透明的信息获取、动态并行行为模拟、自动映射以及自动性能分析和反馈优化支持技术。希望解决超大规模并行程序的调试、数据可视化、性能监测和实时监视等问题。

4.3 高可用性（High Availability）

随着系统规模的不断扩大，超大规模并行计算机系统发生故障的几率呈指数增长。单个模块的故障可以带来整个课题的运行失败，甚至以前所有运行结果付诸东流。SWSE的研究人员目前正在开展针对大规模复杂系统的系统级容错技术的研究，基本的出发点是正视这类系统局部故障多的现实，采用自顶向下的设计方法，研究大规模复杂系统的可用性模型和系统容错模型，从系统级进行可靠性预测和定量分析，科学分配可靠性指标，研究有效的保留恢复、故障抑制和系统级自愈技术，提高系统整体容错能力及可用性。目前，已经在一个大规模并行系统上做了部分研究工作（图5）。

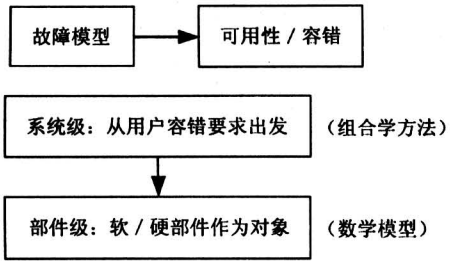

图5 可用性容错模型

Fig.5 The model of usability and fault-tolerance

4.4 平台软件的互操作性

目前高端计算的需求与商品化硬件的融合度愈来愈好（大量可供选用的商品化的器件与部件、构

建高端计算硬件平台的各种标准协议），而支撑高端计算的软件平台总体上缺乏成熟的商品化层次软件的支持以及相关的技术标准（中低端的基础操作系统环境除外），由此带来高端计算平台软件互操作性相对较差的问题。随着高端计算应用领域的扩大，设计满足按需使用、具有个性化、专业化特点的各类平台，首先要解决这个问题。

就操作系统而言，面对多样化的硬件环境，传统分布并行操作系统可重用性、可扩展性以及可移植性差的主要原因在于数据与功能的结构性差，缺乏互通标准。SWSE 的研究人员正在研究采用构架、构件设计方法和插件技术来解决这一问题，设计出根据体系结构可重构的操作系统框架，通过对操作系统层次性划分，抽象并行操作系统结构，独立出与硬件及并行结构有关的部分，设计相关的一套引用的标准，从而实现并行操作系统功能模块构件化，解决不同系统结构的可重用性，使得系统具有良好的通用性、互操作性和可移植性。

5　结语

在高端计算平台软件研发工作中，要着眼于国家重大需求，瞄准国际学科前沿，立足于我国核心软件的研究基础，以分布式并行操作系统结构创新为突破口，以高级并行编程模型与并行编译优化为核心，以高端并行运行控制环境为支撑，以高层算法级并行描述为特色，在高端计算平台软件的各个层次上协同创新，推动高端计算技术的向前发展。

参考文献

[1] Bell G, Gray J. High performance computing: clusters and centers, what next? [R/OL]. Microsoft, MSR-TR-2001-76, http: //research. Microsoft. com /users /Gbell /pubs. htm, 2001

[2] 李国杰，徐志伟. 关于下一代网络体系结构与应用模式的思考 [A]. 中国工程院工程科技论坛：下一代网 [M]. 北京：中国工程院，2002

[3] 陆林生. DPHL 面向科学计算的数据并行高层描述语言 [J]. 计算机研究与发展，2001，38(7)：153～159

[4] Buyyr R (ed). High performance cluster computing [M]. Architectures and Systems, Volume 1, USA, Prentice Hall, 1999

High End Computing Software Platform R&D

Chen Zuoning

(*National Research Center of Parallel Computer Engineering & Technology*, *Beijing* 100080, *China*)

[**Abstract**] High end computing software platform is the basic guarantee for implementing the friendship, availability and high efficiency of the entire machine. With the ever increase of computing scale and the emerging of new computing patterns, such as Peer-to-peer and Grid, the design of software platform faces new challenges. This paper describes and analyses the development of high end computing software platform and the designing difficulties, which it faces. Then the design case of a certain high end computing platform software system will be introduced and discussed here.

[**Key words**] high end computing; MPP; OS; parallel development environment; parallel application environment