

資 料

ペタフロップスマシン技術に関する調査研究 Ⅲ

平成11年3月

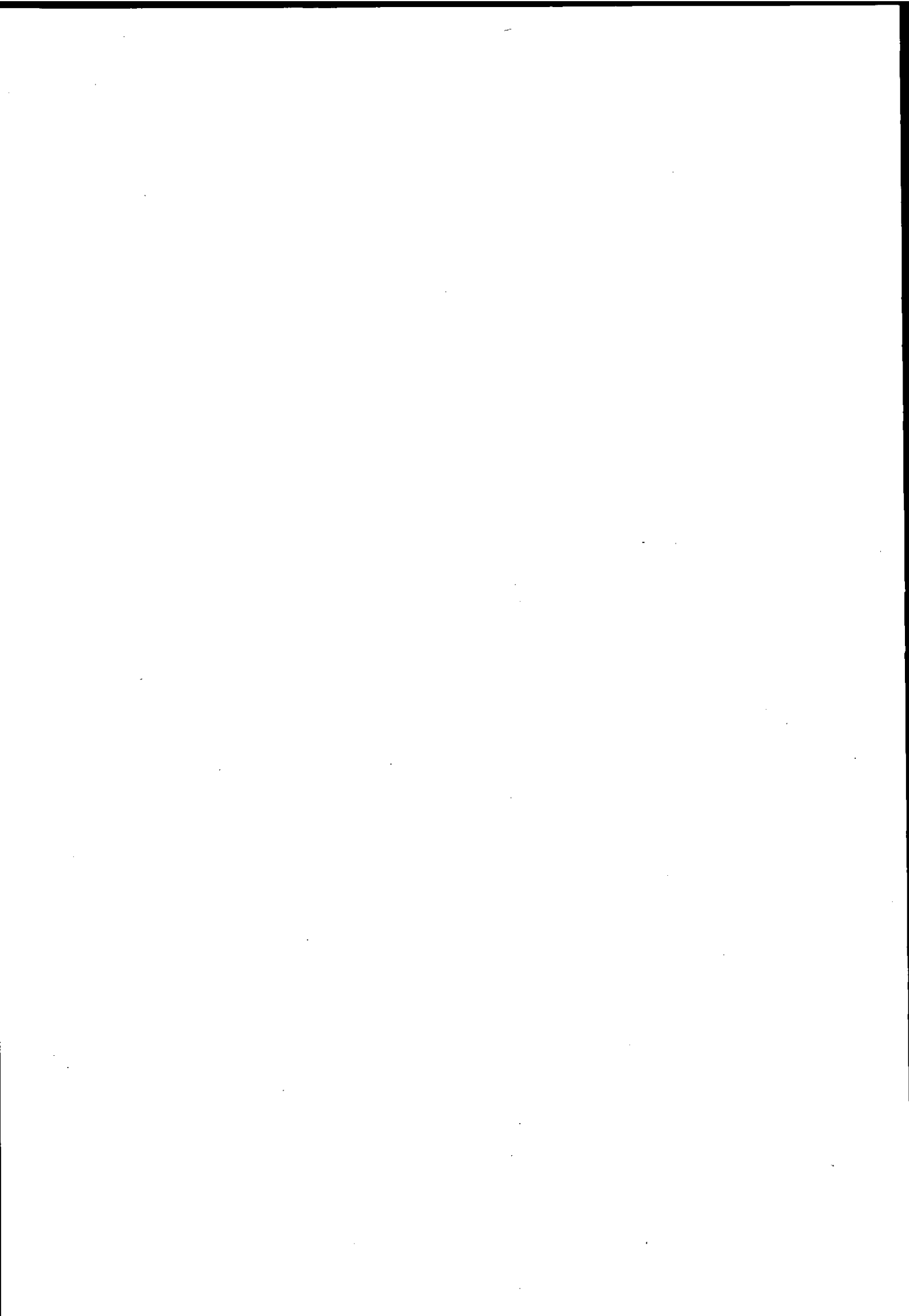
財団法人 日本情報処理開発協会
先端情報技術研究所



KEIRIN

00

この事業は、競輪の補助金を受けて実施したものです。



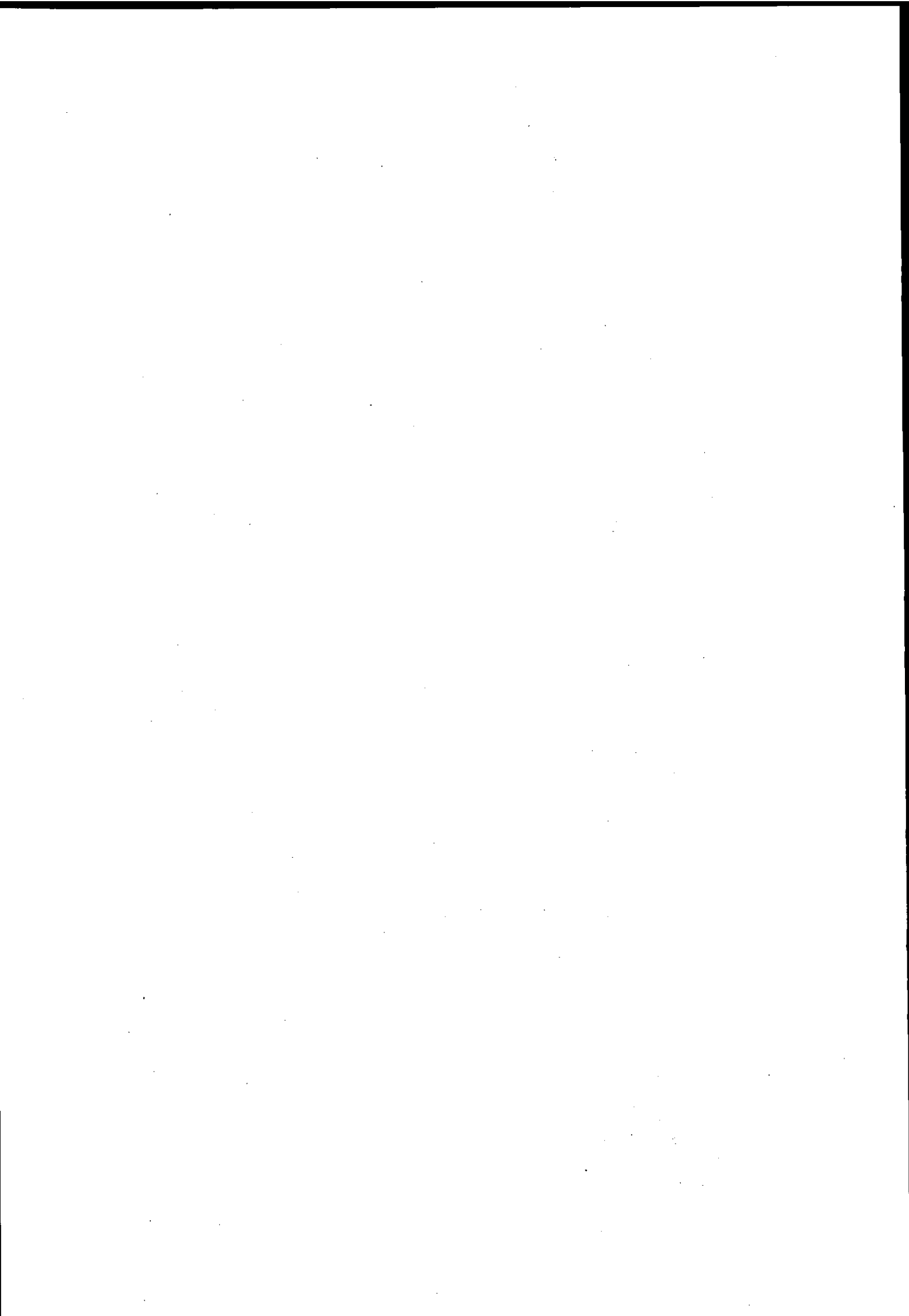
ペタフロップスマシン技術調査ワーキンググループ

- | | | |
|----|-------|--|
| 主査 | 山口 喜教 | 通産省工業技術院 電子技術総合研究所
情報アーキテクチャ部 主任研究官 |
| 委員 | 天野 英晴 | 慶應義塾大学理工学部 情報工学科 助教授 |
| 委員 | 石川 裕 | 技術研究組合 新情報処理開発機構
並列分散システムソフトウェア つくば研究室 室長 |
| 委員 | 笠原 博徳 | 早稲田大学理工学部電気電子情報工学科 教授 |
| 委員 | 久門 耕一 | (株) 富士通研究所 マルチメディア研究所
アーキテクチャ研究部 主任研究員 |
| 委員 | 妹尾 義樹 | NEC C&C メディア研究所 主任研究員 |
| 委員 | 関口 智嗣 | 通産省工業技術院 電子技術総合研究所
情報アーキテクチャ部 主任研究官 |
| 委員 | 中島 浩 | 豊橋技術科学大学 情報工学系 教授 |
| 委員 | 中島 克人 | 三菱電機株式会社 情報技術総合研究所
電子システム部 並列処理チームリーダー |
| 委員 | 濱中 直樹 | (株) 日立製作所 中央研究所
プロセッサシステム研究部 主任研究員 |
| 委員 | 福井 義成 | (株) 東芝 情報システム部 主査 |
| 委員 | 横川三津夫 | 日本原子力研究所 地球シミュレータ開発
特別チーム 副主任研究員 |
| 幹事 | 河西 和美 | (財) 日本情報処理開発協会 先端情報技術研究所
技術調査部 主任研究員 |
| 幹事 | 鳥居 良春 | (財) 日本情報処理開発協会 先端情報技術研究所
技術調査部 主任研究員 |

ペタフロップスマシン技術に関する調査研究Ⅲ

目 次

第1章 まえがき	1
第2章 米国の政府支援ハイエンドコンピューティング研究開発動向	
2.1 はじめに	3
2.2 米国ハイエンドコンピューティング研究開発政府組織	5
2.3 米国ハイエンドコンピューティング研究開発政策	10
2.4 米国ハイエンドコンピューティング研究開発予算	16
2.5 米国ハイエンドコンピューティング研究開発技術動向(1999年度 Blue Book)	19
2.6 ASCIプログラム	30
2.7 ウルトラスケールコンピューティング	44
2.8 おわりに	51
第3章 わが国における研究開発の課題と展望	
3.1 アーキテクチャ	55
3.1.1 VLSI アルゴリズム・アーキテクチャ・リバイバル(天野委員)	55
3.1.2 高性能計算とメタレベル計算(中島浩委員)	63
3.1.3 情報・エレクトロニクス産業振興策およびアーキテクチャ技術 の貢献についての私見(濱中委員)	70
3.2 ソフトウェア	77
3.2.1 Super Simulation Infrastructure(石川委員)	77
3.2.2 自動並列化コンパイラとアーキテクチャ支援(笠原委員)	82
3.2.3 並列プログラミング支援ツールの動向と並列システムの将来(妹尾委員)	94
3.3 アプリケーション	106
3.3.1 高性能計算技術におけるブレークスルー(関口委員)	106
3.3.2 世界における日本の強みと立場を活かした基礎研究開発(中島克人委員)	114
3.3.3 わが国における研究開発の課題と展望(福井委員)	121
3.3.4 大規模科学技術計算のための超高速計算機システム(横川委員)	127
第4章 あとがき	133
付属資料	
付属資料1 SC98 参加報告	141
付属資料2 Frontiers'99 参加報告	152
付属資料3 最先端並列計算機における次世代気候モデル開発に関わる 国際ワークショップ参加報告	156
付属資料4 1999年度 Blue Book にみる米国ハイエンドコンピューティング 研究開発技術動向	165
付属資料5 PACI (Partnerships for Advanced Computational Infrastructures)	191
付属資料6 ワーキンググループ資料リスト	197



第1章 まえがき

本報告書は、先端情報技術研究所(AITEC)内に設置された「ペタフロップスマシン技術調査ワーキンググループ」の委員の方々の意見をまとめたものである。将来のペタフロップスに向かって高性能化していく超高性能計算機の開発に関連した技術開発や技術政策に関して、昨年度の報告書と同様に、各委員の意見を記述していただいた。本ワーキンググループ(WG)は、具体的な調査を行うために、大学、メーカ等の若手研究者、11名から構成されている。委員の方々は、アーキテクチャ、ソフトウェア、アプリケーションの各分野において実際に研究や開発に携わっている方々であるが、多忙な時間を割いて議論に参加していただいた。

昨年度の調査においては、ペタフロップスを念頭におきながらも、近未来、5年ないし7年後くらいを視野に入れた議論を行い、現在あるものの次にくるべきものは何か、または、何がきてほしいかについて報告書をまとめた。今年度のWGの議論や報告書に関しても、昨年度と同様に、近未来の情報処理産業を見据えてどのような技術開発を行うべきかについてヒアリングや議論を行い、報告書としてまとめることとした。ただし、今年度は、WGをスタートするにあたって、予め情報処理技術に関する技術階層(あとがきに記載)を提示し、各階層において、技術的な重要課題はどのようなものであるかという点を念頭において議論をすることにした。それを行うきっかけは、デバイス分野の方々から、21世紀の高速デバイスや新しいデバイスがどのような形で使われることになるかという指針を示して欲しいという要請があったからである。そして、近未来の情報処理におけるアプリケーション分野を考慮にいて、情報処理技術の重要課題について議論をすることとした。そのために、6人の専門家の方々から以下のテーマで話題を提供してもらい、議論を行った。

- (1) 最先端デバイス開発動向とデカナノプロジェクト
- (2) デカナノレベル ULSI 技術開発の現状と問題点
- (3) 音声処理技術研究の現在及び将来
- (4) グローバルコンピューティング技術の現在と将来
- (5) 自動翻訳通信の現状及び将来動向
- (6) データベース処理研究開発の現状と将来

本報告書は、これらのヒアリングや討論をふまえて、我が国の情報処理技術を発展させるためにどのような方策が必要か、また情報処理技術の各階層において我が国にとって重要な技術課題はどのようなものであるかについて、各委員の見解をまとめたものである。そして、記述にあたっては特に次のような点を重視してもらうように要請した。

[1] デバイス分野で、新たなプロジェクトを提起する際の受け皿となりうる研究開発テー

マイメージとは何か。

[2] デバイスから応用ソフト、コンテンツまでの各階層での重要課題を束ねることのできるフラッグシップをたてることの是非、また、そのようなフラッグシップにはどのようなものがあるか。

[3] 新規産業の育成につながる技術アンブレラ（方向性を示す技術的イメージ）とはどのようなものか。

具体的な記述に関しては、昨年度と同様に、各委員の信じる技術論・方法論に関する積極的記述を書いていただくように要請した。そのために、本報告書では、各報告ごとにそれを報告した各委員の記名がなされている。

本報告書が、ペタフロップスに向けた情報処理技術の発展の一助になれば幸いである。

（山口喜教主査）

第2章 米国の政府支援ハイエンドコンピューティング研究開発動向

2.1 はじめに

米国では、情報技術分野で世界を凌駕することは国是であり、政府、大学、民間など、国を挙げて情報技術のレベルアップを行おうとしている。その中で、情報技術のインフラとなりうるハイエンドコンピューティング技術の研究開発は、政府主導の色彩合いが最も強いものの一つである。ペタフロップスコンピューティングは、そのフロントランナーであり、今や国家科学技術委員会(NSTC)の管轄する HPCC(High Performance Computing and Communication)計画の主要な柱を形成している。これらの動きは、大統領直属情報技術諮問委員会(PITAC)の答申、それを受けた2000年度大統領予算教書などで強力に加速されようとしている。

一方、米国に限らず、ハイエンドコンピューティング分野は市場規模が小さく、そこでの研究開発投資がすぐに利益に結びつかないことから、民間での自主研究は成立しにくいことは衆知の事実となっている。ハイエンドコンピューティング研究開発を牽引するアプリケーションとして、従来から存在する気象、航空・宇宙など科学技術計算がいまだに主要な部分を占めていることは間違いなく、それが市場の小ささの原因となっている。

ハイエンドの一手手前のコンピューティング領域では、ビジネスアプリケーション分野でもデータマイニングや可視化技術など、データインテンシブな領域で超高速コンピュータが用いられるようになってきており、すでに出荷台数割合は科学技術用途を凌いでいるとされる。このような領域では、従来のベクトル型コンピュータではなく、多数のスカラプロセッサで構成される SMP クラスタが用いられている。

米国政府の高速コンピューティング領域を育成する戦略は、すべてのアプリケーションに対応できるコンピュータシステムとして、スカラ型並列システムを指定し、その適応市場規模を拡大するとともに、ソフトウェアを初めとし、MPU など構成要素に至る総合的な技術の底上げをねらっている。そして、その最先端領域としてハイエンドコンピューティング研究開発が位置づけられていると考えられる。

上述したように、現状での主要なアプリケーションである気象関連などの科学技術計算では、そのアルゴリズムの特徴やこれまでのソフトウェアの蓄積により、依然としてベクトル型コンピュータの支持者が多いことも事実であり、この解決が(局所的な)大きな問題となっている。ベクトル型コンピュータの有利な領域として、科学技術

計算や可視化があることは間違いないようであり、ソニーのプレイステーション 2(6.2 Gflops)などのようにコンシューマに直結する時代でもある。

一方、米国ではハイエンドコンピューティング環境を、より多くのユーザに提供する仕組みの構築が真剣に考えられている。NSF の PACI をはじめとし、Grid、Globus、Legion などネットワークで計算資源を提供するシステム、ツールなどに関するコンセプト構築、技術開発が盛んに行われている。それらの環境で用いられる言語として JAVA が取り上げられ、それに対する関心は非常に高く、学会などのセッションでは常に参加者を引きつけている。

本章では、米国連邦政府の投資する情報技術研究開発のうちハイエンドコンピューティング分野の動向について、CIC 研究開発小委員会の Blue Book、大統領予算教書、OSTP 発表資料など、インターネット上で入手できる情報、及び海外調査による情報などを参考にして述べる。

内容は、米国政府での研究開発政策を立案、決定、実行する組織形態、また、21 世紀をにらんだ研究開発政策の方向・内容、さらに、その実行状態を示す研究開発予算について述べた後、個々の技術内容の動向について簡単に紹介する。

特に重要であると思われる ASCI プログラムについては、1 項を費やして解説し、さらに、ペタフロップスの 1,000 倍のコンピューティング能力をねらうウルTRASケールコンピューティングを紹介する。NPACI を含む NSF の PACI について、付属資料に掲載する。

2.2.2 テクノロジ委員会 (CT: Committee on Technology)

CT (図 2.1)は、1997 年 12 月に NSTC の改組が行われた際に、NSTC に属していた次の 3 委員会が統合し、再編された。

- ・ CCIC (Committee on Computing, Information, and Communications)
- ・ CTI (Committee on Technological Innovation)
- ・ CTRD (Committee on Transportation R&D)

ここで CCIC は、以前から IT 特に情報通信技術政策を担当していた委員会であり、1991 年に成立した HPC Act に基づく政策立案を行ってきた。

旧 CCIC の委員のうち CTに残った委員は数名で、かなり整理されたという印象である。

CT に属する組織は、以下のように示されている。

(http://www.whitehouse.gov/WH/EOP/OSTP/NSTC/html/committee/ct_subs.html)

- ・ Subcommittee on Transportation R&D
- ・ Subcommittee on Computing, Information, and Communications R&D
- ・ Subcommittee on Building and Construction
- ・ Subcommittee on Materials
- ・ Interagency Working Group for the U.S. Innovation Partnership
- ・ Partnership for a New Generation of Vehicles
- ・ Interagency Working Group on Critical Infrastructure Protection Research and Development (Co-managed with Committee on National Security)

情報通信技術の具体的研究開発政策を立案する CIC 研究開発小委員会は、旧 CIC 研究開発小委員会の委員の大半が引き続き担当しており、審議の報告は、CT に行うこととなった。

2.2.3 CIC 研究開発小委員会 (CIC R&D Subcommittee)

CIC 研究開発小委員会は、CCIC が CT に統合される以前から存在し、次の 5 つの PCA(Program Component Area)を受け持つ WG(Working Group)で構成されていた。

- (1) HECC (High End Computing and Computation)
- (2) LSN (Large Scale Networking, including the Next Generation Internet initiative)
- (3) HCS (High Confidence Sysystems)
- (4) HUUS (Human Centered Systems)
- (5) ETHR (Education, Training, and Human Resources)

現在は、CICがCTに統合された初期段階であるが、上記WGのうちHECCとLSNのミッションが明確になっており、予算が計上されている。他のWGはミッション範囲の決定に何年もかかる可能性が認識されており、現在、詳細な予算の計上は公になっていない。

CIC 研究開発小委員会には新たに FISAC(Federal Information Services and Application Council)を含めるようになった。

FISAC は、CCIC の Applications Council を継承しており、CIC 研究開発小委員会に報告する。FISAC は情報技術の R&D 団体から政府のアプリケーション・ミッションと情報サービス業界への技術の移行を促進し、アプリケーションに関する問題を識別して情報技術の R&D 団体へ知らせる。FISAC は次の組織で構成され、各チームから報告を受けるとされている。

- (1) Digital Government
- (2) FedStats
- (3) Information Technology for Crises Management Team
- (4) Universal Access Team

CIC 研究開発小委員会の組織図を図 2.2 に示す。

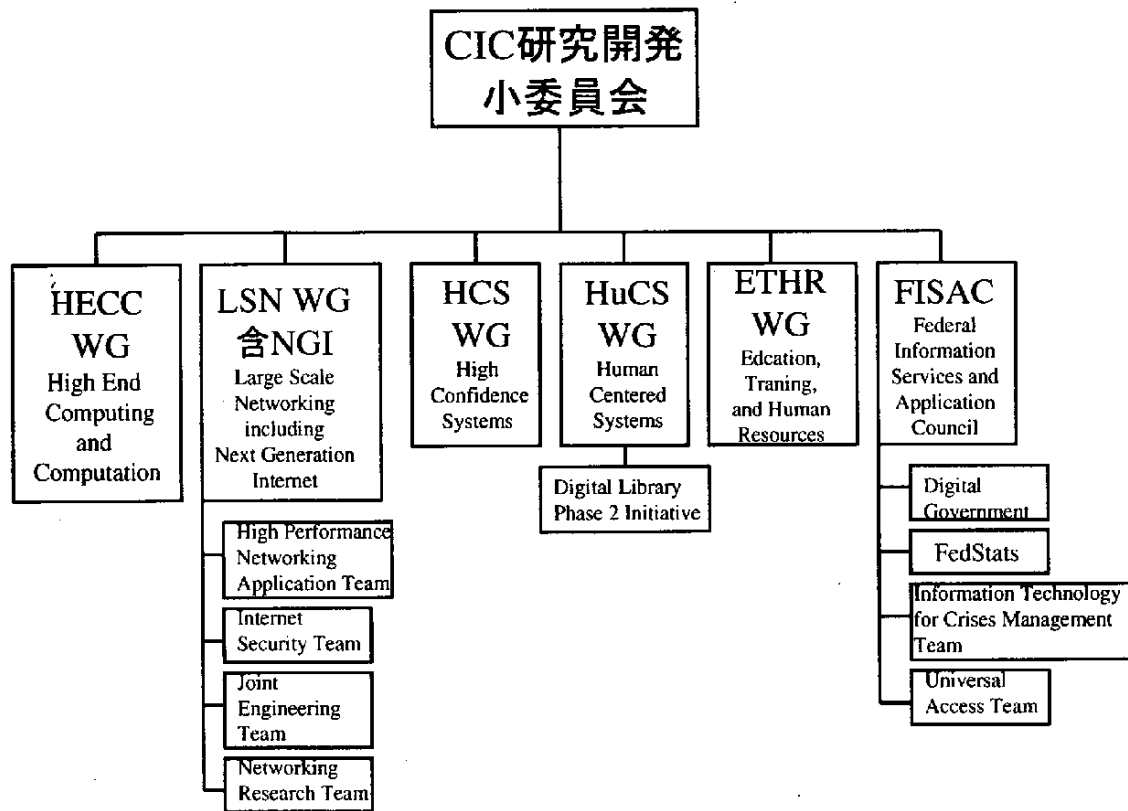


図 2.2 CIC 研究開発小委員会組織図

2.2.4 大統領直属情報技術諮問委員会 (PITAC)

図2.1の中心に図示されている大統領直属情報技術諮問委員会(PITAC: President's Information Technology Advisory Committee) は、1998 年度に設立されたものであり、設立当初は、PAC(Presidential Advisory Committee on High Performance Computing and Communications, Information Technology, and the Next Generation Internet)と呼ばれていたが、1998 年 7 月に改名された。メンバーは共同委員長の Bill Joy、Ken Kennedy を含め、産学から選ばれた 26 名から構成される (法的定員は非連邦政府メンバ 30 名)。

PITAC は、21 世紀に向けた情報技術分野における研究開発において、米国連邦政府がどのような支援をすべきかという政策立案のもととなる、プランおよびその優先順位にかかわる答申を得るべく結成された。元来、このような科学技術に関する諮問委員会としては、「大統領科学技術顧問委員会」(PCAST: Presidential Committee of

Advisors on Science and Technology)が存在し、答申をする役割を課せられているが、今回は、情報技術に特化した政策立案であり、それが民主党政権政策の主要な柱を形成することから、特別な諮問委員会が設立されたと考えられる。

1998年8月6日には、大統領諮問委員会中間報告(Interim Report to The President)が、1999年2月24日には最終報告が提出されている。政策に関しては次項に詳述する。

2.3 米国ハイエンドコンピューティング研究開発政策

2.3.1 これまで

米国政府が実行する政策は、大統領の名の下に各政府機関により立案、執行が行われるが、これまでの政策立案の経緯から、ハイエンドコンピューティング分野におけるゴア副大統領の役割・リーダーシップは特に重要であると思われる。以下に、OSTPの文書に示された内容をもとに、ゴア副大統領関連の政策を列挙し、米国の情報技術政策の変遷を簡単に記述する。

(<http://www.pub.whitehouse.gov/uri-res/I2R?urn:pdi://oma.eop.gov.us/1999/1/25/12.text.1>)

ゴア (Al Gore) は、20 年以上前、議会の一員として、最初に "Information Superhighway" という言葉を普及したと記述されている。

1984 年 高品質インタラクティブな教育ソフトを開発・流通させる法律を提出した。

1989 年 連邦政府の HPCC 研究プログラムを起こす法律を提出した。

1991 年 再提出により HPC Act が成立した。

1993 年 副大統領として、NII(National Information Infrastructure)構想を打ち出した。ここでは、次の項目が提唱された。遠隔通信市場の競争の増加、インターネット上への政府情報頒布、新規無線企業へのより多くの周波数割り当て、個人情報保護強化、インターネットと情報技術の非営利アプリケーションのパイロットプロジェクト。

1994 年 米国中の全ての学級、図書館とインターネットを結ぶことを目標に掲げた。

同年 GII(Global Information Infrastructure)を提唱、G-7、ITU、サミット、APEC で同意を得た。

1996 年 Telecommunications Act により、学校と図書館からのインターネットアクセスを可能とした。ゴアにより、教育テクノロジー(コンピュータ、ソフトウェア、教員訓練)への投資は増加し、1993 年\$23M が、1998 年\$700M となった。

同年 NGI (Next Generation Internet) の名の下に、今日のインターネットの 1,000 倍のスピードアップを提唱。

1998年 NGI 構想は議会承認を受け、法制化された。

同年 個人プライバシー保護のため "Electronic Bill of Rights" を提唱。法律にもりこまれる。

このように、米国の情報技術政策の大きな流れは、ゴア副大統領を中心として策定されてきたと考えられ、彼の去就は、将来の米国情報技術政策の行く末に大きな影響があると考えられる。

ハイエンドコンピューティング分野に関連する政策は、上述の1991年 HPC Act を主なよりどころとして研究開発投資がなされてきたが、政府は、次に示す項目をその主な成果として主張している。

- (a) 高性能コンピューティングの商用基準としてスケーラブルな並列処理を確立した。
- (b) 効果的なハイエンドコンピューティングデバイスとして超並列システムを創出した。
- (c) ハイエンドコンピューティングのための科学的基礎（教育訓練を受けた科学者と技術者、アーキテクチャに関する新しいアプローチ、次世代技術など）を創出した。
- (d) 高性能コンピューティング使用を容易にする20億ドル規模のミッドレンジ高性能コンピューティング市場の技術的基礎を実現可能にするとともに、政府の経費を削減した。
- (e) エネルギー省(DOE)のASCI (Accelerated Strategic Computing Initiative) プログラム のために短期的なコンピューティング技術を実現可能にした。

テラフロップスが達成された現在、米国は次の成果として、1ペタフロップス(Pflops)の持続的処理速度を掲げている。1999年度Blue Bookでは、適切な資金と技術的な注目が集まり、特定の基幹技術が現在の速度で進歩し続ければ、2007年までに目標が達成されると予想されている。

2.3.2 大統領直属情報技術諮問委員会 (PITAC) 報告

(<http://www.whitehouse.gov/WH/EOP/OSTP/NSTC/html/annualrpt98.html>)

1997年2月、クリントン大統領はPITACを設立した(前述したように設立当初はPACと呼ばれていた)。PITACは、科学技術を所管する大統領補佐官でOSTP長官であるNeal Laneを通して、NSTCに高度情報通信及びIT技術の全ての分野に関する方針と勧告を与える責務を負った。PITACの委員は、産業界と大学から広く選出された専門家からなる。

1998年6月、PITACは、大統領に書簡を送り、全ての米国民の生活の質と水準を常に向上させるため、情報通信およびIT技術研究への公的投資を著しく増加すべきであると主張した。そこで、大統領は、科学技術担当大統領補佐官に、情報通信及びIT技術分野で野心的な新研究プログラムを作成するよう指示した。PITACは、1998年8月に大統領に提出した中間報告書において、精緻な意見を述べた。中間報告書では、研究開発投資イニシアティブ(「副大統領1999」で表明されたInformation Technology for the Twenty First Century (IT²))に関して、科学技術補佐官に対する有益な指針が与えられたとされている。

1998年9月、PITACはそのメンバと他の非政府専門家でなる6つのパネルを召集し、中間報告の提案を精密化し、最終報告に加えて、1999年2月に大統領に提出した。

2.3.3 21世紀に向けての情報技術政策

(IT²: Information Technology for the Twenty-First Century)

(<http://www.whitehouse.gov/WH/EOP/OSTP/html/fy2000it.html>)

2000年度大統領予算教書において、クリントン大統領とゴア副大統領は情報技術研究の政府投資の大幅な強化を表明している。特に、従来のHPCC予算とは別枠で、前年度HPCC予算の28%増加に相当する、\$366Mを支出する新イニシアティブを創始する政策を打ち出した。この野心的な計画は「21世紀に向けての情報技術」(IT²: "IT Squared": Information Technology for the Twenty-First Century)と呼ばれ、連邦政府の情報技術研究に関する画期的な新公約であるとされる。PITACとNSTCのアンブレラのリーダーシップのもとで、1999年2月、研究開発担当省庁は "Information Technology for the Twenty-First Century: A Bold Investment in America's Future" のドラフトをまとめた。この文書の付録には各省庁の役割が説明されている。

IT²ドラフトでは、重点項目が以下のように指摘されている。

(1) 長期的情報技術研究

- ・ 情報通信基盤の優位を導く。
- ・ 1960年代に始まる政府の投資が現在のインターネットにつながった

(2) 科学、工学および国家のための先端コンピューティング

- ・ 医薬品の開発期間短縮、低公害・高効率エンジンの設計、竜巻の高精度予測のブレークスルーをもたらす

(3) 経済・社会に生じる情報革命の研究

- ・ 大学での増加する IT 技術者の訓練を助長する努力

IT²の潜在的恩恵について次の3項目が指摘され、強調されている。

- (1) 過去の政府機関の研究結果（インターネット、最初の Web ブラウザ、先端マイクロプロセッサ）は米国の IT 産業分野でのリーダーシップを強化した。また、現在の米国経済成長の 1/3 に達し、私企業平均より 60%以上高賃金の 7.4 百万人の雇用を創成した。米国の全産業は IT を用いることにより、世界市場で競争し、勝利を得ている。米国でのビジネス上の電子商取引だけで、2003 年までに 1.3 兆ドルに達すると予想されている。

- (2) 情報技術は、我々の生活、仕事、学習、コミュニケーションを変える。

IT の進歩により、子供の教育の方法が変わり、障害者がより独立した生活を営めるようになり、遠隔医療により、田舎の人々の医療の質を向上できる。IT 分野での米国優位は安全保障でも本質的に必要である。

- (3) 情報技術は、科学と工学の「黄金時代」に導く。

スーパーコンピュータとシミュレーションとネットワークの進歩は、自然界との新しい窓を創成する。科学上の発見のためのツールとして、理論と実験と同じくらい IT を価値あるものとする。テラフロップスコンピュータによって、科学・工学者は、気候変動のより正確な予測、高効率・低公害エネルギーシステムの設計を可能とし、材料物質の基礎物性に関する新しい知見を得ることができる。

このイニシアティブは、HPCC プログラムや NGI を含む情報通信分野での過去・現在のプログラム*の上に構築されるとみなされている。（*：the High-Performance

Computing and Communications program and the Next Generation Internet)

それは、議会の要求による非政府の諮問委員会である PITAC の答申に応答したものであり、政府は長期にわたり、国民にとって重要な IT 研究に投資すべきことが結論づけられている。産業界・学会の指導者から構成されるこの委員会は、情報革命を持続することが不可欠な IT 分野では、長期的基礎研究投資は私企業には向いていないと結論している。このイニシアティブの意見は、IT が全ての科学技術分野における発見のペースを加速する潜在力を持つという研究コミュニティ内で強く信じられていることを反映しているとされる。

IT²に含まれる省庁は、NSF、国防総省(DARPA を含む)、DOE、NASA、NIH、NOAA である。約 60%の投資は、大学ベースの研究支援となり、先進 IT 技術を有する技術者育成要求に応える助けとなることが主張されている。

IT²の結果としての潜在的ブレークスルーは、次のようなものが考えられる。

- (1) 人間の言葉を話したり、聞いたり、理解するコンピュータは、より使いやすく、リアルタイムで正確に多言語翻訳できる
- (2) 我々のためにインターネット上を動き回る「インテリジェントエージェント」は巨大なデータの海から探している情報を収集し、要約してくれる
- (3) 全世界の全ての研究者にアクセスできるスーパーコンピュータでシミュレーションする事により、広範な科学的・工学的発見が可能となる
- (4) 何千万のコンピュータだけでなく、何千億の機器を接続するようにできるネットワーク
- (5) 今日のスーパーコンピュータの何千倍も速いコンピュータ、バイオもしくは量子コンピューティングのような根本的に異なる技術に基づくコンピュータ
- (6) 高信頼、保守容易な複雑なソフトウェアを開発したり、電話回線、電力網、金融市場、インフラのコア要素をより高信頼にする新しい手法

具体的な研究分野として、次に示すように基礎研究、先端技術、経済・社会への影響が提案されている。

(1) 基礎研究

- (a) ソフトウェア：ソフトウェア工学、エンドユーザプログラミング、コンポーネント化開発、アクティブソフトウェア（アプレット）、自立ソフトウェア（インテリジェントエージェント、ロボット）
- (b) 人間とコンピュータの連関と情報管理：人間の言葉の理解と会話、情報の可視化
- (c) スケーラブルな情報インフラ：膨大な数のデバイスを含むネットワーク、制限のない接続、ネットワークのモデル化とシミュレーションツール
- (d) ハイエンドコンピューティング：ペタフロップスに向けたスーパーコンピュータの性能・効率向上、情報処理網の創成、生物・量子コンピューティング（量子素子技術）など次世代技術

(2) 科学、工学、国民のための先端コンピューティング

- (a) インフラ：非軍事研究者コミュニティに 5 Tflops(2000 年)、40 Tflops(2003 年)のコンピューティング性能の提供
- (b) 科学・工学分野での計算：気象変動予測、高精度天気予報、遺伝子機能解析、計算地震学、エンジン内燃焼、材料開発、宇宙モデル
- (c) コンピュータ科学と技術開発：システムアーキテクチャ/ソフトウェア、開発環境、研究環境、アルゴリズム、可視化/データ管理
- (d) NII(National Information Infrastructure)アプリケーション

(3) 情報技術の経済・社会への適用と労働者教育

- (a) 情報技術の経済的、社会的インパクト：インターネットの非従来型管理、情報技術のもつ遠隔/仮想などの形態の影響、情報技術へのアクセスの不平等性、電子商取引
- (b) 情報技術労働力：大学、大学院レベルの教育システムの改善により世界最高の情報技術労働力を整備

2.4 米国ハイエンドコンピューティング研究開発政府予算

(http://www.access.gpo.gov/su_docs/budget/index.html)

2.4.1 2000 年度政府予算成立プロセス

A Citizen's Guide to the Federal Budget (Budget of the United States Government Fiscal Year 2000) (<http://www.access.gpo.gov/usbudget/fy2000/guide03.html#The President's Budget>)によると、2000 会計年度予算決定、及び執行プロセスは次のように予定されている。

表 2.1 予算策定予定

年月 (暦年)	プロセス
1998 年 2-12 月	President's Budget 作成:各省庁作成の要求を OMB がまとめ、大統領が吟味し予算教書とする。
1998 年 12 月 -1999 年 2 月	予算教書を確定し、議会にゆだねる。
1999 年 3-9 月	議会による予算教書審議:議会は President's Budget を吟味し、独自の予算をつくり、歳入・歳出の法案を承認する。
1999 年 10 月 1 日	2000 年会計年度開始
1999 年 10 月 1 日 -2000 年 9 月 30 日	省庁のプログラムマネージャが法に則った予算を実行する。
2000 年 10-11 月	会計年度にわたる実行された歳入・歳出のデータが得られる。

2.4.2 米国 HPCC 予算の変遷

2.4.2.1 全体の変遷

大統領予算教書の中で、HPCC 予算は、主要な個別予算の一つとして取り上げられており、NSTC 所管となっている。1991 年 HPC Act が成立し、その後具体的に予算が計上されてきた。その変遷を表 2.2 に示す。

多少の増減はあるが、概略年々増額されている。1997 年度以降\$100M~\$150M の増加が達成、もしくは見込まれており、2000 年度の HPCC 予算は\$1,462M(約 1,754 億円、1\$=120 円換算)である。

ちなみに、2000 年度全予算は\$1,766B(約 212 兆円)でそのうち、自由投資できる分は、\$592B(約 71 兆円)である。全研究開発予算は、\$78,242M(約 9 兆 3,890 億円)で、そのうち、非軍事関連予算は\$39,761M(約 4 兆 7700 億円)で 51%を占める。

表 2.2 HPCC 予算の変遷 (\$M)

省庁	1993 Actual	1995 Actual	1996 Estimate	1997 Estimate	1998 Actual	1999 Estimate	2000 Proposed
DOD	298	375	315	357	220	168	207
NIH	47	68	81	97	98	111	115
NASA	82	131	116	128	120	93	136
DOE	100	119	121	152	489	610	659
NSF	233	297	291	294	265	301	314
商務省	12	30	31	35	20	27	27
EPA	--	12	12	25	3	4	4
運輸省	--	24	23	12	--	--	--
教育省	--	16	12	22	--	--	--
Veterans	--	24	21	6	--	--	--
合計	772	1,096	1,023	1,128	1,215	1,314	1,462

2.4.2.2 DOD、DOE、ASCI 予算

1998 年度には、DOE 予算の大幅な増加が見られるが、これは元々 DOE 独自のプログラムであった ASCI が NSTC すなわち CIC 研究開発小委員会所管になり、予算計上されたことが原因であるが、ASCI への予算配分が大幅に増加していることも原因であると推察する。同時に DOD 予算の大幅な減少が見られるが、この原因として、これまでの DOD の研究開発を ASCI に組み入れた可能性も考えられる。

予算配分から、1998 年度において ASCI は HPCC プログラムに組み入れられることになったと解釈できる。ただ、予算教書では、ASCI は「Advanced Strategic Computing Initiative」となっており、「Accelerated」となっていない理由は不明である。

DOE の 1998 年以降の予算内容を表 2.3 に示すが、非軍事予算は 1997 年度とそれほど変化はないが、軍事予算である ASCI への支出が飛び抜けて大きくなっていることがわかる。

表 2.3 DOE の HPCC 関連予算の内訳 (\$M)

予算種別	1998 Actual	1999 Estimated	2000 Proposed
Energy (Civilian Program)	115	126	116
Energy (ASCI)	374	484	543
計	489	610	659

ASCI: Advanced Strategic Computing Initiative * 2000 年度予算教書 114 ページ

2.4.2.3 1999 年度予算案の特殊性

1999 年度予算教書では、NSTC 統括予算中の IT 関連項目に、HPCC としては掲載されておらず、Large Scale Networking and High-end Computing and Computation なる項目で計上されている。この時点での予算案を次表に示すが、2000 年度予算教書では、連続性をもった予算とはなっていない。

表 2.4 1999 年 NSTC 関連予算案(\$M)

省庁	1999 Proposed
DOD	187
NIH	107
NASA	91
DOE	128
NSF	310
商務省	22
EPA	5
計	850

2.4.2.4 IT² 予算

2000 年度には、前出の IT² が新イニシアティブとして新たにスタートし、強力に推進されようとしているが、予算教書には、Information Technology Initiative として HPCC 予算とは別に、表 2.5 に示す総額\$366M の予算が計上されている。

表 2.5 IT² 予算(\$M)

Agency	Fundamental Information Technology Research	Advanced Computing for Science, Engineering, and the Nation	Ethical, Legal, and Social Implications and Workforce Programs	計
DOD	100	---	---	100
DOE	6	62	2	70
NASA	18	9	1	38
NIH	2	2	2	6
NOAA	2	4	---	6
NSF	100	36	10	146
計	228	123	15	366

<http://www.whitehouse.gov/WH/EOP/OSTP/html/fy2000it.html>

2.5 米国ハイエンドコンピューティング研究開発技術動向 (1999 年度 Blue Book)

2.5.1 はじめに

Blue Book は、テクノロジー委員会 (CT、旧 CIC 委員会)が編纂し、大統領予算教書作成のため大統領府に毎年提出される研究開発計画書である。この中で、HECC の研究開発への投資は、ハイエンドコンピューティングにおける米国の指導力の基盤を築き、産官学のアプリケーションと広範な社会的アプリケーションでのハイエンドコンピューティング/計算の利用を促進すると主張されている。また、研究開発分野への政府機関の慎重な投資によって、科学・工学・経済分野での将来の競争力を保証し、国家の安全のニーズを満たすために必要な計算中心型やデータ中心型の分散アプリケーションの開発が可能になるとされている。

HECC の研究開発が貢献する重大な米国連邦政府のミッションニーズとして、具体的に次の項目が挙げられている。

- (1) 国家の安全
- (2) 環境/気候/気象
- (3) 航空探査と宇宙探査
- (4) エネルギー研究 (太陽、燃焼、核融合)

さらに、HECC 推進の原動力となる社会的アプリケーションは、次のような広範な領域に及ぶと主張されている。

- (1) 保健医療
- (2) 危機管理/自然災害警告
- (3) 長期的な環境・エネルギー管理
- (4) 教育と生涯学習

上記の目的を達成するため、HECC 研究テーマを次のように規定している。

- (1) 複雑なシステムにおける物理的・化学的・生物学的モデリングおよびシミュレーションのためのアルゴリズム
- (2) 大量の情報を必要とする科学・工学のアプリケーション
- (3) 今後何年にも渡ってコンピューティング分野の最先端を米国が歩み続けることを可能にする量子・生物・光コンピューティングの先進概念

このような思想のもとに行われてきた連邦政府の HECC 研究開発の成果として、スケラブルな並列処理の確立、超並列システムの創出、科学的基礎（教育訓練を受けた人材、アーキテクチャに関する新しいアプローチ、次世代技術など）の創出、20 億ドル規模のミッドレンジ高性能コンピューティング市場の技術的基礎の実現、政府の経費削減、ASCI プログラムの短期的技術目標実現が挙げられている。

次の目標は、1 ペタフロップスの持続的処理速度の実現である。適切な資金と技術的な注目が集まり重要視されることにより、この目標は 2007 年までに達成されると示唆されている。

Blue Book では、連邦政府の HECC に対する投資をスラストと呼ぶ次の 4 分野に分類して述べている。

(1) スラスト 1: システムソフトウェア技術

テラフロップス規模のシステムの有用性と効果を産官学の各種のアプリケーションについて改善することを目的とし、中期的な技術開発（3～5 年）が中心である。

(2) スラスト 2: 未来世代コンピューティングのための最先端研究

重点目標は、ペタフロップス (Petaflops) の持続的処理速度とエクサバイト (ExaBytes) の記憶容量を達成するための長期的な研究と技術開発である。

(3) スラスト 3: アプリケーションへの新技術導入

多くの政府機関は、開発された新アプリケーションを最初に導入するため、また、問題解決に計算科学のアルゴリズムを使うために、大規模な計算中心型やデータ中心型の省庁ミッション駆動型の科学アプリケーション開発に投資する。

(4) スラスト 4: HECC 研究用のインフラ

最大の処理パワーとバンド幅を備えたバランスのとれた R&D インフラを保証することを目標とし、大規模テストシステム上と大規模・高性能の計算網/ネットワーク上に構築される研究設備を支援する。

次項からは、各スラストに述べられた研究開発テーマをアーキテクチャ、ソフトウェア、アプリケーションに大別し、投資元である省庁ごとに分類した結果を簡単に紹介する。詳しくは、付属資料を参照されたい。

2.5.2 アーキテクチャ

2.5.2.1 DARPA

- (1)現在の半導体ベースのコンピューティングの限界を超えたコンピューティング技術としてウルTRASケールコンピューティング研究開発を推進する。
内容は、量子コンピューティング、分子コンピューティング、DNA 論理演算、アモルファスアルゴリズムなどである。2.7 項で詳述する。
- (2)異機種群を対象とした、安全でスケーラブルなコンピューティング技術と通信技術の基礎（高効率の並列ノード、スケーラブルシステム用の I/O の自動並列化、超低電力システム、セキュリティをサポートするネットワーク分散共有メモリワークステーションクラスタ）。

2.5.2.2 NSF

- (1)非 2 進法光記憶システム
光記録の平方インチ当たりの情報密度を高めるため、光メモリの信号処理コンポーネントの設計を行っている。
- (2)CDMA 通信システムにおけるマルチユーザ検出
デジタル・セルラ通信システムで複数のユーザに同じ周波数を割り当てることを可能にすることを目的としている。
- (3)知識ネットワークキング
大学での次世代相互接続ネットワークのためのデータベースと協調技術の開発。

2.5.2.3 DOE

- (1)MTA (Multi-Threaded Architecture)
ペタフロップスシステムのためのソフトウェアとアーキテクチャを開発するため、Tera Computer システムを用い、最も重要なペタフロップスアーキテクチャである MTA を評価し、かつ DOE ミッション関連のアプリケーション性能評価を行う。

2.5.2.4 NSA

- (1)HTMT(Hybrid Technology Multithreaded architecture)
ペタフロップスの性能を短期間で達成するシステムとして、超伝導技術、高速 VLSI 半導体技術、光インターコネクト、記憶技術などの最先端技術を組み合わせて使う。基本アーキテクチャはマルチスレッド式実行モデルである。

(2) 量子コンピューティング

量子計算理論および実験に関する予備研究を継続する。アルゴリズムと計算量理論、量子誤り訂正 (quantum error correction) 手法、量子デコヒーレンス・プロセス、小規模の研究所の実現を含む。

(3) MARQUISE 計画

(a) 組み込み型高性能ベクトルコンピュータ SOLITAIRE の実機搭載試験

(b) 小型噴霧冷却式組込型ダイヤモンド電源 (電力密度 2 kW/in³)

(c) 組み込み型のスケーラブルノード (MARQUISE の後続アーキテクチャ)

(4) 高速コンピューティング用マイクロエレクトロニクス

合成ダイヤモンド実装技術、全光学式スイッチング、光電子集積回路の実装技術

(a) 電力節減のための使用地点での電源変換

(b) 高性能 MCM の低消費電力版を実現するためのエリアアレイ I/O 設計の研究

(c) 128x128 超伝導クロスバススイッチ

(d) 標準アーキテクチャ用プログラマブルアクセラレータ・プラグイン

(5) 高速回路

テラビットデータ転送用 100 Gb/s 光ファイバリンクが電子回路が扱えるかどうかを調査する。

(6) LOTS

LOTS は、光テープシステムであり、従来の大型記憶装置と比べてテープ当たりの記憶容量は 10 倍、入力データ速度は 10 倍以上優れている。

2.5.3 ソフトウェア

2.5.3.1 DARPA

(1) スケーラブルなソフトウェアと使いやすさ

アプリケーションプログラマがシステムを容易に使えるソフトウェア開発。

(2) オペレーティングシステムの拡張

演算を MPU とメモリ階層に組込まれた論理回路のどちらで実行するかをアプリケーションが指定できるための命令セットの拡張と記憶コンポーネントの設計。

(3) 地球規模のシステムソフトウェア

サービスの品質を念頭に置いた、プロトコル、アーキテクチャ、ネットワーク伝搬技術の開発、粒度間リソース割り当て機構の開発。

(4)情報存続可能性

低電力アーキテクチャ、リアルタイム適応制御、スケーラブルライブラリ技術、防衛関連ソフトウェアを重点項目とする。

2.5.3.2 NSF

(1)時間変化型 QoS 管理

リアルタイムシステムのモデリングやコンピュータネットワーク、分散システムに関する研究を組み合わせる。

(2)認証付きコード

Java/Web ベースのような環境下で、信頼できないソースからダウンロードされたコードを十分に安心して実行できる方法の開発。

2.5.3.3 NASA

(1)ESS (Earth and Space Sciences)

50 ギガフロップスの持続的処理速度をもつテストベッド上で、競争力がまだないプロトタイプシステムソフトウェアを商用化を目標に開発する。

2.5.3.4 DOE

(1)高性能ソフトウェアツール

超並列処理用高効率 OS と I/O、性能監視用ツール、ハイエンド計算結果の管理・視覚化・理解の改善。ACTS (Advanced Computational Testing and Simulation) のプログラム開発環境用/性能監視用ツールも含む。

2.5.3.5 NSA

(1)UPC

UPC は、CCS(Center for Computing Sciences)が開発した AC の特長と、他の 2 つの並列 C 言語、Split-C と PCP の特長を併せ持つ言語である。

(2)RES

「アイドル状態」のワークステーションを活用することを目的としている。

(3)JACKAL

クリシェ (cliche) をベースにしたソフトウェアリバースエンジニアリング研究システムである。

2.5.3.6 NIST

(1) 性能測定技術とテスト技術

LAN 上のクラスタ化ワークステーションのアプリケーションレベルの物理性能モニタとソフトウェアツールの開発を継続する。また、Linux のカーネルと MPI などで性能情報を提供、かつ情報収集できる手法を開発する。光テープサブシステム設計、DVD 媒体用のテスト方法の調査も実施する。

2.5.3.7 EPA

(1) 環境モデリング・ソフトウェアの並列化

クラス固有の自動並列化。現在のシステム (PA-1) は、3 次元の時間マッチング明示差分コードを並列化できる。

2.5.3.8 DOD (正確には CIC には入らないプロジェクト)

(1) HPC Modernization

- ・ 先進のハードウェアやコンピューティングツールを DOD に提供する。
- ・ CHSSI (Common High Performance Computing Software Support Initiative)
スケーラブルなソフトウェアを DOD の計算科学者および技術者に提供することを目的にしたアプリケーションソフトウェア開発計画である。

2.5.4 アプリケーション

2.5.4.1 DARPA

(1) インテリジェントシステム技術

- ・ 知識表現、推論、機械学習が重点項目。コンピュータによる口語、文語、画像の理解を可能にする。
 - (a) 人間とコンピュータの対話技術。自然な形で対話。可能化技術
 - (b) 入手可能なダイアログ文法と理解に重点
 - (c) 戦場での無線会話の自動筆記のための人間とコンピュータの間の対話
 - (d) 高自由度の構成を可能にする回路、ソフトウェアに重点を置いたマイクロエレクトロニクス科学
 - (e) 立案、スケジューリング、リソース割り当ての先進的方法

2.5.4.2 NSF

(1) 複数の物理現象が関わる問題の高性能シミュレーション

- (a) 航空宇宙工学、土木工学、機械工学シミュレーション
- (b) 複数物理現象シミュレーションの理論的基礎と新しいアルゴリズム
- (c) アプリケーション問題用 PSE (Problem Solving Environment)

(2) 複数モデル・複数領域の計算モデル

航空力学分野と音響学分野でのタスク並列型とデータ並列型領域分解方法開発。

(3) スマートアンテナ

セルラ通信分野の圧縮アルゴリズム改善。

(4) 分子生物学に適用される幾何学形状解析

タンパク質の形状の特定と簡略化の証明。

2.5.4.3 NASA

(1) 先進スーパーコンピューティングアプリケーション

地球の中心核とマントルの成長の 3 次元球体シミュレーション、合成開口レーダのインターフェロメトリ、4 次元のデータ同化、大気/海洋力学とトレーサー化学、極微重力環境におけるスケーラブル並列有限要素計算、星の内部の乱流対流とダイナモ、太陽の活動と太陽圏力学、相対論的天体物理学と重力波天文学用の多目的 3 次元コード。

(2) CAS (Computational Aerosciences)

高速民間輸送と高性能航空機に関する研究。高性能コンピューティング技術による画像による航空機のエンジン設計評価。NREN と CAS のテストベッドへの確実なアクセス。

2.5.4.4 DOE

(1) DOE の Grand Challenge アプリケーション

量子色力学ソフトウェアの生の処理効率、相対論的重イオン粒子衝突実験、ASCI (Accelerated Strategic Computing Initiative) アプリケーション、地球規模の変化のモデリングの視覚化。

2.5.4.5 NIH

(1) 分子構造予測とシミュレーション

医薬品業界での使用を対象にしたアブイニシオ構造予測の新方法を改良する。

(2) 3次元PET(陽電子放射断層撮影)画像の再構成

最尤 (ML、maximum likelihood) アルゴリズムによる3次元陽電子放射断層撮影画像の横断面の再構成、複雑な生物医学組織の理論的研究(リポタンパク質の複合体モデル、プリオンタンパク質など)。

(3) 画像管理と通信システム

医療研究・教育用のマルチメディア通信のための高速遠隔医療ネットワーク試作。

(4) RCWS (Radiology Consultation Workstation)

ATM ネットワークを介した遠隔診察のためのリアルタイム遠隔医療診察用と医療教育用に特化された画像ワークステーション。

2.5.4.6 EPA

(1) 環境保護

スケーラブルな計算アルゴリズムにより、他の政府機関が開発した HECC 技術を統合して、環境問題のための新しいパラダイムの実現可能性を調べる。

(2) スケーラブル・ソフトウェアライブラリ

生態系モデルと人間曝露モデルの開発・使用のためのソフトウェアライブラリ、クロスプラットフォーム分散データ管理・処理方法、高性能地理空間解析手法を開発する。

(3) 環境モデリング (Models-3)

地表レベルのオゾン、酸性物堆積、微粒子の都市・地方レベルで大気の状態のシミュレーションを実施する。

(4) 帯水層のモデリング

多孔質媒体帯水層の地域規模の表現に関する研究。事例研究例は、Upper Chesapeake の帯水層での淡水から塩水への変化のシミュレーション。

(5) 大規模環境流と輸送

最小2乗有限要素法と LES (Large Eddy Simulation) を組合わせた乱流のシミュレーション。より大きな流れと輸送の問題を効率的に解決できる。

2.5.4.7 NIST

(1) 計算科学のためのオンライン参照データ

数学ソフトウェア・リポジトリを問題解決環境とする。

(2) 高性能線形代数ソフトウェア

疎行列カーネルの標準インタフェースの仕様を定めている。C++とJavaによる数値ツールキットの開発。

2.5.4.8 NOAA

(1) NOAAにおけるアルゴリズム開発

将来の予報システム最適化のため、従来の共有メモリマシンからスケーラブルシステムへの変換を促進するソフトウェアツールを開発する。

(2) 天気予報の科学の推進

既存の数値天気予報モデルの超並列コンピューティングシステムへの転送を容易とするSMS (Scalable Modeling System) ソフトウェアツールセットの開発。

(3) GFDL (Geophysical Fluid Dynamics Laboratory)

大気中の温室効果ガスのレベルとハリケーン発生との関連のシミュレーション。

2.5.5 インフラストラクチャ

2.5.5.1 はじめに

米国では、連邦政府機関により多数のCIC研究施設が運営されており、その活動は成功していると考えられている。その要因として、つぎの項目が挙げられている。

(1) 多くの可能化技術 (例、高速ネットワーク、スーパーコンピューティングシステム、並列アーキテクチャ、大容量データ記憶、仮想現実表示デバイス) 開発

(2) 研究者や研究施設のスタッフ、ハードウェア/ソフトウェア・ベンダ、業界の外郭団体の献身

また、全研究施設が、幼稚園から大学の学部までを対象にした広範な教育機会と、研究者、大学院生、学部生のための訓練を提供し、またシステムとアプリケーションの両方について最適ベンチマーキングを実施している。

2.5.5.2 研究施設の要件

これらの施設は、次の作業を行う能力を有しているとされている。

(1) 初期のプロトタイプシステムを評価しフィードバックを開発者に返す。

(2) 視覚化システムと仮想現実システムを既存の高性能システムに取り入れる。

- (3)他の施設では利用できないシステムを使ってフルスケールアプリケーションを実行する。
- (4)スケールダウンされたシステムを使って並列ソフトウェアを開発する。
- (5)大規模アプリケーションをリモート接続を介して実行できるようにする、ネットワーク接続性分野の革新的技術にアクセスできる。

2.5.5.3 研究施設の活動資金

これらの研究施設はその資金を主に CIC R&D の政府機関から提供されているが、ハードウェア/ソフトウェア・ベンダからは機器と人材の面で多くの支援を受け、また特定専門分野の政府機関からの資金、州や地元からの資金、業界の外郭団体からの寄付も得ている。これらの施設は、CIC 技術の調査と最終的にはその利用のための低リスクの協力環境を提供している。

2.5.5.4 研究施設が行う活動例

(1)NSF

(a)PACI (Partnerships for Advanced Computational Infrastructure)

標準的な主要大学で利用できる性能レベルより 2 桁優れた高性能コンピューティングを大学の研究者たちが利用できるようにする。(PACI に関しては付属資料 5 を参照)

(2)DOE

(a)HPCRP (High Performance Computing Resource Provider)

ネットワークを用い、DOE のエネルギー科学の研究の計算ニーズを満たすことを目的とする

(b)ACTS (Advanced Computational Testing and Simulation Research) ツールキット

ゲノムシーケンシングとタンパク質折り畳み (protein folding) のためのアルゴリズムの開発・提供に重点を置く

(c)ASAP (Academic Strategic Alliances Program)

ASCI 計画の一部の ASAP は、米国の 5 大学と共同で、高性能シミュレーション性能向上のための活動を展開しており、その詳細は 2.6.3 で述べる

2.5.5.5 CIC研究開発研究施設

ここでは、各種研究施設を主な資金源である政府機関別にまとめている。詳しくは、付属資料4で述べる。

(1)NSF：

(a)スーパーコンピュータセンタ (5 機関)

1985 年度から 1998 年度まで NSF による資金により運営された

(b)PACI センタ (2 機関)

1998 年度から始まる。PACI は NSF Supercomputing Centers を引き継ぎ、HPCC 分野における新規課題をミッションとする

(c)科学技術センタ (3 機関)

(2)NASA：テストベッド (5 機関)

(3)DOE：研究所 (4 機関)

(4)NIH：

(a)システム (2 機関)

(b)National Center for Research Resources' High Performance Computing Resource Centers (6 機関)

(c)National Center for Research Resources' Scientific Visualization Resource Centers (2 機関)

(5)NOAA：研究所 (3 機関)

(6)EPA：システム(1 機関)

2.6 ASCI プログラム

2.6.1 はじめに

1995年6月、10年で10億ドルの予算支出を予定したASCI(Accelerated Strategic Computing Initiative)プログラムは、4年目を迎え、2004年100 Tflopsに向けて順調にマイルストーンを達成している。米国は、核反応の未臨界実験を繰り返しており、それと相補的な関係にあるASCIプログラムの達成は、最重要戦略的政策の一つといえるであろう。大統領の予算教書によると、1998、1999、2000年度の3年間の総計だけで約14億ドルを支出予定となっている。1997年度の予算要求が約1.22億ドルなので、予算面で大幅なAcceleratedになっている。このことから、ASCIが米国のハイエンドコンピューティング研究開発を強力に牽引している様子がうかがえる。

以下に、ASCIマシン開発の現状とソフトウェア、アプリケーション開発状況を記述する。

2.6.2 ASCIマシン開発の現状

2.6.2.1 ASCI Red (Sandia 国立研究所)

Intelが担当しているSandia国立研究所のRedは、ASCIが最初に開発したマシンで、唯一、分散メモリ型MIMDアーキテクチャを採用している。



図 2.3 ASCI Red

(<http://www.sandia.gov/ASCI/images/RedPictures.htm>)

1998年11月のトップ500では、Rmax1.34 Tflops (理論ピーク性能1.83 Tflops)で世界最速である。(http://www.netlib.org/benchmark/top500/top500.list.html)

このマシンは、ユーザから見るとUNIXマシンとして使用できる。実際のOSは、PragonのOSであるSUNMOSをベースとしたCougar(Pumaのインテル版)を用いている。

現状の性能を以下にまとめる。

(http://www.sandia.gov/ASCI/TFLOP/Home_Page.html)

- (1)ピーク性能：1.8 Tflops
- (2)構成：9,326 Pentium Pro/200MHz (4,536 compute nodes)
- (3)メモリ：0.68 Tbytes
- (4)ストレージ：39×1.2 Tbytes
- (5)消費電力：??Mwatts
- (6)ファイバケーブル長：??km(miles)
- (7)床面積 (ディスク領域をのぞく)：149 m²(1,600 ft²)
- (8)キャビネット数：84(CPU)、10(disk)

2.6.2.2 ASCI Blue-Pacific (Lawrence Livermore 国立研究所)

IBMが1996年に\$94Mで開発を契約したBlue-Pacificは、1998年10月、その完成がゴア副大統領により直々に発表された。その諸元は以下の通りである。

(http://www.news.com/News/Item/0,4,28051,00.html?st.ne.2.head)

(http://www.llnl.gov/asci/platforms/bluepac/sstgallery/)

(http://www.rs6000.ibm.com/resource/features/1998/asci_oct/asci_fact.html)

- (1)ピーク性能：3.88 Tflops(Livermore codeによる測定) (1998.10.28)
- (2)RS/6000SPで構成：1,464 nodes (4-way SMP：5,856 PowerPC 604e/332 MHz)
- (3)メモリ：2.6 Tbytes
- (4)ストレージ：75 Tbytes (米国国会図書館蔵書が全て格納できる)
- (5)消費電力：1.0 Mwatts (本体は0.5Mwatts?)
- (6)ケーブル総延長：6.4 km(4 miles)以上のNo.4ワイヤと
80 km(50 miles)のファイバケーブル
- (7)床面積：900 m² (8,000 ft²)
- (8)重量：52.5 tons(105,000 pounds)

(詳しくはhttp://www.llnl.gov/asci/news/phase1_SST.htmlを参照)

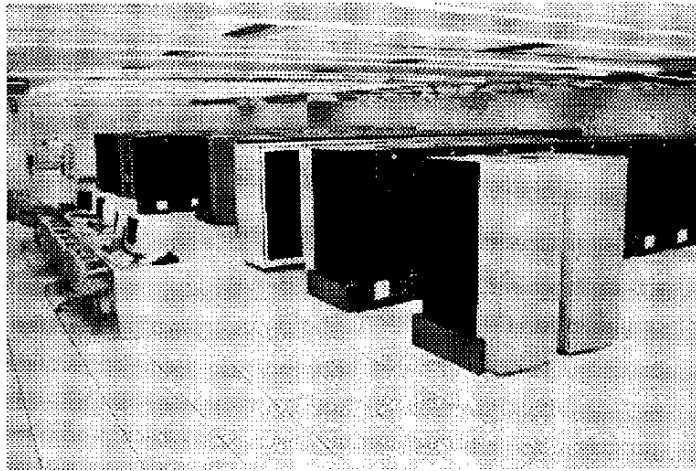


図 2.4 ASCI Blue-Pacific

(<http://www.llnl.gov/asci/platforms/bluepac/sstgallery/>)

このマシンは、核兵器シミュレーションの3次元化に用いられるが、ゴア副大統領は医療、製造、航空機、気象の分野への適応も指摘している。IBM は、スーパーコンピュータのより広い応用として、データマイニングやウォールストリートでのポートフォリオ最適化などのビジネス分野を見ている。

RS/6000 SP のより小さなタイプは Schwab のような会社で用いられており、そこでは、Blue Pacific の 1/14 の 100-node システムが Web-based stock trading のために稼働している。また、United Airlines では、旅客輸送管理に用いられている。現在、全世界で 5,000 台の RS/6000 SP システムが稼働している。

2.6.2.3 ASCI Blue-Mountain (Los Alamos 国立研究所)

3 研究所の中では最も情報の開示が少ない。現在、Los Alamos 国立研究所の Web ページ(<http://www.lanl.gov/asci/>)の中に開設された Strategic Computing Complex なるページが充実されつつある(<http://www.lanl.gov/asci/scc/>)。

一方、HIPPI ネットワーク(<http://ext.lanl.gov/lanp/hippi-maps.html>)と ATM ネットワーク(<http://ext.lanl.gov/lanp/atm-maps.html>)については、詳しい記述がある。

1998 年 11 月に、Silicon Graphics により開発された Blue-Mountain の実性能が以下のように発表された。

(<http://www.lanl.gov/projects/PA/News/111298.html#anchor263034>)

(http://www.lanl.gov/asci/bluemtn/ASCI_fly.pdf)

- (1) ピーク性能：1.6 TeraOps
- (2) 構成：48 Origin2000 (128-Way SMP:全 6,144 processor/250MHz)
- (3) メモリ：1.5 Tbytes
- (4) ストレージ：76 Tbytes
- (5) 消費電力：1.6 Mwatts
- (6) ファイバケーブル長：766 km(476 miles)
- (7) 床面積：929 m²(10,000 ft²)
- (8) キャビネット数：384(CPU)、48(meta router)、96(disk)、8(HIPPI switch)



図 2.5 ASCI Blue-Mountain

(http://www.lanl.gov/asci/bluemtn/ASCI_fly.pdf)

Blue Mountain は、1999 年中に nuclear stockpile に関連した何千ものシミュレーションを計算し、 80×10^{18} ops を実行することが期待されている。

2.6.2.4 ASCI White (Lawrence Livermore 国立研究所)とそれ以降の計画

IBM が受注した ASCI White は、10.2 Tflops 達成をめざした RS/6000SP であるが、2000 年半ばに Livermore 国立研究所に納入される予定である。512 nodes を有し(表 2.6 参照)、全必要電力 6.2 Mwatts、設置面積 1,579 m²(17,000 ft²)を必要とする。このため、新建屋を建築している。(<http://www.llnl.gov/asci-scrapbook/>)

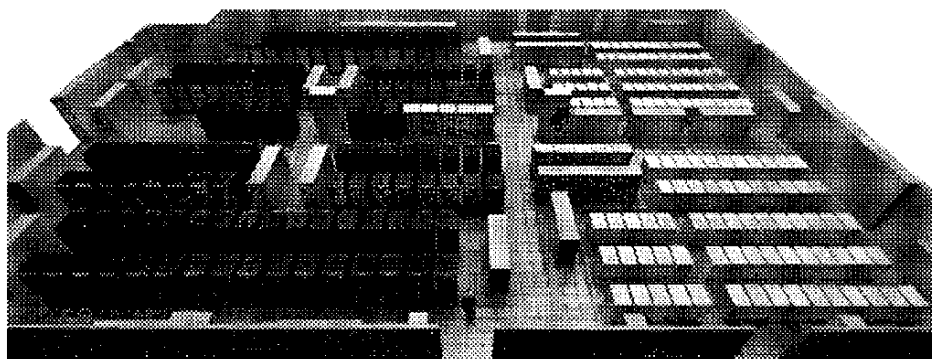


図 2.6 ASCI White (予定図)

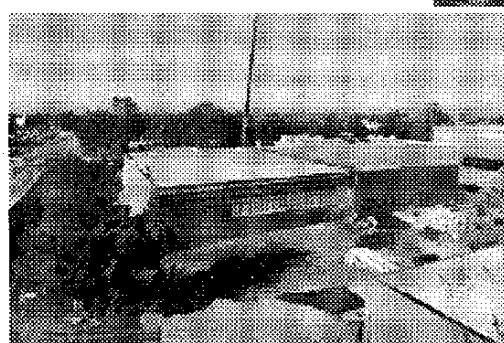
(<http://www.llnl.gov/asci/images/option-white-large.jpg>)



July 23, 1998



September 25, 1998



January 8, 1999

図 2.7 建築中の ASCI White 用建屋
(<http://www.llnl.gov/asci-scrapbook/>)

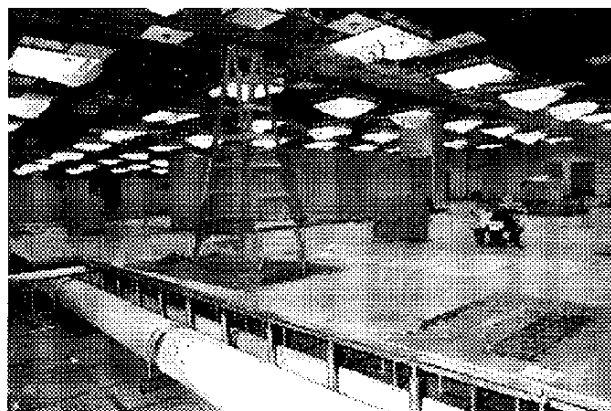


図 2.8 ASCI White 用建屋内部

(<http://www.llnl.gov/asci-scrapbook/990129/slide07.html>)

次の 30 Tflops マシンは Los Alamos 国立研究所で開発される予定であるが、IBM が公表する ASCI Blue 以降のハイエンドマシン性能の開発ロードマップは、表 2.6 に示すように、よりアグレッシブな計画となっている。

表 2.6 IBM の ASCI マシン開発ロードマップ

	1999 年	2000 年	2002 年	2004 年
総合演算速度	3-4 Tflops	10+ Tflops	60+ Tflops	120+ Tflops
デバイス	1 GF/proc 1 GF/chip	1.3-1.6 倍	3+ 倍 6+ 倍	2 倍
SMP 構成	8 procs/SMP	2 倍	2 倍	同左
ノード性能	6-8 GF/node	2.5-3 倍	6-8 倍	2 倍
ノード数	512 nodes	同左	同左	同左

After G.Wenes et al. Proc. on Int. Workshop on Next Generation Climate Models in the Advanced Computing Facilities, March 1999.

Los Alamos 国立研究所の SCC(<http://www.lanl.gov/asci/scc/>)の文書によると、2004 年 100 Tflops マシンまでのシステム要求環境が表 2.7 のように明示されている。これは、ASCI Blue-Mountain を構成する Origine 2000 をベースにした見積もりである。

表 2.7 ASCI マシン環境ロードマップ

System	6 Tflops	30 Tflops	100 Tflops
Date	2000 年 1 月	2002 年 1 月	2004 年 9 月
Power(CPU+disk)	5 MW	7.5 MW	9.0 MW
Heat dissipation	16 M BTU/hr	25 M BTU/hr	30 M BTU/hr
面積(CPU+disk)	~12,000 ft ²	>40,000 ft ²	>40,000 ft ²
Power(cooling)	3.4 MW	4.5 MW	5.0 MW

2.6.3 ASAP (Academic Strategic Alliances Program)

CIC 研究開発小委員会の計画には、DOE の ASCI 計画の一部である ASAP が組み込まれている。現時点では、ASAP は米国の大学と共同で、高性能シミュレーションの性能向上のため、次に示す研究開発を行っている。

(1)Stanford 大 -- CITS (Center for Integrated Turbulence Simulations) は、ガスタービンエンジン設計用のシミュレーション技術を開発する。このプロジェクトの 1999 年度の目標は、ロータ/ステータ流 (rotor-stator flow) のシングルステージの非定常シミュレーションである。(http://www-fpc.stanford.edu/CITS/)

(2)California 工科大 -- Computational Facility for Simulating the Dynamic Response of Materials は、高性能爆薬が引き起こす衝撃波がさまざまな相（すなわち気体、液体、固体）の物質に与える影響を調べる。1999 年度の目標は、気体相の爆発をベースに実験上の反応速度を明らかにすることである。

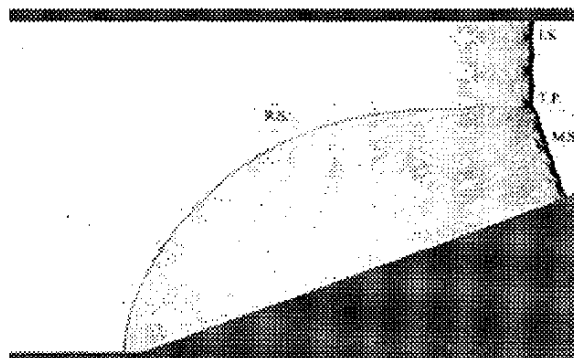


図 2.9 爆発による衝撃波の相図

(http://www.cacr.caltech.edu/ASAP/)

(3)Chicago 大 -- Center for Astrophysical Thermonuclear Flashes は、天体物理学的热核融合発光をシミュレーションと解析によって調べる。1999 年度の目標は、超並列マシン上で FLASH 問題を解決するための適応グリッド方法を開発することである。
(<http://www.asci.uchicago.edu/>)

(4)Utah 大/Salt Lake -- Center for Simulation of Accidental Fires and Explosions は、偶発的な火災と爆発（特に、高引火性物質の取り扱いと貯蔵の際に発生するもの）の数値シミュレーションのために最新科学をベースにしたツールを提供する。1999 年度の目標は、延焼シナリオのシミュレーションの実施である。



図 2.10 シミュレーション対象の火災・爆発
(<http://www.csafe.utah.edu/>)

(5)Illinois 大 (Urbana-Champaign) -- Center for Simulation of Advanced Rockets は、正常/異常な動作条件下での固体燃料ロケットの詳細な完全システムシミュレーションに重点を置く。1999 年度の目標は、異常なシナリオの予備調査である。

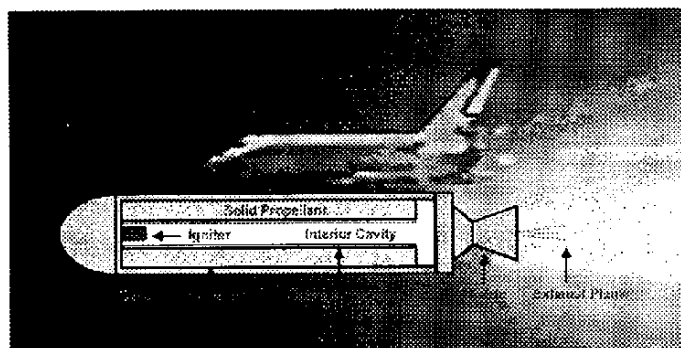


図 2.11 シミュレーション対象の固体燃料燃焼
(<http://www.csar.uiuc.edu/>)

最初の 5 年間の協力期間中、各大学は DOE の Los Alamos 国立研究所、Lawrence Livermore 国立研究所、Sandia 国立研究所と連携し、国立研究所に設置されたハイエンドコンピューティングシステムの一部を利用する。

2.6.4 ASCI 関連ソフトウェア/アプリケーションプロジェクト

(<http://www.llnl.gov/asci/sc97fliers/snl/ASCIAO.html>)

ASCI アプリケーション用コードは、2010 Stockpile Stewardship を達成するキーとなる要素であり、現実のコストで核実験を必要としない核管理プログラムである。

ASCI は、核兵器の性能、安全性、信頼性、製造可能性をシミュレーションする能力を開発する。シミュレーションは、現実 に即した物理モデルに対して、より高い解像度、3 次元計算をフルシステムで行わなければならない。

図 2.12 にはこのようなシミュレーションを行うプログラムを開発するのに必要なソフトウェア（開発環境）のロードマップを示す。

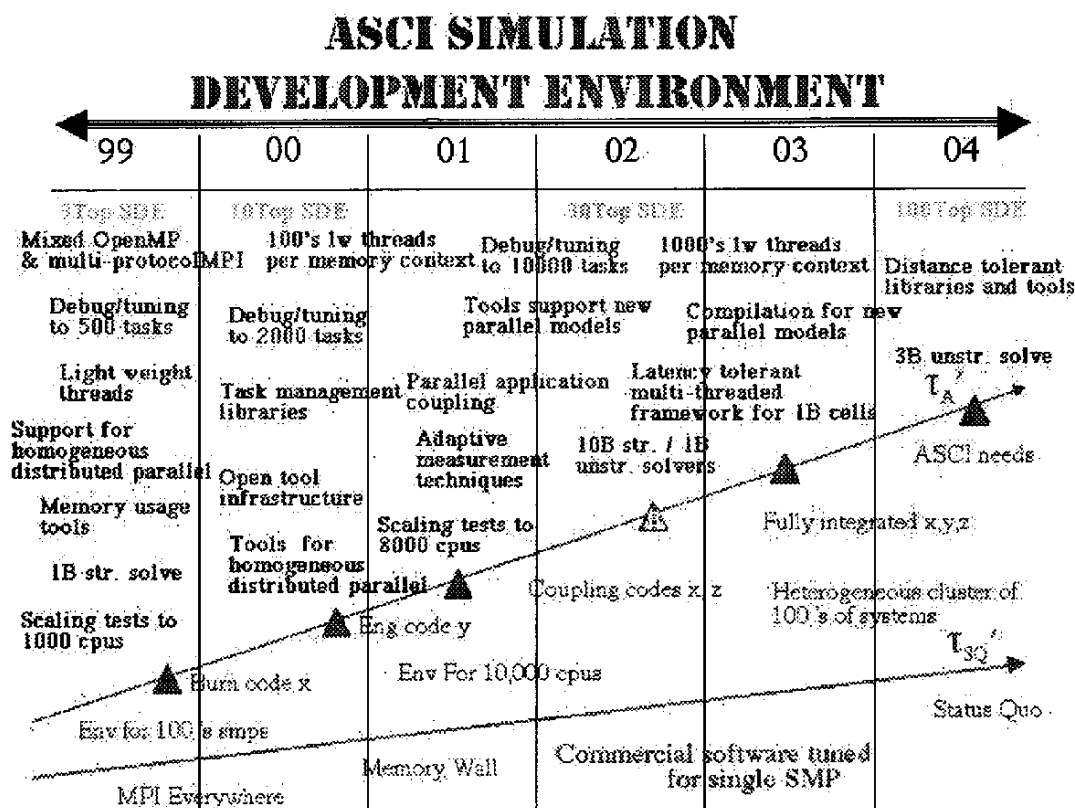


図 2.12 シミュレーション開発環境

(http://www.llnl.gov/asci/pse/ads/sde.curve.jpeg)

次に、ASCI 関連ソフトウェアを開発しているプロジェクトについて、関連する DOE 3 研究所ごとに簡単に紹介する。

2.6.4.1 Sandia 国立研究所

(1) AZTEC (A Massively Parallel Iterative Solver Library for Solving Sparse Linear Systems) (<http://www.cs.sandia.gov/CRF/aztec1.html>)

Aztec は、線型システム解法のための並列繰り返し法用のライブラリであり、使いやすさ、効率の良さを特徴とする。global distributed matrix を用いることで単純化が行われている。

関連アプリケーションとして、MPSalsa (Reacting Flows)、GOMA (Free Surface Moving-Mesh)、COYOTEII (Heat Transfer)、Salinas (Structural Dynamics)、CUERVO (E&M)、SUBWAY (E&M / PRONTO)、ALEGRA (E&M)がある。

(2)MPSALSA(Massively Parallel Numerical Methods for Advanced Simulation of Chemically Reacting Flows) (<http://www.cs.sandia.gov/CRF/MPSalsa/>)

ここでは、先進超並列数値アルゴリズムを開発、解析、実現しており、流体の流れと化学反応モデリングの高解像3次元シミュレーションを対象としている。

GaAs MOCVD: Simulation

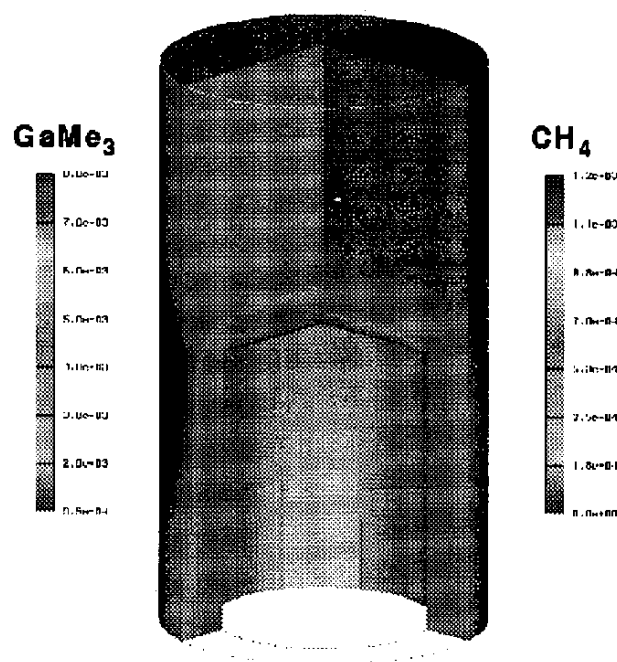


図 2.13 原料トリメチルガリウムと生成物質メタンの質量比等高線
(http://www.cs.sandia.gov/CRF/MPSalsa/main_pictures.htm)

図 2.13 には、トリメチルガリウムを原料とし、メタンが生成される場合の両物質の質量比等高線を示す。物質生成面で、トリメチルガリウムが減少し、メタンが生成されるのが示されている。

(3)CUBIT (Mesh Generation Research Project) (<http://endo.sandia.gov/cubit/>)

CUBIT は、2D、3D の有限要素法用のメッシュ生成ツールを提供する環境で、四角形・六角形要素のメッシュを生成する。

- (4) ALEGRA (Arbitrary-Lagrangian-Eulerian General Research Applications)
(<http://sherpa.sandia.gov/9231home/alegra/alegra-frame.html>)

ALEGRA は、3次元、多種物質、Multi-Material を対象とした固体力学用コードで、一世代上の大きな変形と、衝撃を取り扱う。

HAMMER、EMMA、HYRUM、HELIOS、などのプログラムから構成される。

- (5) Computational Sciences, Computer Sciences, and Mathematics Center
(<http://www.cs.sandia.gov/>)

このセンターは、DOE 内外のユーザのために、広範な各種プログラムの研究開発のサポートを行い、超並列コンピューティング分野を担当している。Massively Parallel Computing Research Laboratory (MPCRL) 内にある。

- (6) Equation Solver Interface (ESI) Standards (<http://z.ca.sandia.gov/esi/>)

方程式ソルバとその要素の標準セットの開発を目標としている。ASCI の問題を解くために、スケーラブルな解法、要素、ツールの開発が対象である。

2.6.4.2 Laurence Livamore 国立研究所

- (1) ALE3D (Arbitrary-Lagrangian-Eulerian)
(<http://www.llnl.gov/bdiv/ale3d/>)

ALE3D は、非定形のグリッドでの流体や弾性・非弾性の応答を扱うコードである。

- (2) ASCI Turb (<http://www.llnl.gov/casc/asciturb/>)

これは、3次元圧縮性乱流のシミュレーションで Large-Eddy Simulation (LES) 法を用いる。シミュレーション対象は、次の項目である。

- (a) 衝撃波通過を含む多様な力学的メカニズムに対する一様、等方な乱流
- (b) 高度に二層化された媒体で、連続的(Rayleigh-Taylor)もしくは瞬間的(Richtmyer-Meshkov)に増殖する乱流

図 2.14 には、Rayleigh-Taylor 不安定に乱流が混合する様子を示す。全領域は各軸方向に512の立方体格子で分割され、128 ノードを有する ASCI Blue-Pacific ID System により計算された。

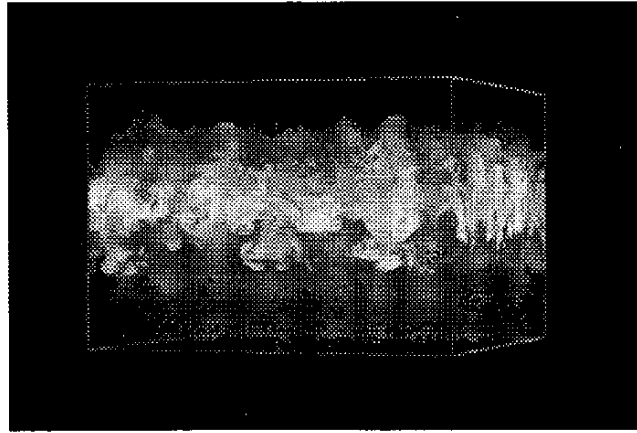


図 2.14 Rayleigh-Taylor 不安定性と乱流混合
(http://www.llnl.gov/casc/asciturb/images/turb_fig3-lge.gif)

(3) The Center for Applied Scientific Computing (CASC) (<http://www.llnl.gov/casc/>)

CASC は、高性能コンピュータと高効率の新しい計算法を必要とする科学研究を先導する。

(4) CASC data sheets (http://www.llnl.gov/CASC/data_sheets/)

CASC のデータは数値ライブラリで、パブリックドメインとして提供される。
ライブラリ例を次に示す。

Aztec 1.1、BlockSolve95、BPKIT、DAGH、Diffpack、KeLP、PETSc 2.0、PGSLib、
PIM、PLAPACK、POOMA、PVODE's SPGMR Solver

(5) MeshTV (<http://www.llnl.gov/bdiv/meshtv/>)

MeshTV は、2、3 次元メッシュ上のデータを図示や解析を行う双方向のグラフィカルな解析ツールである。多くの異なる種類のメッシュを取り扱え、データをいろいろな方法で可視化でき、元々の可視化装置と同じ速度でデータを図示することができる。

(6) Python framework (<http://www.python.org/>)

Python は、インタープリタで、双方向なオブジェクト指向の言語であり、Tcl、Perl、Scheme、Java. と比較される。Python は著作権で保護されるが、自由に用いられ、頒布でき、商用に用いても良い。

(7)PACT library (http://www.llnl.gov/def_sci/pact/pact_homepage.html)

PACT (Portable Application Code Toolkit) は、統合されたポータブルなソフトウェア開発環境である。これにより、過剰に存在するハードウェアアーキテクチャと OS 及び標準のない状態を克服することができる。

2.6.4.3 Los Alamos 国立研究所

(http://www.llnl.gov/asci/applications/apps_lanl.html)

Los Alamos 国立研究所でも次のようなプロジェクトが存在することが Web で紹介されているが、詳細の記述がない。

(1)Antero : プロジェクトリーダー名 Ken Koch、Joyce Guzik

(2)Blanca : プロジェクトリーダー名 Stephen Lee、Art Dana

(<http://www-xdiv.lanl.gov/XCI/PROJECTS/BLANCA/>)

Blanca は、Los Alamos 国立研究所の ASCI のみのプロジェクトであり、完全に真のオブジェクト指向パラダイムを用い、C++で書かれている。

(3)Crestone Leaders: プロジェクトリーダー名 Mike Clover、Bob Weaver

(4)Shavano: プロジェクトリーダー名 Don Burton、John Pedicini

(5)Telluride: プロジェクトリーダー名 Doug Kothe

2.7 ウルtrasケールコンピューティング (UltraScale Computing)

(<http://www.darpa.mil/ito/research/ultra/index.html>)

2.7.1 はじめに

米国ハイエンドコンピューティング研究開発政策の根幹は、呼称はどうあれ、HPC Act で規定されてきた HPCC 計画と新設の IT² イニシアティブであり、その現在の情報処理能力目標は、ペタフロップスコンピューティングを実現することである。このため、米国政府は、ASCI(Rainbow)マシンに代表される、多数のスカラプロセッサで構成される超並列コンピュータの開発に注力する一方で、超伝導素子 RSFQ を MPU とし、各種の半導体・光技術を結集した HTMT アーキテクチャによる実現にも投資をし、目標達成に邁進している。

ペタフロップスマシンは、一般にはまだ遠い目標であると考えられているが、米国は早ければ 2007 年に目標を達成する準備をしている。さらに米国は、すでにペタフロップスの次をめざす研究にも着手している。その目標は、エクサフロップス (Exaflops=1,000 Pflops) である。

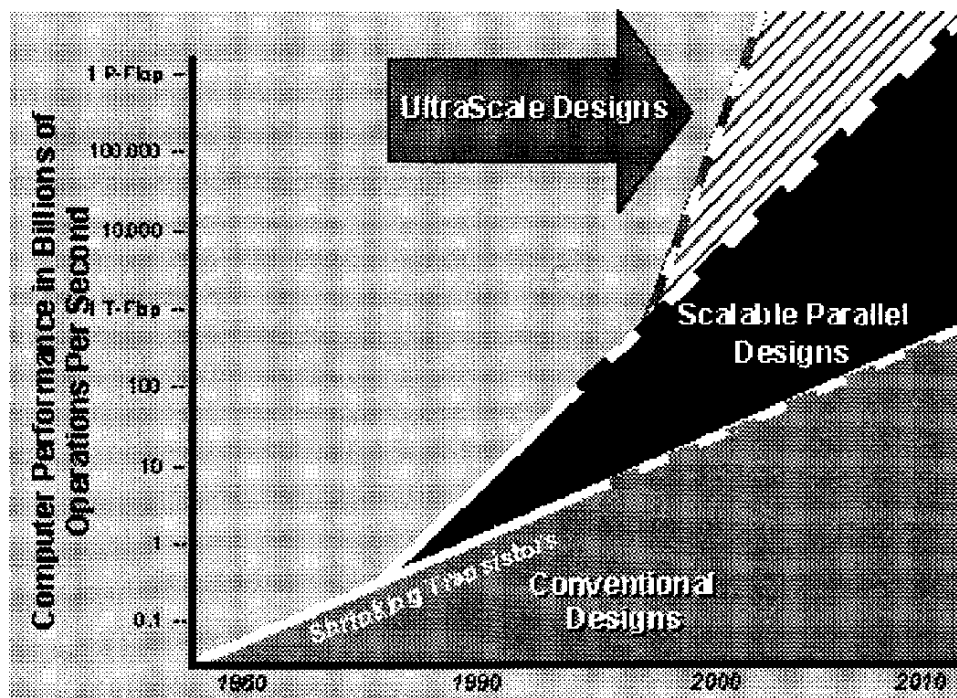


図 2.15 DARPA のウルtrasケールコンピュータ計画の位置づけ

(<http://www.sainc.com/arpa/ultrascale/perform.htm>)

DARPA が開発投資をしているウルトラスケールコンピューティング研究計画は、現在の半導体技術をベースとしたコンピューティング技術(COTS: Commercial Off-The-Shelf)とは異なる方法を開発することにより、より早く、より高いコンピューティング能力を実現しようとしている(図 2.15)。

この研究計画の記述は、NCO が発行する 1998 年度 Implementation Plan と 1999 年度 Blue Book から登場している。すなわち、この名称で HPCC 予算が割り振られている。両報告書は、ともに 1998 年 8 月に発行された。1998 年 Implementation Plan には、ウルトラスケールコンピューティングの 1997 年度のマイルストーンが掲載されているが、1998 年度 Blue Book、1997 年度 Implementation Plan には、それに関する記述が無い。このことから、この計画は、1998 年度の 1997 年 11 月から 1998 年 8 月までの間に、これまでのテーマを整理統合してできたと考えられる。

1998 年度 Implementation Plan では、ウルトラスケールコンピューティングは、DARPA のスケーラブルシステム/ソフトウェアの項に、スケーラブルコンピューティング、スケーラブル I/O コンソーシアム、Quorum とともに含まれている。スケーラブルシステム/ソフトウェアは、NSF、NASA、NSA をパートナーとしており、1998 年度予算要求額は、\$40.2M である(内訳: HECC=\$35.2M、HCS=\$5.0M)。

2.7.2 研究開発の目標 (<http://www.sainc.com/arpa/ultrascale/slides/index.htm>)

数値目標が次のように規定されている(図 2.16 は概念図)。

- (1)性能: 1 Exaflops (= 1,000 Petaflops)以上
- (2)コンピューティングユニット数: 10^{11} processors 以上
- (3)素子の大きさ: 原子間距離レベル
- (4)消費電力: 10^{10} ops/joule 以下
- (5)複雑さ: 手に負えない複雑さを扱うことができるようにする
- (6)賢さ: 推量や創造性をもたせる

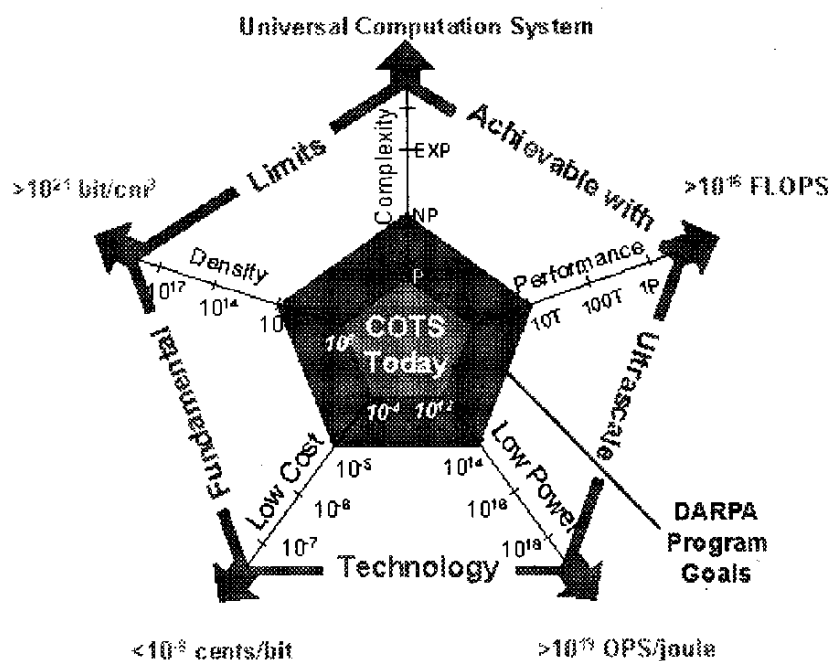


図 2.16 目標概念

(<http://www.sainc.com/arpa/ultrascale/limits.htm>)

- 2.7.3 研究内容 (<http://www.sainc.com/arpa/ultrascale/detail.htm>)
(<http://www.sainc.com/arpa/ultrascale/slides/index.htm>)
(<http://www.darpa.mil/ito/research/ultra/challenges.html>)

プログラムの研究分野は3つに分類され、各分野での研究目標と達成のための研究項目が次のように提案されている。

(1) 計算方式の新モデル

$10^4 - 10^5$ プロセッサは、固定アレイ、Adaptable Software の適用、 $10^6 - 10^9$ プロセッサは、Address-Free のハードウェアの適用、 2^N プロセッサは量子コンピューティングを適用するとされている。

(a) 並列コンピューティング (Continuum Computer Architecture: CCA)

CMOS の可能性探求、CC アーキテクチャによる 100 Tflops 実現、 10^{12} 演算素子
(<http://www.sainc.com/arpa/ultrascale/caltechc.htm>)

(b) 群コンピューティング (Swarm or Amorphous Computing)

偏微分方程式を解くための百万要素を持つ自動機械のシミュレーションとテスト
(<http://www.sainc.com/arpa/ultrascale/mit.htm>)

(c) 量子コンピューティング (QUIC Quantum Information and Computation)

量子状態を利用して情報の書き込み、処理、読み出しを行う
(<http://www.sainc.com/arpa/ultrascale/caltechq.htm>)

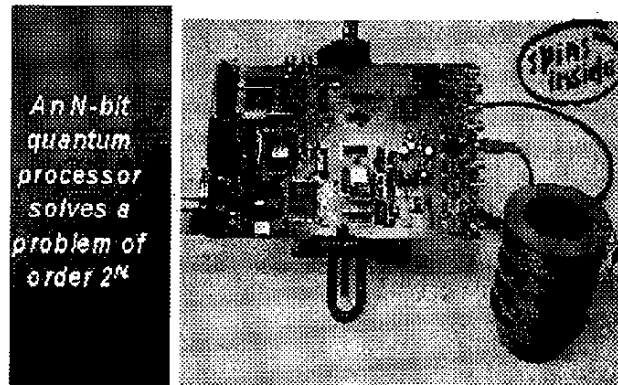


図 2.17 量子コンピューティング

(<http://www.sainc.com/arpa/ultrascale/slides/index.htm>)

(2) 物理的新メカニズム

(a) DNA コンピューティング (DNA Computation of Surfaces)

DNA もしくは他の生体分子中にデータを蓄えたり、分子中で演算を行ったりする。 10^{21} Bits/cm³ 以上の密度の実現

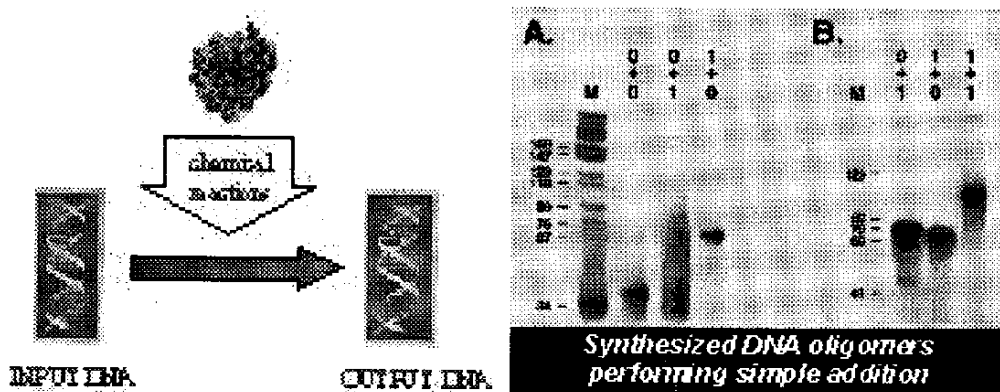


図 2.18 DNA コンピューティング

(<http://www.sainc.com/arpa/ultrascale/slides/index.htm>)

(<http://www.sainc.com/arpa/ultrascale/uw.htm>)

(b)細胞工学 (Cellular Engineering)

バクテリアの遺伝子の配列やその書き換えを用いて有限状態機械を実現する。
超低コスト実現

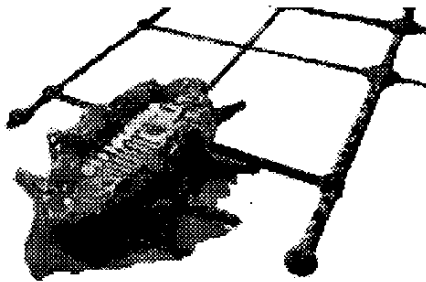


図 2.19 細胞工学概念図

(<http://www.sainc.com/arpa/ultrascale/cellular.htm>)

(c)ニューラルネットワーク (Cultural Neural Networks)

ニューラルネットワークプロセッサのアーキテクチャ研究を進め、脳型コンピューティングを確立を探索する。例としてシリコン上に培養した神経物質

(<http://www.sainc.com/arpa/ultrascale/neural.htm>)

(3)ハイブリッド情報装置

(a)情報処理

各種信号を化学プロセスに変換するバイオ技術による双方向通信

(b)周辺機器

生物的なセンサやアクチュエータと通常のコンピュータとの接続

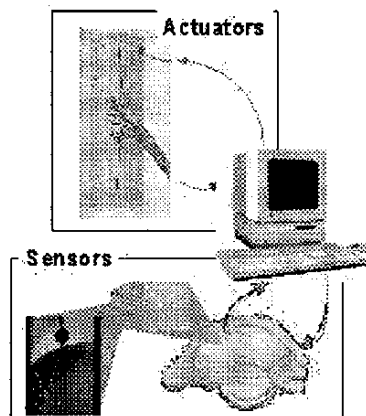


図 2.20 Hybrid Peripherals 概念図

(<http://www.sainc.com/arpa/ultrascale/slides/index.htm>)

(c) ストレージ

器官、細胞、組織で形成されるメモリへの入出力機構

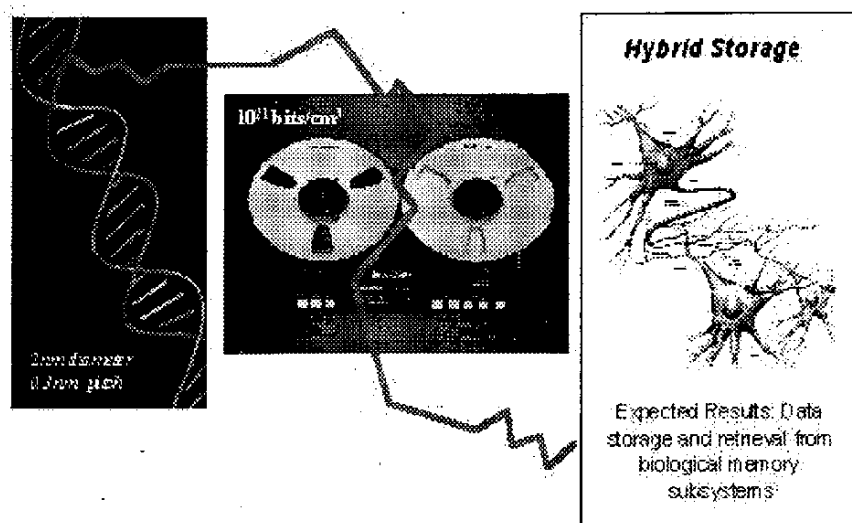


図 2.21 Hybrid Storage 概念図

(<http://www.sainc.com/arpa/ultrascale/slides/index.htm>)

2.7.4 実動プロジェクト

(<http://www.darpa.mil/ito/research/ultra/projlist.html>)

(<http://www.darpa.mil/ito/research/ultra/whatsnew.html>)

1996 年度より 1998 年度までになされた研究を研究機関別に以下に示す。

(1) California 工科大:

(a) Continuum Computer Architecture

(b) Quantum Information and Computation

(2) Cornell 大: Neuronal Cells on Silicon

(3) Duke 大: Prototyping Biomolecular Computations

(4) Harvard Medical School:

(a) Ensemble Quantum Computing by NMR

(b) Genome Engineering of Bacteria

(5) Harvard 大: Oscillating Channels and Sensors

(6) Massachusetts 工科大: Amorphous and Cellular Computing

(7) Molecular Sciences Institute: Elemental Computation

- (8)North Carolina 州立大 : Strongly Coupled Computing Systems
- (9)Princeton 大 : Quantum Communications and Computing
- (10)Raytheon Systems Company : Artificial Nervous System
- (11)Science Applications International Corporation(SAIC) : Fabrication of a Novel
Neuroelectric Device
- (12)Stanford 大 :
 - (a)Bulk Quantum Computation with NMR
 - (b)Genetic Regulatory Networks:Analysis and Simulation
 - (c)New Strategies and Architectures for Brain Imaging
- (13)The Johns Hopkins 大 Applied Physics Laboratory : Neuronal Networks for
Information Processing Applications
- (14)Delaware 大 : Nanoscale Field Localization for Manipulation and Probing
of Computationally Interesting Biomolecules
- (15)Southern California 大 :
 - (a)Theoretical and Experimental Aspects of Biomolecular Computing
 - (b)Computational Aspects of Molecular Science
- (16)Wisconsin 大 : DNA Computation on Surfaces

1999 年度には、次に示すテーマが新たに採用された。

- (1) Delaware 大 : Nanoscale Field Localization for Manipulation and Probing of
Biomolecules
- (2) The Molecular Sciences Institute : Elemental Computation
- (3) Stanford 大 : New Strategies and Architectures for Brain Imaging
- (4) MIT : Interfacing with Large-Scale Neuronal Ensembles
- (5) California 大-Irvine : Hybrid Computation Systems Based on Cultured Brain
Slices
- (6) USC : A Hybrid Neuron-Silicon Computational System for Pattern Recognition
- (7) USC : Computational Aspects of Molecular Science
- (8) NanoDynamics: A Hierarchical Key Structure for DNA Code Storage and Retrieval
- (9) HNC Software : Neuromodem

2.8 おわりに

第2章を終わるに当たり、これまで記述してきたことを筆者の主観によってまとめるとともに、感想を述べたいと思う。

米国における情報技術研究開発政策は、大統領の指導力のもとに、最終的には各省庁が実行する。当然、予算執行も各省庁が行う。この意味で、省庁の持っている権限は強大で、かつ縦割りの弊害が生じるのは必然である。しかしながら、優先項目の選定、政策の整合性、予算配分などの決定を NSTC のテクノロジー委員会や CIC 研究開発小委員会に権限と責任を持たせて決定させることにより未然に防止しようとしている。権限は、委員会の答申が直に政策に反映されることでその効力を発揮する。責任は、委員個人の社会的評価に負うところが大きい、最終的な責任を大統領がとることが特徴で、それによって米国ではすべてが解決されているように思われる。

現在のクリントン政権の特徴である情報技術強化政策の推進は、ゴア副大統領のイニシアティブによるところが大きいと思われるが、今や米国の遠い将来をもにらむ基本方針になっており、基本となるシステム/ソフトウェア指向は政権が変わっても変わることはないであろう。また、研究領域としてのハイエンド/フロンティア分野は、政治、軍事、経済の各分野において、米国が世界で常に優位を保つために必要不可欠であり、PITAC 答申や IT² イニシアティブで主張される長期的な研究開発重視政策は時の情勢によりトーンの違いは出てくるものの、他国に比べ研究投資を緩和することは予想しがたい。

2000 年度大統領予算教書によると、HPCC 予算が\$1,462M(約 1,754 億円)、IT² 予算が\$366M(約 440 億円)である。HPCC 予算のうち 1/3 強の\$543M は DOE の軍事的計画である ASCI に支出され、さらに、DARPA の\$207M を差し引いた残りは、\$712M(約 854 億円)となる。このような投資が一時的な景気対策ではなく、各省庁ミッション実現に必要な技術開発のため、ポリシーを持って継続的になされるところに米国情報技術研究の強みがあると考えられる。

米国ハイエンドコンピューティング技術のめざすところは、この先 10 年を見通すと 1 Pflops の計算能力であり、そのためのソフトウェア/ハードウェアの研究開発に注力している。その実現の原動力は DOE の ASCI であることは間違いないが、さらに NSA、NASA、DARPA は超伝導素子や PIM(Processors-In-Memory)で構成される HTMT アーキテクチャによりペタフロップスを短期に達成しようとする動きもある。これは、暗号解読、ネットワーク情報データマイニングなどのグローバルネットワーク戦略へのペタフロップスマシン早期適用を目論んでいるのではないかと推察され、国家ミッション

としての研究開発テーマの存在が、研究開発推進にいかに関与しているかが示されている。さらに、量子コンピューティング、DNA コンピューティングなど、次世代のコンピューティングアーキテクチャ研究をひとまとめにし、1 Exaflops を目標とした「ウルtrascaleコンピューティング」を DARPA が推進するなど、フロンティアを極める努力を怠っていない。

ここで、情報技術研究開発において、米国が最も意識しているのは日本であることは間違いない。それは、組織、予算、研究開発テーマ等が揃っているように見えるからである。例えば、組織では、表 2.8 のように比較できる。

表 2.8 科学技術政策

米国	大統領	OSTP	NSTC	PCAST	PITAC
日本	総理大臣	科学技術会議 (科技厅)	科学技術会議	科学技術会議	(高度情報通信社会推進本部?)

表 2.8 は、科学技術庁科学技術政策研究所の調査資料 No. 55 を参考にしたものであるが、() 内は、筆者が書き加えた。これから、科学技術会議の責任は非常に重く、重要な位置にあることがうかがえる。

政府支出の情報技術研究開発予算は、そのようにまとめた費目がないので明確ではない。ただし、科学技術基本計画により平成 8 年度から 5 年で 17 兆円を予定し、平成 11 年度には科学技術全体で 3 兆 1,552 億円を支出する力は侮れないと見られて当然である。

研究テーマでは、科学技術庁主導の 2001 年度にピーク性能 40 Tflops で稼働し、気象・気候・地殻変動を計算機上に実現する「地球シミュレータ」計画や、2004 年度に 1 Pflops の速度で分子動力学計算や生体化学反応計算ができる専用計算機実現をめざす理化学研究所の計画があり、これら計画自身は米国の HPCC や ASCI に勝るとも劣らないものである。

このような観点に立つと、わが国の情報技術開発は順風満帆であるが、この分野で、将来の不安を持つ関係者が多いのはなぜであろう。専門家の意見は、本 WG の山口主査や第 3 章の委員の方々の記述をご覧頂きたいが、筆者は次の点を危惧する。

- (1) 組織・予算で、科学技術会議は、省庁横断的な科学技術振興調整費を有しているが、その額は科学技術全体で 270 億円(1998 年度)で、米国の HPCC 予算(非軍事)と比べても約 1/3 である。また、省際調整の実施段階がトップダウンでなく、省庁間

のバランスを崩すような運用はやりにくいのではないか。

- (2)研究プロジェクトテーマが単発で、その先をにらんだグランドデザインが見当たらない。地球シミュレータや理研のコンピュータは、次にどのようにつながっていくのか明確にすべきではないか。

米国の情報技術研究開発政策に関する情報は、インターネット探索だけで明確なコンセプトが入手でき、さらに、Implementation Planのように、HPCC関連の研究開発内容、テーマ毎の参加省庁名、予算の配分などが親切に記述されている文書が公開されており、上述のような危惧が生まれる余地を未然になくしている。

わが国においても、インターネットによる省庁の情報開示は、最近とみに著しいが、米国のように納税者への説明責任を前提とし、全体像とその根拠となる基礎データをわかりやすく提供しようという意欲は、結果として伝わってこない。

ハイエンドコンピューティング研究開発推進には、国民合意形成が最重要であることは言うまでもなく、それを達成するためには、政府から各研究機関に至る全ての階層で、より進んだ情報公開の努力が必要不可欠なものとする。

第3章 わが国における研究開発の課題と展望

3.1 アーキテクチャ

3.1.1 VLSI アルゴリズム・アーキテクチャ・リバイバル

天野英晴委員

3.1.1.1. はじめに

1980年代のはじめにVLSI アルゴリズム・アーキテクチャの研究が盛んに行われた。これらの研究は、それまで既存のアルゴリズムやアーキテクチャをいかにLSI上に実装するか？を考えてきたのに対して、むしろLSIに適合したアルゴリズム・アーキテクチャを開発しようとする動きであった。

これらの研究を通して、(1)同一セルの繰り返しパターンから成る。(2)セル間 コミュニケーションがローカルで済む。(3)入出力ピンが少なく済む。アルゴリズム、アーキテクチャが提案され、特にH.T.Kungの提案したシストリックアレイは、その高い並列性とデータ流に対する巧妙な処理によって注目を集め、実際にシステムも実装された。

しかし、これらのVLSI アルゴリズム・アーキテクチャは、結局多くチップが実装されることなく、1980年の中ごろ以降、RISCに代表される汎用マイクロプロセッサの性能の飛躍の陰に隠れるように姿を消してしまった。

なぜ、VLSI アルゴリズム・アーキテクチャは勢いを失ってしまったのだろうか？本節では、まず80年代当初の研究を振り返り、この原因を考える。そして、これらの研究に光を当て、新しい状況に即したアルゴリズム・アーキテクチャの可能性について考える。

3.1.1.2 VLSI アルゴリズム・アーキテクチャとは？

登場の背景

1980年代はじめ、LSI技術は着実に発展を続け、32bitプロセッサがLSI上に実装可能になった。RISCが提案され、LSIに適したプロセッサは何か？ということで議論が始まりつつあった。この時代はCPUを始めとする様々なシステムがLSI上に搭載できる可能性が現れる一方で、LSIのプロセス実装技術ではなく、設計上のコストの大きさ、困難さが始めて

認識されつつあった時代であった。この辺、システム LSI が実現化しつつある現在と多少似たような状況が見られたのである。

今までボード上で実装されていたシステム、たとえば VAX などの複雑な構造のプロセッサを LSI 上に実装しようとする、設計は困難を極めた。当時はもちろんハードウェア記述言語もなく、CAD 技術は未熟、設計はゲートレベルで行うしかなかったし、ジャンパやリメイクが容易なボードレベルの実装に対して、チップ内で設計が決まったら修正が利かないというプレッシャーは大きい。それで困難の末チップ上に搭載してみても、結局思ったより性能が上がらないのである。

これは、既存の構造のシステム、アルゴリズムを無理やり LSI チップに合わせたのが悪かったのではないか？LSI に合わせてアルゴリズム・アーキテクチャ自体を変える必要があるのではないか？このような考え方で始められたのが一連の VLSI アルゴリズム・アーキテクチャ研究である。(様々な文献があるが、ここでは代表例として[1][2]を挙げるに止める)

この研究では VLSI チップについての以下のような認識に基づいている。

(1)VLSI チップに関して

- (a)チップ内の配線は面積を大きく消費する。
- (b)配線上の伝播遅延は処理速度に対して無視できないほど大きい。
- (c)チップ外部との入出力用のピン数は限られている。

(2)VLSI の設計に関して

- (a)VLSI の設計、テストについてのコストはチップ自体の製造コストに対し無視できぬほど大きい。
- (b)開発期間を短縮することにより新しいプロセスを利用することができ、有利になる。

このことから VLSI に適したアーキテクチャは以下の特徴を持つことが好ましいとされた。

- (1)単純なセルの繰り返し等の規則的な構造を取る。このことにより、設計を容易にし、設計・テストのコストを削減する。
- (2)通信コストを重視した構成を取る。したがって、特に密なデータ交換を必要とする結合は隣接セルのみの間で行われることが望ましい。
- (3)単純セルの繰り返しで高い性能を得るためには、並列処理、パイプライン処理を効果的に利用することのできるアーキテクチャが望ましい。

- (4) 計算操作と入出力操作のバランスをうまく取り、入出力ピン数の制限が性能に影響を与えないアーキテクチャが望ましい。

シストリックアレイ

VLSI アルゴリズム・アーキテクチャの代表選手はH. T. Kung の提案したシストリックアレイである。このアーキテクチャは、単純な計算能力を持つセルを規則構造に接続し、並列かつパイプライン的に動作するものである。外部との通信は境界セルのみで行われ、セル全体は単一のクロックに同期して計算とデータ交換を行う。ちょうど心臓の鼓動に合わせて血液が流れるようにリズムカルに計算とデータ交換を行うことからシストリック（鼓動）アレイと呼ばれる。

シストリックアレイの最も簡単な例が図1に示すパターンマッチ用のアレイである。この例ではセルは一次元に配列され、各セルは1文字の比較を行う。図に示すようにアレイの一端からはテキストを、他の一端からはマッチするパターンを入力すると、各セルは並列に比較を行い、一回パターンとテキストを流してやれば、全照合が終了する。各セルは1クロックごとに「比較動作」とアイドルを繰り返しており、これがシストリック的な動作である。

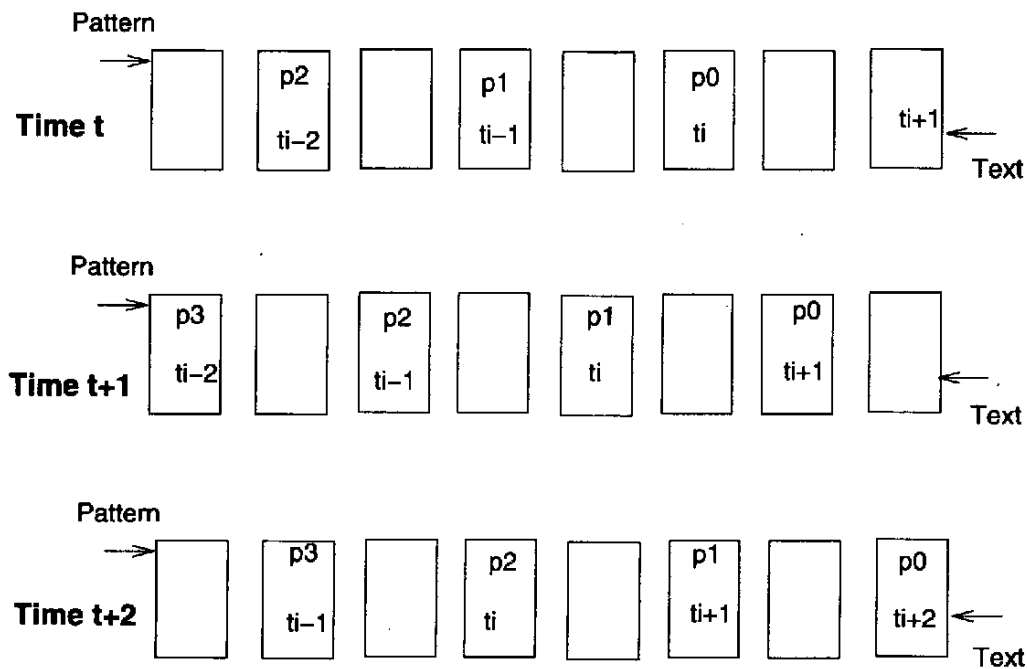


図1 Systolic Array for pattern matching

図2は、幅 W を持つ $N \times N$ の帯行列と N 次元ベクトル x の積を求めるアレイである。この例では一端から x を入力し、上方から帯行列要素を入力してやると、データがぶつかったセルで積和（ $y = y + a x$ ）が取られ、それが流されることによって答え y が出力されていく。必要なセル数は $W/2$ 、実行時間は $2W \cdot W$ である。

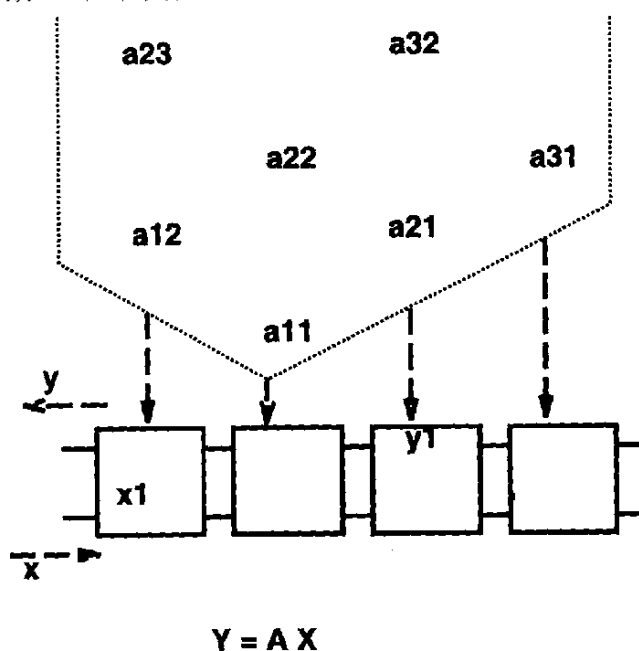


図2 Systolic Array for Matrix vector multiplication

他にも6角アレイによる行列の積計算用アレイ、コンボリューション、フーリエ変換などに対して様々な構造のアレイが提案された。

シストリックアレイは、通信が完全に局所的に行われること、セルの規則構造、並列、パイプライン処理の利用、入出力と計算のバランスの点で先の条件を良く満足している。しかし、すべての応用分野についてこのようなアルゴリズムを構築することは難しい。そこで、シストリックアレイにグローバルバスを付加したセミシストリックアレイも研究された。この構造により FIR フィルタ等の分野にも適用が可能になる。図3は一種のグローバルバスを持った FIR フィルタのセミシストリックアレイによる実現例を示す。

ここでは省略するが、他にも配列形式を持つ演算回路、正方格子状のアレイを持つ計算システム等がVLSI アーキテクチャとして盛んに研究された。

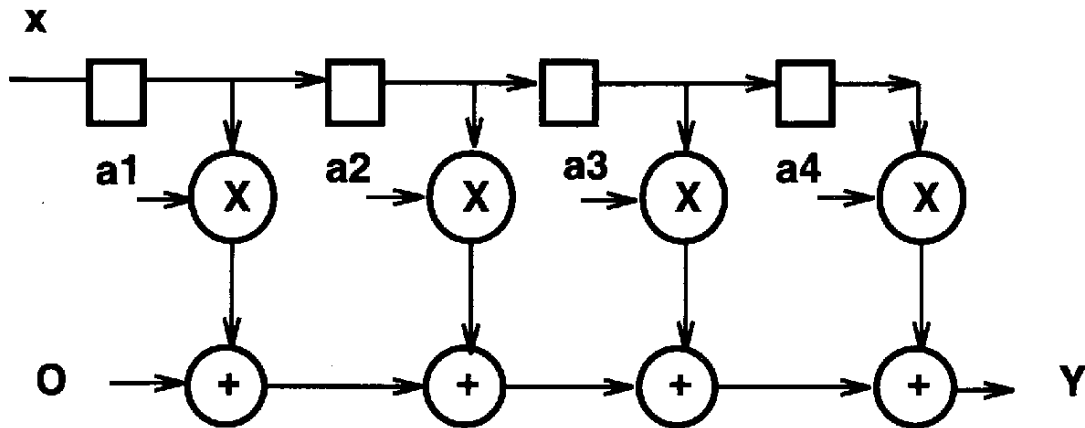


図3 Semi-Systolic Array for FIR filter

3.1.1.3 VLSI アルゴリズム・アーキテクチャはなぜ姿を消したか？

VLSI アルゴリズム・アーキテクチャは1985年くらいまでは盛んに研究がおこなれたが、1980年後半には勢いを失った。この原因としては様々なものが考えられるが、私見として以下の点を挙げておく。

- (1) VLSI 設計上の困難がCADの進展によって大幅に緩和されたこと。特に1980年代後半から急速に発達した論理合成・圧縮技術によって、ハードウェア記述言語によるRTLレベルの設計が可能になった。このことにより、ある程度複雑なアーキテクチャであっても短期間で市場に投入できるようになり、結果として単純セルの組み合わせのメリットが目減りした。
- (2) 同様にCADの発達により、配線上の遅延が予想したより性能に影響しなかった。チップ全体に渡るバスは要所にレジスタを入れたマルチプレクサとして実装されるようになり、グローバルバスの実装もさほど気にならなくなった。
- (3) パッケージ技術の発達により、予想以上にLSIのピン数を大きくすることができるようになった。依然としてLSI間のデータ交換は性能に大きな影響を与えるが、実はシストリックアレイも構成によってはかなり入出力を要するものもあり、ピン制限を与える性能に対する影響という点で大きく優位に立つことができなかった。
- (4) シストリックアレイ自体が当時のチップ上に実装可能なものが少なかった。1980

年代はじめのLSI技術では、セルには単純なビット演算や比較演算しか入れることができなかった。このため、シストリックアルゴリズムの中でかなりの割合を占める浮動小数点演算を伴うアルゴリズム（行列演算、FFT など）を1チップ上に実装することができなかった。

- (5)シストリックアーキテクチャはセル間の通信路、セルの構成が固定されてしまっており、変更を行うことが難しかった。このためきわめて特定されたアルゴリズムに制限されてしまった。

このような状況で、VLSI アルゴリズムのFlagship的な存在であったシストリックアルゴリズム、アーキテクチャの失速によって、VLSI アルゴリズム、アーキテクチャの研究は下火になり、その考え方や成果が忘れ去られようとしている。（もちろんシストリックアレイは最近でも一部で研究、実装が行われている。、またシストリックアレイを実証するために開発されたWARP[3]は iWarp の実現に関連して VLIW アーキテクチャという凄い副産物につながっている）

3.1.1.4 ネオ VLSI アーキテクチャの可能性

VLSI アルゴリズム・アーキテクチャ登場の背景には、

- (1)半導体の面積を効率的に利用するためのアーキテクチャの設計能力が追いつかなくなった
- (2)半導体上に様々なシステムが搭載される期待と必要性が広がった

の2点があり、現在、回りまわってかなり似通った状況になっている。半導体の面積の増加は、発達した CAD を用いたとしても、効率的に利用する設計を行うことが困難になりつつあるし、様々なシステムを単一チップに搭載するシステム LSI への期待が広がっている。したがって、今 VLSI アルゴリズム・アーキテクチャに再び光が当たる可能性はあるといえる。

まず、単純に考えて状況が変わったことによって以下の点が有利になっている。

- (1)現在複数の浮動小数点演算器をセルとして単一のLSIに実装可能となっている。
このことにより長年実現が困難だった数値演算向き VLSI アーキテクチャが実際に構築可能になっている。
- (2)LSI-IPの発達により単純なセルの構築がきわめて容易になっている。

応用分野によっては、昔のアルゴリズムを掘り出して実装するだけで、ある程度の成功を収めることができるかもしれない。しかし、それでは本格的なインパクトを与えることは難しい。そこで、さらに状況の変化を考えて、VLSI アルゴリズム・アーキテクチャを練り直す必要がある。

手品の種は以下の通りである。

- (1) DRAM の混載技術：シストリックアルゴリズムは基本的にメモリは外にあり、データは流し込むもの、として設計されている。これは当時 LSI チップ中に大容量の DRAM を混載させることが困難であったためである。ところが、シストリックアルゴリズムの中で最も巧妙な方法のうちのいくつかは、かなり大容量の、しかも固定したデータを繰り返し流し込む必要が生じてしまう。図1の照合パターンや図2の行列がこれに当たる。これらを DRAM に格納しておき、アラインメントを考えてカラムバッファに対して一度に読み出すことによって、シストリックアレイの使いやすさと応用分野を広げることのできる可能性がある。もちろんシストリックだけではなく、他の VLSI アルゴリズムと DRAM 混載技術の組み合わせも魅力的である。
- (2) FPGA との組み合わせ：VLSI アーキテクチャの弱点は往々にして、その構造の柔軟性が不足している点にある。FPGA と計算セルを効果的に組み合わせることにより、Reconfigurable な VLSI アーキテクチャを考えることができる。最近の Reconfigurable Systemあるいは Custom Computing Machine の研究の発達[4]は、この方向性に対して大きな助けとなる。
- (3) CPU コア、DSP コアとの組み合わせ：1980 年代と異なり高性能の CPU や DSP が IP として容易に使える以上、セルを専用のハードウェアにする必然性は少ない。IP を有効に利用することにより柔軟性の乏しさを補うことができる可能性がある。

以上、種はあるが、有効な組み合わせには、かなりの研究を要し、我々にとっての大きな課題である。

3.1.1.5 おわりに

本節ではネオ VLSI アルゴリズム・アーキテクチャについての可能性を論じたが、残念ながらこのような観点は現在の所全く個人的なもので、研究としても立ち上げ段階である。この研究領域は研究プロジェクトとしてはかなり広い領域となる。

- (1) 混載 LSI 技術、IP 技術とリンクする。

- (2)アーキテクチャ技術がもちろん研究領域として考えられる。
- (3)アルゴリズムの研究が必要である。
- (4)アルゴリズム記述に関するソフトウェア的な研究が必要である。
- (5)アーキテクチャレベルに及ぶ高位論理合成技術が必要である。

残念なことに、現在 Flagship 的な構成やキラーアプリケーションが存在しないのが問題であり、すぐに大プロジェクトとしてスタートさせるのは困難であるが、少なくともシステム LSI 時代に向けて、当時の研究を振り返って考えることは必要だと思われる。

参考文献

- [1]H. T. Kung: "Why Systolic Architecture?" IEEE Computer, Vol. 15, No. 1, pp. 37-46, 1982.
- [2]C. L. Seitz: "Concurrent VLSI Architecture," IEEE Trans. On Computers, Vol. C-33, No. 12, pp. 1247-1265, 1984.
- [3]M. Annaratone et al, "The Warp Computer: Architecture, Implementation and Performance," IEEE Trans. On Computers, Vol. C-36, No. 12, pp. 1523-1538, 1987.
- [4]天野: "直接解法マシン: 打倒プログラム格納型計算機への最後の希望," ペタフロップスマシン技術に関する調査研究 II 平成 10 年 3 月.

3.1.2 高性能計算とメタレベル計算

中島 浩 委員

3.1.2.1 はじめに

昨年度、高性能計算の key technology が計算の高精度予測であることを述べ、その中でハードウェア／ソフトウェアが協調した予測のようなメタレベル計算を実現する枠組が存在しないことを指摘した[1]。本節ではこのメタレベル計算を対象に、以下のような議論を行なう。

(1) メタレベル計算とは

高性能計算、特に並列計算に関するメタレベル計算としてどのようなものが存在するかを明らかにし、それらの必要性、重要性について議論する。

(2) メタレベル計算の枠組

メタレベル計算を実現するための枠組として考えられる候補をいくつか取り上げ、それらの利害得失について議論する。

(3) メタレベル・ハードウェア機構

メタレベル計算を支えるハードウェア機構として、どのようなものが考えられるかを議論する。

3.1.2.2 メタレベル計算とは

プログラムは一般に、問題を解くためのアルゴリズムを記述することと、プログラムを実行するプロセッサに対して指令することの、二つの側面を持っている。この後者の指令のほとんどは、前者のアルゴリズムを実現するために本質的に必要な操作、すなわちオブジェクトレベル計算のために行なわれる。しかし中には計算の効率を上げるなどの目的で、正しい解を得るためには論理的には必要ではない操作、すなわちメタレベル計算のための指令も存在する。

たとえばソフトウェアによるキャッシュ・プリフェッチは代表的なメタレベル指令であり、この操作が論理的には不必要であることは明らかである。また並列計算においては、並列実行により計算の効率を上げることがそもそもの目的であるため、メタレベル計算の頻度が逐次計算に比べて必然的に高くなる。その代表例は動的負荷分散であり、計算負荷の測定あるいは推定、それに基づく負荷の割付処理などは、全てメタレベル計算である。

これらのメタレベル計算は、プログラムに陽に書かれていたり、あるいは特定の機械命令を実行する形で行なわれるといった、狭義のものであると考えることができる。これに対して以下のような処理や操作を、広義のメタレベル計算としてとらえることができる。

コンパイラによる最適化、並列化

コンパイラに本質的に求められる処理が、プログラムの論理的な意味を充足するような機械命令列の生成であると考えれば、最適化のような処理は全てメタレベルのものであると考えることができる。すなわち、最適化はプログラムに対する実行前のメタレベル計算であり、並列化も一種の最適化である以上やはりメタレベル計算となる。この最適化や並列化がコンパイラの処理の中に占める割合は非常に高く、また今後も増大の一途をたどるものと予想される。

ハードウェアの高速化機構

キャッシュ、命令レベル並列処理、分岐予測など、高速化のためのハードウェア機構のほとんどは、メタレベルの操作を数多く含んでいる。たとえばキャッシュに関しては、データアレイのアクセスがオブジェクトレベル操作と、またタグアレイのアクセスやその結果に基づくヒット／ミスの判定などがメタレベル操作と、それぞれ考えることができる。昨年度の報告[1]でも述べたように、これらの「付加的ハードウェア機構」に費やされるトランジスタの割合は年々増加している。

また、昨年度の報告[1]で述べたように、将来の高性能計算では高精度の予測に基づく計算の効率化など、ソフトウェア／ハードウェアの協調による高度なメタレベル計算が求められる。すなわち上記の二階層の連携によるメタレベル計算が、今後の key technology となる。

一方、以上に述べた各階層におけるメタレベル計算（処理、操作）とその比率は、図 1 に示すようなものとなっている。なお比率については観念的なものであるが、機械命令レベルの階層でのメタレベル計算の比率が、その上下階層に比べて極めて小さいことが示されている。すなわち、現状の機械命令セットや命令セット・アーキテクチャ(ISA)は、オブジェクトレベル計算に特化されており、メタレベル計算、特にソフトウェア／ハードウェア協調メタレベル計算には不向きな枠組であることが示唆されている。

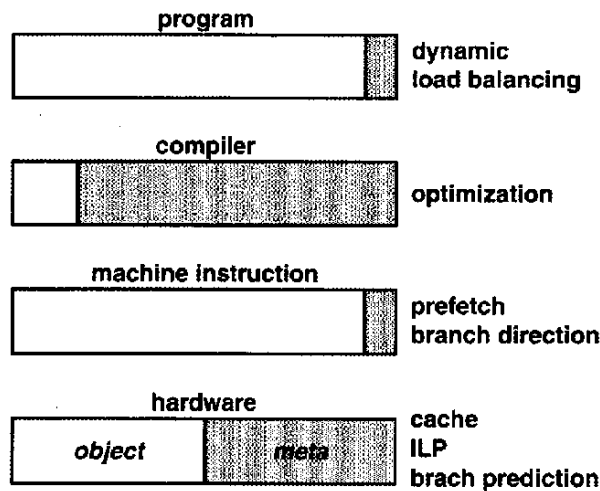


図1 メタレベル計算とその階層

3.1.2.3 メタレベル計算の枠組

[1] メタレベル ISA の現状

前節で述べたように、現状の ISA はメタレベル計算に十分に対応できているとは言い難いが、メタレベルの操作命令が皆無であるわけではない。将来のメタレベル計算の枠組を展望するための基礎として、ここではキャッシュ操作と分岐関連操作に関するメタレベル命令について述べる。

キャッシュ操作 前述のようにキャッシュ・プリフェッチは代表的なメタレベル操作であり、Alpha のように陽にプリフェッチ命令が存在するものや、zero register へのロードがプリフェッチとして機能するものなどが商用レベルで存在する。より細かい操作、たとえばソフトウェア・キャッシュ・コヒーレンスのための無効化などの操作は、実装に依存した特殊な（特権）命令を必要とするのが一般的である。一方、個々のデータをキャッシュすべきか否かを動的に制御することの有効性は既に知られており、たとえば HPPA7200 の assist cache では命令レベルでの選択が可能となっている[3]。

これらの種々のキャッシュ操作を表現するための統一的な枠組は今だ確立されていないが、命令コードに手を加えずに実現する方法の一つとして JUMP-1 のアプローチがある。これは物理メモリアドレスの上位ビットを、キャッシュに関する詳細な制御コードとして利用するものである[4]。したがって特定のアドレスに関して種々の操作が実現できる反面、

利用可能な物理メモリ領域の削減、TLB エントリの消費、アドレス生成の複雑化などの問題点がある。これらの内、アドレス生成以外の問題点は仮想アドレスのビットを利用することにより解決可能であり、64 ビットアドレスの上位が未使用である現状を考えると、一つの解とみなすことができる。

分岐関連操作 分岐アドレスの方向あるいは命令コードに埋め込まれた hint bit により、分岐方向の静的な予測結果を示唆する方法は、いくつかのプロセッサですでに採り入れられている。また Alpha のように、間接分岐アドレスの予測に関するヒントを与えるものもある。

この他、分岐を越えた危険な命令移動をサポートする枠組として、non-excepting 命令やそれを拡張した sentinel scheduling などが知られているが[5、6]、これらは Intel と HP が共同開発中の IA-64 に採り入れられている[7]。

[2] メタレベル ISA の将来

前項で述べた現状のメタレベル ISA は、図2(a)に示すように、コンパイラによるメタレベル計算（計算状況の予測）の結果を、命令の annotation という形でハードウェアのメタレベル操作機構に伝える静的な一方通行の枠組である。

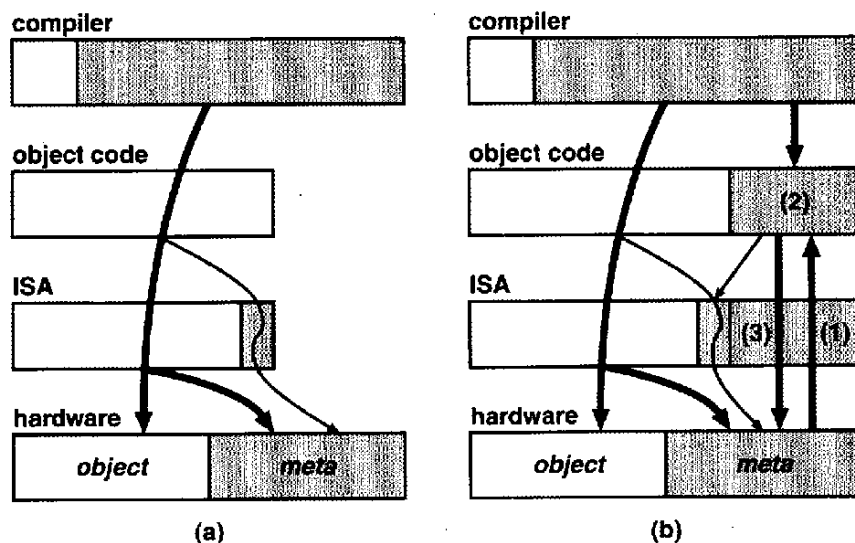


図2 メタレベル ISA の現状と将来

一方、将来の高性能計算においては同図(b)に示すように、動的かつ双方向の枠組が求められる。

たとえば実行時に高精度の予測を行ない、その結果に応じて計算の効率化を図る場合、以下のような枠組が必要である。

- (1) 予測を行なうために必要な統計データなどを、ハードウェアのメタレベル機構から収集するための枠組。
- (2) 予測に必要なメタレベル計算が、オブジェクトレベル計算に与えるオーバーヘッドを最小化するための枠組。
- (3) 予測結果によって動的に変化するメタレベル指令を、ハードウェアに伝えるための枠組。

上記のような枠組を実現する方法として、図3に示すように、以下の三つが考えられる。

- (a) 従来のISAを水平方向に拡張し、(1)や(3)のメタレベル操作を行なうための命令を追加する。(2)については命令レベル並列処理あるいはマルチスレッドなどの、既存のプロセッサ内並列処理技術によって対応する。

ISAの連続性の面ではメリットが大きい、ある程度の一般性と安定性を持った拡張ISAの構築が困難であることが予想される。また簡潔なマシンモデルという、ISAの一つの重要な側面を損なうおそれ強い。

- (b) 従来のISAよりも高いレベルに仮想ISAを、また同等あるいは低いレベルに(1)や(3)を含む実装ISAを設定し、(2)は仮想／実装ISAの中間に位置するインタープリタやダイナミック・コンパイラによって行なう。このアイデアはUniv. of WisconsinのJ. Smithらによるものであり[8]、ある意味ではマイクロプログラムの復権である。実装に依存しがちなメタレベル操作をある意味で隠蔽できるため、仮想ISAの一般性や安定性は確保できる。問題点としては、仮想ISA以下の階層の構築コストの他に、仮想ISA (SmithらはJAVA VMを想定) を介して、上位階層のコンパイラがメタレベル情報をうまく伝達できるかどうかが挙げられる。

- (c) メタレベル計算に専念するコプロセッサによって(1)～(3)の(ほとんど)全てを行なう。コプロセッサは単純なRISCで十分であり、メインプロセッサとの緊密な結合によってオブジェクトレベル計算への適切な介入を実現する。

(a)と同様にISAの連続性は保たれ、また(b)と同様にメインプロセッサISAのマシンモデルとしての簡潔性が保たれる。問題点としては、生成コードの二重化に伴うコンパイラの負荷増大が最大のものであろう。

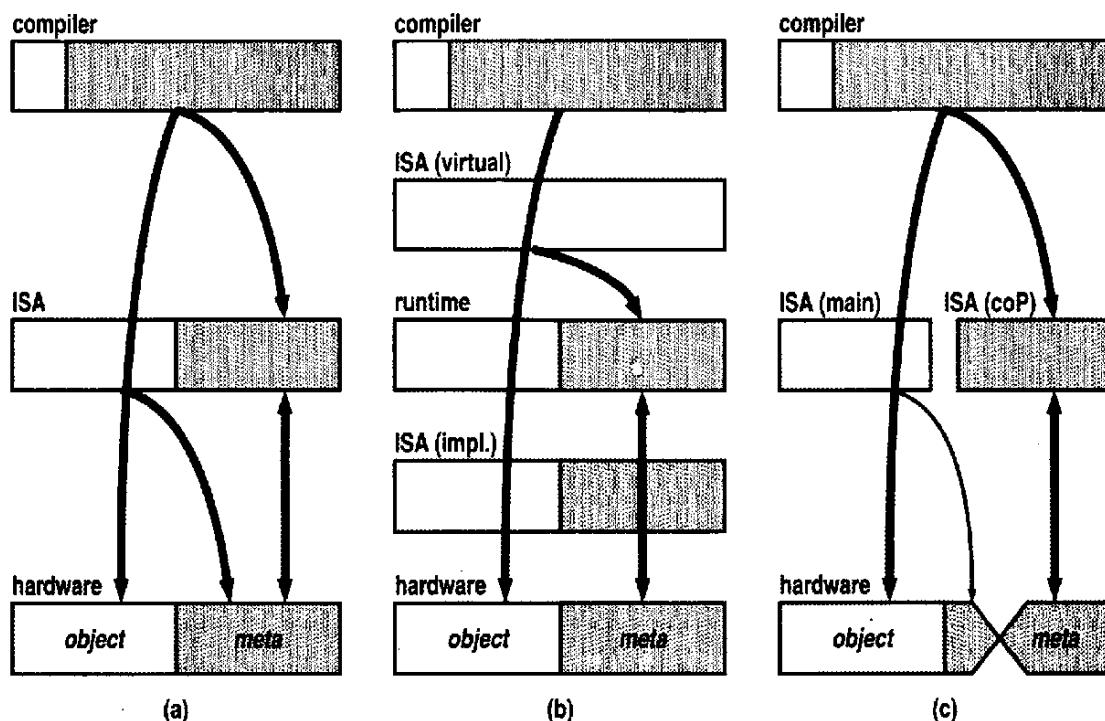


図3 メタレベル ISA の将来像

3.1.2.4 メタレベル・ハードウェア機構

ソフトウェア／ハードウェア協調型の高度なメタレベル計算を行なうためには、3.1.2.2 節で述べたような既存のメタレベル機構に加えて、新たなハードウェア機構が必要となる。しかし分岐予測投機機構のように複雑なハードウェアを新たに導入することは、トランジスタと設計の両面から cost effective ではない可能性が高い。したがってソフトウェアとの協調を第一義的に考え、以下の3点に集約することが得策であると考える。

統計情報収集機構 ソフトウェアによって処理すると最もオーバーヘッドが大きそうなものが、プログラムの挙動を動的に把握するための統計情報の収集である。一方ハードウェアにとっては、あらかじめ定義されたイベント集合に対して、個々の要素の生起を検出して計数することは比較的容易である。また統計情報であるという性質から、その計測結果は完璧なものである必要はなく、時間的あるいは計数そのものの誤差も一定範囲で許容される。

計算状態保存機構 予測と投機のように、オブジェクトレベルの計算状態に一定の影響をもたらすような方式は、メタレベル計算による効率化の重要なターゲットとなる。このために必要となる計算状態の保存は、やはりソフトウェアが不得手とする操作であり、ハードウェアによるサポートの効果は極めて高い。なお昨年度の報告でも述べたように、柔軟性や一般性の確保のために時間コストの最小化が困難である場合には、時間的成功コスト（すなわち保険コスト）を最小化する方向でチューニングすべきである。

一般的計算機構 単純なプロセッサは比較的低コストで設計／実装が可能であり、ソフトウェアとの協調動作という点でも適している。前節で述べたコプロセッサ型のメタレベルISAではもちろん、他のタイプのISAにおいてもメタレベル計算を部分的に担当するようなコプロセッサを導入し、ソフトウェア／ハードウェアの両面で設計コストの大幅な節約を図るのが得策である。

参考文献

- [1]中島 浩. 高精度予測と高性能計算. ペタフロップスマシン技術に関する調査研究 II, pp.49-57, JIPDEC/AITEC, March 1998.
- [2]中島 浩. 投機に投資しよう. 情報処理, Vol.40, No.2, pp.195-201, February 1999.
- [3]K. K. Chan, et al., Design of the HP PA7200 CPU, In *HP Journal*, Vol.47, No.1, February 1996.
- [4]S. Tomita, et al. Hardware Architecture, In *The Massively Parallel Processing System JUMP-1*, pp.169-224, OHMSHA Ltd., February 1996.
- [5]R.P.Colwel, et al. A VLIW Architecture for a Trace Scheduling Compiler, In *Proc. ASPROS'87*, pp.180-192, April 1987.
- [6]S.A.Mahlke, et al. Sentinel Scheduling for VLIW and Superscaler Processors, In *Proc. ASPROS'92*, pp.238-247, October 1992.
- [7]R.E.Curry. IA-64 Architecture Delivers New Levels of Performance and Scalability, In *Platform Solution*, Intel Corp.,
<http://developer.intel.com/solutions/archive/issue6/stories/ia64.htm>
- [8]J.E.Smith and S.Sastry, Achieving High Performance via Co-Designed Virtual Machines, In *Proc. IWIA'98*, October 1998.

3.1.3 情報・エレクトロニクス産業振興策および アーキテクチャ技術の貢献についての私見

濱中直樹 委員

3.1.3.1 情報・エレクトロニクス産業の現状

1970年代から1980年代にかけて我が国の情報・エレクトロニクス産業は、主として米国の情報・エレクトロニクス産業をキャッチアップしようと猛追を試みた。その結果、世界トップクラスの情報・エレクトロニクス産業が確立した。この過程においてハードウェアからソフトウェアに至るまでの全てを一社でカバーする「総合電算機メーカ」が誕生した。

この猛追が成功した背景には色々な要素があろう。ただ、猛追の方向を示した通産省の指導や、それを具現するための高い目標を掲げた国家プロジェクトが大きな役割を果たしたことは周知の通りである。

1990年代には、まず米国にて高性能マイクロプロセッサ技術とインターネット技術に立脚する新規産業が急速に進展し、世界規模で急速に普及しつつある。これにより電子計算機に対する世界観は激変、これに伴い情報・エレクトロニクス産業の供給する商品構成も大きく影響を受けた。また、ドッグイヤーと称する極めて短ピッチの商品サイクルが定着した。

この過程においては、米国を中心とする専業メーカが、なかでもベンチャ企業が果たした役割が大きいことは周知の通りである。なぜベンチャ企業が躍進したかについての詳細な分析は諸処で語られているので省略する。ひとことで言うなら、(1)過去の経緯に縛られない自由な発想、失敗してもやり直せる社会環境、損失をベンチャキャピタルが負う仕組み等によって、アイデアの芽を軽快なフットワークで育てられる雰囲気があること、(2)新規産業であるがゆえに成功すれば莫大な利益を享受できるため所謂アメリカンスピリットに合致すること等が考えられよう。

現在までの延長線で推移するならば、今後の情報・エレクトロニクス産業は次のようになるであろう。

- (1) マイクロエレクトロニクス分野での巨額の先行投資を緩和するための企業間アライアンスが活発になる。先行投資に耐えられない企業はこの分野からドロップする。
- (2) 先進技術に即時追従するための企業間アライアンスやベンチャ等専業メーカの買収が活発になる。先進技術に即時追従できない企業は利益を上げられずドロップする。
- (3) 企業間アライアンスにおいて有利な技術、すなわち利益に直結するような技術の開発

がますます活発になる。これを開発できない企業は企業間アライアンスに参画できず、最終的にはドロップする。

3.1.3.2 デバイスから応用ソフト・コンテンツまでを束ねるフラグシップの意義

上記のように推移していくと、今後次の様な問題が発生すると考えられる。

(1) 一部ユーザのためだけの情報革命

情報化投資によって恩恵を受けたユーザが再度情報化投資を行なうという循環が発生する。メーカ側は利益を生まねば生き残れない境遇にあるので、声の大きいユーザの意向を優先せざるを得ない。そのため、情報技術の恩恵はビッグユーザに偏る。

(2) 先行技術開発の弱体化

アライアンスや企業買収によって「先進技術は購入により調達」という風潮が高まると、技術やノウハウの蓄積が困難になる。そのため先行技術開発も困難になる。また、商品サイクルの短ピッチ化により、次機種の完成は5年後ではなく1、2年後になる。それゆえ「将来」に対するイメージは現在に向かって引き寄せられ、長期的視野に立った組織的な基礎技術開発は困難になる。そのため、順次必要になる将来技術の開発において、未着手のまま放置される大きな空白地帯が発生する可能性がある。

上記のような問題点があるため、「情報インフラ」は水道・電力・電話のようなレベルの「社会インフラ」に到達しているとは言い難い。また、技術開発自体が利益中心で推移している現状では、「社会インフラ」に向かっていくための大きな方向性が徹底されているとも言い難い。

情報技術が人々の暮らしを豊かにすることは明らかであろう。それゆえ、この状況を打破し、「情報インフラ」を「社会インフラ」のレベルに進化させることを通じ、万人が情報技術の恩恵を十分に享受できるようにする必要がある。そのためには、利益中心では推進しにくい分野に投資や政策面で国家が支援するのが効果的である。支援によって得られる成果は、国内のみならず海外にも適用できるようにすべきことは言うまでもない。しかし、このような支援を散発的にせず、大きな効果をもたらすためには、方向性の提示、すなわち、フラグシップが必用になる。

このフラグシップは、「社会インフラ」の整備に関連するので、遠大な計画と多額の投資を伴うことになる。そのため、ここで必用なフラグシップは、国民的理解を得やすく、

かつ、短期間で陳腐化しにくいことが必須である。1960年代、米国では「人類を月へ」というキーワードのもとに技術開発が行なわれ大きな成果があったのは周知の通りであるが、これに匹敵するレベルが必用になると思う。例えば「情報の下の平等」のような。

3.1.3.3 フラグシップの下での産業振興ならびに新規産業育成

上記のようにして設定したフラグシップを効果的に活用して産業振興ならびに新規産業育成に結び付けるためには、以下が必用になるだろう。

(1) 政策の明示

フラグシップの明示ならびに周知徹底と、長期にわたる推進計画、中でも投資計画を明示し、世間が短期的な利益中心のみで行動しないようにする必要がある。

(2) 制度支援

フラグシップに向かって行動していく企業が、現代の厳しい競争の中でフェードアウトしていかないようにするためには、現在行なわれつつある産業界再編や国際的アライアンスを推進するような制度面での支援が必用になる。そのためには、例えば、独占禁止や知的所有権に関する法令ならびに輸出入や公害防止等の規制を諸外国との協議のもとに共通化し、企業が世界を舞台にして迅速に行動できるようにする必要がある。また、企業が必用な人材を速やかに集められるようにするために、労働市場確立を推進する必要がある。労働市場確立を国家が直接遂行することは困難であろうが、企業間での人材流動の抑制要因のひとつと考えられる年功主義や退職金制度緩和のため、雇用契約に成果主義を選択できるよう義務づけるなど、間接的には工夫の余地があると思う。

また、新規産業の育成にはベンチャ支援が必須である。そのために、ベンチャ自体あるいはベンチャキャピタルに対する税制支援等が考えられる。

(3) 計画的な要素技術先行開発の推進

フラグシップを完遂するためには、それに必用な全ての技術を首尾良く開発していかなければならない。必用な技術の中には、それ自身が大きな利益に結びつかない技術、産業界から見ると遠い将来に見えるが学術界の興味を引きにくい技術など、地味ではあるが地道に推進しなければならない技術も含まれるだろう。これら技術に対してのプロモート活動を通じ、先行開発に空白地帯が生じないようにする必要がある。

3.1.3.4 技術開発の推進イメージ

以下では、フラグシップの下での技術開発に論点を絞る。フラグシップとしては先に例示した「情報の下での平等」を仮定する。

上記フラグシップを仮定すると、その恩恵を受けるユーザとしては、特に情報・エレクトロニクスに明るい訳ではない一般的な人々を想定する必要がある。このようなユーザの特性としては、(1)多量の情報受信と少量の情報発信、(2)身近な情報・エレクトロニクス機器には家電的な操作性を要求、という事項を想定する必要がある。また、情報の洪水に溺れぬようにして真の恩恵を受けるためには、受信する情報の種類として大多数の平均的な人々相手の情報ではなく、各個人に最適化された情報を得る必要がある。すなわち、(3)マスメディアからパーソナルメディアへの移行、も想定すべきである。

陳腐な言い方ではあるが、これらの要件を標語的に表現するならば、「世間に存在する多量の情報の中から、自分の欲しい情報を、いつでも、どこでも、だれでも、直ちに利用可能」ということが目標になるであろう。この目標を分解して技術と対応付けると次表のようになる。フラグシップ完遂のためには、表に掲げた各技術の将来を見据えて用意周到に推進していく必要がある。

表1 目標と技術の関係

目標	対応する情報・エレクトロニクス技術
世間に存在する多量の情報	大規模データベース、分散データベース
の中から、自分の欲しい情報を	データ検索、データマイニング
いつでも、どこでも利用可能	モバイル端末、無線/有線 NW コンピューティング
だれでも利用可能	高度ヒューマンインタフェース
直ちに利用可能	高性能 DB/DM サーバ

3.1.3.5 アーキテクチャ技術の貢献

以下では更に論点を絞り、フラグシップ完遂のために必用な技術の中で、アーキテクチャ技術が貢献すべき項目と、貢献にあたって強く意識しなければならないと考える点を述べる。

まず、アーキテクチャ技術という用語をどのような意味で使うかについて述べる。

近年、アーキテクチャという用語は多様な意味で使われるようになった。この中で、本稿での主張の位置付けを明確にするために、以下では「ソフトウェアから見たハードウェアの仕様」、すなわち「ソフトウェアとハードウェアの界面」に関連するハードウェア技術をアーキテクチャ技術と呼ぶことにする。

フラグシップを完遂するために必用な技術として前節に提示した情報・エレクトロニクス技術の中で、上記定義に基づくアーキテクチャ技術が貢献すべき分野と具体的な主要技術は表2の通りであると考ええる。

表2 アーキテクチャ技術の貢献分野と主要技術

アーキテクチャ技術の貢献分野	主要技術
高性能データベースサーバ	高性能マイクロプロセッサ
	高速半導体メモリ利用メモリサブシステム
	マルチプロセッサ構築用高性能チップセット
モバイル端末	低電力マイクロプロセッサ
	低電力システム LSI

表2に示した主要技術は、現在の技術分野としてはそれぞれ独立した専門領域と見なされがちであるが、LSI 技術をどのように利用するか、という点では共通項が存在する。

今日のアーキテクチャ技術を推進するにあたっては、いかなるアーキテクチャ技術であっても、LSI 技術と密接な協調関係無くしては成立しない時代になっている。そして、LSI 技術の進化に伴ない、これまでは LSI 技術の中での課題がアーキテクチャ技術にも見えるようになった。

(1) 高速化に伴う課題

これまでは LSI 技術のトレンドに沿って高速化されとの考えで多くのアーキテクチャ技術は進展してきた。確かにゲート遅延は順次低減しているが、LSI の微細化に伴ってゲート遅延よりもむしろ配線遅延が支配的になってきている。また、LSI の内部周波数の著しい向上と比べ LSI 間の信号伝送周波数の向上は穏やかである。そのため、LSI 内部・外部を含め、実現するための物理構造を強く意識してアーキテクチャ設計を行わないと実用に耐えるアーキテクチャとは言えない時代が来ている。このような傾向は従来からも超高性能計算機の分野には存在したが、近年この傾向がますます強まっている。

(2) 微細化に伴う課題

微細化によって LSI に搭載可能な回路数は増大している。また、高速化に伴い回路あたりの消費電力は増大する傾向にある。そのため、LSI あたりの消費電力は増加の一途にある。特に、高性能向き LSI ではこの傾向は顕著であり、いずれ空冷の限界に到達すると考えられる。そのため、冷却に関連する物理構造を意識する、あるいは、アーキテクチャ的な技法による回路動作のオン/オフを意識するような要求が高まる。

また、微細化に伴って回路各部の挙動を巨視的な電流としてモデル化することは次第に困難になり、個々の電子の量子力学的な振る舞いを考慮すべき時代が迫っているとも聞く。このこともアーキテクチャ技術に大きな衝撃を与えると考えられる。

上記のような課題に伴い、アーキテクチャ技術の発展を考慮するにあたっての LSI 高度活用がますます大きな位置を占めるようになる。

以前は設計効率追求のために設計の階層化が重要視された。その中で、アーキテクチャ技術を考える上では、ハードウェアはデジタル回路として把握すれば済む時代が続いてきた。しかし、これからのハードウェア、特に、高性能あるいは低消費電力というような極限を求める分野においては、ハードウェアをアナログ回路的なセンスで見ることが生じている。つまり、従来の様な設計階層に固執してはいられない時代が来ている。

そのために必用なことは、LSI の特性を活かしたアーキテクチャ設計である。具体的には、ソフトウェアのセンスで応用を分析し、電子回路あるいは実装構造のセンスでインプリメントをイメージするような手法が必用になる。しかしながら、LSI 技術の細部までアーキテクチャ技術者が立ち入ることは困難である。そのため、アーキテクチャ技術と LSI 技術との間での Give and Take の関係作りが重要になる。

アーキテクチャ技術から LSI 技術への Give の第一歩としては、フラグシップ完遂のためにアーキテクチャ技術が推進すべき項目を明確にした上で、これを LSI 技術の言葉に翻訳して LSI 技術の目標を明確にすることが挙げられる。すなわち、アーキテクチャ技術からの必然的な要求を LSI 技術に対してロードマップとして提示することである。そのロードマップには、必要になるチップサイズ、ピン数、ゲート遅延、配線層数などが表現手段として登場するだろう。

このような表現手段は必ずしも新しい表現方法ではなく、これまでしばしば使われてきた。しかし、今後は、これまで以上に要求の意図を明確にしてこれらの表現を使用する必要がある。なぜなら、先述の通り、LSI と応用の関係が密接になってきているからであり、それが故に、一般には非常に大きな投資が必要になる LSI の技術開発の方向性決定において、応用技術からの意見の影響度が高まってくるからである。そして、応用技術および LSI

技術双方の事情を勘案して適切なソリューションを提供するのがアーキテクチャ技術の貢献であろう。

以上、アーキテクチャ技術がフラグシップ完遂に向けて貢献すべき方法について概観した。最後に、上述のような貢献を遂行できる人材について考えてみる。

同じ電子計算機技術と言えども、応用技術から LSI 技術に至るまで研究開発分野が細かく分業化されてしまった今日、上記のような役割を遂行できる人材は極めて限られた存在になってきている。そして、そのような人材は、組織的に育成されているのではなく、当事者の興味と能力によって言わば自然発生的に育っているに過ぎない。

今後アーキテクチャ技術が世の中に貢献していくためには、このような人材を多量に育成する必要がある。このような人材の育成には、高い目標を掲げある程度の期間継続して取り組めるような環境を設定する必要がある。しかし、今日の情勢の中では、一般企業の中ではこのような環境を設定することは難しい。そのため、このような環境を設定し、安定して維持して行くことが、フラグシップ完遂のため国家に期待されるもうひとつの大きな役割なのであろう。

3.2 ソフトウェア

3.2.1 Super Simulation Infrastructure

石川 裕 委員

3.2.1.1 はじめに

今後の産業および科学技術を支えるためのインフラとして、また、新しい市場の創成のために、その原動力となる環境整備が重要である。その一つが計算機によるシミュレーション技術であると考ええる。

産業および科学技術では計算機によるシミュレーション技術が重要である。例えば電子回路のシミュレーション等シミュレーション時間の短縮は製品開発の短縮につながり競争力を高めるものとなる。自動車の車体設計、エンジン開発でもシミュレーション技術は欠かせない。新薬の開発でも計算機による分子動力学シミュレーションによるたんぱく質構造予測技術の期待が高まっている。

さらに物理現象のシミュレーション技術は科学教育の教材としても利用できる。実世界をモデル化して実時間アニメーションも可能となるだろう。実時間アニメーションが可能になれば、静的な文字や絵による伝達手段からアニメーションによる伝達手段へのパラダイムシフトが起こると考える。そして、ドキュメント、マニュアル、教科書がインタラクティブ・アニメーションで実現されると予想される。これにより、現在の並列処理市場が拡大されると考える。

このような世界に向けて技術開発を牽引するには、まず、従来限られた研究機関でしか設置できなかったテラフロップス級計算機を一企業でも保持できるワークステーションサーバ程度のコンパクトさと価格にして、ハイエンド並列計算機の利用者層の拡大をはかる必要がある。すなわち、テラフロップス級ワークステーションからテラフロップス級パーソナルコンピュータを実現する計算機ハードウェアの開発である。

コンパクトかつ安価な計算機開発は、国策として何もしなくても推進されていくだろうか？既に市場が出来あがっているテレビゲームの世界では、ハイエンド計算機並あるいはそれ以上の性能を持つかも知れないプロセッサの開発が進んでいる。しかし、それは、市

場に最適化される方向での開発でしかない。ここで想定している科学技術計算に必要な能力を持った計算機開発は、市場主義に基づいた技術開発では不可能であろう。なぜならば、市場が膨らむのに必要なソフトウェアが不足しているからである。

国による開発により開発コストを低減してコンパクトかつ安価な計算機を実現すると共に、システムソフトウェア、ライブラリ、応用プログラムの充実が必要である。本稿では、このような観点からプログラミング言語処理系、オペレーティングシステムについて考察する。

3.2.1.2 並列プログラミング言語処理系

今後のハイエンド並列計算機は、複数の SMP を高速ネットワークで接続された SMP クラスタになるだろう。ハードウェア的には、SMP 内でのマルチスレッド共有メモリ実行モデルと SMP 間での通信あるいは分散共有メモリという 2 タイプの実行モデルが混在する形になる。このような異種性のあるハードウェアでは、並列アプリケーションのタスクおよびデータの分散を工夫しないと並列性能を向上させることは出来ない。

最近では、HPF のようにプログラマに対して分散方法を指示させる言語処理系が制定されている。しかし、このような言語は次の 2 点の問題がある。

- (1) 並列計算機の仕組みを熟知していないアプリケーションプログラマに対してデータ分散の仕方を教育する困難さ。
- (2) 全てのデータの分散方法を決めない限り、並列化できないという障壁。

(2) は別の言い方をすれば、逐次プログラムを並列化するときには少しずつ並列化していくということが出来ず、逐次から並列への敷居を高くしている。

2 年前から SMP 向けに openMP というコンパイラ指示文の規格化が行われ、現在、製品が出荷され出している。SMP を想定しているのでデータの分散方法を指定する必要はない。openMP では、どのループが並列化できるのか等の指示をコンパイラに与える。従って、逐次プログラムを徐々に並列化していくことが可能となる。openMP の持つ徐々に並列化するという特徴を生かしつつ、SMP クラスタ上で利用できる処理系の開発が重要となる。

プログラムの記述性保守性の向上を目的に C++ プログラミング言語等、オブジェクト指

向言語が利用されてきている。C++プログラミング言語は、言語が持つ数式記述能力と効率良い処理系の出現により、数学者、物理学者の間でも使われるポピュラーな言語になってきている。最近では、C++の持つオブジェクト指向プログラミング支援機能を生かした並列アプリケーション記述のための研究が盛んである。

例えば、米国 Los Alamos 国立研究所で進められている POOMA プロジェクトでは、POOMA は流体力学やプラズマシミュレーションのような計算科学を解くためのフレームワークを構築している。POOMA はローカルデータ、並列抽象化、グローバルデータ、コンポーネント、アプリケーションの5つのレイヤを提供している。並列計算機に固有のコードはローカルデータと並列抽象化レイヤによって実現されている。グローバルデータレイヤは、計算機独立の大域的なデータアクセスを提供している。コンポーネントレイヤはアルゴリズムの集合であり、FFT とか共役勾配法等が提供されている。

グローバルデータレイヤの例として Field クラスおよび Index クラスの使用例を以下に示す。Field クラステンプレートはグローバルデータを表現するためのものであり、FieldLayout クラスは仮想プロセッサ間の Field の分散を管理する。Index クラスは、Field クラスのデータを参照するためのものである。

```
1: Index          I(100), J(100);
2: FieldLayout<2>    layout(I, J);
3: Field<double, 2>  A(layout), B(layout);
4: assign(A[I][J], 0.25*( B[I][J] + B[I+1][J]
                        + B[I][J+1] + B[I-1][J]
                        + B[I][J-1] ));
```

上記プログラムにおける4行目のように、assign 関数テンプレートを使うことにより、ループ制御構造を記述せずに緩和プログラムが記述でき、アルゴリズムの可読性が増す。assign 関数テンプレートの実現は、Template Closure (あるいは Expression Template と呼ばれている) 技法を用いることにより、効率よいコードが生成される。

このように POOMA を使うと、アプリケーションを記述するユーザに対して並列計算機の構造を意識したマルチスレッドプログラミングやメッセージパッシングプログラミングを記述せずにアルゴリズムの記述に専念することが可能となる。今後の並列処理ユーザ層

の拡大のためには、このような科学技術計算のためのライブラリだけでなく、広く使われるシミュレーション技術を体系化して再利用可能な並列ライブラリを構築して、応用プログラムの充実を図る必要があるだろう。

最近、Java を高性能計算分野で利用できるように Java Grande という組織が結成されている (<http://www.javagrande.org/>)。Java Grande では、高性能計算のための Java で記述されたライブラリの規格化や Java で書いた高性能計算を必要とするプログラムが効率良く最適化出来るように拡張を提案していく。また、C++ のようなテンプレート機能は、並列処理を抽象化するのに非常に良い道具となっている。C++ において蓄積されてきている技術を Java に適応していこうという動きもある。

3.2.1.3 スケーラブルオペレーティングシステム

SMP クラスタ上で並列アプリケーションが効率良く実行できるオペレーティングシステムが必要である。1 テラフロップス級 SMP クラスタの構成要素である 1 ノードに 2 ギガフロップスのプロセッサを 4 個搭載していると仮定すると 128 ノード必要となる。128 ノードでローカルなオペレーティングシステムが稼働し、全てのノード間で資源の大域管理を行い、高速通信機能や分散共有メモリの支援、ジョブスケジューリング、ファイルシステム、並列 I/O、メンテナンス、管理ツールを実現しなければならない。

実現方法には、Linux 等の既存のオペレーティングシステムカーネルを利用し改造していく方法と新規開発していく方法がある。本格的 SMP クラスタ構築と今後の基盤技術の整備という観点からは、SMP クラスタのためのオペレーティングシステムを開発していくのが良い。なぜならば、メモリ管理機構に踏み込んだカーネルの修正をする場合に大幅な修正が必要となり、元にしたカーネルが改良される毎にそれを反映させなければならない、コストがかかるからである。

新規オペレーティングシステムカーネルの開発は危険性が伴うが、それを軽減することは可能である。UNIX の API を採用することにより、ほとんどのシステムソフトウェアを新規に開発しなくても GNU や Linux のコミュニティが開発したものが使える。開発したカーネルをオープンにしてユーザ層だけでなく開発や支援する層を広げることが出来る。これは、GNU や Linux が行ってきた手法である。

3.2.1.4 おわりに

本稿では、今後の産業および科学技術を支えるためのインフラとして、そして、新しい市場の創成には、(1)従来限られた研究機関でしか設置できなかったテラフロップス級計算機を一企業でも保持できるワークステーションサーバ程度のコンパクトさと価格にして、(2)産業界で使われるシミュレーションやインタラクティブ・アニメーションを可能とするソフトウェアを実現することにより、ハイエンド並列計算機市場を作り出せると考え、そのために必要な技術研究開発の中からプログラミング言語処理系、オペレーティングシステムについて考察した。

3.2.2 自動並列化コンパイラとアーキテクチャ支援

笠原博徳 委員

3.2.2.1 はじめに

現在日本のスーパーコンピュータメカは世界トップのハードウェア技術を有し、現時点でのピーク性能は 1TFLOPS を越え、21 世紀初頭には数十 TFLOPS 以上のピーク性能をもつマシンが開発されると予想される。しかし、現在のスーパーコンピュータは、ピーク性能の向上と共にプログラムを実行したときの実効性能との差が大きくなっている、すなわち価格性能比が必ずしも優れていると言えない状況になっている。また、使い勝手的にも、ユーザは問題中の並列性を抽出し、HPF、MPI、PVM などの拡張言語あるいはライブラリを用いハードウェアを効果的に使用できるようなプログラムを作成しなければならず、一般のユーザには使い方が難しいあるいは使いこなせないという問題が生じている。さらに、これらにも起因して、世界の HPC 市場を拡大できないということが産業界にとって大きな問題となっている。

この価格性能比、使いやすさの問題を解決し、スーパーコンピュータのマーケットを拡大するためには、ユーザが使い慣れている Fortran 等の逐次型言語で書かれたプログラムを自動的に並列化する自動並列化コンパイラ[4-10、18]の開発が重要となる。

一方、汎用マイクロプロセッサの分野では、我が国の産業界が世界をリードしているとは言いがたい状況にあることが認識されている。しかし、従来我が国の電気産業を支える一つの柱となっていた DRAM による利益確保が将来的に困難と予想される状況では、より付加価値の高い汎用マイクロプロセッサの開発を検討することが必要である。

その際、海外企業が大きなシェアをもち、さらに命令レベルの並列性[1-3]の限界から将来的な実効性能の向上が難しいと予測されるスーパースカラや VLIW ではなく、21 世紀初頭の有力なアーキテクチャの一つとなると考えられるシングルチップ・マルチプロセッサ[2、24]について検討を行うことは、産業界がこの分野で一定のシェアを獲得するために重要と考えられる。また、このようなシングルチップ・マルチプロセッサに関する検討は、上述の HPC の価格性能比向上に対しても重要な役割を果たすものと思われる。

ただ、このようなシングルチップ・マルチプロセッサの研究開発を行う際にも、単に従来の主記憶共有アーキテクチャでプロセッサを集積しただけでは、近隣諸国の技術レベルの向上も著しい現在、世界に対する国産プロセッサの優位性は得られない。21 世紀に向け十分な利益を上げることを可能とするためにはアーキテクチャの独自性、高性能化、低価格の達成が必須である。

このためには、プログラム中の命令レベル並列性、ループ並列性、粗粒度並列性をフルに使用できるマルチグレイン並列処理[4]のように、真に実行すべき命令列からより多くの並列性を抽出し、システムの高価格性能比の達成すると共に、誰にでも使えるユーザフレンドリなシステムの構築を可能とする新しい自動並列化コンパイル技術と、それを生かせるようなアーキテクチャの開発が重要であると思われる。

本報告書ではこのマルチグレイン並列化コンパイラと、それをサポートするマルチプロセッサアーキテクチャについて述べる。

3.2.2.2 マルチグレイン並列化コンパイラ

マルチグレイン並列処理[4、8、25]とは、サブルーティン、ループ、基本ブロック間の粗粒度並列性[16]、ループイタレーション間の中粒度並列性（ループ並列性）[5-9、18、19]、ステートメントあるいは命令間の（近）細粒度並列性[11、14]を階層的に利用する並列処理方式である。この方式により、従来の市販マルチプロセッサシステム用自動並列化コンパイラで用いられていたループ並列化、あるいはスーパースカラ、VLIWにおける命令レベル並列化のような局所的で単一粒度の並列化とは異なり、プログラム全域に渡るグローバルかつ複数粒度によるフレキシブルな並列処理が可能となる。

(1) 粗粒度並列処理

単一プログラム中のサブルーティン、ループ、基本ブロック間の並列性を利用する粗粒度並列処理は、マクロデータフロー処理とも呼ばれ[8、16、17]、OSCAR マルチグレイン Fortran 並列化コンパイラで世界で初めて自動並列化が実現された[4、25]。

OSCAR コンパイラでは、粗粒度タスク（マクロタスク）として、単一の Fortran プログラムより RB (Repetition Block)、SB (Subroutine Block)、BPA(Block of Pseudo Assignment Statements)の3種類のマクロタスクを生成する。RB は各階層での最外側ナチユラルループであり、SB はサブルーティン、BPA はスケジューリングオーバーヘッドあるいは並列性を考慮し融合あるいは分割された基本ブロックである。

次にマクロタスク間の制御フロー[7、8]とデータ依存[5、7、8]を解析し、図1のようなマクロフローグラフ(MFG)を生成する。MFGでは、各ノードがマクロタスク(MT)、点線のエッジが制御フロー、実線のエッジがデータ依存、ノード内の小円が条件分岐文を表している。また、MT7のループ(RB)は、内部で階層的にMT及びMFGを定義できることを示している。

次に、マクロタスク間制御依存[7、8]及びデータ依存より各マクロタスクが最も早く実行できる条件（最早実行可能条件）[4、8、26]すなわちマクロタスク間の並列性を検出す

る。この並列性をグラフ表現したのが図2に示すマクロタスクグラフ(MTG)である。MTGでも、ノードはMT、実線のエッジがデータ依存、ノード内の小円が条件分岐文を表す。ただし点線のエッジは拡張された制御依存を表し、矢印のついたエッジは元のMFGにおける分岐先、実線の円弧はAND関係、点線の円弧はOR関係を表している。例えば、MT6へのエッジは、MT2中の条件分岐がMT4の方向に分岐するか、MT3の実行が終了した時、MT6が最も早く実行が可能になることを示している。

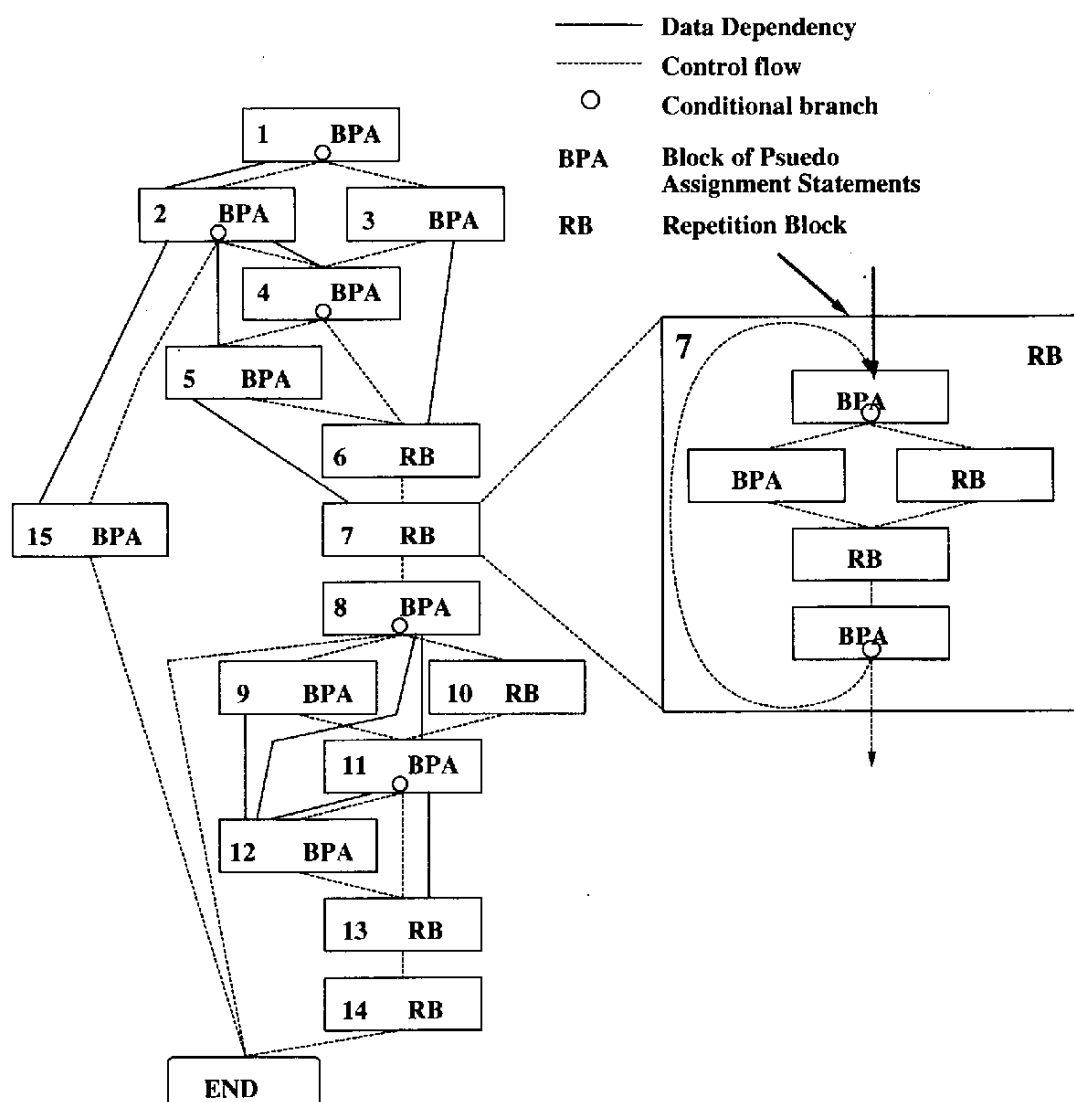


図1 マクロフローグラフ(MFG)

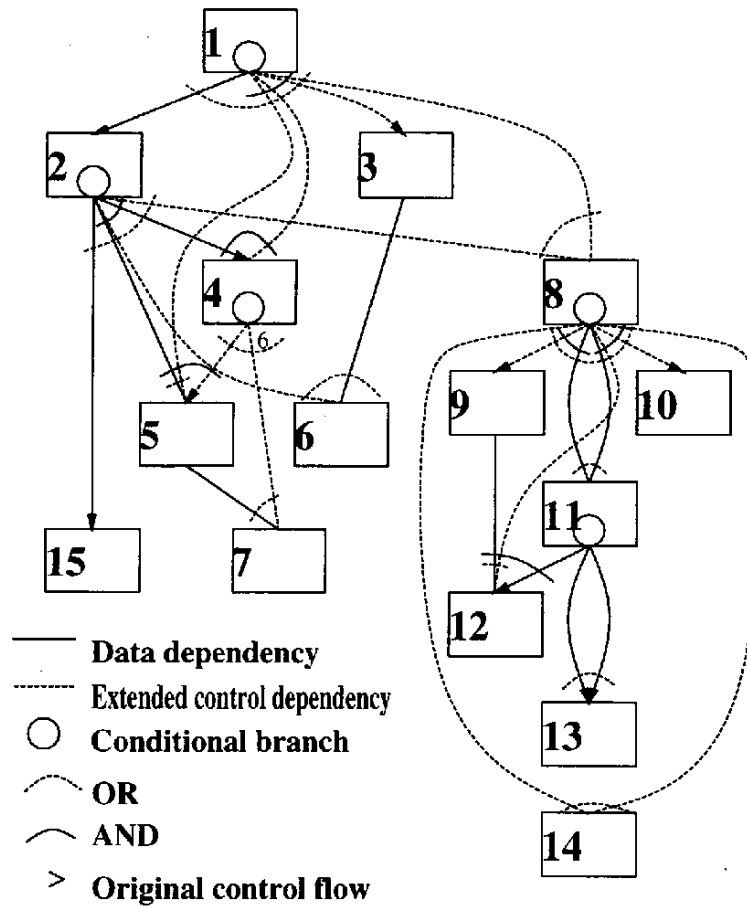


図2 マクロタスクグラフ(MTG)

次にコンパイラは、MTG上のMTをプロセッサクラスタ（プロセッサのグループ）への割当てを行う（スタティックスケジューリング）か、実行時に割当てを行うためのダイナミックスケジューリングコードをDynamic CPアルゴリズムを用いて生成し、これをプログラム中に埋め込む。これは、従来のマルチプロセッサのようにOSあるいはライブラリに粗粒度タスクの生成、スケジューリングを依頼すると数千から数万クロックのオーバーヘッドが生じてしまう可能性があり、これを避けるためである。

また、このスタティックスケジューリング及びダイナミックスケジューリングコードの生成の時には、各プロセッサ上のローカルメモリあるいは分散共有メモリを有効に使用し[19-22]プロセッサ間データ転送量を最小化するためのデータローカライゼーション手法[17、27]も用いられる。

データローカライゼーションは、MITG 上でデータ依存のある複数の異なるループにわたりイタレーション間のデータ依存を解析し（インターループデータ依存解析）、データ転送が最小になるようにループとデータを分割（ループ整合分割）後、それらのループとデータが同一のプロセッサにスケジューリングされるように指定するパーシャルスタティックスケジューリングアルゴリズムを用いてダイナミックスケジューリングコードを生成する。

またこの際、データローカライゼーションによっても除去できなかったプロセッサ間データ転送を、データ転送とマクロタスク処理をオーバーラップして行うことにより、データ転送オーバーヘッドを隠蔽しようとするプレロード・ポストストアスケジューリングアルゴリズムも使用される[15]。

(2) ループ並列処理

マルチグレイン並列化では、マクロデータフロー処理によりプロセッサクラスタ(PC)に割当てられるループ(RB)は、そのRBがDoall、あるいはDoacrossループ)の場合PC内のプロセッサによりイタレーションレベルで並列処理される。

ループ・リストラクチャリングとしては、以下のような従来の技術をそのまま利用できる。

- (a) ステートメントの実行順序の変更
- (b) ループディストリビューション
- (c) ノードスプリッティング、スカラエクスパンション
- (d) ループインターチェンジ
- (e) ループアンローリング
- (f) ストリップマイニング
- (g) アレイプライベタイゼーション
- (h) ユニモジュラー変換（ループリバーサル、パーミュテーション、スキューイング）

また、ループ並列化が適用できないループに関しては、図3の階層的なマクロタスクの定義のように、ループボディ部を後述する（近）細粒度並列処理か、ボディ部を階層的にマクロタスクに分割しマクロデータフロー処理を適用する。

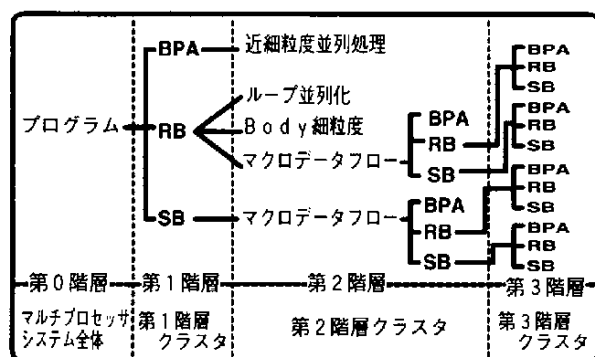


図3 階層的なマクロタスクの定義

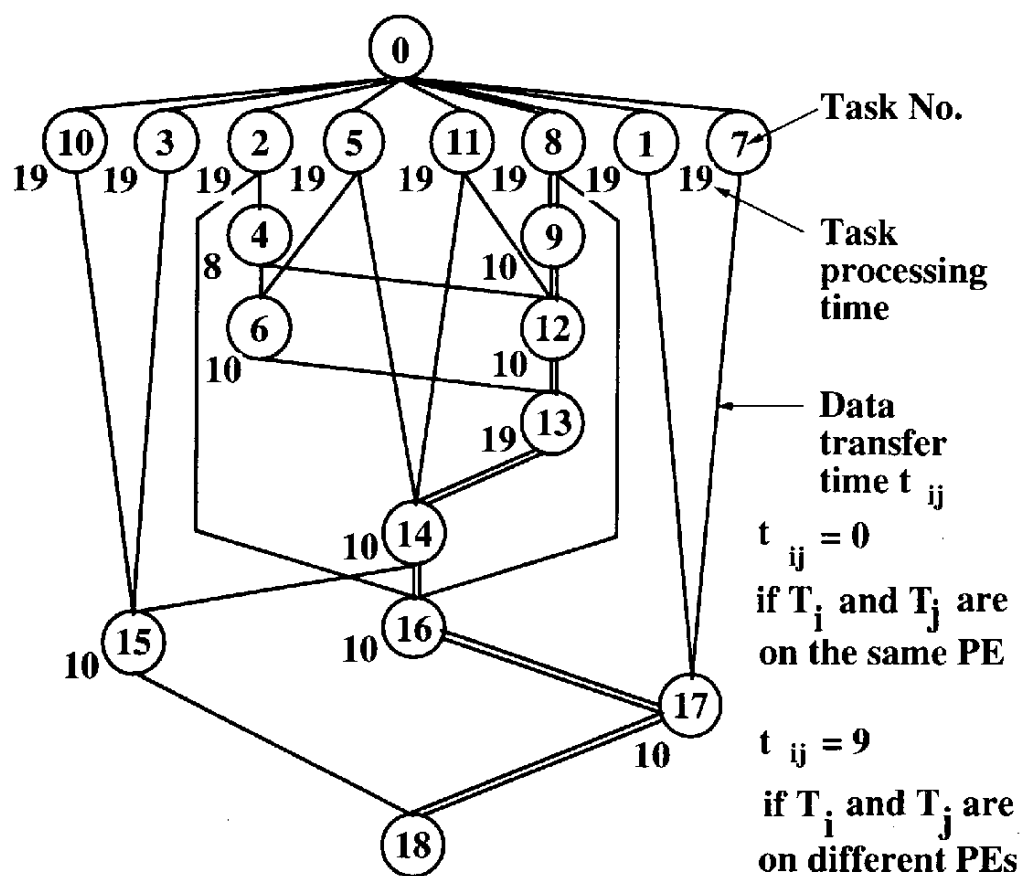


図4 近細粒度タスクグラフ

(3) (近) 細粒度並列処理

PC に割当てられる MT が BPA、またはループ並列化あるいは階層的にマクロデータフロー処理を適用できない RB 等の場合には、BPA 内部のステートメントあるいは命令を (近) 細粒度タスクとして PC 内プロセッサで並列処理する。

マルチプロセッサシステムあるいはシングルチップ・マルチプロセッサ上での近細粒度並列処理[11、14]では、プロセッサ間の負荷バランスだけでなくプロセッサ間データ転送をも最小にするようにタスクをプロセッサにスケジューリングしなければ、効率良い並列処理は実現できない。さらに、この近細粒度並列処理で要求されるスケジューリングでは、図4に示すようにタスク間にはデータ依存による実行順序制約があるため強NP完全な非常に難しいスケジューリング問題[13]となるが、CP/DT/MISF (Critical Path/Data Transfer/Most Immediate Successors First)法等のスタティックヒューリスティック・アルゴリズムの使用により実マルチプロセッサシステム上でも効率良い並列処理が行えることが確かめられている。

次にコンパイラは、図5に示すようにタスクスケジューリング、データ分割・配置決定後、各プロセッサで実行される並列化マシンコードあるいは拡張 Fortran 言語等を生成する。各プロセッサで実行されるべき並列マシンコードは、そのプロセッサに割当てられたタスクコード、プロセッサ間でのデータ転送のためのコード、及び同期のためのコード等から構成される。

このマシンコード生成時には、同一プロセッサに割当てられたタスク間でのデータ授受のためのレジスタ利用の最適化、データ転送オーバーヘッド最小化を目指したプロセッサ間データ転送モードの最適化、同期オーバーヘッドの最小化を目指した冗長な同期コードの削除等が行われる。特にシングルチップ・マルチプロセッサでは、コード生成時に厳密なコード実行スケジューリングを行うことにより、実行時のデータ転送タイミングを含めた全ての命令実行をコンパイラが制御し、全ての同期コードを除去して並列実行を可能とする無同期並列化[11]のような究極的な最適化も行える。

3.2.2.3 アーキテクチャサポート

3.2.2.2 で述べたマルチグレイン並列処理をマルチプロセッサシステム上で実現しようとする時に望まれるアーキテクチャサポートについて述べる。

(1) 集中・分散共有メモリとローカルメモリ

粗粒度並列処理では、条件分岐に対処するためにダイナミックスケジューリングが使用

される。この場合、マクロタスクがどのプロセッサで実行されるかはコンパイル時には分からない。したがって、ダイナミックにスケジューリングされるマクロタスク間の共有データは（集中）共有メモリに配置できることが好ましい。

また、粒度によらずスタティックスケジューリングが適用できる場合には、あるマクロタスクが定義する共有データをどのプロセッサが必要とするかはコンパイル時に分かるため、生産側のプロセッサが消費側のプロセッサ上の分散共有メモリにデータと同期用のフラグを直接書き込めることが好ましい。これにより集中共有メモリを介してデータ転送する場合に比べ同期を含めたオーバーヘッドが1/2以下に軽減される。

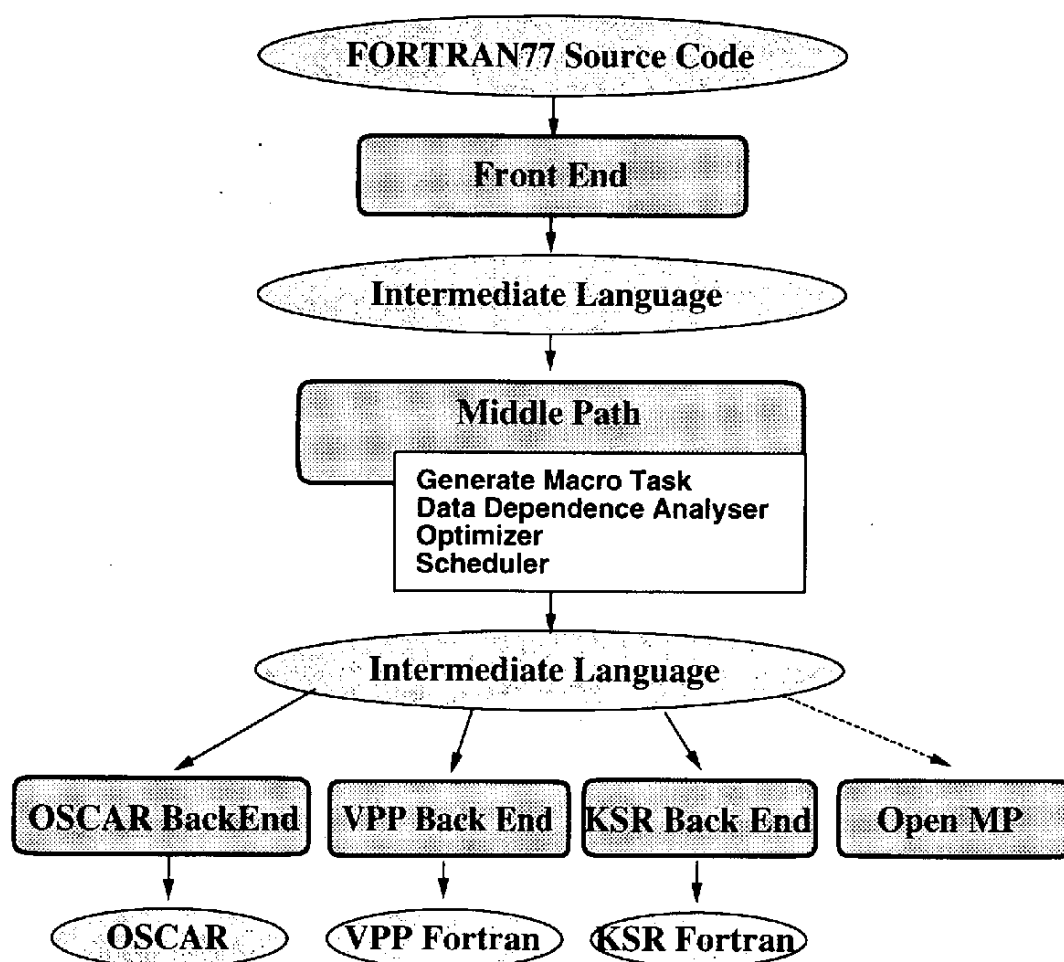


図5 OSCAR Multigrain Compilerの構成

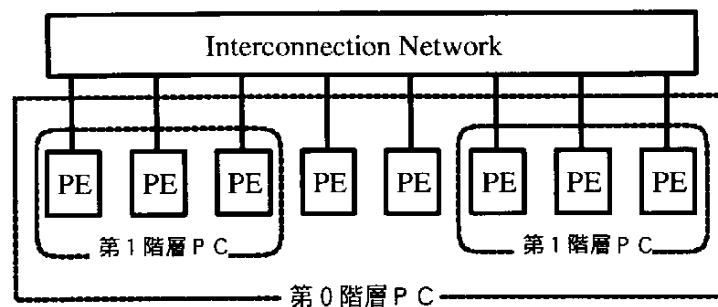


図6 プロセッサクラスタの階層的定義

さらにタスクローカルなデータあるいはデータローカライゼーション手法によりマクロタスク間でローカルにデータの授受が可能な場合には、各プロセッサ上のローカルメモリあるいは分散共有メモリにそれらのデータを割当てることによりデータ転送オーバーヘッドを顕著に削減できる。

また、プログラムコードはデータと比べ小さいことが多く、さらにスタティックスケジューリングを行う場合には各プロセッサ毎に別々のプログラムを生成した法が効率良い並列処理が行えるため、できる限りローカルメモリにおくことが望ましい。

(2) 可変プロセッサクラスタリングと同期

マルチグレイン並列処理では、図3のようなプログラム中の何階層目にどの粒度の並列性がどの程度あるのかによって、階層的なプロセッサクラスタの構成（図6）もソフトウェア的に変えられるクロスバのようなネットワークが望ましい。また、それに応じクラスタ内でバリア同期が高速にとれること（可変バリア）が望ましい。

また、近細粒度並列処理においては分散あるいは集中共有メモリ、共有キャッシュ、共有レジスタ等を用いた細粒度データ転送及びデータ同期を高速にとることを可能とするアーキテクチャサポートが必要である。

さらに、プロセッサをクラスタリングした場合でも各プロセッサから集中共有メモリへ直接アクセスできることが望ましい。

(3) 処理とデータ転送オーバーラッピング用データ転送ユニット

データローカライゼーションあるいはデータ転送を考慮したスケジューリングで除去できなかったプロセッサ間データ転送は、マクロタスクの実行とオーバーラップして行なわ

れる。すなわち、スタティックスケジューリングモードでも、ダイナミックスケジューリングモードでもマクロタスクの実行が始まる前にデータが自プロセッサ上のメモリにロードされていることが好ましい。このためには、コンパイラが制御できる高機能なデータ転送ユニットが必要である。

(4) ソフトウェア制御キャッシュ

マクロデータフロー処理のようなダイナミックスケジューリング環境下での、データローカライゼーション及び高度データ転送ユニットを用いた処理とデータ転送のオーバーラッピングスケジューリング技術の融合は、コンパイラとコンパイラがプログラム情報を埋め込んだダイナミックスケジューリングコードによって、ノーマスヒットのキャッシュの実現にもつながると考えられる。

現在筆者らは、以上のような要求をほぼ満たしているマルチプロセッサ OSCAR をベースに、次世代のスーパーコンピュータアーキテクチャ及びシングルチップ・マルチプロセッサの評価・検討を行っている[23、24]。

3.2.2.4 まとめ

本報告書では、今後シングルチップ・マルチプロセッサからスーパーコンピュータまで適用可能なフレキシブルかつ強力な並列処理手法であるマルチグレイン並列コンパイルーション技術と、そのためのアーキテクチャ・サポートについて述べた。

マルチグレイン並列化は、データローカライゼーション（データ自動分散技術）、データプレロード・ポストストアスケジューリング技術（処理とデータ転送のオーバーラッピング技術）等の技術と共に、マルチプロセッサシステムの実効性能を向上させシステムの価格性能比の向上を可能とするだけでなく、逐次プログラム全域から細粒度から粗粒度に至る全ての並列性を自動的に抽出することを可能とする使い易いマルチプロセッサシステムの構築を可能とする。

このようなマルチグレイン並列化コンパイラとそれを効果的に実現するアーキテクチャの実用化により、ハイパフォーマンスコンピュータの市場を拡大、及びシングルチップ・マルチプロセッサによる汎用マイクロプロセッサ市場への参入を可能とすると考えられる。

特に、現在劣性の汎用マイクロプロセッサ市場への参入し、21世紀に情報産業で世界をリードする技術を得るためには、デバイス技術、アーキテクチャ技術、コンパイラ技術、OS技術等をバーティカル・インテグレーションする新しい形態の国家プロジェクトの推進が望まれる。

参考文献

- [1] D.Burger, J.R.Goodman, "Billion Transistor Architectures," IEEE Computer, Vol.30, No.9, pp.47-49, Sep.,1997.
- [2] L.Hammond, et.al, "A Single-Chip Multiprocessor", IEEE Computer, Vol.30, No.9, pp.79-85, Sep.,1997.
- [3] M.H.Lipasti, J.P.Shen, "Superspeculative Microarchitecture for Beyond AD 2000," IEEE Computer, Vol.30, No.9, Vol.30, No.9, pp59-66, Sep.1997.
- [4] H.Kasahara, et.al., "A Multi-Grain Parallelizing Compilation Scheme for OSCAR (Optimally Scheduled Advanced Multiprocessor)," Proc. Languages and Compilers for Parallel Computing, Lect. Notes Comput Sci., Springer-Verlag, Vol.589, 1992.
- [5] U.Banerjee, Loop Parallelization, Kluwer Academic Pub., 1994.
- [6] D.J.Lilja, "Exploiting the Parallelism Available in loops," IEEE Computer, pp.13-26, Vol.27, No.2, Feb.1994.
- [7] M. Wolfe, High Performance Compilers for Parallel Computing, Addison Wesley, 1996.
- [8] 笠原, 並列処理技術、コロナ社、1991年.
- [9] 笠原、情報処理学会編情報処理ハンドブック第3編計算機アーキテクチャ 7章・並列計算機 6節・基本ソフトウェア、オーム社、1995年.
- [10] 笠原、"並列処理のためのシステムソフトウェア"、情報処理 Vol.34, No.9, 1993年9月.
- [11] 笠原、"マルチプロセッサシステム上での近細粒度並列処理"、情報処理、Vol.37, No.7, Jul. 1996.
- [12] B. Blume, R.Eigenmann, D.Padua, et.al., "Polaris: Then Next Generation on Parallelizing Compilers", 7th Annual Workshop on Languages and Compilers for Parallel Computing, 1993.
- [13] H.Kasahara, S.Narita, "Practical Multiprocessor Scheduling Algorithms for Efficient Parallel Processing," IEEE Trans. on Computers, Vol.C-33, No.11, Nov. 1984.
- [14] H.Kasahara, H.Honda, et.al., "Parallel processing of Near Fine Grain Tasks Using Static Scheduling on OSCAR (Optimally Scheduled Advanced Multiprocessor)," Proc. of IEEE Supercomputing '90, Nov. 1990.
- [15] 藤原、笠原等、"データプレロードおよびポストストアを考慮したマルチプロセッサスケジューリングアルゴリズム"、電子情報通信学会論文誌D-1、Vol.75,

- No.8, pp.495-503, 1992年8月.
- [16] 笠原、吉田、本多、合田等、"Fortran マクロデータフロー処理のマクロタスク生成手法"、電子情報通信学会論文誌D-1、Vol.75, No.8, 1992年8月.
 - [17] A.Yoshida, H.Kasahara, et.al., "Data-Localization for Fortran Macrodataflow Computation Using Partial Static Task Assignment," Proc. ACM Int. Conf. on Supercomputing, May 1996.
 - [18] U. Banerjee, R. Eigenmann, A. Nicolau, D.Padua, "Automatic program parallelization," Proceedings of IEEE, Vol.81, No.2, pp.211-243, Feb. 1993.
 - [19] P.Tu and D.Padua, "Automatic Array Privatization," 6th Annual Workshop on Languages and Compilers for Parallel Computing, 1993.
 - [20] M.Gupta and P.Banerjee, "Demonstration of Automatic Data Partitioning Techniques for Parallelizing Compilers on Multicomputers," IEEE Trans. on Parallel and Distributed System, Vol.3, No.2, pp.179-193, 1992.
 - [21] J.M.Anderson and M.S.Lam, "Global Optimizations for Parallelism and Locality on Scalable Parallel Machines," Proc. of the SIGPLAN '93 Conference on Programming Language Design and Implementation, pp.112-125, 1993.
 - [22] A.Agrawal, D.A.Kranz, V.Natarajan, "Automatic partitioning of parallel loops and data arrays for distributed shared-memory multiprocessors", IEEE Trans. on Parallel and Distributed System, Vol.6, No.9, pp.943-962, 1995.
 - [23] 小幡、笠原等、"科学技術計算プログラムにおけるマルチグレイン並列性の評価"、情報処理学会第56回全国大会、2E-07, Mar. 1998.
 - [24] 木村、笠原等、"マルチグレイン並列処理用シングルチップマルチプロセッサアーキテクチャ"、情報処理学会第56回全国大会、1N-03, Mar. 1998.
 - [25] H.Kasahara, et.al, "OSCAR Multigrain Architecture and its performance", Proc. International Workshop on Innovative Architecture for Future Generation High-Performance Processors and Systems, IEEE Press, Oct. 1997.
 - [26] 本多、笠原等、"Fortran プログラム粗粒度タスク間の並列性検出手法"、電子情報通信学会論文誌D-1 Vol.73, No.12, pp.951-960, 1990年12月.
 - [27] H.Kasahara, A.Yoshida, et.al, "A Data-Localization Compilation Scheme Using Partial Static Task Assignment for Fortran Coarse Grain Parallel Processing", Journal on Parallel Computing (Special Issue on Languages and Compilers for Parallel Computing), May. 1998.

3.2.3 並列プログラミング支援ツールの動向と並列システムの将来

妹尾義樹 委員

おそらくは、数万台規模以上の並列システムになるであろうペタフロップスマシンの性能を十分に活用するアプリケーションプログラムの開発は、非常に困難な作業になることが予想される。本節では、この超並列プログラミング作業を支援するツールについて、その要件と現状を米国を中心に行われている PT00L コンソーシアム[1]の活動を例として参照しながら論ずる。また、このようなツール開発が並列システムの将来に与えるインパクトについて考察する。

3.2.3.1 並列プログラミング支援ツールの重要性と問題点

並列プログラミング[1]において、MPI[2]、HPF[3]そして OpenMP[4]などの並列化記述インタフェースが重要であるのはもちろんであるが、並列プログラミングツールもこれらと同様に大きな役割が期待される。コンパイラによる自動並列化のみによって利用者が期待する並列性能が得られることが少ない現状では、プログラマが明示的に並列化の記述を行ったり、チューニングを行う必要がある。並列プログラミングの作業は逐次の場合のそれに比べて非常にプログラマへの負担が大きいからである。並列マシン（特に分散メモリの高並列システム）が実用システムとして、なかなか一般ユーザに普及しない原因として、MPI は低レベルすぎて一般プログラマが使いこなすのは困難であるとか、HPF は利用範囲が狭く、またコンパイラの安定性に問題があるなどの種々の要因があるが、並列プログラミングツールが未整備であることも大きな要因となっている。並列プログラミング種々のフェーズにおけるユーザの負担と期待されるツール機能について現在利用可能なツールと対比しながら考えてみる。

(1) 並列プログラム開発（コーディング）

対象となる並列システムのアーキテクチャや用いる並列化記述インタフェースにより違いはあるが、一般に並列プログラミング（ここでは逐次プログラムの並列化を念頭におく）においては、並列アルゴリズムの選択、データの分割（分散メモリへのデータレイアウト）、処理分割（ループ繰り返しや手続き処理などのプロセッサへの割り当て）、同期/通信の挿入などを考える必要がある。これらの作業は、対象となるプログラム構造および対象並列システムへの深い理解が不

可欠であり、専門化にとっても複雑で困難な作業である。

一方で、自動並列化コンパイラや HPF コンパイラは、これらの並列化作業を自動的に行うが、その完成度はいわば幼稚園児なみといわざるを得ない。ただし、どのような並列化を行うかの指示さえあれば、プログラムの定型変換を自動的に処理でき、利用者の負荷を大きく軽減できる。そこで、これらのコンパイラ機能にユーザと対話できるインタフェースを設けインタラクティブな並列化支援ツールを構築することが重要である。データパラレル並列化支援として、FORGEexplorer[5]、プログラム構造変換の Foresys[6] などがある。

ただし、並列化作業の下流工程（ループリストラクチャリングやデータマッピング、計算マッピングの決定など）の作業のツールによる支援は実現可能性が高いと思われるが、最も困難な上流工程（アルゴリズムの選択、データ構造の選択など）についての支援については、まだ明確な筋道が見えていない。今後の重要な研究課題の一つである。

(2) チューニング

並列化作業において、作成した並列プログラムが一度で所望の性能を達成することはほとんどない。通常は、利用者がプログラムの並列実行の様子をなんらかのプロファイリングにより把握し、並列実行のボトルネックを一つずつ解消していくということになる。通信量の削減、通信の効率化、逐次実行部分の削減、負荷分散などが主な作業である。現在の並列システムでは、独 Pallas 社の Vampir[7] (図 3.2.3.1 参照)、米 PGI 社の PGPROF[8]、イリノイ大の Pablo[9] などのプロファイリングツールがサポートされており、プログラム中の負荷の分布や、通信の様子などを知ることができる。しかしこれらの情報からボトルネックの要因を判断したり、ソースコードの修正に結びつけるなどの支援はない。これらはすべてユーザの仕事となる。このユーザの作業を支援するツールの研究開発が求められる。

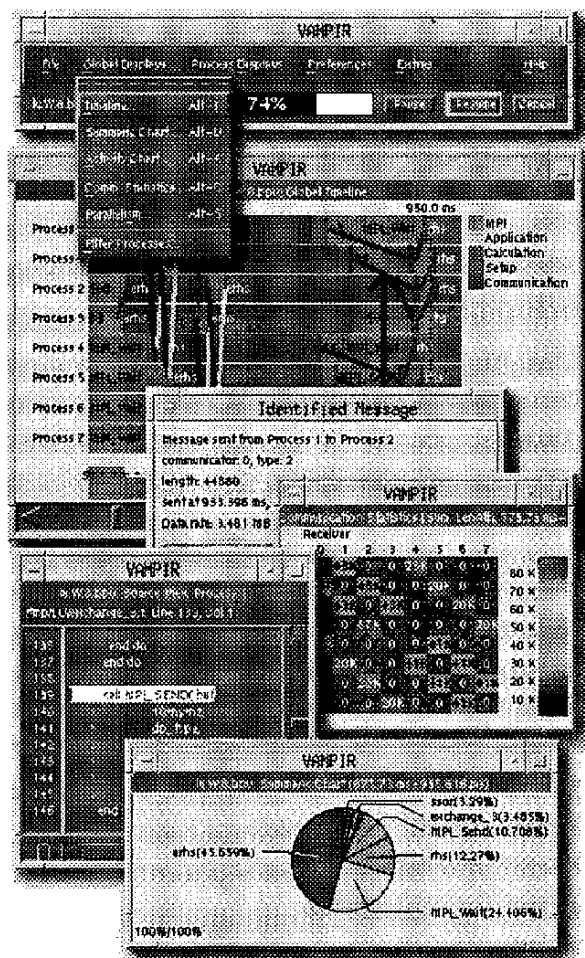


図 3.2.3.1 Vampir の表示例

(<http://www.pallas.de/pages/vampir.htm>から引用)

上から2つ目のWindowで複数プロセスの実行履歴 (Timeline 表示と呼ばれる) を表示している。この中の矢印は通信を示し、これをクリックすることで通信内容や、対応するプログラム個所を表示できる (すぐ下、左下のWindow)。またその時点での全プロセスの通信の様子 (右下) やプログラム全体の要因別コスト分布 (一番下) が表示できる。

(3) デバッグ

並列プログラムのデバッグは、並列化の作業の中で最も困難であると言われている。複数プロセッサによる複数の処理の流れを追う必要があり、また同期忘れなどによるバグには再現性に乏しいものがあり、たとえば100回の実行で1回発生するエラーなども存在するからである。また、100プロセッサ以上の高並列システムにおいては、複数の処理の流れを追うことすら現実的でないケースもある。また、自動並列化コンパイラやHPFは、大規模なプログラム変形を行うため¹、入力プログラムとオブジェクトコードの対応をとるのが困難で、ユーザが記述したプログラムレベルでのデバッグに制限が生ずるという問題点もある。デバッグは、OSやハードウ

¹ コンパイラがワーク変数を導入したり、新たに手続きを生成したりする。

エアに依存する部分が多いため、通常はハードウェアごとに、これを製品化しているベンダからサポートされることが多いが、最近では、米 ETNUS 社の TotalView[10] が多くの米国 ASCI の高並列システムなど広範なプラットフォームでサポートされ、標準の並列デバッガとしての地位を確保しつつある。TotalView は、マルチスレッドプログラムを対象とするグラフィックインタフェースのシンボリックデバッガでありで最小のキーストロークでデバッグ作業を行えるようインタフェースが設計されている。

(4) その他

実際の並列プログラミングにおいて、上記のそれぞれのフェーズ(作業)は独立に行われることは稀であり、相互に密接な関連を持っている。しかしながら、現在利用可能なツールは、特定機能に限定されたものが独立に存在するだけで、プログラマは、インタフェースの異なったこれらツールを使い分けながら、プログラミング作業を進める必要がある。場合によっては、これらツールがそれぞれ別のソフトウェアベンダから提供されることも稀ではない。たとえば、デバッガは Total View、動的性能プロファイラは Vampir という具合である。これらツールを統合し、たとえば、プログラム構造の解析ツールを参照しながらデバッグ、チューニングを行ったリ、さらに、これら作業と並行してプログラム修正、再コンパイル、実行などを行える枠組みの構築が望まれる。

また、ツールインタフェースの標準化についてはほとんど未整備といってよい状況であり、対象となる並列システムが変われば、それぞれ別のツールインタフェースを習得する必要があるという問題点がある。

3.2.3.2 PTOOL コンソーシアム[11]

PTOOL コンソーシアム (The Parallel Tools Consortium) は、1993 年 11 月に設立された米国の産、官、学を中心とするコンソーシアムで、Dept. of Computer Science, Oregon State University の Cherri M. Pancake をリーダーとして、並列化ツールの標準化、開発、普及を目的として活動が続けている。本コンソーシアムの最大の特徴は、活動がツール開発サイドのメンバよりもむしろツールを利用する側の研究者を中心として行われている点である。このことにより、ツール開発側のシーズに基づく論理ではなく、並列プログラムを実際に開発する経験から導かれた並列化ツールに対するニーズが検討され、活動の成果がより直接的に利用者にフィードバックされる結果に結びついている。以下で、PTOOL の主要な活動について述べ

る。

(1) ツールに関する要件の洗い出しと、インタフェース標準化

本活動は、PTOOL と National Coordination Office for Computing, Information, and Communication. が支援する Task Force on Requirements for HPC Software and Tools[12]として行われ、スポンサーは米国 ARPA、DOE、NASA、NSF、DOD である。まず、ツールの要件がユーザを中心としてまとめられ、これが PTOOL のベンダメンバーに提示され、実現可能性について議論される。ここで実現可能と判断されたそれぞれの要件は標準インタフェースとともに文書化され、計算機導入の際の要求仕様としての利用が推奨される。たくさんの主要ユーザから、ここでまとめられた要求仕様が用いられることにより、ツールインタフェースのデファクト標準の確立を目指している。要求されるべき機能は、それぞれの重要度に応じて、Essential for ALL users/ Essential for MANY users/ Essential for SOME users/ will be essential in the future の4段階に分類されている。

ツールという枠組みで検討されているが、エディタや Unix コマンド、言語インタフェース、OS インタフェース、通信ライブラリ、数学ライブラリ、グラフィックス関連とユーザと HPC システムとの接点となる機能はすべて盛り込まれている。これらは ASCI プロジェクト[13]での採用も考慮されており、標準化されれば、そのインパクトは非常に大きい。

(2) Lightweight Corefile Browser (終了済みプロジェクト)

並列プログラムの異常終了時に、その原因とプログラムソース上の発生箇所、異常終了を引き起こしたプロセッサなどの情報を収集するためのツールとそのインタフェース。具体的には、LCF (Standardized Lightweight Corefile Format) と呼ばれる、異常終了時に出力する Core ファイルのフォーマット、Core ファイル生成の API、Core ファイルから必要な情報の取り出し方 (グラフィックベースおよびキャラクタ端末ベース) が規定されている。また、本仕様に基づく試作ツールのソースコードが公開されており、ベンダが開発時にこれを用いることができる。

図 3.2.3.2 にグラフィックインタフェースによる表示例を示す。

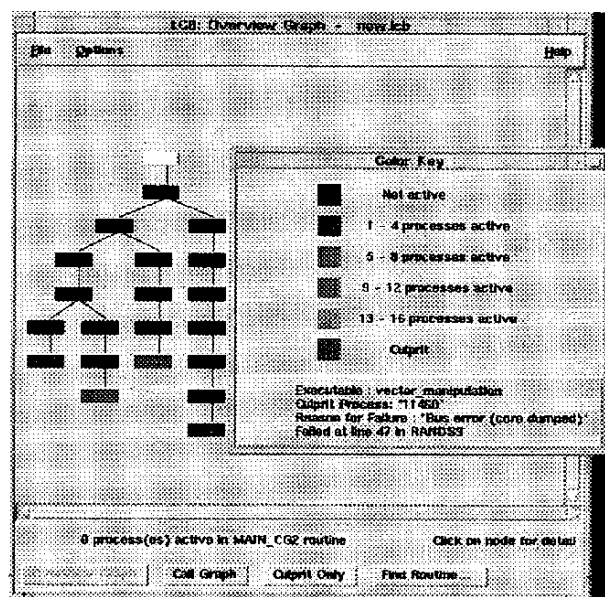


図 3.2.3.2 Lightweight Corefile Browser の表示例

異常終了時の複数プロセスの制御がどこにあったかを手続きの Invocation Tree 上で示している。制御があった手続きが、そこにいたスレッドの数に応じて色分けして表示される。

(<http://www.ptools.org/projects/lcb/demo/> から引用)

(3) Parallel Print Function (終了済みプロジェクト)

MPI を利用した並列プログラムにおいて、並列に動作するプロセッサ（プロセス）の協調動作を記述できる出力文のインタフェースを提供。

たとえば、PPF_print という関数インタフェースが用意されており、これを用いて／各プロセッサが非同期に実行する／プロセッサのMPIでのrank順に出力する／指定した特定のプロセッサだけが実行する／出力をマージして特定のプロセッサだけが実行する／などの出力制御を行うことができる。

(4) Performance API

並列プログラムのチューニング作業には、並列実行の実行時情報（CPU 時間、キャッシュミス、データ転送時間…）の把握が不可欠である。この実行時情報の取得インタフェースを定義するのがこのプロジェクトの目的である。取得情報としては、キャッシュミス関連、分岐予測情報、MIPS 値、FLOPS 値、実行命令数、クロックカウンタ値などがある。

(5) Dynamic Probe Class Library

並列プログラムのInstrumentationについて実行時にON/OFFができることを特徴とするモニタリングツールのAPIを定義する。インタフェースはモニタリングdaemonにサービスを要求するためのクラスライブラリとして定義されており、軽量かつflexible、そしてマシン不依存になっている。

(6) Portable Timing Routines

経過時間やCPU時間などを可能な限り低オーバーヘッドで計測するタイマルーチンのAPIを定義している。ハードウェアで用意したタイマレジスタを読み出すだけの機能から、タイマレジスタ長の桁溢れを考慮して正しい時間を返すものまで種々の機能が用意されている。

(7) Message Queue Manager (MQM)

主に分散メモリマシンにおけるプロセッサ間通信を低レベルでコントロールするインタフェースを提供する。デバッグ時の利用を想定しており、通常のデバッグが変数の値を参照したりセットしたりできる機能をメッセージバッファ(Message Queue)のデータに対して行えるようにすることを目的としている。

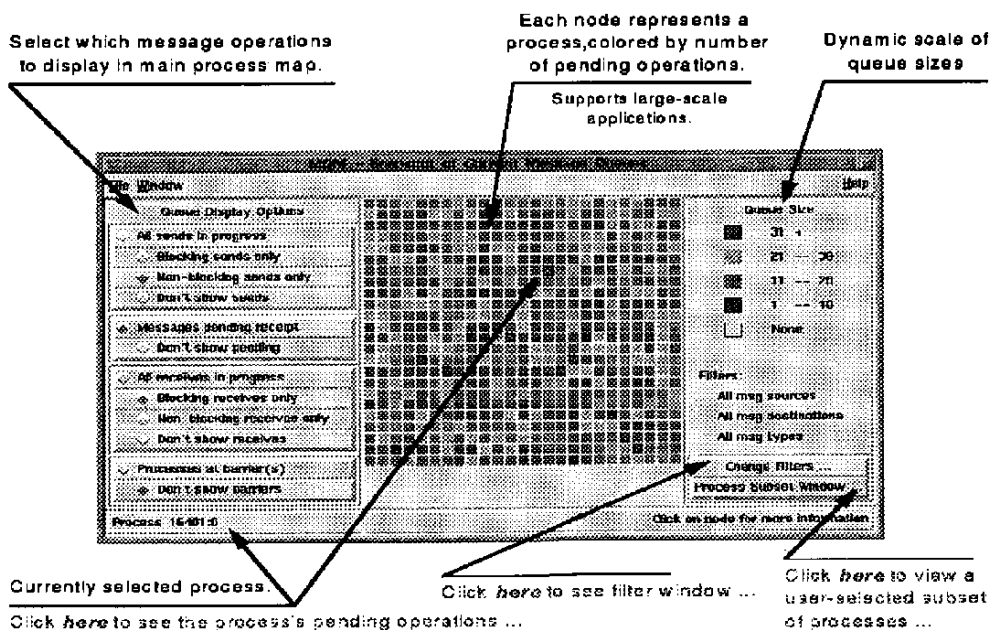


図 3.2.3.3 Message Queue Manager の表示例(次ページへ)
(<http://www.ptools.org/projects/mqm/overview.html> より引用)

中央のそれぞれの四角は各プロセッサのメッセージバッファを示し、キューにたまっているメッセージの数で色分けされている。メッセージタイプや送信プロセスなど、種々の情報によりフィルタリングすることができる。またそれぞれの四角をクリックすることによりキューの中身のデータを表示することができる。

(8) Memory Utilization Tracking Tool (MUTT)

並列プログラムのメモリ利用情報を得るためのインタフェース (API) の定義を目的としている。並列処理においては、コンパイラ、OS、ランタイムライブラリなどがそれぞれ複雑にメモリを利用するため、ユーザが実行時のメモリ利用を把握するのが困難である。このプロジェクトでは、プログラムからこれらの情報を得るための API および、採取したモニタリング情報を可視化するインタフェースの構築を目標として活動している。

(9) Distributed Array Query and Visualization

並列処理において大規模な配列をプロセッサに分散配置して処理した場合に、利用者がその配列のグローバルなデータ内容を把握することが困難となる。これは、一般に配列が大規模であることと、たとえば MPI で記述されたプログラムの場合、物理的にあるまとまった意味を持つ配列データ (逐次プログラムなら 1 つの配列で宣言される) を分割し、このそれぞれを各プロセッサがローカルアドレスイメージで参照するためである。このようなケースにおいて、HPF で行われるような、グローバルイメージのデータの分散配置をユーザが定義し、この情報に基づいて、ツールにローカルイメージのデータをグローバルイメージでアクセスする機能

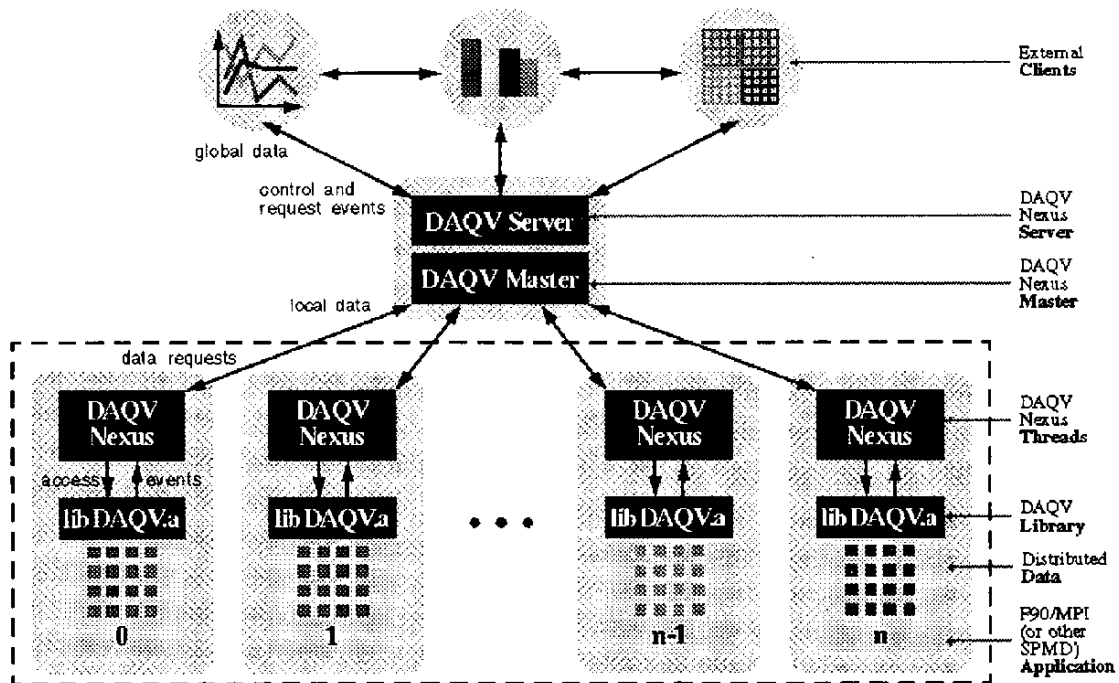


図 3.2.3.4 DAQV システムの構成例

(<http://www.cs.uoregon.edu/research/paracomp/proj/ascii/daqv/overview/>より引用)

一番下の各プロセッサにおいて MPI などを利用される分散配列（ローカルイメージ）の全体構造を DAQV Master で管理し、グローバルイメージでの Client からのデータ可視化要求に答える。

を提供することを狙っている。また、大規模データを生データの羅列ではなく、グラフィックイメージによる表示機能の標準 API の検討を行っている。

(10) High Performance Debugging Forum (<http://www.ptools.org/hpdf/>)

この Forum は、直接の PTOOL プロジェクトではなく、PTOOL がスポンサーとなる活動と位置付けられている。活動の目的は、並列デバッガの標準インタフェースを構築することである。インタフェースとして、CUI (Character User Interface) ベースのものと、GUI ベースのものと両方の標準化を目指している。

具体的活動としては、Conceptual model for parallel debugging、General Interface、Program Location Information、File/Function/Variable Names、Expressions and Language Compatibility、Process/Thread Groups、Data Display

and Manipulation、Actionpoints、Process Control、Process Initialization and Destruction の 10 の Working Groups を設け、それぞれの WG においてユーザからの要求をベースにインタフェース案が構築されている。

現在のところ、CUI ベースの標準インタフェースのドラフトが作成され、参加者でレビューするとともに、Reference Implementation が行われている。これは完成すれば公開し、ベンダによる本仕様に基づくデバッグの早期開発を加速することを狙っている。

(11) Parallel Unix Commands

対象とする並列システムの各プロセッサに並列に Unix コマンドを適用するインタフェースの検討を行っている。たとえば、それぞれのプロセッサがローカルディスクを持ち、あるディスクの内容を他の複数のディスクにコピーしたいような場合、このような処理を 1 つのコマンドで記述できると便利がよい。また、この処理において、1 つのプロセッサが順次コピー操作を逐次的に行ったのでは効率が悪く、できれば、メッセージパッシングのブロードキャストのように、2 分木方式で並列にコピー操作を行いたい。そこで、並列 Unix コマンドの検討ということになったわけである。ファイル操作の他に、各プロセッサのプロセスやスレッドの状況を表示する機能なども検討されている。

3.2.3.3 並列プログラミング支援ツールで今後取り組むべき課題

並列プログラミングインタフェースについて、現状およびその問題点、そして米国 PTOOLS の検討状況について述べてきたが、これらを踏まえて、今後わが国で取り組むべき課題について私見を述べる。

(1) インタフェース標準化

米国のここ数年の HPC 分野での活動を振り返ると、標準化活動に非常に大きなウエイトをおいていることがわかる。たとえば、並列化インタフェースでは、MPI や HPF そして OpenMP などの仕様標準化を行ってきたし、ここで取り上げた PTOOLS コンソーシアムの活動の最大の目的もツールの標準インタフェースの構築にある。この他、Internet2 プロジェクトにおける広域分散処理研究などでも標準インタフェースの構築が重要なテーマとなっている。

これら、標準化を目指す活動の共通の特徴は、

(a) エンドユーザ、官／学の研究者、ベンダなど広範な人々が参加するフォーラム

形式で活動

- (b) 活動を牽引するのは、官／学の研究者
- (c) 国が資金面でバックアップするが、活動そのものは世界にオープン
- (d) 官／学でリファレンスインプリメントを行い、ベンダ（SWベンチャーが多い）の製品開発を加速（ある意味で新規産業の育成につながっている）

などである。これに対し、日本では標準化を意識した、こういったタイプのプロジェクトがあまりにも少ないように思われる。今後の大規模プロジェクトでは、テーマにかかわらず、技術の核となるべき標準インタフェースの構築をフラグシップにすることが重要だと考える。

(2) HPC ユーザ層の拡大

PTOOLS の活動は、ある意味では非常に泥臭いものである。華やかな新規技術を追求するのではなく、既存技術に立脚した地道なインタフェースを構築している。たとえば、Lightweight Corefile Browser や Parallel Print Function、そして Portable timing Routines などがその典型である。これらは非常に地味ではあるが、一方で標準化が実現されれば、HPC ユーザにとっては大きな利益になるものである。技術的に新鮮味が薄くとも、先端の現場で毎日 HPC システムと格闘している多数ユーザからの要望をもとにしたインタフェースを構築することで、自らが使ったことのないベンダ任せのシステムよりも、はるかに使い勝手のよいものができる。そして、これらツールの整備により、HPC ユーザ層を拡大することができる。

また、これらの活動形態にはもう一つの利点がある。すなわち、ユーザとベンダが、利用インタフェースに関する議論をすることにより、ベンダにはユーザの立場の情報が、またユーザにはベンダのインプリメントにかかわる技術情報が双方向に流れる。ユーザはシステムについてのより深い知識を吸収することにより、システム利用技術を高めることができる。この点で、ツールインタフェースというのは非常に良い切り口である。HW アーキテクチャに対して注文をつけるのはデバイステクノロジーに関する知識も必要であり、かなり難しいが、いつも使っているツールに関する問題点はユーザが一番良く知っている。

HPC 分野でのわが国の最大の問題点はユーザ層の薄さにある。一部先端ユーザは存在するが、その絶対数および裾野が小さい。特に、HPC システム構築技術に精通したエンドユーザが欧米に比べて少ないように思う。このため、ユーザからベンダへの注文が少なく、Seeds 指向あるいは欧米発祥のツールを押し付けられることになる。インタフェース標準化をフラグシップとした産官学が一体となったプロジェ

クトにより、ユーザ層の拡大を目指すことは重要であると考え。米国のあと追いになるからと、これら活動を眺めているばかりだと、米国との差は広がる一方である。

(3) ペタフロップスマシンにおけるツール整備

ペタフロップスマシンを実現したとして、この上でどのようなツールが必要になるかについてあらためて考えてみると、実は良くわからないというのが実情である。米国の活動も、ASCIなどを視野においているが、どちらかといえば、市場の大きい中並列の SMP が主対象であるように思える（特にベンダの興味はここに集中している）。HPC システム（特にペタフロップスマシンのようなハイエンド機）ではハードウェアはできるだけシンプルな作りにして、その性能を追求する。複雑なことは、可能な限りソフトウェアに任せるとというのが今後の流れのように思う。ペタフロップスマシンを目指すプロジェクトを考えるなら、システムソフトウェア、特にツールの研究開発にもある程度のウェイトをおいたものにする必要がある。

参考文献 (URL)

- [1] 湯浅他編、「はじめての並列プログラミング」、Bit 別冊、共立出版、1998 年 5 月
- [2] The Message Passing Interface standard <http://www.mcs.anl.gov/mpi/>
- [3] High Performance Fortran Forum <http://www.crpc.rice.edu/HPFF/>
- [4] OpenMP <http://www.openmp.org/>
- [5] FORGExplorer: http://www.apri.com/apr_forgex.html
- [6] Foresys: <http://www.simulog.fr/foresys/>
- [7] Vampir(Pallas 社): <http://www.pallas.de/pages/vampir.htm>
- [8] PGPROF: http://www.pgroup.com/ppro_docs/pgiws_ug/pgi30ul5.htm
- [9] Pablo: <http://www-pablo.cs.uiuc.edu/>
- [10] TotalView(ETNUS 社): <http://www.etnus.com/tw/tvover.htm>
- [11] The Parallel Tools Consortium: <http://www.ptools.org/>
- [12] Task Force on Requirements for HPC Software and Tools:
<http://www.nacse.org/projects/HPCreqts/>
- [13] ASCI: <http://www.llnl.gov/asci/index.html>

3.3 アプリケーション

3.3.1 高性能計算技術におけるブレークスルー

関口智嗣 委員

3.3.1.1 概要

通商産業省がとりまとめを行っている「経済構造改革アクションプラン」「コンピュータ関連基盤技術開発プログラムの策定について」において、下記の3つの点が今後の重要施策項目として検討が行われている。

- (1) 電子商取引等、ネットワークを通じた様々なサービス等の実現のための基盤として情報通信ネットワークの一層の高度化
- (2) ビジネスにおける情報処理ニーズの増大やシミュレーション等の科学技術計算分野からの要求に加えて、動画像等のマルチメディア処理等を用いたビジネスの発展等も見込まれることから情報処理の高速化・大容量化
- (3) 情報通信システムを学生から高齢者まで様々な者が自由に使いこなせるような情報家電等のシステムの普及等の観点から、使いやすさの向上への対応

ここには、情報通信技術の技術的限界を突破して将来に向かっての可能性を拡大するものであって、新たな情報通信技術の応用分野を創出して我が国経済の発展、国民生活の向上に資することが期待される重要技術分野を抽出し、それらの研究開発の方向性を示すことにより情報通信の研究開発及び技術革新の促進を目的とした技術研究開発を行うという背景がある。これに対応して、高速ネットワーク、大規模大容量、ユーザビリティという観点から研究項目の検討がなされている。特に、「大規模大容量」においては次世代のスーパーコンピュータや超高密度記録システム等に関連して下記の3項目に整理されるであろう。

- (a) 超高性能計算機利用技術、革新コンピュータ技術等の HPC (High Performance Computing) 技術
- (b) 新薬、新材料等の設計、環境等先端シミュレーション技術といった高速化・大容量化が求められるアプリケーション技術
- (c) 光デバイス技術（光磁気、オプトエレクトロニクス技術等を含む）、超高速デバイス技術（省エネルギー技術を含む）、高密度実装技術等のデバイス関連技術

情報技術発展の方向性を整理し、計算科学技術に代表される高性能計算技術ならびにネットワーク利用技術について動向を述べる。

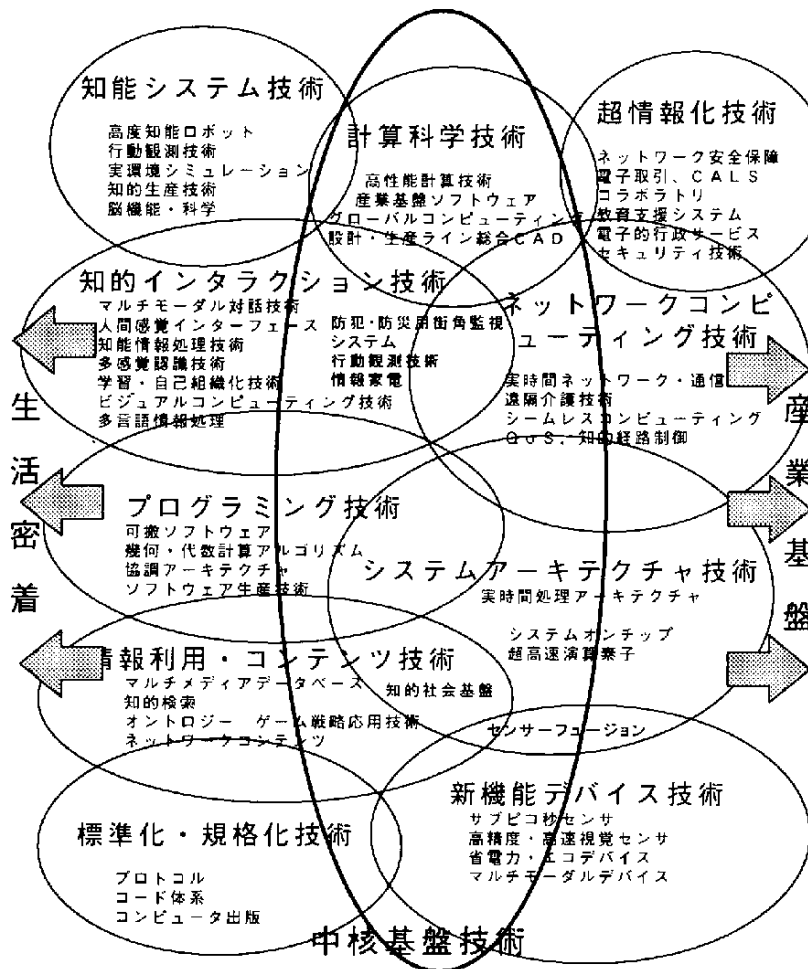


図1 情報技術研究の方向性

図1に情報技術研究の方向性について整理を行った。大きく、生活密着型（コンシューマプロダクト）と産業基盤型（インフラ）という軸を与える。技術の分類とそれぞれを代表するであろうキーワードを割り振った。縦軸としてはデバイス技術に始まり、応用システムへの階層を想定した。このうち、いくつかのキーワードについてはすでにプロジェクト化が行われている。

これまでに幾度となく議論されているが、計算科学技術または情報技術的にいえばハイパフォーマンスコンピューティング (HPC) 技術は基盤技術として、産業応用を支えている (図2)。HPC 技術により、新機能材料、環境資源、半導体、航空宇宙などに応用できるソフトウェアが開発され、また広く使われている。もちろん、これらがすべて完成した技術ではない。従来のソフトウェアは従来のハードウェア・システム資源に依存した設計となっており、必ずしも、今後の新たな環境で高性能を発揮するものではない。



図2 HPC 関連技術

もう少し具体的な研究課題を考えるために、電子技術総合研究所の HPC グループを例に取った HPC 技術の関連を図3に示す。すなわち HPC 基盤システムアーキテクチャとして「ネットワークコンピューティング」と「クラスタコンピューティング」の両システムを対象のプラットフォームとしている。いずれも、今後の高速化されるネットワークとプロセッサの進展を取り入れることが容易なシステムアーキテクチャであり、発展が期待されている。計算科学のアプリケーションとしては、複数のスキームを同時に扱うマルチスケール手法やさらに複雑なフルフィジックスモデルが今後の対象となる。しかし、これらの問題点については後述する。さらに、プラットフォーム技術とアプリケーションの双方と深く関連するのが性能評価技術である。プラットフォームの特微量ならびにアプリケーションの特微量を与える性能評価モデルを設

定することが重要な点である。性能評価を通じることにより、プラットフォームとアプリケーションが関連付けられ、「最適化」という言葉が意味を持ち、アルゴリズムやプログラムなどの良否を議論することが可能となる。

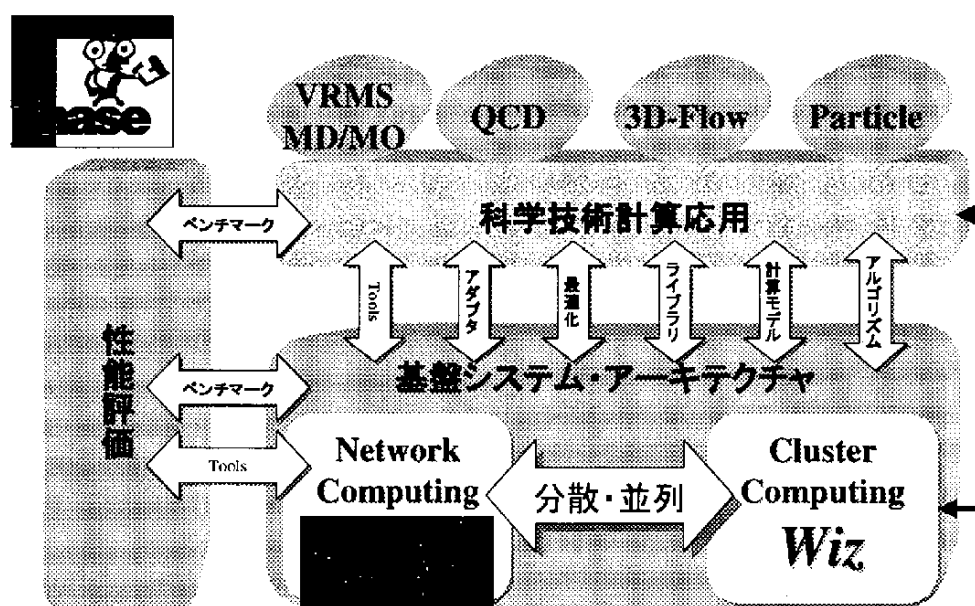


図3 HPC 関連研究展開

3.3.1.2 HPC 基盤システムとしてのグローバルコンピューティング

例えば、計算化学や構造設計などを実用規模で計算するためには、現存のスーパーコンピュータを越える計算性能が必須となる。さらに、ユーザにおいては様々なパラメータ設定を行い効率的にアルゴリズムを検証するために、従来のスーパーコンピュータのようにバッチシステムとしてバックエンドコンピュータとして利用するのではなく、自分のデスクトップからインタラクティブに解決したいという要求を持つ。これらの問題を解決するために、高速ネットワークで接続された複数の計算資源をデスクトップから自由に組み合わせて大規模計算資源として利用可能とするようなデスクトップネットワークコンピューティング環境を実現することが必要になる。これらの要求を満たすために、ネットワーク中の巨大計算資源や遊休資源をデスクトップから

自由かつ容易に組み合わせて経済的に有効利用できるデスクトップネットワークコンピューティングの実現を計る。同様のことが、物質構造計算や流体力学など多くの応用にもいえる。これらの実現には、これまで未解決であった Network Enabled Server、Federated Resource Access、Network Transparent Programming、Performance Portability などの新しい技術要素の開発が、不可欠である。

研究開発途上のグローバルコンピューティング基盤システムはインハウスの実験運用だけではなく、フィールドにおける実践的な運用の中から新たな一步を踏み出す時期に来ている。セキュリティ、運用管理、スケジューリング等でまだ多くの研究課題を抱えている。また、分散メモリ型の並列計算機に対する効率のよい RPC を実装する必要もある。さらに、様々なグローバルコンピューティングと連携するためにはプロトコルの標準化が不可欠である。こうした実験に共同の実践の場が必要となってきた。

米国では官学連携の GRID というグループが徐々に勢力を持ってきていることが知られている。GRID は GRID という単一プログラムが存在する一枚板のプロジェクトではなく、実際にはそれぞれの要素技術研究の集合体である。これを上手にコーディネートして、全体の流れを作っていく、大きな技術潮流を作っていくかという点が心憎いばかりにうまい。今後のネットワークアプリケーションを考えた場合にこうしたグローバルコンピューティングの研究は必須のものとなるであろう。さらに、日米間、日豪間などに国際高速回線が整備され、これを使った真の共同研究が求められるような状況である。国内でも高速ネットワークの整備が進み、ネットワーク距離は徐々に狭まってきている。こうした高速ネットが手近に利用可能となると、グローバルコンピューティングの現実味が帯びてくる。しかし、グローバルコンピューティングはネットワークの管理設置、計算資源の提供、複数の協力者、がどうしても必要である。そのため、単独の研究拠点では展開することが困難である。例えば、卒業論文でグローバルコンピューティングに関係する研究をやろうと思いついても、なかなか研究室では立ち上げるのが難しい。そこで、こうしたグローバルコンピューティングに携わる人の裾野を広げること、グローバルコンピューティングの実験環境を整備し研究を加速することを目的としたグローバルコンピューティングテストベッド (GCT) が必要である。こうした活動は公的資金で支援すべき性質のものであると考える。現在は特別な運営団体や運営費を持たないボランティアベースである。このテストベッドのユーザとして期待されているのは、基盤システム開発者、ツール群開発者、応用ソフト群開発者である。エンドユーザ向の単なるサービスは提供されないであろう。特に、こうした基盤システムやツールを用いた応用ソフトウェアの開発には世界中から多い

に期待される。

3.3.1.3 マルチスケール計算技術

近年の計算機能力の向上に伴う計算の大規模化と、計算手法、近似の信頼性の向上は、計算機シミュレーションにより「空間と時間についての強力な顕微鏡」として充分なところまで到達しつつある。材料設計や構造設計において今以上に、実験主導のスタイルから計算科学主導のスタイルにするためには、実験に伴われる不確定要素に起因する現象をも含んだ、起こりうる現象のできる限りの再現能力が必要である。計算科学の特色を生かしたより有効な最適化（設計）技術・解探索技術の確立が必要であり、今後のより一層の研究が望まれる。具体的なマルチスケール問題の典型例を図4に示す。

- (1) ミクロでの現象のより精密な記述（量子力学など）をはかり、
- (2) マクロにおける実時間、実スケールでの現象取り扱い

が必要となるが、(1)との連続性の確保が必須となる。これにより、将来的には生体内反応など、より高次の現象の記述が可能となる。

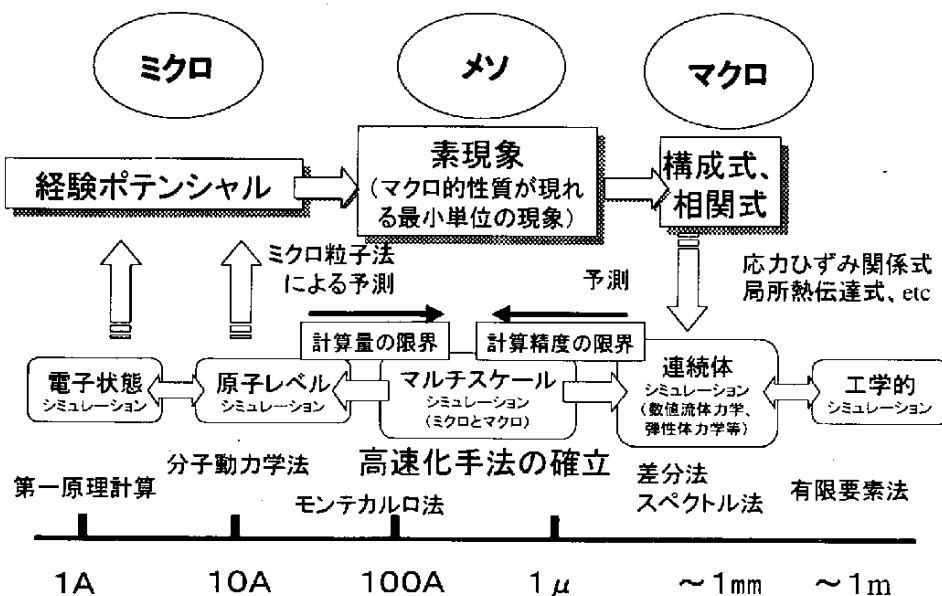


図4 マルチスケール計算の例

材料の性質として強度や破壊を見る分野では別のマルチスケール技術が用いられている。ここでは引張強度が理論値と大きく異なることを理解するために発展した線形破壊力学理論、せん断強度が理論値と大きく異なることを理解するために発展した転位論である。これらの理論は連続体モデルに基づいて、有限要素法や境界要素法等が用いられる。さらに、物質の特性の記述ために、経験値（実験値）に基づいた構成方程式が必要となる。しかし、一般的に構成方程式では材料内部の原子スケールの構造を正確に記述できない。そこで、よりミクロなスケールからの原子モデルを用いることで構成方程式を構成することなくできる限り厳密に解析することが可能となってきた。まだ、計算機資源の問題やメモリの問題があり、十分に大きな系は扱えないが、今後は大規模アプリケーションのひとつとして期待される。実用材料の内部構造を十分正確に記述できるようになれば、それを元にして合金設計やさらにもっと広く材料設計への応用が可能となり、新材料の開発の効率化が期待される。

一方で計算科学として考慮すべき点は、例えば材料科学的にも次のようなことがある。情報技術としては直接扱う話ではないが、HPC 技術の促進という意味では支援が必要である。まず、分子動力学やモンカルロ法など、すでに開発されている手法を用いて分子レベルで現実的な複雑な系について材料設計を行おうとすると、信頼できるポテンシャル関数や力場パラメーターが手元にないことが多い。これにはポテンシャルに関するデータベースを作成することが研究効率の向上に有効である。次に、長時間（マイクロ秒かそれ以上）における凝縮系の動力学の計算は誤差の問題、計算時間の問題などがあり、新たな方式が期待されている。

しかしながら、マルチスケールにおけるモデルの融合とその際のアルゴリズム的発展の期待は非常に高まってきていると言える。この分野でのソフトウェア開発が先行して可能であれば、現在諸外国に先行されている産業基盤ソフトウェアでの立場が逆転することが期待される。

3.3.1.4 終わりに代えて

HPC 技術の動向と情報技術における位置づけを記した。特にネットワークコンピューティングとマルチスケール計算技術の2点について状況を記した。公的資金による研究開発ならびにその準備としての各種委員会についてであるが、プロジェクトの包括的な方向性をどこかで議論を行い、その具体的な中身について研究者、行政側、大

学等と議論を行う場が必要である。すなわち、個々の要素技術の寄せ集めではなく、グランドデザインがあり、これが産官学共同で立案できる性質のものが望ましい。そのためには、現在の委員会形式で年に数回議論を行い、報告書を作成するのではなく、1週間程度の議論・立案・作文を中心としたワークショップを合宿で実施し、一気にプロジェクト立案とそれぞれの参加者による想定される貢献を記録するところまで推進することを提案する。その後は、既存のワークショップ等を活用してアップデートを行っていけばよい。そうすることによって、アカデミアとして一体のプロジェクトが計画・実施できるのではないだろうか。

3.3.2 世界における日本の強みと立場を活かした基礎研究開発 中島克人 委員

3.3.2.1 はじめに

日本経済の長期低迷により企業の設備投資は冷えきっており、本来はリストラや業務効率化に必要な情報処理関連投資も、米国製のプロセッサと米国製の OS を搭載した安価なパソコン以外は極端な落ち込みとなっている。企業では研究開発投資も新人採用も抑制基調のままで推移している。日本の特に情報関連の産業および技術の地盤沈下はどんどん進んでおり、今後景気が好転したとしても、取り返しのつかない状況に陥っているのではないかと危惧する。

国の施策としてプロジェクトを起こし、この地盤沈下に歯止めをかけ、更に浮揚を図るためには、良いテーマと良い枠組み(人・組織・法律)と国民や企業の期待・理解などの条件を揃えなければならない。グローバル化したこの世界の中で、「日本で行うことにより成功した」と振返ることが出来るようにするためには、日本の特徴、長所・短所等の現状を考慮したプロジェクト立案が必要である。

本報告ではまず、ケーススタディとして、筆者が考えるプロジェクト候補技術分野について述べる。将来社会のニーズは勿論のこと、日本の得意とするデバイス開発の要素があるのか、標準仕様を策定する要素があるのか、標準仕様の下にデバイスから応用ソフトまでの縦横に広がりを持ち得る領域であるのか等の観点で議論する。次に、基礎研究開発プロジェクトの立案・選定において考慮すべき日本の特殊性について検討し、最後にまとめる。

3.3.2.2 ケーススタディ

ここでは、計算パワーを必要とする候補領域として、「情報セキュリティ」と「分散シミュレーション」の2つを取りあげる。

(1) 情報セキュリティ

インターネットの一般庶民への普及により、エレクトリックコマースが急拡大しつつある。従来の企業や個人のプライバシー保護以上に電子マネーの安全利用は重要視されるため、情報セキュリティ技術の利用・普及は急ピッチである。エレクトリックコマース等で

は多数のクライアント側と小数の売り手や認証センターにおけるサーバ側で情報セキュリティのための先進技術が必要となる。

クライアント側での本人認証には指紋・顔・虹彩・声紋等の生体情報を利用することが研究されているが、そのためには認識装置が必要である。安価で誤認証の少ない高速な装置とするために専用デバイスの開発が必要かも知れない[1]。また、パスワードや注文内容の暗号化を高速に行うための仕掛けも必要であり、現実的には解読不能とするためには複雑で時間のかかる暗号アルゴリズムを採用しなくてはならなくなっている。

暗号には秘密鍵を受発信双方で共有する秘密鍵暗号(共通鍵暗号)と、受信者が公開した鍵で送信者に暗号化して貰い、受信側で秘密鍵で復号化する公開鍵暗号がある。後者の方が安全性の高いと言われるが、暗号化に前者の 10 倍以上の時間(ワークステーションで数十 ms のオーダー)がかかるため、ある程度の分量のコンテンツは秘密鍵暗号方式で通信せざるを得ず、その秘密鍵の配送のみを公開鍵暗号で送信することが多い。

秘密鍵暗号の安全性は鍵長が大きな指標となるが、米国で普及している 56 ビット鍵長の DES(Data Encryption Standard)ではもはや安全ではないと認識されだした。というのは、RSA が主催する DES 解読コンテスト(DES チャレンジ II)において、1997 年 6 月にはインターネット上の 70,000 台の計算機で手分けして 32 日で解読が成功し、翌年の 1998 年 7 月には EFF(Electronic Frontier Foundation)が開発した DES-Cracker という専用マシンによりわずか 56 時間で解読されてしまった[2]からである。DES-Cracker では専用開発された LSI が 1800 個用いられているが、サーバタイプの筐体数台分に収納されており、決して大袈裟な並列マシンではない。日本でも広く利用されている FEAL のような 64 ビット鍵長でも、LSI 技術の長足の進歩と解読ノウハウの進歩により、全く安心ができない状況になって来た。

暗号技術は盾と矛の関係にあり、暗号の安全性評価、即ち、暗号の破られにくさの研究は、暗号解読を試みることにより行われる。情報化社会の健全な発展に寄与するために、米国の NSA(National Security Agency)に対応するような組織を作り、利用側技術(=盾)として公開鍵暗号や鍵長 64 ビットを超える秘密鍵暗号の高速化のための専用デバイス開発、安全性評価技術(=矛)として LSI 技術を駆使したより強力な暗号解読マシンの開発を指揮・統合し、標準化を推進すると共に、認証業務、クラッキングの情報収集・蓄積なども行っではどうであろうか。生体情報認識も古くて、かつ、今でもチャレンジングな研究テーマである。インターネット上での認証だけでなく、一般のセキュリティ関連への応用も広く、市場は小さくない。

(2) 分散シミュレーション

マイクロプロセッサの長足の進歩により、文房具や玩具としてのパソコンやゲーム機の性能はかなり満足の出来るレベルに達したのではないだろうか。プロセッサの能力が上がればそれだけ重いソフトウェアが現れると言われて来たが、数万円のゲーム機で高解像3次元グラフィックがリアルタイムで滑らかに動くようになると、これ以上を望む人は一般大衆の中では少数派となろう。即ち、プラットフォームとしては成熟期を迎えつつあるのではないだろうか。

計算機的高速化・大容量化が現状でもまだまだ望まれ、当面それが際限なく続くと思われるものの代表例は科学技術(物理・化学・環境等)・産業(設計・製造)・経済・政治・教育／訓練等におけるシミュレーションであろう。

平成8年度の報告書で、仮想実験室としてのシミュレーションが今後とも有効である分野を「倫理的理由」「社会的理由」「実現性の理由」「費用・開発速度的理由」に分けて議論した。シミュレーションにおいてはモデリングが重要であるのは間違いないが、これはそれぞれの応用分野の専門家に譲るとして、ここではシミュレーションのためのプラットフォーム技術に関して述べる。

この分野では精度向上・規模拡大・機能拡張のための有効な手段は並列処理もしくは分散処理である。特に最近ではスケーラビリティとコスト価格比を指向する分散シミュレーション技術、即ち、複数のワークステーション(WS)やパソコン(PC)上にそれぞれ搭載されたシミュレータを連係動作させる技術が注目を集めている。同じ構成要素を持つシミュレーションシステムでも、その目的やデータにより要求されるメモリ容量や応答性能が異なる。マシン単位のスケーラビリティ、もしくは、機能モジュールのプロセス単位のコネクタビリティが確保されれば、ソフトウェアの再利用性と拡張性が高まり、システムの開発コスト低減に対する効果も高い。

2000年にIEEE標準化が準備されている分散シミュレーション接続仕様HLA(High Level Architecture)[4]は、そういった目的のものであり、米国では当初国防総省の強力なバックアップの下で技術開発が開始されたが、現在では宇宙防衛産業界に限らず、シミュレータ業界からネットワークゲーム業界に至るまで一丸となってIEEE標準化を進めている。

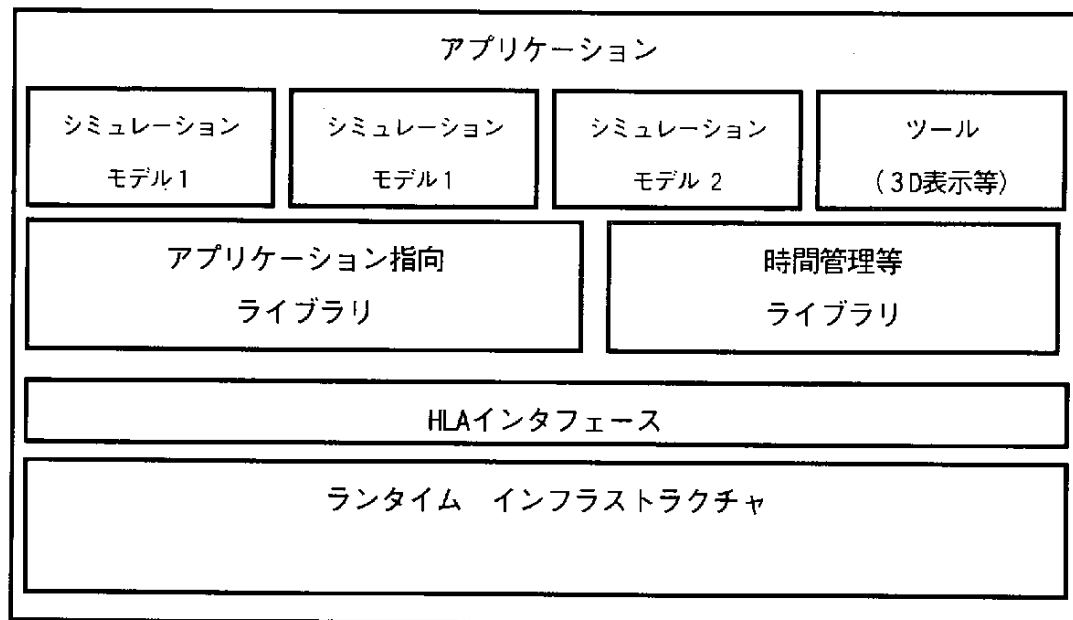


図1 HLAに基づく協調型シミュレーションシステムの構成例

図1はHLAをインターフェースとして採用した場合の、異種連係型のシミュレーションシステムの構成例である。HLAはインターフェースを定義しているに過ぎず、それぞれのシミュレータの構成は各自の創意工夫に任されているが、HLAに準拠することにより、モジュラリティの高い設計が促進される。例えば、複数のシミュレーションモデル間の通信や同期実行を仲介するランタイムインフラストラクチャや、シミュレーションの時刻進行を制御するライブラリ、例えば対戦型ゲームといったある応用のタイプに共通なライブラリなど、多くの共通ミドルウェアを最終的な応用とは独立に用意することが可能となった[3]。

HLAの策定は米国の特定のプロジェクトにおいてなされているものではない。3年以上に渡る春秋のワークショップとメール等による密な議論を重ねることにより、標準化と周辺技術開発、そして、HLAに準拠するコミュニティの形成を成し遂げて来ている。ただ、民間のボランティアではこれだけ短期間に仕様の策定を行うのは不可能であろう。米国でこれが可能だったのは、'96年に米国防総省が発表した、2001年以降のHLA非準拠シミュレーション製品の調達禁止宣言であろう。シミュレーション機能を備えてない防衛用システムはゼロに近く、また、米国の製造業で防衛システムを担当していないメーカは少数派であることから、短期間に主要な製造業がHLAの利用技術を身につけさせるのに、米国政府は成功したように思える。

Information Super Highway 計画にせよ、HLA にせよ、米国では官側が国家が目指す理想を提示し、技術開発に必要なインフラに対する資金の投入は惜しまず、しかしそれに向けての技術開発は民側にまかせ、産業と経済を独自に活性化させることによって、結果的に国家全体の利益につながる、というビジネスモデルの上に、国家プロジェクトが成り立っている。

我国でも、上記のような枠組みの上で、HLA のような標準仕様をベースにその周辺の基盤・共通・応用技術を発展させるようなプロジェクトを設定すべきである。もしこの分散シミュレーション領域自身であれば、HLA に準拠しつつ更に应用到に近い部分での新たなレイヤを策定できるかも知れない。ランタイムインフラストラクチャや共通性の高いライブラリをエンハンスする専用デバイスの開発等に民側が競って参加できるようなものに出来るに違いない。

3.3.2.3 日本の特殊事情とそれを考慮したプロジェクトについて

この 20 年間に世界で成功している日本の製品・作品には、一般大衆向けの製品、小さくて高機能・高性能な製品、TV ゲームやアニメやタマゴッチのような情感に訴えるものが多い。高平均点を目指す均質な日本の受験教育は先端デバイスによる高品質製品を説明できても、TV ゲームの世界ヒットは説明できない。理由は分からないが、とにかくこれを日本人の特性と信じて、これらの特徴を備えたプロダクトが誕生または派生するかどうかをプロジェクトのテーマ設定における一つの指標としても良からう。

一方、日本人は論理的構成力に弱いと言われているが、その結果として、小異を越えた高邁なビジョンの下に標準化を進めるような活動が苦手だと思う。少なくとも情報処理分野では日本発の標準仕様やデファクトスタンダードは殆んどない。また、日本の受験教育および寄り道を不利にする年功序列システムのせいで、複数専攻を持つ人が非常に少ない。情報処理技術は言わば組合せ技術であり、学際的能力が活かせる分野であると思うが、日本で育った者に多くは期待できない。これらの弱点を避けて行くべきか、むしろ克服すべくチャレンジすべきかは判断が分かれるところである。

更に、情報処理の世界における日本の最大の不利は日本語の特殊性であろう。それに加えて、奥ゆかしい国民性からか、情報発信の能力も劣っていると思う。村八分を恐れる強い協調性はあくまでも受け身のものである。そのため、外部もしくは異種のものとの相互作用／切磋琢磨の中から新しい製品を生み出すというよりは、与えられた（多くの場合は欧米製の）モデルを暖め磨いて行くことに長けている。しかし、この言語障壁はどうしても乗り越えなくてはならない。この課題を持たない英語圏の国よりは絶対に技術進歩する

筈である。

さて、前章にてスタディした情報セキュリティにおいては、クライアント側の生体情報認識や暗号化に日本の得意とするデバイス開発の余地がある。情感に訴えるようなコンテンツとしては、このセキュアなインフラの下、新規パソコンユーザの旗手である主婦層向けのバーチャルモールなどが考えられる。しかしながら、日本が得意としない暗号アルゴリズムや認証の仕組み等の標準化の要素が重要であり、これは避けて通れない。日本語処理としては、声紋認識等に日本語の特殊性の考慮が必要となろう。

もう一方の分散シミュレーション分野では、デバイス開発に発展するかどうかは不明である。特定のシミュレータエンジンやユーザ端末側の対話機能のために専用デバイスが必要となるかも知れない。標準化については最重要課題である。応用としては、日本が得意とする対戦型ゲーム等をそのまま含み、多くのアイデアが試されることが期待される。

3.3.2.4 おわりに

経済力と技術力を背景に世界経済の三極の一つと見なされて来た日本は、経済の長期低迷に陥り、今後もその地位を確保できるのか怪しくなってきた。企業は技術開発投資の縮小を余儀なくされ、日本を世界の一流としていた根拠である経済力と技術力の両方を同時に失いつつあるのではないだろうか。

政府は景気刺激策としての補正予算を次々と打ち出すが、従来型の土木・建築等と異なり、情報処理技術分野では予算を有効に活用する仕組みが非常に乏しい。例えば、情報インフラを整備するタイプの研究開発事業は、技術進歩のみならず産業育成に大変有効であるが、省庁の縦割り論理や官民役割分担の建前論により、本来素晴らしいテーマが却下されたり歪曲されたりすることもあると聞く。予算の使途についても、理不尽とも言える足枷が多くあり、融通のなさからかえって無駄が生じる可能性も高い。

規則は人が作り、人を拘束する。しかし、規則を変えることが出来るのも人である。幸い、世論を動かし得るインターネットという手段を大衆は手にした。情報処理分野発展のためには、情報処理技術を武器として既成の制度や考え方などの足枷を取り払うべく、官民の有志が一致して努力していきたいものである。

参考文献

- [1] 久間、田中、太田、田井、岩附：人工網膜チップの開発と事業化、応用物理、pp.424-430, Vol.67, 1998.

第3章 わが国における研究開発の課題と展望

[2] RSA Laboratories - DES Challenge II, DES Challenge II home page, RSA Data Security, Inc.

Available at <http://www.rsa.com/rsalabs/des2/>

[3] 並列分散協調型ウォーゲームシミュレーション構築環境の実現と評価、古市 昌一 他、
並列処理シンポジウム JSPP'98, 1998.

[4] IEEE Std P1516 (draft), Draft Standard for Modeling and Simulation High Level Architecture
(HLA) - Federate Interface Specification", <http://www.sisostds.org/stdsdev/hla/>, July, 1998.

3.3.3 デバイス技術の向上で実現して欲しいことと PetaFlops 計算機で欲しい技術

福井義成 委員

3.3.3.1 はじめに

進展著しいデバイス技術で今後実現して欲しいこと（あるべき姿）とHPC分野で欲しい技術について述べる。また、記号処理についてもふれる。

PetaFlops 計算機を理解しやすい応用分野は昨年までに報告した。概要は下記の通りである。

(1) ミクロな原理からマクロな現象を解明する場合

原子を支配する基本法則（第一原理）だけから物質の特性を計算して求める場合などがある。原子の基本方程式を対象とする原子数に対して計算し、そこから物質としての性質を求める分野である。現在は計算機的能力から原子レベルからの計算も色々な仮定を置いて計算を進めることが多いが、それでは、仮定条件以外の答えが出てくる可能性はない。なにも仮定を置かない第一原理からの計算ではこれまで人間が考えつかなかった答えが出てくる可能性がある。この場合、扱える原子数がどれくらいかで扱える現象の範囲が決定する。多くの有用な現象は、その原子が純粋な塊（多くの場合、結晶）でどの方向へも同じ原子が無限に続いている場合ではない。また、有用な現象は物質の表面や2つの性質の異なる物質の境界面でおきる。触媒の場合は物質の表面であり、半導体では2つの性質の異なる物質の境界面である。現在の計算機的能力では、関係する原子の数が $10 \sim 100$ のオーダー程度が無理なく計算できる範囲である。1モルの分子数は 10^{23} のオーダーの原子数であり、現在は計算不可能である。全ての欲しい物質の性質を出すのには、 10^{23} のオーダーの原子数は必要はないが、現在の計算機の少なくとも 1000 倍以上の能力は必要である。

(2) 非線型3次元問題の時間応答問題

電子計算機（コンピュータ）の多くのアプリケーションは偏微分方程式を数値的に解くという観点から始まっている。偏微分方程式を離散化して解く場合、1次元から始まり、

2次元、3次元と進んでいる。今後も、詳細な結果が必要な分野では3次元の取り扱いが不可欠となる。3次元の場合、各次元の分割数を増やすと必要となるメモリーが急激に多くなる（3乗に比例）。近似モデルも線形で済んでいた（我慢していた）時代から、非線型が必要な（扱える）時代へと変化している。時間的な面も定常解で済んでいた（我慢していた）時代から、非定常解が必要な（扱える）時代へと変化している。

(3) リアルタイム性が必要な分野のシミュレーション

シミュレーション結果のリアルタイム性が重要な分野では実時間よりも速くシミュレーション結果を得て、対応策を検討・適用するため、高速な計算機が必要になる。計算結果だけを得るには現在の計算機でも可能であっても、リアルタイム性を実現するためには、より高速な計算機が必要になる。気象予測、地震予測、原子力発電所の制御などが考えられる。また、化学反応、溶鉱炉、半導体製造などで、製造段階の情報からシミュレーションを行い、結果を予測し、その後の工程を制御することにより、品質向上などが可能になる。

(4) 多くのケースの計算が必要な場合

1ケースの計算には高速な計算機は必要ではないが、シミュレーションのケース数が数千から数万ケースの場合は全体として処理能力の高い計算機が必要である。計算機的能力が低かった時代には、1ケースだけでも結果が得られることが大切であったが、計算機的能力が大きくなるにしたがって、多くのケースの計算が可能になり、研究・開発の最適化が可能になる。

3.3.3.2 デバイス技術の向上で実現して欲しいこと（信頼性の向上）

計算機の利用の変遷を考えると、最初はアセンブラあるいは機械語でプログラムを書いていた時代から、コンパイラ言語でプログラムを書く時代になり、現在では既製のアプリケーションを利用することが多くなっている。これを計算機の利用効率の面から見ると、利用率は低下の傾向にある。別の表現を使えば、計算機のハードの性能向上を利用効率だけではなく、使い易さにも振り分けてきたと言える。

近年の計算機の処理能力の向上、コストパフォーマンスの向上には目をみはるものがある。今後のデバイス技術の進展で処理能力はさらに向上するはずである。その処理能力は

前に述べた大規模計算に適用するのが1つの手段である。一方、向上した処理能力を信頼性の確保にも振り向ける必要がある。今後は人間生活のあらゆる分野に計算機が使われるようになる（コントローラ等として）。そのような状況では計算機システムの信頼性が重要になる。個人的な経験であるが、数年前、自動車を運転していて、踏み切りで突然エンジンが停止したことがあった。急いでエンジンを再起動したところ問題なく動いた。恐らく、エンジン等の機械系には問題なく、マイクロコンピュータを使った制御系の誤動作のようであった。そのことを定期点検時にメーカーに連絡したところ、エンジン制御用のコンピュータを交換してくれた。現在では、普通の自動車にも多くのマイクロコンピュータが入っているとのことである。パソコンをワープロや表計算に使っているのであれば、多少、信頼性がなくてもリカバリーができるが、人間の命に関わる場面では信頼性が重要になる。

信頼性を向上させる1つの方法は1チップに搭載できるトランジスタ数が大幅に増加しているため、1チップの上に複数のプロセッサを搭載して演算結果を多数決で決めする方法がある。システム全体の信頼性を向上させるにはボトルが無いようなシステム設計を行わなければならない。デバイス技術の向上はシステムの信頼性を向上させるためにも有効である。

また、信頼性を高めるためのソフトウェア技術の開発も重要な項目となる。

また、デバイス技術の向上により、計算機システムが安価になれば個々の機器の疲労の監視なども可能になるであろう。たとえば電車自身に疲労情報を蓄積するシステムをつけることにより、ある車両だけが疲労が増していることなどが検出可能になり、社会システムの安全性を向上させることができる。この場合も疲労情報を蓄積するシステムの部品点数が多くなることにより、このシステムの信頼性を高める必要がある。

「社会の信頼性を向上させるようなシステムは信頼性の高い計算機で構成しなければ無意味である。」

3.3.3.3 PetaFlops 計算機で欲しい技術

(1) 計算モデルや計算手順を間違いにくくする技術

計算機のユーザが問題を解決するために計算機を使う場合、モデル（プログラム）記述が容易なことが大切である。ユーザにとって、演算が速いことではなく、解きたい問題が速く解けることが大切である。PetaFlops 計算機は並列計算機と考える。現在、普通に使われている手続き型言語（Fortran、C等）はノイマン型計算機を前提にしている。プログ

ラム（モデル）はどのデータをどのように加工するかの手順を計算機に指示することにより構成されている。現在のアプリケーションも多くのはアルゴリズムもノイマン型計算機を前提にしている。

手続き型言語での並列プログラムの作成はユーザにとって負担が大きい。並列計算機では、単独プロセッサでは問題とならない複数プロセッサ間のタイミングの問題も生じる。タイミングにからむ問題は再現性が容易でないことが多く、これが手続き型言語での並列プログラミングの困難さの一因である。

並列計算を使い易くするためには、並列計算機向きのモデル記述手法・アルゴリズムがあるべきである。ユーザにとって、手続き型言語で並列プログラムを書くのは負荷が大きく、それが可能な人口は限られてしまう。ユーザにとって、モデル記述が容易で、モデルの変更も容易で、なおかつ、並列性の抽出が計算機システムにとって容易であることが望ましい。

PetaFlops 計算機的能力を十分に発揮させるためには、並列計算機向きのモデル記述手法・アルゴリズムの開発が不可欠である。

(2) データの移動の構造を記述する手段

現在の計算機で大規模な計算（特に、ランダムなデータを扱う場合）を行っている場合、最大の問題は演算ではなく、データの移動である。ノイマン型計算機ではメモリ内の番地を指定し、そのデータをメモリからプロセッサ内のレジスタに移動する。レジスタ内のデータ同士を演算し、結果を別のレジスタに格納し、その後、レジスタからメモリにデータを転送するのが一般的である。演算は最近のプロセッサの高速化により、ほとんど問題とならない。メモリとレジスタ間の転送時間が最大の問題である。これはプロセッサに比べて、メモリの遅さに原因がある。演算や分岐制御は完全にメモリとレジスタ間の転送に隠れてしまうプログラムもある。プロセッサとメモリの速度差があることを是認すれば、この速度差を避ける手段を考える必要がある。普通、よく行われる計算機アーキテクチャ的解決方法はメモリ遅延の隠蔽である。

ソフトの面から考えると現在のプログラミング環境にはデータの移動の構造を明示的に記述する手段がない。そのため、必要以上にデータ移動の回数（データを示す番地データも含めて）が増えてしまうことがある。データの移動構造を明示的に記述できれば、計算機アーキテクチャ的にメモリ遅延の隠蔽も容易になる。

これを実現するにはソフト的な開発とそれを支援するハード的な開発が必要となる。

(3) 可変データパス計算機

前項で述べたハード的な開発に利用できそうな技術がFPGAであろう。FPGAを利用したシステムでは命令体系等を可変にする研究は目にしているが、アプリケーションの性能を上げるためには、命令体系だけではなく、データパスを可変にした計算機が望まれる。可変データパスとは表現しているが、本当にデータパスを変更しないまでも、キャッシュの制御アルゴリズムやプリフェッチのアルゴリズムを指定できるだけでも非常に有効であろう。

3.3.3.4 記号処理について

記号処理の場合、浮動小数点演算は非常に少ないので、文字処理が高速に行える計算機であることが必要である。具体的にはLISPのような言語が高速に処理できる計算機である。記号処理の1分野である数式処理においては、ベクトル計算のような連続的なメモリアクセスは少なく、多くの場合、ランダムなメモリアクセスを行なわなければならない。数式処理ではランダムなメモリアクセスが高速な計算機が望ましい。

数式処理にはアルゴリズムを書いて計算するものと、「公式」を用意しておき、パターン認識で行うものがある。後者の場合、パターン処理が高速な計算機が必要になる。この点は翻訳や音声認識と共通な部分があるのではなかろうか。

大規模な科学技術計算に戻って考えると、数値的処理だけでは収束等に非常に時間のかかる計算も数式処理を上手く組み合わせることにより、収束性などを大幅に改善できる場合もある^{1), 2)}。

3.3.3.5 まとめ

PetaFlops 計算機を使用するであろうユーザが本質的（直接的）に欲しいものは下記のようなものである。

- (1) 速く正しい結果の得られる道具
- (2) 問題の処理過程全体が速くなること

これを実現するためには、PetaFlops 計算機のような速い計算機とモデル（プログラム）記述手法の革新が必要である。高度な計算（情報処理）には数値的処理だけではなく、記

号処理のような非数値処理の高速化・高度化も重要である。

参考文献

1. Lecture Notes in Computer Science Vol.378 (Ed.)J.H.Davenport, Springer-Verlga 1989. pp.163-171. M.Suzuki, T.Sasaki, M.Sato and Y.Fukui : A Hybrid Algebraic-Numeric System ANS and Its Preliminary Implementation.
2. Journal of INFORMATION PROCESSING, Vol.13, No.4, pp.529-533, 1990. T.Sasaki, Y.Fukui, M.Suzuki and M.Sato: Proposal of a Scheme for LInking Different Computer Languages - from the Viewpoint of Algebraic-Numeric Computation - .

3.3.4 大規模科学技術計算のための超高速計算機システム

横川三津夫 委員

ペタフロップスマシン技術に関する調査研究も3年目が終了しつつある。筆者は、前回の報告書において、並列型の形態を取らざるを得ないペタフロップスマシンは大規模科学技術計算においてこそ重要であり、一般に普及される必要はなく特殊な実験装置として特定の研究者、技術者に提供されるべきであるとの主張を述べた。したがって、ペタフロップスマシンの一般的需要が見込めないことから、商業ベースの開発によって達成することが極めて困難であり、日本の科学及び産業技術の基盤技術を維持、向上させるためにもペタフロップスマシンの開発は国がリードしていくべきであると考えている。産業界へのインパクトは、ペタフロップスマシンの開発に伴う波及効果として捉えるべきであろう。

大規模科学技術計算においても、MPPが有効であるアプリケーションは当然存在することから、ペタフロップスマシンの形態はベクトル並列とスカラ並列の両方が協調して動作するハイブリッド型が良いと筆者は考えている。しかし、本稿では、最近コストパフォーマンスの低いベクトル計算機の重要性の認識が低下していることから、ベクトル計算機に対するいくつかの意見を概観するとともに、地球変動研究のために開発されているベクトルプロセッサを採用した超高速並列計算機システム「地球シミュレータ」の概要を述べることにする。

3.3.4.1 ハイエンドコンピューティングを巡る日米欧の状況

ハイエンドコンピューティング技術は、国家の安全保障上極めて重要であると認識されており、東西冷戦時において米国はベクトル型計算機維持のために豊富な資金を投入した。しかし、冷戦終結とともに、コストパフォーマンスのよいコモディティ・プロセッサ等商用品をハイエンドマシンに利用するASCI計画が進められており、米国のスーパーコンピュータ市場は大きな方針転換がなされたと言われている[1]。LINPACKベンチマーク性能に基づく計算機のランク付けであるTOP500においても、米国のSMPやMPPが上位を独占するようになってきた[2]。しかし、産業用の実アプリケーションコードでは、LINPACKのようにうまく並列化できないことが次第に明らかとなり、数千台の並列計算機のスケラビリティを生かす並列プログラムの開発が極めて困難であることが認識されている。実際、大統領の諮問委員会であるPITACの報告書においても、スケラブルな並列アーキテクチャが大規模アプリケーションに効果的に対応できないことが述べられており、革新的なアーキテクチャの開発が必要であると結論づけている[3]。

ASCI計画に投入した資金とその結果を見れば、米国の開発路線は誤った方向に向いていたと言って良い。

また、欧州においては、すでに数千個のプロセッサから構成されたMPPは、HPCアプリケーションでは役に立たないとの意見が述べられている[4]。すなわち、日常的に利用する多くの並列アプリケーションプログラムは、高々32並列までがスケーラブルな性能が得られる範囲であり、特に構造解析などの大規模な既存の産業用コードは32並列以上の並列化は不可能である。したがって、並列台数は少ないが単体プロセッサの性能が高い並列ベクトル計算機の方がHPC分野には適合しており、高い実効性能を達成することが可能である。SMPやMPPは、いわゆるスループットマシンとしての位置づけに留まるであろうとの考えが示されている。

一方、アプリケーション分野の動きを見ると、地球温暖化防止に関する京都会議を契機に地球環境問題が広く取り上げられており、地球変動に関する数値シミュレーション分野が世界的にも注目されている。科学技術庁においては、地球規模の複雑な現象を解明することが重要であるという認識の下に、地球変動プロセスに関する基礎研究、地球規模の観測、数値シミュレーションの三位一体による研究体制を推進している。このうち、数値シミュレーション分野の推進では、地球規模の複雑な諸現象をシミュレートするための超高速並列計算機システム「地球シミュレータ」の開発と高度なアプリケーション・ソフトウェアの開発を目標とする「地球シミュレータ」計画が策定された[5]。

気象・気候分野のシミュレーションでは、並列ベクトル計算機とキャッシュベースのマイクロプロセッサを要素計算機とするMPPでの実効性能を比較した結果、並列ベクトル計算機の並列化効率が格段に高いことが明らかとなった。従って、気象・気候分野のシミュレーションを高速化することを目的とする地球シミュレータのハードウェア構成では、ベクトル計算機を要素計算機とする並列計算機を採用している[6]。

この計画に刺激を受けた米国は、DOEの新しいプロジェクトであるACPI (Accelerated Climate Prediction Initiative) 計画を検討しており、気候変動シミュレーション研究を推進する報告書が提出された[7]。報告書では、気候モデリング分野における米国の計算機資源が他の国と比較してかなり後退してしまったとの危機感が述べられているが、計画している40テラフロップス計算機はASCI計画で得られた成果を基に開発するとしており、この計画でベクトル計算機を採用することはなさそうである。しかし、連続体モデルに基づく気象・気候分野のシミュレーションでは、単体プロセッサとしてのベクトル計算機の優位性は明らかであり、米国がACPI計画においてどういう新規のアーキテクチャを採用するかは注目に値する。

3.3.4.2 地球シミュレータの概要

超高速並列計算機「地球シミュレータ」は、ベクトルプロセッサを要素計算機とした分散主記憶型並列計算機であり、その基本設計が終了した。ここでは、基本設計に基づく地球シミュレータの概要について述べる。ただし、ここで述べるハードウェアの構成は、今後の開発の進展により変更される可能性があることをご承知願いたい。

地球シミュレータは、640 台の計算ノードをクロスバネットワークで結合させた分散メモリ型並列計算機である。計算ノード (PN: Processor Node) は、ピーク性能 8Gflop/s のベクトル型計算プロセッサ (AP: Arithmetic Processor) 8 台が主記憶装置 16GB を共有する共有メモリ型並列計算機となっている。したがって、全体では AP が 5120 台、ピーク性能は 40Tflop/s、主記憶容量 10TB となる (図1)。

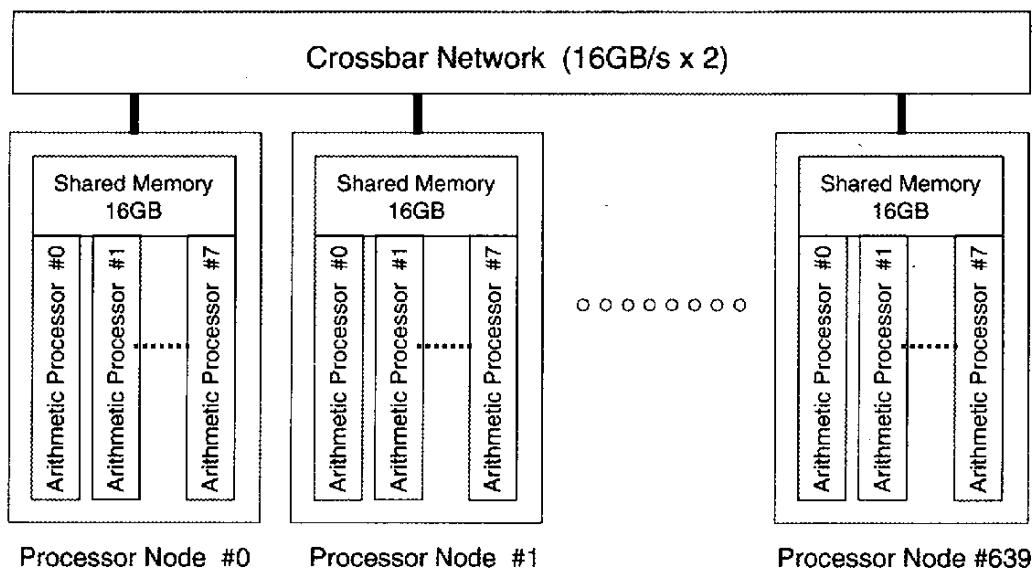


図1 地球シミュレータの全体構成

各 PN は、8 台の AP、32 台の主記憶ユニット (MMU: Main Memory Unit)、リモート制御装置 (RCU: Remote Control Unit) 及び入出力プロセッサ (IOP: I/O Processor) から構成されている (図2)。

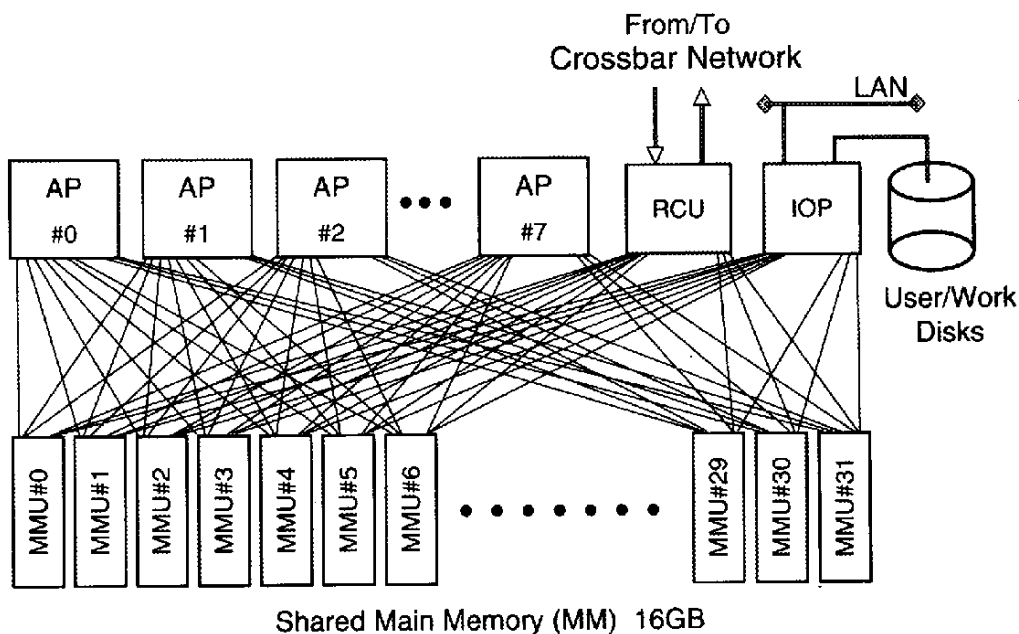


図2 計算ノードの構成

AP は、ベクトル処理部 (VU: Vector Unit)、スカラー処理部 (SU: Scalar Unit) 等が 1 つの LSI 上に実装され、主要クロック周波数 500MHz で動作する。SU は、4 ウェイのスーパースカラであり、128 個の汎用レジスタ、2 ウェイセットアソシアティブ方式の命令キャッシュとデータキャッシュをそれぞれ 64KB ずつ実装している。VU は、6 種類 (加算、乗算、除算、論理、ビット列論理、ロード/ストア) のベクトル演算器と 72 個のベクトルレジスタからなるベクトル演算器セット 8 個で構成され、最大 8Gflop/s の性能を有する。32 台の MMU には、主記憶素子として 128Mbit の高速 RAM を採用し、2048 バンク構成を取っている。各々の AP は、主記憶システムとの間に 32GB/s のバンド幅を持っており、1PN で 256GB/s を確保している。

RCU はクロスバネットワークと直接接続され、クロスバを介した送信、受信を AP と独立に動作させることができる。クロスバネットワーク (IN: Internode Network) は、実際のデータが転送されるデータバス部 (XSW: Internode Crossbar Switch) と、PN 間の転送経路の予約、競合調停等を行う制御部 (XCT: Internode Crossbar Control Unit) から構成されている。IN 及び RCU は、PN 間データ転送機能、PN 間同期等の機能を持つ。データ転送機能では、同期型転送と非同期型転送がサポートされており、主記憶上の連続した領域のデータを転送するブロック転送の他に、非同期型転送としてストライド付きベクトル転

送、リストベクトル転送が可能である。データ転送のスループットはハードウェアの最大性能で送信、受信とも 16GB/s である。

地球シミュレータのシステムソフトウェアは、既存の Unix 系オペレーティングシステム (OS) をベースに地球シミュレータ全体を管理、制御するための種々の機能を実現する予定である。特に、大規模分散並列処理に対応するために、高速ファイルアクセスが可能なファイルシステムを導入するとともに、並列ファイル入出力機能を提供する予定である。地球シミュレータ上の並列プログラミングでは、AP 内ベクトル処理、PN 内並列処理 (共有メモリ型) 及び PN 間並列処理 (分散メモリ型) が可能な 3 階層の並列実行機能を有効に活用できる環境を用意する予定である。基本的なプログラミングスタイルとして、Fortran90 または C をベースにメッセージパッシングライブラリ MPI2 を利用した並列処理形態を用意する。また、ユーザの利便性と従来のプログラム資産を考慮し、HPF2 を拡張した言語体系も用意する予定である。

地球シミュレータは、平成 13 年度内の完成を目標に本体製作に着手することになっており、ベクトル型計算機の性能を再考する上でも重要な試金石となる。過去、大規模科学技術計算分野においては、汎用コンピュータがベクトル計算機に置き代わっていったのと同じように、並列計算の技術が習熟した後は、ベクトル計算機を要素計算機とした並列ベクトル計算機へと移行していくのが自然な流れと思われる。しかも、同じ性能を達成するための並列台数が少ないため、ソフトウェア開発は MPP と比較して格段に容易であることが予想される。米国においてはベクトル計算機技術を再構築するのが極めて難しい状況にある中で、日本においてペタフロップスマシンがベクトル計算機の技術で達成されれば、十分なインパクトを持つであろう。

3.3.4.3 おわりに

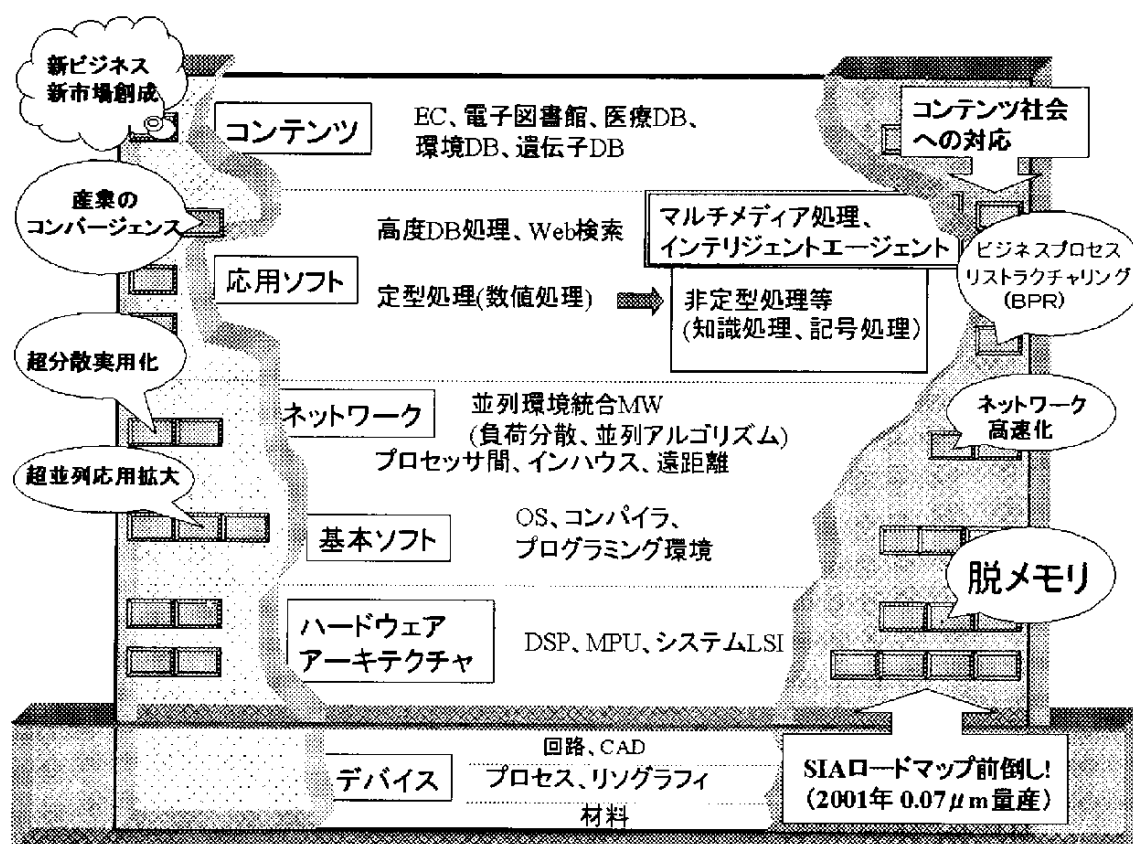
ペタフロップスマシンの技術調査も 3 期めが過ぎ、すでに調査に関する意欲が失われつつあるように思う。ペタフロップスマシン開発に向けた実際の動きが少しでも前向きであればさらに調査課題も絞り込めるものと思われるが、その動きすら感じられない状況において何を議論すべきであろう。米国では、MPP だけを利用するという極端に片寄った政策を取っておりその成果が疑わしくはあるが、少なくとも 100 テラフロップスマシン開発を念頭においたロードマップを描いており、ペタフロップスマシン開発にも意欲的である気がする。一方、日本においてはペタフロップスマシンに向けた産・官・学の体制さえまとまっておらず、今後のハイエンドマシンの開発について非常に危惧している。本報告書によって、ペタフロップスマシン開発の機運がさらに高まることを期待するものである。

参考文献

- [1] 宍戸、"アメリカの方針転換で変わるスパコン市場"、情報処理、Vol.40, No.3 (1999).
- [2] <http://phase.etl.go.jp/TOP500>
- [3] "Information Technology research: Investing in Our Future," National Coordination Office for Computing, Information, and Communications, February (1999).
- [4] "Europe can take the lead again in supercomputing – vectors can be the answer", <http://www.hoise.com/primeur/99/articles/live/AE-PR-01-99-5.html>.
- [5] 「地球シミュレータ」計画の推進について、科学技術庁計算科学技術推進会議、平成9年7月.
- [6] 横川他、"地球シミュレータ用性能評価システムの開発、" 情処研報、Vol.99, No.21 (1999).
- [7] "Capacity of U.S. Climate Modeling to support Climate Change Assessment Activities," National Academy Press, Washington, D.C. (1998).

第4章 あとがき

今年度の本ワーキンググループの本報告書は、昨年度と同様に、ペタフロップスという超高性能計算機を視点に入れながら近未来の情報処理産業を見据えてどのような技術開発を行うべきかについてヒアリングや議論を行い、まとめたものである。今年度は、ワーキンググループをスタートするにあたって、下記の図に示すように、予め情報処理技術に関する技術階層を提示し、各階層において、技術的な重要課題はどのようなものであるかについて議論を行うことによって、より問題点が明らかになるように努めた。



本報告書における各委員の見解については、例年と同様に、各委員独自の見解を表明してもらっているため、全体として必ずしも整合性がとられているわけではないのでその点はご了承願いたい。また、これも昨年と同様に、意見の整理のために各委員の見解をアーキテクチャ関係、ソフトウェア関係、アプリケーション関係という3つに分類して掲載した。ただし、委員の意見はそのような分野にとどまらず、もっと幅広く情報処理技術全般に渡っている。しかも、各委員の意見はかなり集約

されてきているように思われる。

ここでは、各委員の見解をもとに、私見を交えて技術課題をまとめてみることにする。最初に、我が国における情報処理産業振興のためのプロジェクトの形態について考えてみる。まず、従来型のプロジェクトのように、フラグシップとなる大きな目標を立て、これに向けてハードウェアからアプリケーションまで、技術開発を行うことが考えられる。この場合の目標としては、長期的な視野に立った基礎技術の開発が適切であろう。これには、環境問題など全世界的な問題の解決を高度な情報処理によって支援するなどといったテーマが挙げられる。次に、「情報インフラ」の整備などをからめて、これからの高齢化社会を情報化によって支える技術や一般的な個人に対して「情報化社会」の恩恵を与えるための技術の開発などが考えられる。またもう1つ別の目標としては、情報処理技術のいくつかの階層に関して、我が国の技術力を高めるべき課題を設定して、それらの技術開発を支援する方向が考えられる。現在の情報処理技術は種々の階層の連携が重要になってきているので、このような形態のプロジェクトの場合においては、情報処理技術におけるいくつかの基礎技術の連携を密にし、それらの相互作用によって新たな技術の展開を図るようなプロジェクト形態を構築する必要がある。ここでは、特に後者に関して、各委員の報告を参考にして考えてみることにする。以下では、デバイス技術の階層から情報処理の技術開発課題を考えた場合と、逆にアプリケーションの階層から考えた場合との2つのケースについてまとめてみる。

[1] ボトムアップに捉えた場合

数年前までは、半導体の製造技術については、最先端のメモリ技術を先導してきた日本が米国をリードしてきたと言えるが、現時点では逆に、微細加工技術などにおいても米国のプロセッサメーカーに先を越されているという観測がある。さらに悪い状況として、日本経済全体の景気の低迷やメモリ業界の先行きの不安などから、民間の設備投資が落ち込んでいることがある。近い将来の微細加工のための製造技術に関する投資は莫大になることが予想されるため、国などのサポートによる組織的な開発計画や資金的なバックアップが必要となってきた。その場合には、世界的な企業間の競争において、これらの最先端の微細加工技術をどこに適用すべきかという点も考慮しておかなければならない。我が国の状況を考えると、これからは「脱メモリ」を主体的に考えなければならないという点が、衆目の一致するところである。現在の「メモリ」に代わるものとして、産業的および社会的な状況から考えて現時点でのマイクロプロセッサ産業への参入が困難であることを考慮する

と、以下のようなものが考えられる。

- ① システム LSI
- ② シングルチップ・マルチプロセッサおよびそれらを利用した並列システム
- ③ 丸書き換え可能なプロセッサや VLSI アルゴリズムに基づいたアーキテクチャ

システム LSI に関しては、組み込み用の LSI としてシステムオンチップ、メモリ混載プロセッサなど、我が国の技術力が発揮できる分野であり、この技術開発への支援は利益が大きい。具体的な技術開発の項目として、メモリ混載技術などのデバイス製造技術はもちろんであるが、さらに重要なことは、LSI 設計用 CAD 技術の高機能化を図るとともに、LSI の部品となる様々な IP に関して、その品種を揃えるとともに標準化などに積極的に関わっていくことが重要である。システム LSI に対するもう 1 つの需要としては、たとえば家庭用テレビゲーム機用のプロセッサなどがあげられる。このようなデバイスでは、汎用のプロセッサをコアとしながらもそれに様々な特殊機能を付加することによって、ハイエンド計算機並あるいはそれ以上の性能を持たせることが可能となる。将来的には、このような機器が、現在のパーソナルコンピュータに対抗するあるいは置き換わるような勢力を持つことも考えられるため、このようなコンパクトかつ安価な情報処理機器の開発に対しても国の政策としても何らかの方向性を示しておくことが必要である。

次の、シングルチップ・マルチプロセッサの分野は未だ未開拓であり、その研究開発に早急に取り組むことによって、21 世紀のプロセッサ産業の勢力分野において、我が国がマイクロプロセッサ産業が現在とは異なるシェアを確保できる可能性が残されている。さらに、この分野は将来の並列処理計算機の基幹となるマイクロプロセッサとしても重要な位置を占めると考えられる。21 世紀において、国際的に競争力のあるシングルチップ・マルチプロセッサを産み出すためには、まずハードウェア製造技術としてさらなる微細加工などのデバイスの技術開発が必要である。さらに、このようなデバイス技術による大量の論理素子やメモリを生かすためのアーキテクチャレベルでの独自性に加えて、並列化されたハードウェアアーキテクチャを生かすための並列処理コンパイラ技術との協調がより重要になってくる。したがって、アーキテクチャの開発と自動並列化コンパイラ技術などとを一体化した研究開発、あるいは計算状況の予測（メタレベル計算）などを効率的に処理するハードウェア機構を含んだハードウェア／ソフトウェア協調などの研究開発が必要である。

また、高性能処理を実現するための手段として並列処理を導入することがより一般的になっていく状況において、ソフトウェア技術として並列処理の実効性能を高めることが最重要課題であるという点で各委員の意見は概ね一致している。ただ、そのために必要とされる技術として、

- ・並列プログラミングツールの開発や標準化の重視
- ・自動並列化コンパイラの研究開発を重視

と多少意見が異なっている部分もある。共に必要な技術であると考えられるが、前者に関しては必ずしも国家プロジェクトのような大規模な研究開発ではなく、個々の企業での研究開発を情報交換や標準化のための組織立てがあれば良いと考えられる。したがって、私見では、後者の自動並列化コンパイラに関して、より大規模で英知を結集するようなプロジェクトが必要であると考えられる。

3 番目の書き換え可能な論理デバイスを用いた reconfigurable なプロセッサの研究開発や VLSI アルゴリズムに基づいたアーキテクチャの研究開発は、現在のプロセッサ構成やそれを前提にしたコンパイラ技術などとは異なるアーキテクチャやソフトウェア技術を産み出す可能性がある。しかしながら、その応用分野が未知数である点や VLSI アルゴリズムやそのソフトウェア技術、さらには新たな論理合成手法の開発など、幅広い領域における基礎研究が必要であるなどの点から見て、現時点で大きなプロジェクトの柱としてうち立てることは難しい。しかしながら、21 世紀の新技术を開拓するためには、何らかの形でそのような基礎研究を活性化させることが重要である。

[2] トップダウンに捉えた場合

アプリケーションの方面から情報処理技術を捉えた場合には、高性能計算機のあるべき姿と同時に現在あるいは将来の高度なネットワーク社会における情報処理の姿を考える必要がある。

まず、高性能計算機のアプリケーションに関しては、大規模な数値計算を主体にした「シミュレーション」を重視するべきであるという意見と非数値的な計算を重視すべきであるという意見の2つに集約されている。

前者の例としては、ミクロな原理からマクロな現象を解明するアプリケーション、各種の物理現象のシミュレーション（地球変動に関する数値シミュレーション、気象予測、地震予測など）、さらには、自動車開発、医薬品開発、新材料開発、半導体製造などで製品開発の短縮につながり競争力を高めるものとなる各種のシミュレーション（たとえば、計算機による分子動力学シミュレーションによるたんぱく

質構造予測技術)などが挙げられる。数値計算の応用をサポートするための計算機システムについて考えてみると、汎用のマイクロプロセッサをベースにしたいいわゆる汎用(超)並列計算機とベクトルプロセッサを主体にしたベクトル並列計算機に大別される。最近では、汎用のSMP計算機を高速ネットワークで接続したクラスタ計算機をハイパフォーマンス計算機として使用するための研究が進んでおり、将来的にもこのような傾向は強まると思われるが、このような計算機も前者の分類に含まれる。米国では、主として汎用(超)並列計算機システムがハイパフォーマンス計算を行うために用いられており、価格性能比が高いだけではなく、いくつかのベンチマークテストなどにおいては、高い性能を引き出すことが実証されている。このタイプの計算機システムにおいては、コンパイラ技術や最適化技術、あるいは並列化のためのプログラミングツールなどの果たす重要性が高い。このことによって、米国では並列化のためのソフトウェア技術や並列化されたプログラムのライブラリなどが蓄積されてきており、わが国はこの点で遅れをとっているのではないだろうか。ただし、すべてのプログラムやアルゴリズムが効率よく並列化されるというわけではなく、プロセッサ数が多くなるに従って、むしろ効率的な並列化ができないアプリケーションが一般的になってきているという事実もユーザの間で認識されているとのことである。これに対して、従来から我が国のスーパーコンピュータの主流であった並列ベクトル計算機は、超並列計算機に比べて並列化の効率が高いという利点がある。また、同じ性能を達成するための並列台数が少ないためにソフトウェアの開発も容易である。このようなことから、気象・気候分野のシミュレーションを高速化することを目的とする「地球シミュレータプロジェクト」では、ベクトル計算機を要素計算機とする並列計算機が採用されている。米国ではベクトル計算機の技術開発がストップし、それを再構築することが困難な状況にある中で、我が国でのこのようなプロジェクトによってベクトル型計算機の技術が継承されていくことは意義がある。しかしながら、情報処理産業を全般的に捉えたと、いわゆるスーパーコンピュータに対する需要が相対的に低下する傾向は防ぎようがないため、我が国としても、より汎用的な技術である並列化ソフトウェア技術に関して積極的な支援を行う必要がある。

後者の非数値的な計算の例としては、データベース処理やデータマイニングなどのいわゆるデータインテンシティブな応用、情報セキュリティ、暗号解読などが挙げられる。現在および数年先の情報処理産業の状況を考えた時には、ビジネスにおける情報処理ニーズとして数値計算などに基づくシミュレーションよりもデータインテンシティブな応用の方がマーケットとしてさらに拡大していくことが予想

される。このような状況を勘案すると、ハードウェア的には高性能なデータベースサーバや大容量記憶装置、あるいは暗号処理の高速化のための専用デバイスの開発などがあり、ソフトウェア的には、並列計算機や並列データベースサーバを効率よく処理するための並列ソフトウェア技術やこのようなシステムを誰でも簡単に使用できるようにするためのユーザインタフェース技術などの開発が重要になるであろう。

次に、将来の高度ネットワーク社会における情報処理技術について応用面から考えてみると、以下の点が指摘できる。まず、このようなネットワークを一般のユーザが容易に利用できるようにするためのいわゆる「情報インフラ」の整備があげられる。早急に、全国的な規模で高速なデジタルネットワーク網を整備する必要があるであろう。さらに、このような高速ネットワークを利用したサービスや処理形態を開拓する必要がある。高速ネットワークを利用して、複数の計算資源を組み合わせ、大規模な計算処理を行うことのできるグローバルコンピューティング、電子商取引、マルチメディア情報の配信などに関するソフトウェア技術が必要とされている。これらを含めて、具体的な応用形態などを想定して、様々な側面から研究開発を行うプロジェクトの立案が必要であろう。

最後に、我が国における情報処理分野における全般的な研究開発の政策に関してまとめておく。以下の点は、昨年の報告書においても記述したことではあるが、情報処理技術分野において共通した仕様や共通のインタフェースの抽出とそれをいかに効率よくインプリメントするかといった戦略の必要性があるのではないだろうか。21世紀に向かって、戦略的な技術開発がより重要になると考えられる。米国では、情報処理の分野が非常に活気があり米国の経済の牽引車となっている感があるが、その米国においてさえも、情報処理技術開発をさらに積極的に進めるべきであるという大統領に対する提言がまとめられている。これは、情報技術に関する大統領諮問委員会 (PITAC: President's Information Technology Advisory Committee) の提言である。この委員会では、1998年の8月に中間報告書が提出され、最終報告書が1999年2月24日に大統領に提出された。PITACは、Rice大学のKen KennedyとサンマイクロシステムズのBill Joyの2人が委員長を務め、その構成委員として、HPC関連の企業 (SGI, Intel, IBM)、ネットワーク関連の企業 (3Com, MCI, AT&T, Qualcomm)、ソフトウェア関連企業 (Microsoft, PointCast, Walt Disney Imagineering)、ベンチャーキャピタル (CenterPoint) などから成っている委員会である。ここでの提言のポイントは、GDPで30%を占めるIT (情報産業) には、政

府予算の75分の1の研究開発予算しかついておらず不十分であり、これを5年間で倍増すべきであるという指摘である。しかも、その投資対象として現在の予算がショートレンジのR&Dにつき込まれていることを批判し、よりロングレンジでリスクの大きな研究開発に対して優先的に予算を割り当てるべきであると主張している。そして、具体的な研究分野として、「ソフトウェア」、「インターネットに代表される大規模ネットワークなどのインフラストラクチャ」、「ハイエンドコンピューティング」を挙げて、それぞれの分野における問題点と忠告を指摘し、各分野に今後5年間、どのように予算を割り当てるべきかについても言及している。これらの詳細なレポートは、<http://www.ccic.gov/ac/report/>より入手できる。

米国は、情報産業における short term の技術においては、かなりの成功を収めている。したがって、PITAC の主張は、short term の技術については民間企業やベンチャーに任せて、政府としては、もっと long term でリスクの大きな技術開発を行うべきであるとも捉えられる。我が国としては、米国のこれらの動向を見ながら、独自の立場で我が国としての情報処理技術に対する戦略を立てるべき時に来ているのではないだろうか。これからの我が国の施策としては、個々の企業の機動性を高め、競争によってポテンシャルを高めることを可能とする施策を行うとともに、政府としては日本全体の先端技術、特に情報処理技術の競争力を高めるために必要となる施策を早急に講ずるべきである。そのことによって、我々が当初描いていたベタフロップスマシンという超高性能計算機に向かう道がよりはっきりと見えるようになるはずである。

(山口喜教主査)

付属資料1 SC98 参加報告

電子技術総合研究所 高木浩光

フロリダ州 Orlando の OCCC (Orange County Convention Center) で11月9日から13日まで開催された、国際会議「SC98」に参加したので、その報告を行う。また、続けてバージニア大学の Legion グループを訪問し、グローバルコンピューティングシステムについて調査を行ってきたので、その報告も行う。

1. SC98

この会議は、かつて「Supercomputing'XX」と呼ばれていたものが、'97年より、名称を「High Performance Networking and Computing Conference」と改めたものである。Supercomputing'XX のころより通称として「SCXX」と呼ばれていたことから、今回も通称として「SC98」と呼ばれている。

SC98 は今回で、第11回目であるが、10周年にあたり、第1回の開かれた同じ Orlando で開催された。これまでの開催都市、展示を含めた参加者数を以下に示す。

		開催都市	参加者数(人)
1回	1988年	Orlando	1495
2回	1989年	Reno	1926
3回	1990年	New York	2303
4回	1991年	Albuquerque	4442
5回	1992年	Minneapolis	4636
6回	1993年	Portland	5196
7回	1994年	Washington	5822
8回	1995年	San Diego	5772
9回	1996年	Pittsburgh	4682
10回	1997年	San Jose	5436
11回	1998年	Orlando	5750

今後の開催地の予定は

12回	1999年	Portland
13回	2000年	Dallas

とのことである。これらについては、SC Conference Series のホームページ

<http://www.supercomp.org/>

で確認することができる。

今年の参加者は、technical program への有料登録者は 2,025 人、展示だけの入場者を含めると 5,750 人とのことである。前々回やや少なかったのを除けば、ここ 6 年間はほぼ横ばいといったところであろう。ホームページには、開示を認めた参加者 3,200 人の名簿が掲載されているが、日本からの参加者（日本の研究機関所属または日本名の企業関係者）は約 200 人であった。

会議は、日曜から月曜まで、チュートリアルと Educational Program が開かれた後、月曜夜の「Gala Openings」で展示会場が公開されて始まった。そして火曜から木曜までの 3 日間にわたって technical program や招待講演、パネル、展示、Exhibitor's Forum などのセッションが平行して開かれ、最終日の金曜の午前にパネルセッションが開かれて終わった。

筆者ら、電総研のグループは、前回に続き、研究展示(Research Exhibits)としてブースを出した。今回は十分な広さを確保して本格的な出展となった。筆者は、ブースの設営とブースでのプレゼンテーションに追われ、多くのセッションを見ることができなかった。以下は、主に展示開場にて調査した情報と、最終日のパネルセッションの内容を元に報告する。

1.1 展示

大規模な展示はこの会議の特色の一つとなっている。今年も広大な会場に多くの企業の展示が派手に設営されていた。今年は、10 のハードウェアベンダー (Compaq/Digital, Dell, Fujitsu, HP/Convex, Hitachi, IBM, NEC/HNSX, SGI, Sun Microsystems, Tera) を始めとして、ソフトウェア、大容量記憶装置、ネットワーク、出版、学会など 69 の企業等が出展した。最近になって会議の名前に「Networking」の文字が含まれたことから想像されるように、ネットワーク関連の展示が増えてきている。

大学・研究所の展示も、年々盛んになってきており、今年は 56 の展示があった。NASA、ANL、ASCI などの大研究所は例年のようにベンダー並みの展示を出していた。

日本からは、電子技術総合研究所、原子力研究所計算科学推進センター、航技研、RWCP、RIST、九州大学、埼玉大学、筑波大学の 8 つのブースの出展があった。

特に今回、電子技術総合研究所と筑波大学とは、「Tsukuba HPC Alliance」と称して、連続した区画を確保し、共同でセミナースペースを作り、関係者やゲストのセミナー

を定期的に行った。外部からの講演者として、J. Dongarra 教授、R. Eigenmann 教授、大阪大学の門林助手らが招かれた。筑波大学は、PAX の 20 年の歴史、CP-PACS のアーキテクチャ、応用などのパネルを示し、日本にある CP-PACS とネットワークで繋いで、動作モニターとデモ計算を表示していた。電総研は、Ninf と Global Computing Testbed を展示し、日本におけるグローバルコンピューティング研究のアクティビティを世界に印象づけることができた。

RWCP (新情報処理研究組合) は、128 台の PC クラスタや 32 台の Alpha クラスタを持ち込んで、企業並みの規模の展示をしていた。九州大学は、PPRAM について展示、応用の一つとして、分子軌道計算エンジンの模型を出展していた。埼玉大学は毎年 SC に参加しており、VRML などを用いた可視化による教育をテーマにしていた。

日本から以外の展示に目を向けると、応用に関する展示が多かったように思われる。一部には、「どう high-performance computing に関係するの?」と首を傾げたくなるものもあるくらいだった。一方、基礎的な研究に目を向けると、そのほとんどが、グローバルコンピューティングあるいは Java 関連のものだったように思う。もはや昔の「Supercomputing」というイメージではなかった。

そのグローバルコンピューティング関連の最大の展示といえば、iGRID (The International Grid) である。これはイリノイ大学シカゴ校とインディアナ大学との共同で、vBNS の高速ネットワークを活用して、世界中の様々な分野の研究者と、広域分散処理、バーチャルリアリティ、大型データベースなどの技術を使って共同研究を行うプロジェクトである。

これは、昨年の SC97 でデモが行われた「GUSTO」と呼ばれたグローバルコンピューティングテストベッドを発展させて、国際的なものにしたものと言える。アメリカ国内だけでなく、カナダ、シンガポール、日本ともつながっている。「GRID」とは、広域ネットワークを介して組織間で互いの計算機を利用しあうメタコンピューティングのためのインフラの総称であるが、同様の目的を持つ研究機関の集会として「GRID meeting」が、この年の 7 月にシカゴで開かれた国際会議「HPDC」(High Performance Distributed Computing)と併設して開催されている。今回の「iGRID」も同様に、いくつかのプロジェクトが共同でブースを出したものと言える。

この iGRID には日本からも参加している。日本とは今回の展示に間に合わせるかたちで、TransPAC (Asia-Pacific Advanced Network Consortium) のネットワーク (70Mbps) で結ばれた。日本との共同のデモとしては、大阪大学のグループにより、大阪大学に設置されている 3MV ultra-high voltage 電子顕微鏡を SC 会場からリモート操作する

デモが行われた。また、慶応大学 SFC の村井純助教授が、デジタルビデオのインターネット経由の伝送技術のデモとして、慶応大学のキャンパスに対して「情報処理系論」の授業を展示会場から行った。

グローバルコンピューティング関連で他には、バージニア大学の Legion のグループが比較的広い区画を確保してデモを行っていた。Legion は既に実用の域に達しているようで、企業から製品化の話も出ているとのことである。

他には、今回、Java 関連の研究の展示が目立った。Emory University の IceT のグループ、UC Berkeley の Ninja プロジェクトなどである。

IceT は、Java を用いた分散計算環境のひとつである。Java を用いた分散計算環境の研究はじつに多数存在するが、IceT の特長は、JNI (Java Native Interface) を活用している点である。すなわち、計算ルーチンそのものは、ネイティブコード(各 CPU アーキテクチャ用バイナリコード)で作成するものとしており、計算のプログラムは Java で書かれるわけではない。多くの Java による分散計算研究が、マルチプラットフォーム性を活用するために Java を採用しているのに対してである。ならばいったい何のために Java を使うのか? 以前よりこの疑問を持っていた筆者は展示ブースでこの疑問をおつけてみた。回答はこうだった。まず、Java による数値計算は現状では実行速度の点で難がある。したがって計算ルーチンはネイティブコードとする。Java を使うのは、分散計算環境を実現するためのジョブ発行や負荷分散といったシステムのコア部分に対してであり、これは Java のプログラミングのしやすさからという点と、JNI を様々な言語とのインターフェイスとして利用しているとのことだった。なるほど、たしかに JNI を多言語との汎用的なインターフェイスとして使うというアイデアはおもしろい、他にも活用できそうだと感想を持った。ここで、セキュリティの確保はどうするのかと尋ねたところ、それは今後の課題であると、現在は解決法を持っていない様子だった。

UC Berkeley のブースの近くを通りかかったところ、「系列」という日本語の文字が目をついた。これは何か? と尋ねたところ、Ninja プロジェクトのデモのひとつであるとのことだった。「Ninja」は、Matt Welsh 氏が開発を進めている Java による分散コンピューティング環境に関する様々なツール群である。そのコアとなるのは「NinjaRMI」という、Java RMI と同一の機能を一から実装しなおしたものである。なぜ実装し直しているのかと尋ねたところ、Sun の RMI は自由に改造できないので、自由に改良できるものを持っておきたかったからとのことである。その上、RMI との互

換性を維持していきたいとのことだった。これをベースにして、「iSpace」と「MultiSpace」という、分散環境を構築中とのことだった。その内容をきくと、Sun の JavaSpaces がやろうとしていることにたいへん近いようだった。JavaSpaces との違いは?と尋ねると、プログラミングモデルに工夫があり、「Path」という、データフローモデル的なアプローチをベースとしているとのことであった。なお、「系列」というネーミングは変だと伝えておいた。

1.2 パネルセッション

最終日(金曜)の昼で SC98 は終幕となったのだが、その午前中を通して Java に関するパネルセッションが設けられていた。そのタイトルは「Java Grande I: Rationale, Status and the Forum」と「Java Grande II: Issues and Futures」であった。同時に開かれている他のセッションに比べて聴衆の数は多かったとのことで、Java による高性能計算に関する関心の高さがうかがわれた。

Java Grande は、1998 年 3 月に Stanford 大学で開かれたワークショップ「Java for High Performance Networking and Computing」の席で、Syracuse 大学の Geoffery C. Fox 教授が音頭をとって始められた、Java を高性能計算に向けたものとするための様々な問題を解決していくことを目的としたフォーラムである。これまで、2 つのワーキンググループ、「Numerics」と「Concurrency and Applications」に別れて議論が行われてきており、今回のパネルセッションの前半では、その成果報告が行われた。

Numerics ワーキンググループでは、Java を数値計算に利用する際に問題となる 5 つの点を指摘している。それは、(1)複素数計算を実現しようとしたときに十分な性能が得られない点、(2)クラスをまたがった最適化が簡単ではない現状ではより高速化の簡単な「Lightweight」が必要だとする点、(3)基本型以外の値に対する演算を(メソッド呼び出しではなく)演算子によって記述できるためにオペレータオーバーローディングをサポートすべきだとする点、(4)浮動小数点演算ハードウェアの性能を完全に活かすことができるように精度に関する言語仕様を緩和するべきだとする点、(5)現状の Java の仕様である「配列の配列」によるものではない真の多次元配列が必要だとする点であった。このうち、(4)に関して提案された手法(の一部)は、既に Sun の JDK 1.2 に採り入れられている。また、(1)と(2)については、Java 言語設計者である、James Gosling 氏や Guy Steele 氏らも検討するつもりがあることを表明している。(3)については Gosling 氏も Steele 氏も、言語が複雑になると難色を示しているが、今後この

ワーキンググループの提案がどのように扱われていくかは興味深い。

Concurrency and Applications ワーキンググループでは、ベンチマーク等によって、RMI の性能の問題点を指摘している。RMI の性能が低いために、現状では並列計算には使えないとのことである。その性能の低さの原因を、オブジェクト serialization と、RMI 自体の性能とに分けて議論されている。パネルでは、これらの性能評価の結果と、解決法に関するいくつかの提案をしていた。その他、MPI (Message Passing Interface) を Java で利用することの標準化に関する議論がまとめられていた。

後半の Java Grande II では、今後の展望について4つの講演があった。そのうち特に目をひいたのは、新たなコミュニティ「Datorr」の立ち上げを呼び掛けるものであった。Datorr とは「Desktop Access TO Remote Resources」の略で、グローバルコンピューティング、メタコンピューティング、および Web コンピューティングに関する研究者が一堂に会してなんらかの標準化のための活動を行っていかうというもので、Argonne National Laboratory の Gregor von Laszewski 氏が音頭をとっていた。既に、第一回のワークショップが、1998 年 10 月 8 日から 9 日にかけて、同研究所で行われたとのことだった。これは Java Grande の、concurrency / applications ワーキンググループのミーティングと連続開催されそうで、Java のコミュニティから派生したとすることができる。このことから、グローバルコンピューティングやメタコンピューティングとの Java の関わりは大きいことが伺い知れる。この第一回ワークショップでは、まず、Datorr に関連すると思われる現存の研究プロジェクトのリストアップが行われたそうである。それらはタイトルだけを列挙すると以下の通りとのことだった。

- ・ Akenti - A Distributed Access Control System
- ・ Arcade: A Web-Java Based Framework for Distributed Computing
- ・ CIF - DOE2000 Collaboratory Interoperability Framework Project
<http://www-fp.mcs.anl.gov/cif/>
- ・ Common Component Architecture Forum (CCA Forum)
<http://www.acl.lanl.gov/cca-forum/>
- ・ CUMULVUS <http://acts.nersc.gov/cumulvs/main.html>
- ・ CPU-Usage Monitoring with DSRT System
<http://dast.nlanr.net/Projects/monitor.html>
- ・ Distributed Resource Management for DisCom2

- Condor <http://www.cs.wisc.edu/condor/>
- DiscWorld <http://www.dhpc.adelaide.edu.au/projects/dhpci/index.html>
- EveryWare: A Toolkit for Building Adaptive Computational Grid Programs
<http://nws.npaci.edu/EveryWare/>
- Globus: The Globus Grid Computing Toolkit
- GlobusJ
- Habanero
- Harness: Dynamic Reconfiguration and Virtual Machine Management
in the Harness Metacomputing System
- IceT <http://www.mathcs.emory.edu/icet/>
- JINI <http://java.sun.com/products/jini/>
- Llava http://www.tc.cornell.edu/UserDoc/SP/Llava_info.html
- Ligature: Component Architecture for High-Performance Computing
<http://www.acl.lanl.gov/ligature/>
- Legion <http://www.cs.virginia.edu/~legion>
- Ninja <http://ninja.cs.berkeley.edu/>
- NCSA Workbench Project
- Netsolve <http://www.cs.utk.edu/netsolve/>
- Northrhine-Westphalian Metacomputer Taskforce
<http://www.uni-paderborn.de/pc2/nrw-mc/>
- PAWS <http://acts.nerisc.gov/paws/main.html>
- PARDIS <http://www.cs.indiana.edu/hyplan/kksiazek/pardis.html>
- POEMS <http://www.cs.utexas.edu/users/poems/poems.html>
- Sweb -- The Virtual Workshop Companion: A Web Interface for Online Labs
<http://www.tc.cornell.edu/~susan/webnet98/>
- UCLA Java Effort For Collaborative Computing
<http://www.math.ucla.edu/~anderson/JAVAclass/CAMJava.html>
- Teraweb <http://www.arc.umn.edu/structure/>
- UNICORE: Uniform Access to Computing Resources
<http://www.fz-juelich.de/unicore>
- WebSubmit: A Web Interface to Remote High-Performance Computing
Resources
<http://www.itl.nist.gov/div895/sasg/websubmit/websubmit.html>

・WebFlow: Web Based Metacomputing

<http://www.npac.syr.edu/users/haupt/WebFlow/>

筆者らのNinfプロジェクトもまさにこのDatorrがスコープとする研究であり、今後このDatorrの活動に参加していくと約束してきた。

1.3 全般的な感想

SCは、Supercomputingという元の名前から想像されるように、以前はスーパーコンピュータのためのものであったが、応用面以外の研究の流れは、もはや大部分がネットワークコンピューティングの方向に向かっているようであったし、企業展示も、科学技術計算だけでなく、ハイエンドのインターネットサーバといった分野も参入してきており、もはやこの流れは止まらないように感じられた。

2. バージニア大学 Legion グループ訪問

2.1 チュートリアル

SC98が閉幕した翌週、筆者らはVirginia州 Charlottesville市にあるバージニア大学のAndrew Grimshaw教授らのグループを訪問し、Legionプロジェクトに関して調査と討論を行ってきた。以下はこれについて報告する。

到着一日目は、まず、Legionチームのメンバーから、Legionについてのチュートリアルを受けた。

Legionとは何かを一言で表現するなら「メタ・オペレーション・システム」であるとのことだった。デスクトップのマシンから分散した資源を簡単にリモートに使えるようにするものとのことである。Legionは、分散ファイルシステムもサポートしており、他のグローバルコンピューティングシステムに比べると、もはや分散OSの世界に踏み込んでいるように感じられた。

Legionは「オブジェクト・ベース」なシステムだと強調されており、計算機やユーザといった資源のすべてが「オブジェクト」として統一的に抽象化されている。オブジェクトには、ディレクトリ構造で名前が付けられており、コマンドレベルでこのデ

イレクトリ構造の操作ができる。すべてがオブジェクトで表現されていることから、素性はとてもよさそうだが、スピードと安定性が気になった。実行速度に関しては何も見せてもらえなかったが、システムの立ち上げの様子を見せてもらったが、非常に遅かった。いつまで経っても終わらず、話題がなくて気まずい思いをしたほどだった。これは、ブートストラップとしてシステムのオブジェクトを構築しているからのようで、一度構築してしまえば後はこんなには遅くはないとのことらしい。

今回受けたチュートリアルでは、セキュリティに重点が置かれていた。すべてのオブジェクトのリファレンス(LOID)に公開鍵が含まれており、それを用いた強固で柔軟なセキュリティが実現されているそうである。しかし、すべてのメッセージ通信において、公開鍵暗号を使った認証が行われるとすると、実行速度に問題があるように思われた。

スケジューリングに関しては、クラス・オブジェクトが行うとのことだった。各クラスオブジェクトの実装によって、スケジューリングポリシーをクラス毎にき決められるからとのことらしい。しかし、この方法では、スケジューリングポリシーがオブジェクトクラスごとに固定してしまうので、スケジューリングオブジェクトは外付けにしようということになりつつあるようだ。

GC はしないとのことだった。クラスオブジェクトがつねにインスタンスへのポインタを握っているので、インスタンスへのリファレンスがなくなることはあり得ないというのが一つの理由。使われないオブジェクトは自動的に inart になり vaults に入ってしまうからいいのだ、というようなことも言っていた。しかし、長期的に安定して動かすためには何らかの GC 的なものがないとまずいのではないかと感じた。

テネシー大学の Jack Dongarra 教授らの NetSolve プロジェクトと共同で、Legion <-> NetSolve との協調処理を実験しているとのことだったので、詳しく話をきいたところ、NetSolve の API を実装しているだけで、NetSolve のコンポーネントやプロトコルを使っているわけではなかった。つまり、Legion の実行モジュールのひとつとして NetSolve のクライアントが位置付けられていた。

同様の方法で、MPI2 や PVM、CORBA へのラッパーが用意されており、既存のコードを Legion システムに載せて動かすことは簡単だと強調されていた。このため、計算プログラムの記述に利用可能な言語としては、FORTRAN、C、C++、Java などと多くのものがカバーされていた。

2.2 セキュリティに関する議論

翌日は、それぞれの分野を担当している方と、詳細なディスカッションを行った。筆者はセキュリティを担当している Adam J. Jerrari 氏とお話した。

まず、速度の問題はないかとたずねた。特に、鍵の生成に非常に時間がかかるはずで、オブジェクトを作成するたびに、鍵を生成するのでは、実用にならないのではないかとたずねた。その点は課題だと認識しているとのこと、前日のデモで立ち上がりが非常に遅かったのもそれが原因であるとのことだった。改善策として、アイドル状態のときにあらかじめ鍵を生成しておく方法などを検討しているとのことだった。

次に、鍵生成時に使用する乱数のランダム性が不十分だと、鍵が破られてしまうのではないかと、という点を指摘してみた。ウェブブラウザなどの人間の操作が介入するものでは、そのタイミング等によってランダム性を実現できるが、Legion のように、機械的に連続して鍵を生成するシステムでは、そこが問題になるはずだというわけである。この点については現状では対策していないとのことだった。そのためには、プラットフォームとなる OS の力を借りる必要があって、いくつかの OS では、そういった要求に応えられるランダム性を持つ乱数器の機能を持っているが、多くの OS が備えていない点が問題だとのことだった。

次に、セキュリティホール of 危険性について議論となった。ウェブブラウザのようにたくさんの人が使うソフトウェアになれば、多くの人がセキュリティホールがないかと調べてくれるが、そうでない限り、セキュリティホールは発見されにくい。そこまで普及すればうれしいのだが…という話題になった。

彼は、元々はモバイル・エージェントの研究をしていて、分散プログラムでのセキュアな実行について関心を持ったとのことだった。モバイル・エージェントの研究をやめてしまったのは、それにはアプリケーションがないからとのことだった。その点で、Legion は、分散計算というはっきりとした目的があるでやりがいがあるとのことだった。

2.3 感想

Legion は既に実用化に向けて動き出しているようだった。そのためにセキュリティに対してこのところ力を入れているとのことだ。こうしたシステムには、他にも Condor や Globus などがあり、乱立状態であるが、他のシステムとのコラボレーションを重視しているようで、各システムとのゲートウェイの開発の計画があるそうだ。

そろそろ何かしらの標準化が必要ではないかと感じたが、しばらくはこのまま乱立状態が続くように思われた。

謝辞

本稿をまとめるにあたって、東京大学の小柳教授、電子技術総合研究所の中田研究官からいろいろとご教示頂いた。ここに感謝する次第である。

付属資料 2 Frontiers'99 参加報告

電子技術総合研究所 関口智嗣

IEEE Computer Society が主催する Frontiers シンポジウムは今回で7回目である。開催地は前回と同じく Maryland 州の州都 Annapolis である。Annapolis はワシントン DC の郊外約 50km、最寄りには Baltimore 空港からタクシーで約 30 分ほどの距離にある。今回の会場は Marriott Waterfront で入江に面した風光明媚なホテルであった。約 100 名の参加者があった。講演者としてでもあるが、予想以上に著名な参加者があった。Foster (ANL)、Reyndors、Lubeck、Hoisie (LANL)、Reed (Illinois)、Pozo(NIST) など、日本からは中村氏 (RIST)、雨宮教授 (九大)、山名氏 (電総研)、村岡教授 (早大) をはじめ我々を含めて 8 名であった。

旅程の都合により 2 日目から参加をした。予定が変更になり、当日の基調講演は Paul Messina になった。氏はカリフォルニア工科大学の教授であるが、現在、エネルギー省に出向し、ASCI プロジェクトの技術ディレクタを任じられている。講演の中身は ASCI におけるシミュレーション技術という観点からの最新情報の提供であった。2004 年まで 国防プログラムの一環として Office of Defense Programs を設け、Office of Research, Development and Simulations (Director Dr. Gilbert Welgand) その下に Computing and Simulation Modeling、Computing Simulation Environment、Integrated Computer Systems (visualizations)、の 3 つのサブプログラムが設けられた。ここでは、2005 年以降の統合的計算機シミュレーションシステムの話に焦点があり、このため、あまり現実性は感じられなかった。非常に大規模な問題であることは間違いがないが、このような核実験代替といった応用例でないと予算が獲得できないところが悲しい。日本のような純粋な計算科学や地球シミュレータ等のプロジェクトでは一致して大規模ソフトウェアを開発ということにならないのであろうか。ただ、プロジェクトのスキームとしては国立研が主体になり、大学 (U. of Illinois、Caltech、U. of Utah、Stanford、U. of Chicago) と連携し、さらには産業界を競争的に開発意欲を持たせるところは業界全体を元気づけることになる。

イリノイ大学の Dan Reed 教授は「Delphi: Integrated、Language-Directed Performance Prediction、Measurement、and Analysis」という標題で講演を行った。これには同じくイリノイ大 Padua 教授、インディアナ大学の Gannon 教授、アルゴンヌ国立研究所の Foster 氏、ウイスコンシン大学 Millar 氏などが参加している。

要旨は下記のようなものである。コンパイラと性能評価モデルとの統合は以前からいわれているが、具体的な話はまだ実現していない。この Delphi はコンパイラ、ツール、性能モデル、のメンバーが共同して開発を行う。プログラムに関してコンパイラで得られた静的情報と、ツール群が獲得した動的情報をデータベースを介して解析を行うことにより、より最適なプログラム実行環境を得る。

翌日 25 日は Deep computing という標題で William Pulleyblank、Dir. Math. Sci., Watson Research Ctr. の講演があった。IBM の製品戦略を Deep Blue によるチェスゲームマシンの逸話にはじまり、アトランタオリンピックの際の短期間、局所予報についての実績、フロリダ州立大とのハリケーン進路予想の共同研究、電力負荷予想に基づき東西の時差を用いた負荷分散、電力マーケットによる実時間の売買について、またピーク使用量の予想は気温に基づくことから、ピーク使用量を予想する。10-22 のシナリオを用意し、10% ほどのサイクルコストは \$20M の節約になる。UPA (Underwriting Profitability Analysis) におけるデータマイニングの実例により利益をもたらすための分析方法を例示した。バイオメトリクス (音声認識、指紋、手書き文字、網膜、顔、手のサイズなど) 先端的な Id 技術に用いられる。指紋については古い歴史があり、ridge のパターンとして分類される。不変的な情報を用いた例。データベースとのマッチングをとるのに、多くのアルゴリズムが開発されている。高速検索、聖書の全文の中から様々なテキストパターンを抽出するこうした技術はバイオメトリクスに非常に有効であるなどの話であった。

JAVA に関しては Geoffrey Fox が Java Grande: Is Java Useful in MPC/HPCC: <http://math.nist.gov/javanumerics> として講演を行った。JAVA Grande が提案された経緯を述べた。その中で C++ に期待されたが、仕様が複雑になりすぎたことにより HPC では用いられなくなったとのこと。Java は基本的にインターネットベース、スレッドベース、分散計算が自然に取り込まれている点であるとした。Java Grande は高性能計算とネットワーク分散計算を目指した新たな規格の提案と実現を目指している。Fox 氏はそのリーダーとして、Dattor (Desk top access to remote resources) など様々な活動を紹介した。

さらに Marc Gupta が Java for Numerically Intensive Computing: from Flops to Gigafllops について講演を行った。普通のプログラムの場合、例えば流体の計算で Fortran が 40 倍高速であり、行列乗算の場合、500x500 において 50 倍高速であっ

た。これは以下の理由がある。

- 厳密な配列バウンドとヌルポインターチェックによる厳密な例外処理
- 高次のループ変換は許されていない
- 加算・乗算のフュージョンができない

ループの外で配列範囲チェックすることにより、内部でのチェックを減らすことができ、これにより、1.6 Mflops であったものが、100 Mflops となった。さらに、加算・乗算のフュージョンや配列の配置を行うことができるような Fortran 90-like array package が必要であると認識している。このほか、必須の型として、doubleArray2D、intArray 1D、ComplexArray3D standard complex class などが必要である。結論としては、Java は Fortran 数値計算においても Fortran 並の性能を出せることが示されたことと加算・乗算の融合命令が必須であることがわかった。

Frontiers'99 の後、西海岸・カリフォルニア州サンディエゴに向かった。サンディエゴスーパーコンピュータセンター (SDSC) において NPACI の状況と Tera Computer の状況について現地調査を行うことが目的であった。会合を Associate Director、Center for Advanced Computational Science and Engineering の Wayne Pfeiffer ならびに Margaret Simmons と行った。また、現地に滞在中の田浦 (東大)、中田 (電総研) 両氏も議論に加わった。米国 NSF の計算センターの縮小に伴い、最終的には NCSA (イリノイ) ならびに SDSC が残ったこと。これまでの計算センターとは NPACI (National Partnership for Advanced Computational Infrastructure) を通じて一体的な運用を行うことを目指している。ある意味での仮想大規模計算センターとして位置づけられている。また、SDSC はその中でもハイエンドの計算資源を提供するセンターとして、大規模並列機 (T3E) や SP-2 ならびに Tera Computer などのシステムを導入してきている。これらをユーザが透過的に利用できるように環境の構築を目指しており、そのための必要な技術 (Thrust area) の目標を設定して、それぞれの研究プロジェクトに発注をしている。かなり細かなミッションを与えて、具体的なソフトウェアとして研究成果を利用する環境を提供している。物理的に離れている計算センターの (例えば) SP-2 を利用するのに同じユーザが状況を見て、バッチキューを選べるようになっていること、またそのために分散ファイルシステムを構築していることなどが印象的であった。計算センター運用に関しても模範となる様式を揃えていると言えるであろう。

Tera の MTA に関しては、非常にユニークな形状をしている。実装もガリウム砒素を使い続けており、そのために不安定なところも依然としてある。訪問の数日前に 4 台 CPU に増強されていたが、一般のユーザはまだ付いていないようであった。筐体は被せたものであり、実際のシャーシとは独立して設置されていた。性能に関しても、ベクトル化が困難なとある計算化学系のベンチマークでのスケーラビリティが良いことを主張していた。説明は Feo 氏という Tera から派遣されているソフトウェアの責任者であった。氏はもともとローレンスリバモア国立研究所において並列言語 SISAL の研究グループに長らく属していたそうだ。安定稼働にはまだ時間がかかるという印象を受けた。

以上、Frontiers'99 ならびに SDSC の出張について得られた成果についてまとめた。ASCI をはじめとして、情報技術ならびに産業が相変わらず元気であるのが印象的であった。ホテルにもインターネットサービスがあたりまえになってきていたのには改めて驚かされた。

付属資料 3 最先端並列計算機における次世代気候モデル開発 に関わる国際ワークショップ参加報告

1. ワークショップの主旨

財団法人 高度情報科学技術研究機構は、科学技術庁の委託事業として、関係大学及び研究機関と共に、平成10年度より「科学技術振興調整研究：高精度の地球変動予測のための並列ソフトウェア開発に関する研究」を開始している。この研究は、大気海洋及び固体地球の超長期にわたる変動に関し、並列ソフトウェア技術を駆使して大規模シミュレーションを行う環境を構築する研究である。

今回、大気・海洋分野のこれまでの研究成果を基に、「高性能並列計算機における次世代型気候、気象モデルの構築」とするテーマで、今後、発展する高性能大規模並列計算機等の資源を有効に利用しつつ、高精度の次世代型気象、気候モデル開発の展望について、振興調整費研究者と海外参加研究者の研究成果を相互に発表し合い、討論するワークショップを開催した。

2. ワークショップ開催日時、場所および調査員

開催日時： 平成11年3月1日（月）－3月3日（水）

会議場所： 米国ハワイ州ホノルル市、ハワイ州立大学 東西交流センター

調査員： 鳥居 良春（当協会先端情報技術研究所技術調査部 主任研究員）

河西 和美 当協会先端情報技術研究所技術調査部 主任研究員）

3. 主催者等

主 催： 財団法人 高度情報科学技術研究機構
（科学技術振興調整費研究の委託事業として）

後 援： 科学技術庁

4. 総合的な感想

日米の気象に関する研究者、スーパーコンピュータのベンダー関係者が、80 数名集まり、主に気象のモデルとその数値計算の実行方法について議論を深めた。

日本は、すでにハードウェア・プラットフォームとして日本電気の地球シミュレータ用スーパーコンピュータを利用する方向が定まり、高度情報科学技術研究機構を中心に研究体制が固まりつつある。

一方、米国においては NCAR, UCAR, LANL などの国立研究所で気象の研究が進められているのに加えて、海軍、空軍などの研究機関、地方大学に置ける地域に密着した気象の研究が進められているなど、進んだ研究がなされてはいるが、まとまりに欠ける感じを受けた。ハードウェア・プラットフォームについても、IBM SP、Cray J-シリーズ、SGI オリジンシリーズと機種が豊富でまとまりに欠けている。

しかし、米国においても研究の統合（モデルの統合）を必要とする声が上がっており、ハードウェアについても実力から IBM の MPP (Massively Parallel Processing) に統合されていくような雰囲気である。

ワークショップにおいては、日米の協力が議論とはなっているが、現状では、単なる意見を述べ合ったにとどまり、共同研究への話にまではいたらなかったと感じている。

米国は、あまり日本の成果（ハードウェア・プラットフォームを含めて）を期待していないようだ。

余談ではあるが、NCAR において NEC のスーパー導入に尽力してくれた Buzbee 博士は、このワークショップにおいても、ベクトルマシンの効率の良さ（ピーク性能の 35% を引き出せる：MPP 機では 10% 前後しか引き出せない）をアピールしてくれた。彼は、先ごろ NCAR を退職し、Buzbee Enterprises というコンサルタント会社を起こしたようだ。

5. ワークショップの背景

この項においては、日米の気象モデル研究の紹介とカナダ気象庁の気象予報を例として報告する。ワークショップの内容については、<http://www.tokyo.rist.or.jp>に詳述されているので参照されたい。

5.1 日米の気象システムモデル研究の紹介

5.1.1 米国 NCAR の気象システムモデル

NCAR は、気候関連のリサーチを大学、企業にまたがって計画し、組織し、指導する機関であり、ここでは、CSM(Climate System Model)プロジェクトが実施されている。

CSM プロジェクトは、1994 年 1 月に、物理と生物地理化学とを含んだ総合的な気象モデルを構築する長期プロジェクトとして始まった。

1998 年度において、CSM プロジェクトは、モデル開発の第一段階を完了させることに注力している。今現在、課題となっている事項は、実行可能な物理気象システムの構築と評価である。特に、物理的な気象システムモデル（年単位とか 10 年単位のサイクルを表現できるモデル）の評価に取り組んでおり、生物地理化学的な要素が気象システムに与える影響に着いては、次のステップで盛り込むことを計画している。

初期モデルが完成した後は、硫黄のサイクルとかグリーンハウス（温室）効果とか、海洋モデルの水平方向の解像度を上げた短期の気象変動などを取り込んでいく考えである。

総合的な CSM 構成は、独立した 4 つ（大気、海洋、地表、海氷）の基礎的なモデルによりできおり、個々はお互いにメッセージを伝達しあっている。

大気モデルは NCAR の CCM3 (Community Climate Model version 3) であり、海洋モデルは NCOM (NCAR CSM Ocean Model) であり、海氷モデルは CSIM (CSM Sea Ice Model) であり、地表モデルは LSM (Land Surface Model) である。LSM は、最近 CCM3 に取り込まれている。

プロジェクトを開始した当時、上記の 3 つのモデルは非常に良い特徴を持っていたが、お互いのモデルを統合したときに問題となる重大な欠陥を含んでいた。従って、CSM の開発は、これらの統合を低解像度のレベルでテストしてお互いの影響を調べるトラックと、欠陥を改善した（モデル自身の見直しを含む）モデルを構築するトラックの 2 つを設けて開発を進めた。

(1) 大気モデル

CCMモデルはこの18ヶ月の間に大幅に改定された。主要なグリーンハウス・ガス（水蒸気、二酸化炭素、オゾンに加えて、窒素酸化物、メタン、CFC11、CFC12）は、ロング・ウェーブ・ラディエーション・コード（プログラム）に追加された。境界レイヤーの式は、タワーデータの新たな解析と詳細なレイヤーモデルにより改定された。雲の光学的な特性の診断も改善され、降水の蒸発による影響もモデルに含められた。より一般的な重力波のパラメータも考慮されている。地表と海洋との相互作用の係数も改善されている。大気対流に関するパラメータにいても Zhang and McFarlane スキームが選択され、その還流の効果をテストしている。

(2) 海洋モデル

海洋モデルである NCOM も、ここ数年で大きく改定されている。高解像度（公称2度：ソーンス解像度2.4度、子午線方向の解像度1.2度-2.3度、垂直方向45レベル）のモデルは、CSMに対応した初めのモデルとなるであろう。海洋モデルは、Gent-McWilliams パラメータを数値解析の改善を図るために利用している。単純な生物地理化学のモデルが開発され、テストを実施している。

(3) 地表モデル

地表モデルは、地表のエネルギー、水の理論、炭素化合物などを植物種の生態系の差異とか土壌のタイプ熱・水的差異などを考慮して計算している。

地表モデルは現在 CSM に組み込まれているが、今後は分離されたモデルとして開発される予定である。

(4) 海水モデル

CSM に組み込まれる初めの海水モデルは CSIM モデルであり、CSIM モデルは従来 CSCORG (Climate Sensitivity and Carbon Dioxide Research Group) により使用されていたモデルを改善した。今後は、より現実的な氷の物理現象や熱力学に基づいた開発が計画されている。

(5) フラックス・カプラー

フラックス・カプラー に関する初めのプログラムは、1994年に開発されテストされた。このプロトタイプ・プログラムは標準的な地表温度予測を使用した CCM2-R15 (rhomboidal 15 version of CCM2) と 3X4 度の海洋モデルと、気候学上の海水を使用した。このプロトタイプを用いて、過去25年分の気象シミュレーションを実施しデバッグをした。カプラーは改良に改良を重ね、その性能を大幅に改善した。

フラックス・カプラーは、今後、ハードウェアのネットワーク性能の向上とソフトウェアのメッセージパッシング技術の向上に伴って、グリッド間の補間などのアルゴリズムが改善され、大いに開発が進むことが期待されている。

このフラックス・カプラーは、エネルギー省のコンピュータ・ハードウェア—Massively Parallel Computers の接続モデルとなる CHAMMP (Advanced Mathematics and Model Physics) プログラム—にも採用される予定である。

5.1.2 日本の気象システム・モデルの研究

ここでは、科学技術庁が中心となって推進している“地球シミュレータ”プロジェクトで計画されている『雲解像非静力学モデルの最適化並列プログラム構築に関する研究』について紹介する。

(1) 研究目標

数千 km 四方の領域を対象に個々の雲を解像して降水過程を再現する高精度の高分解能非静力学大気モデルを開発する。地球シミュレータでの高速シミュレーションを可能とするため、モデル全体を分散主記憶型並列計算機に対応できるようにするとともに、高精度領域数値予報モデルや全地球規模の高分解能プラットフォームと結合させ、より広域の気象条件を取り込むことが出来るようにする。これによって、集中豪雨・豪雪、雷、ダウンバースト等の局地的顕著気象現象や台風の微細構造を数值的・力学的に再現し、それらのメカニズムを解明するとともに、局所的な気象災害の迅速な予測のために役立つ技術情報を提供する。

(2) 研究項目の現状

気象研究所予報研究部では、地形を含み降水現象の再現実験の可能な、気象研究所非静力学メソスケールモデルを開発している。気象研究所台風研究部は、2 ウェイの多重格子台風モデルや非静力学対流モデルを開発した実績があり、静力学モデルによる台風の3次元シミュレーションを行っている。

これまで、日本では非静力学モデルを用いて千 km 四方以上の領域を対象に個々の雲を解像して降水過程や台風を扱った3次元シミュレーションは行われておらず、世界的にもまだ殆ど例がない。

(3) 研究計画の概要

(a) 概要

気象研究所予報研究部で開発された非静力学メソスケールモデルをベースに、非静力学プラットフォームの基礎となるように物理過程の強化、力学部分の高速化など、プログラムの最適化を行うとともに、数千 km 四方の領域や台風のシミュレーションにも対応できるようにプログラムの整備を行う。モデル全体の書換えを行い、分散主記憶型並列計算機上で高速動作する雲解像非静力学モデルの最適化並列プログラムを構築する。

(b) 年次計画

(第 I 期)

平成 10 年度

- ・物理過程の強化や力学フレームの検討によるモデルの最適化
- ・基本性能評価のための予測精度の検証と再現実験
- ・UNIX への対応と並列アルゴリズムの検討

平成 11 年度

- ・分散主記憶型並列計算機に対応させるためのプログラムの書き換え
- ・並列計算試験装置によるテスト
- ・台風の 5 km メッシュ級シミュレーションへの対応と側面境界条件の改良

平成 12 年度

- ・モデル全体の最適化による第一世代並列化非静力学モデルの構築
- ・千 km 四方領域を対象とするシミュレーションによるモデル性能の検証

(第 II 期)

平成 13 年度

- ・移動格子系や 2 ウェイなど、新しいネスティング技術の開発
- ・全球プラットフォームとの結合
- ・並列化処理の高度化と地球シミュレータへの対応

平成 14 年度

- ・最適化並列プログラムによる雲解像非静力学プラットフォームの構築
- ・地球シミュレータによる数千 km 四方領域を対象とするシミュレーション
(集中豪雨・豪雪、台風等の再現実験)

5.2 カナダ気象庁の気象予報

<http://www.hpc.comp.nec.co.jp/sxwj/no21/example.html>

5.2.1 概略

カナダ気象庁では、NEC スーパーコンピュータ SX-4/32 を使用し、気象予報計算を行っている。1997 年に北米全体(米国・カナダ本土)を 10km メッシュ($753 \times 510 \times 31 = 1190$ 万 4930 点の格子)にきり、初のリアルタイム 24 時間予報を実現させている(計算時間: 40 分)。使用予報モデルは、メソスケール地域モデル

(MC2: Mesoscale Compressible Community2) の断熱計算カーネル部と数値予報研究所 (RPN: Recherche

en Prevision Numerique) のフル物理パラメタリゼーションモデルとを組み合わせている。計算結果は、マシンの理論最大性能の 40% 以上の実効性能を示している。

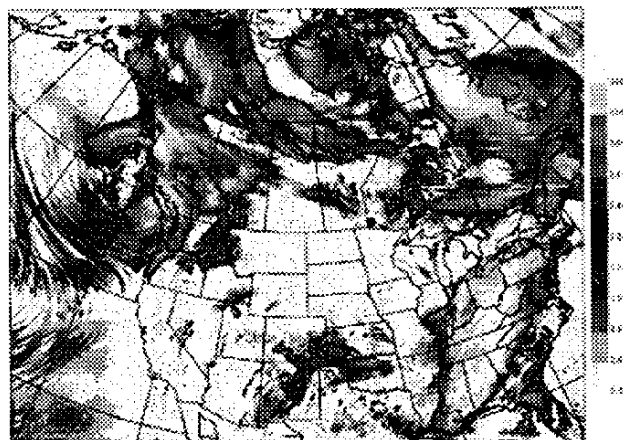


図1 10km メッシュモデルによる 24 時間降水予報

5.2.2 メソスケール圧縮性地域モデル (MC2)

メソスケール圧縮性地域モデル (MC2) は、1980 年代中盤に数値予報研究所とケベック州モントリオール大学 (UQAM) で開発された完全圧縮性地域モデルの拡張版である。カナダでの気象モデリングのコンセプトの実証と大気乱流の研究やプラネタリー境界層に対する新パラメタリゼーションスキームの開発には、大幅な計算能力が必要であり、そのため分散メモリ並列処理対応の MC2 が開発されている。

5.2.3 MC2 の力学モデル

MC2 の力学モデルは、単一プロセッサでの性能を最大限とするよう最適化されている。非対称行列の解法には、前処理付き GMRES 楕円型ソルバが導入されている。現在、急峻な地形の効果をうまく扱うためのインプリシット解法定式化が行われている。

5.2.4 MC2 の物理過程

MC2 モデルでの物理モデルでは、乱流運動エネルギーに基づくプラネタリー境界層スキーム、相似理論に基づく接地境界層スキーム、日射・赤外放射、大域的な降水、対流性の降水、重力波による地表面抵抗が含まれている。将来的には、新たな接地気層スキームの開発とともに、微物理過程モデルのテストも計画されている。

5.2.5 性能測定

SX-4/32 SMP30 プロセッサシステム上で、30 個のユニットからなる領域分割法を用いた。時間刻み幅は 3 分。24 時間 (480 ステップ) の予報計算での実効性能は 24 GFLOPS (ピーク時) であった。

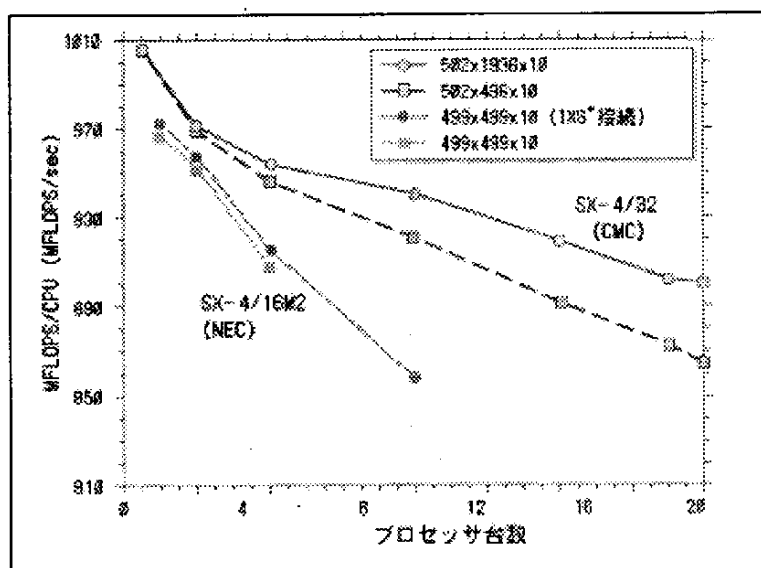


図 2-1 プロセッサあたりの実効性能 (MFLOPS)

SX-4/32 (カナダ気象庁) および NEC による計測結果。凡例の数値は計算格子点数

* IXS: ノード間接続装置

5.2.6 MC2 のソフトウェア

分散メモリ、メッセージパッシング対応の MC2 バージョン 3.5 には、MSG (構造格子に対するメッセージパッシングツール) が用いられ、これはモントリオール大学計算機応用研究センター (CERCA : University of Montreal, Center for Research on Computation and its Applications) が開発した。ここでは、MPI 標準の持続型通信チャンネルおよびパイプラインメッセージ (MPI_sendinit、MPI_recvinit、MPI_startall、MPI_waitall、MPI_request_free) が用いられている。

5.2.7 リアルタイム予報能力

1 タイムステップの計算時間は、5.5 秒(モデル計算：4 秒、物理過程処理に伴う並列 I/O：1.5 秒)である。SSRAM メモリの追加により、物理過程部分の計算が加速され、1 タイムステップを 4 秒で処理でき、I/O 処理を含めて 32 分(480 ステップ×4 秒=1920 秒)で実行可能と予測されている。

6. 気象モデルと計算機能力の関連を表す図

<http://www.kuamp.kyoto-u.ac.jp/labs/fndsys/ESP/HTML/poster-31.html>

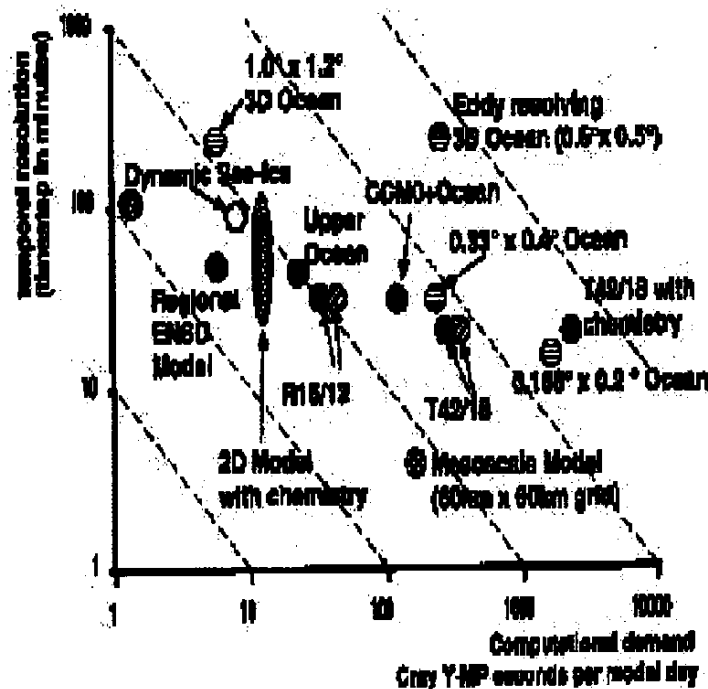


図3 典型的な気候モデルの時間解像度と計算機能力
縦軸は時間解像度、横軸は Cray Y-MP 計算機使用で必要となる計算時間。R15 は CCM1、T42 は CCM2 のパラメータ。

付属資料 4 1999 年度 Blue Book にみる

米国ハイエンドコンピューティング研究開発技術動向

1999 年度 Blue Book を参照し、米国政府が推進するハイエンドコンピューティング研究開発テーマを、アーキテクチャ、ソフトウェア、アプリケーションに大別し、投資元である省庁ごとに分類した結果を詳述する。

1. アーキテクチャ

1.1 DARPA

(1) 現在の半導体ベースのコンピューティングの限界を超えた性能/コスト/サイズ/重量/消費電力の改善のための新しいコンピューティング技術について研究開発を実施する。

「ウルTRASケールコンピューティング」と呼ばれるテーマの研究開発計画を次に示す。

1998 年度	1999 年度
アモルファス・アルゴリズム計算 (amorphous algorithmic computation) のため、量子とシリコンの間のハードウェア/ソフトウェア・インタフェースと言語について設計・モデリング・評価を行う。	大型のアモルファス・アレイ (amorphous array) をベースにした計算モデルのシステムレベルの設計とシミュレーション研究を実施し、要素数が 1,000 を超えるプロトタイプアレイのデモを行い、量子コンピューティング研究媒体の開発における核磁気共鳴技術の役割を確立する。
256 個の要素によってアドレス指定される分子計算機構アレイ (array of molecular computational mechanisms) のデモを実施し、シリコン上に神経要素を培養するための表面パターンニング機構を評価する。	DNA ベースの論理演算、セルベースの計算、新しい通信路と、防衛用アプリケーションで、これらの手法のスケラビリティを重点とする。
ウルTRASケールコンピューティング手法を使うインプリメンテーションに適した計算モデルを調査する。	ウルTRASケールコンピューティングモデルのデモと検証を実施する。

(2) 異なる性能のシステムの全てに適した、安全でスケラブルなコンピューティング技術と通信技術の基礎を創造する。以下の技術項目があげられている。

- (a) 高効率の並列ノード
- (b) スケラブルシステム用のファイル I/O の自動並列化
- (c) 超低電力システムの最初のノードレベル性能

- (d) セキュリティをサポートする新しいバックプレーンネットワークの性能
- (e) ハードウェアで高速化された分散型の共有メモリのワークステーションクラスタ上での性能

1.2 DOE

(1) MTA (multi-threaded architecture)

Tera Computer システムが、最も重要なペタフロップスアーキテクチャにである MTA の評価に使われる。

1998 年度	1999 年度
プロトタイプシステムの評価として、1 組のアルゴリズムと DOE ミッション関連のアプリケーション性能評価を Tera Computer システム上で実施する。	ペタフロップスシステムに必要なソフトウェアとアーキテクチャに関する新しい開発を評価する。

1.3 NSF

(1) 非 2 進法光記憶システム

Georgia 工科大では、Optex Communication Corporation と協力して、光メモリの信号処理コンポーネントの設計をすすめた。M=6(「M 進法」)として光記録の平方インチ当たりの情報密度を高める。この研究は、PECASE (Presidential Early Career Award for Science and Engineering) からの資金提供を受けている。

(2) CDMA 通信システムにおけるマルチユーザ検出

CDMA (Code Division Multiple Access) によって、デジタルセルラ通信システムで複数のユーザに同じ周波数を割り当てることを可能にする。最高クラスのマルチユーザ検出は、現時点では実現が困難であるが、New Jersey 工科大、North Carolina 州立大、Ohio 州立大、Rice 大、Colorado 大では、複雑度が低いアルゴリズムと関連ハードウェアを開発している。

(3) 知識ネットワークキング

これは、大学での次世代の相互接続ネットワークおよびそれに関連するデータベースとコラボレーション技術に重点を置いた計画である。

1.4 NSA

(1) HTMT (Hybrid Technology Multithreaded architecture)

- (a) 目標：従来のアプローチよりもはるかに短時間でペタフロップスレベルの性能に到達できる超高性能コンピュータシステムを構築する。次の 10 年間に米国の戦略上重要ないくつかの高性能アプリケーションにこのシステムが使われることが期待されている。
- (b) 構成：マルチスレッド式実行モデルをベースにする HTMT は、超伝導技術、高速 VLSI 半導体技術、光インターコネクト、記憶技術などの最先端技術を組み合わせて使う。
- (c) 研究機関：Caltech/JPL、Delaware 大、SUNY Stony Brook、Notre Dame 大、Princeton 大に加えて、他の政府/メーカーの研究所から構成される機関。

HTMT 概念に関する初期の研究は、1996 年～1997 年に NSF と NASA から資金提供を受けた (Point Design Study)。HTMT の包括的な設計調査は、現在は DARPA、NSA、NASA がスポンサーであり、Caltech と JPL によって管理されている。Delaware 大で実施されている研究は、HTMT のプログラム実行とアーキテクチャモデルに重点を置いている。

(2) 量子コンピューティング

1998 年度と 1999 年度に、NSA は、量子計算理論および実験に関する基本的な予備研究を継続する。これらは、アルゴリズムと計算量理論、量子誤り訂正 (quantum error correction) 手法、量子デコヒーレンスプロセス、小規模の研究所の実現を含む。

(3) MARQUISE 計画

- (a) 組み込み型高性能コンピュータ SOLITAIRE の実装試験を、米国空軍機を使って実施する。

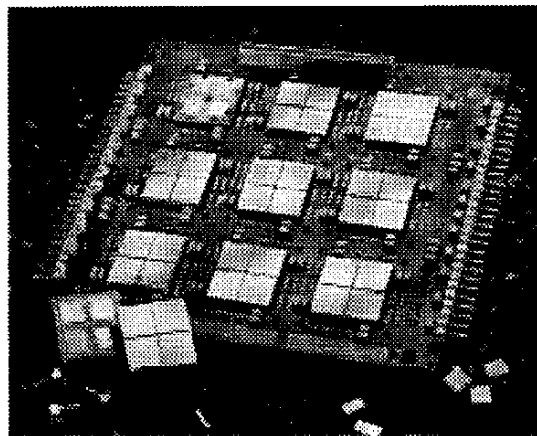


図 1 SOLITAIRE 基板

SOLITAIRE の基板は、4 個のベクトルプロセッサ、ネットワーククロスバ、I/O、0.5 G バイトの DRAM を含む完全な SGI/CRAY J90 System Element を搭載している。

基板の表面にはダイヤモンドベースのダイキャスト MCM (multichip module) が 9 個実装され、裏面にはダイキャスト MCM が 8 個実装されている。SOLITAIRE には両面基板の格子状の部品取り付けに関する新しい成果が取り入れられており、その特長はモジュールと基板間接続の数が 25,000 を超えることである。再実装されたこの System Element は、製品化されたものよりも 80% 小さく 75% 軽い。10 平方インチの基板に 500W の電力が集中する。

- (b) 小型化された噴霧冷却式の組み込み型ダイヤモンド電源に関する研究で、電力密度 2 kW/in^3 の $48 \text{ V}/2.8 \text{ V} @ 200 \text{ A}$ の高効率電源コンバータの発表を予定している。

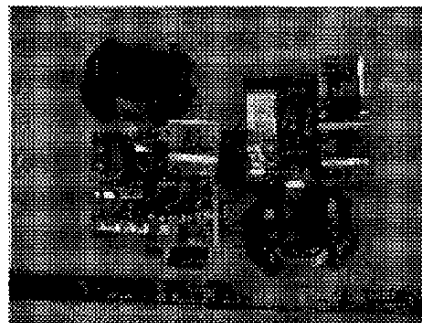


図 2 200 W/in^3 の電力密度を達成した $28 \text{ VDC}/4 \text{ VDC}$ 噴霧冷却式コンバータ

- (c) 組み込み型のスケーラブルノード (MARQUISE の後続アーキテクチャ) の研究を継続する。商用コンピュータの発表後 6 カ月で 100% ソフトウェア互換の小型高性能コンピュータを利用可能にする再実装インプリメンテーションを 3 つ選択した。

(4) 高速コンピューティングに使われるマイクロエレクトロニクス

合成ダイヤモンド実装技術、全光学式スイッチング、光電子集積回路の実装技術などの分野の研究開発項目を次に示す。

- (a) 電力節減のための使用地点での電源変換
- (b) 高性能 MCM の低消費電力版を実現するためのエリアアレイ I/O 設計の研究
- (c) 標準アーキテクチャ用の非常に高度なプログラマブルアクセラレータプラグイン

図 3 は、 128×128 超伝導クロスバスイッチである。スイッチのマトリクスは直径 4 インチの MCM に取り付けられ、絶対温度 4 度に維持される。室温環境との接続には、8 マルチチャネルの電気ケーブルが使われる。

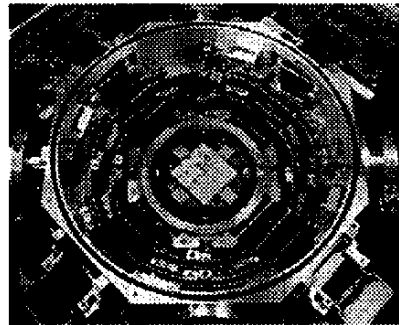


図 3 128x128 超伝導クロスバスイッチ

(5) 高速回路

テラビットデータ転送のため、100 Gb/s のシリアルデータ速度の光ファイバリンクを電子回路が扱えるかどうかを調べる研究を実施している。

(6) LOTS

開発中のこの光テープシステムは、従来の大型記憶装置と比べてテープ当たりの記憶容量は 10 倍、入力データ速度は 10 倍以上優れている。

2. ソフトウェア

2.1 NSF

分散コンピューティング、分散マルチメディアシステムに関する NSF（全米科学財団）支援の活動は、次の開発を目標としている。

- (a) 証明可能なサービス品質 (QoS, Quality of Service) を提供する方法
- (b) 異種ネットワーク上で実行されるマルチメディアアプリケーションに対する QoS の保証
- (c) 生のオーディオ/ビデオストリームや保存されたオーディオ/ビデオストリームの送信用に最適化されたアルゴリズムとプロトコル
- (d) ルーティングとリソース予約のアルゴリズム
- (e) トラフィック管理の統合手法

(1) 時間変化型 (time-variant) QoS 管理は、リアルタイムシステムのモデリングやコンピュータネットワーク、分散システムに関する研究を組み合わせる。必要なタイミング制約

条件の交渉・再交渉を実行するための仲介手順 (brokerage procedure) と、送信中に時間変化型 QoS の提供や適応を満足させるための分散制御機構とが開発段階にある。Maryland 大、Texas 大、Illinois 大で実施されている。

(2) 認証付きコード

Carnegie-Mellon 大では、Java/Web ベースのような環境下で、信頼できないソースからダウンロードされたコードを十分に安心して実行できる方法を開発した。この新しい技術は、コードと一緒に認証をダウンロードするもので、受信側では定理証明ソフトウェアが受信された認証をコードと比較して、そのコードが本当に問題ないことが保証される。この活動はプログラミング言語のセマンティクス、定理証明、コンパイラ手法、形式的方法 (formal method) に関する研究を利用している。この研究の初期の成果には、比較的効率の良い証明チェック機構や、C++ の重要なサブセットを対象とした「保証 (certifying)」コンパイラがある。

2.2 DARPA

(1) スケーラブルなソフトウェアと使いやすさ

DARPA (米国防総省高等研究計画局) 支援のプロジェクトは、アプリケーションプログラマがシステムを容易に使えるように調整されたスケーラブルなソフトウェアを開発する。

1998 年度	1999 年度
タスクとデータの両方の並列性をサポートする HPC++ 言語とランタイムサービスのリリース。	作業負荷や利用可能なリソースの変化に対応する適応型のリソース割り当ておよび消費をサポートするプロトタイプサブシステムのリリース。
先進の分散シミュレーションなどの並列アプリケーションをサポートする言語とランタイムサービスの開発。	組み込みシステムを使った試験的なスケーラブル画像処理アプリケーションの発表。
エコー面積 (radar cross section) のモデリングのための新しい反復ソルバ (iterative solver) の試験バージョンの発表。	

(2) オペレーティング・システムの拡張

1999 年度の DARPA の計画は、演算を中央のプロセッサとメモリ階層に組み込まれた論理回路のどちらで実行するかを防衛分野のアプリケーションが指定できるようにするための、命令セットの拡張と記憶コンポーネントの設計である。

2.3 DOE

(1) 高性能ソフトウェアツール

重点目標は、科学的問題の解決での高性能システムの使用を促進するソフトウェア開発である。開発内容は、超並列処理用の効率的なオペレーティングシステムと I/O、アプリケーションコードの基礎となるハードウェアの詳細から分離する枠組み、科学分野のアプリケーションの性能監視用ツール、ハイエンド計算の結果の管理・視覚化・理解の改善など多岐に渡る。

DOE のプロジェクトには、ACTS (Advanced Computational Testing and Simulation) のためのプログラム開発環境用と性能監視用のソフトウェアツールが含まれている。目標は、危険すぎる、費用がかかりすぎる、アクセスできない、政策的に実現可能でないなどの理由で実際の実験が困難な場合に、実験の代わりにコンピュータシミュレーションを使えるようにするアルゴリズムやソフトウェアツール、インフラを統合された形で開発することである。

1998 年度	1999 年度
オブジェクト指向の計算 (numerics) のためのスクリプティングインタフェース言語の開発。	計算網アプリケーションをサポートするために、広域実行用に SciTL を拡張する。
ACTS Toolkit 用の並列プログラム解析の枠組みの完成。	
SciTL (Scientific Template Library) 用の高性能ソフトウェアコンポーネントアーキテクチャの開発。	
共有メモリアクセスプログラミングモデルによる SciTL の拡張。	

2.4 NASA

(1) ESS (Earth and Space Sciences)

NASA の ESS スケーラブルテストベッドの 1998 年度の目標は、50 GFlops の持続的処理速度である。この開発のために、NASA は、大規模な生産環境で実証された高い可用性と移植性を提供する、競争力がまだないプロトタイプシステムソフトウェアを、最終的に商用化することを目標にして開発する。NASA の持続的処理速度 100~250 GFlops のスケーラブルテストベッドの 1998 年度の目標は、100 GFlops の持続的処理速度である。

2.5 NSA

(1) UPC

NSA の CCS (Center for Computing Sciences) は、California 大 Berkeley および Lawrence Livermore 国立研究所と共同で UPC を開発している。UPC は、明示的並列 C 言語の標準となる可能性がある基本言語を開発する試みである。UPC は AC を拡張したものだが、次の 3 つの点が AC と異なる。

- (a) 安全性と性能をバランスさせるためにメモリの一貫性をプログラマがコントロールできるようにするキーワードが追加されている。
- (b) 特定のアプリケーションについて性能とプログラミングの容易性を改善するために、プロセッサ間でアレイのブロッキングを実行できる。
- (c) 並列に実行可能なループ内部での負荷のバランスをとるために「forall」概念を導入している。

(2) RES

RES は、1991 年～1992 年に NSA の HPCF 資金によって開発されたシステムで、「アイドル状態」のワークステーションを活用するために設計された。普及の度合いをますます高めている RES は、継続的な強化の対象になっている。

(3) JACKAL

JACKAL は、クリシェ (cliche) をベースにしたソフトウェアリバースエンジニアリング研究システムである。JACKAL への入力、ソースコードか逆コンパイル済みコードを高度な抽象言語で翻訳したものである。JACKAL は、ツリーマッチングか文字列マッチングのアルゴリズムを使ってクリシェを識別する。

2.6 NIST

(1) 性能測定技術とテスト技術

NIST は、LAN でクラスタ化されたワークステーションを対象にしたアプリケーションレベルの測定手法 (物理性能モニタとソフトウェアツールを含む) を継続して開発する。これらは、オペレーティングシステム、通信ライブラリや、それらに関するデーモンからの性能情報の収集を含む。NIST は Linux のカーネルと MPI などのメッセージ転送ライブラリを調査して、性能情報を提供するとともに、ユーザが情報を選択的に収集できるような手法を開発する。その他には、光テープサブシステム設計や、デジタルビデオディスク (DVD) 媒体用のテスト方法に関する調査も実施する。

2.7 EPA

(1) 環境モデリング・ソフトウェアの並列化

各種の環境モデリングコードを並列化するための枠組みが、EPA の支援を受ける Iowa 州立大で開発された。主な特長は、クラス固有の自動並列化で、そこでは特定クラスに関する高度な知識が、効率の良い並列コードに到達するために使われる。このアプローチは、エキスパートシステムとコンパイラの技術を組み合わせて、手作業によるコーディングを自動化する。現在のシステム (PA-1) は、3 次元の時間マッチング明示差分コードを並列化できる。PA-1 を使うと、Mesoscale Meteorology モデル (MM5) は 2 週間で並列化された (手作業では 3 年)。スケラブルな 2 次元の局所的酸性降雨モデルは 3 週間で並列化された。手作業での並列化の場合は、もっと単純な 1 次元のモデルでも 10 カ月かかる。

2.8 DOD

(1) HPC Modernization

正式には CIC R&D 予算配分の一部ではないが、DOD の HPC Modernization 計画 (HECC Working Group が実施) では、先進のハードウェアやコンピューティングツール開発、技術習得訓練を行い、戦闘員の支援に用いられる。高性能コンピューティングをシステム設計プロセスに取り入れ、米国の武器体系設計分野での技術的優位性を維持する。また、高性能コンピューティングをシステム収集の初期段階で使用することで、新しい戦闘支援システムの総ライフサイクルコストが削減される。

(2) 地球規模のシステムソフトウェア

1998 年度	1999 年度
サービス品質交渉プロトコルと適応型リソース発見プロトコルを開発する。	複数属性サービス品質仕様言語アーキテクチャの統合。
最高時の 30~50% の性能を達成する粗い粒度と細かい粒度の性能駆動型リソース割り当て機構のデモンストレーションを実施する。	層やネットワークの境界にまたがるサービス品質制約条件の経路ベースの伝播のデモ。

(3) 情報存続可能性

情報存続可能性に関し、次の活動に重点を置く。

- (a) 低電力の構成可能な計算要素のアーキテクチャの開発。
- (b) リアルタイムの適応制御とリソース管理のデモンストレーション。
- (c) スケラブル・ライブラリ技術をベースにした、防衛にとって重要なソフトウェアのリリース。

1999 年度の DARPA の計画を次に示す。

- (a) 自分で構成する自律システムのための新しい制御機構を調査する。
- (b) 低電力の構成可能なアーキテクチャを検証し、支援ソフトウェアを開発し、500,000 要素の自動化マッピングのデモを実施する。

3. アプリケーション

3.1 DARPA

(1) CHSSI (Common High Performance Computing Software Support Initiative)

DOD の HPC Modernization 計画の一環として実施される CHSSI は、スケーラブルなコンピューティングシステム用のソフトウェアを DOD の計算科学者および技術者に提供することを目的にしたアプリケーションソフトウェア開発計画である。これらの成果は、DOD の科学技術、開発テスト、評価計算を改善する。コアアプリケーションは、改善された 21 世紀の軍事機能に重点を置く。

(2) インテリジェントシステム技術

インテリジェントシステム技術分野での DARPA の活動の重点は、知識表現、推論、機械学習のための先進テクニックに置かれている。これによって、コンピュータによる口語、文語、画像の理解が可能になる。インテリジェントシステム技術の研究内容を次に示す。

- (a) 人間とコンピュータの対話技術。重点は、人間とコンピュータがより自然な形で対話できるようにする設計方法と可能化技術。
- (b) 堅牢な口語/文語技術のプロトタイプ。入手可能なダイアログ文法と理解に重点 (1998 年度)。
- (c) 危機管理と戦場での無線での会話の自動筆記を対象にした人間とコンピュータの間の対話 (1999 年度)。
- (d) 構成の自由度が高い計算要素と記憶要素を可能にする回路とソフトウェアに重点を置いたマイクロエレクトロニクス科学。
- (e) 立案、スケジューリング、リソース割り当てのための先進的方法。

3.2 NSF

(1) 複数の物理現象が関わる問題の高性能シミュレーション

NSF は、物理学者が基本構造から特性と機能を予測できるように、物理的に実現可能なシステムのシミュレーションを支援する。この共同研究には、Colorado 大のコンピュータ科学と計算機数学の研究者たちが関与している。

(a) 工学分野の研究開発(航空宇宙工学、土木工学、機械工学分野)

- ・モデリング方法、問題分解方法、並列シミュレーション方法の共通基盤
- ・広範な科学的視覚化
- ・共通ソフトウェア/ハードウェア・インフラ

(b) 計算機数学に関する研究の重点

- ・複数物理現象シミュレーション (multiphysics simulation) の理論的基礎
- ・インタフェース条件や高速非線形ソルバ、Staggered Multi-field Time Stepping に関する新しいアルゴリズムの設計

(c) コンピュータ科学研究の重点

アプリケーションの PSE (Problem Solving Environment) の開発に重点を置く。PSE は、大規模な複数物理現象シミュレーションの並列インプリメンテーションに適している。主な目標は、プロセッサ間通信、視覚化、デバッグ、性能監視などの、異種システムや分散システム上での大型の並列アプリケーションの実行に必要なリソースのすべてに対し、トランスペアレントでポータブルでスケーラブルなアクセスを実現することである。

(2) 複数モデル・複数領域の計算モデル

Old Dominion 大、Colorado 大、Notre Dame 大、New York 大 (NYU)、Boeing 社、Argonne 国立研究所が協力し、タスク並列 (粘性境界層と非粘性自由流などの複数のモデルが関係) 型とデータ並列 (複数の副領域が関係。付加的並行性のため) 型の領域分解方法を使って、航空力学分野と音響学分野の問題のためのソフトウェアラボラトリを開発する。

(3) スマートアンテナ

Brigham Young 大、Texas 大、Purdue 大では、セルラ通信など、デジタル無線アプリケーション用の基地局アンテナ関連研究開発を行っている。既存の商用システムの性能と理論上の最高性能の食い違いを明らかにし、圧縮アルゴリズムを改善する。アップリンク/ダウンリンク・シナリオを実際の環境でテストする。

(4) 分子生物学に適用される幾何学形状解析

Illinois 大で実施されている NSF の資金提供を受けた複数年の計算幾何学プロジェクトでは、分子生物学の研究グループが光収獲系タンパク質のビルディングブロックを調べるためにアルファ形状アプローチを使った。このアプローチによって、凝集に関与する面の位相的特性を失わずに、タンパク質の形状をユニークに識別し簡略化することが可能なことが証明されている。

3.3 NASA

(1) 先進スーパーコンピューティングアプリケーション

NASA は、先進スーパーコンピューティングアプリケーションに関する研究開発を支援する。研究分野を次に示す。

- (a) 地球の中心核とマントルの成長の3次元球体シミュレーション。John Hopkins 大との共同研究。数百年から数百万年の期間に渡って地球内部の進化を引き起こし、その結果、地球の表面を形成する、混沌としたプロセスをシミュレートする。
- (b) 合成開口レーダ (SAR, Synthetic Aperture Radar) のインターフェロメトリと画像科学への先進コンピューティング技術の適用。JPL (Jet Propulsion Laboratory) との共同研究。衛星が収集した SAR データの処理・視覚化を複数のスーパーコンピュータを使って行くと、アルプス型氷河やプレートテクトニクス、降雨林における地域的な変化を精密に監視できる。
- (c) 4次元のデータ同化 -- 高性能コンピューティングと現在のアルゴリズムの調査。
NASA の Goddard Space Flight Center と Maryland 大で実施。観測結果と気候モデル予測を組み合わせ、NASA の Earth Observing System 用の堅牢なデータ解析スキームを作成し、大気の正確な画像をあらゆる空間・時間について生成することに重点を置く。
- (d) 地球系モデルの開発 -- 大気/海洋力学とトレーサー化学 (tracer chemistry)。UCLA (California 大, Los Angeles) との共同研究。地球の気候のリアルな描写を目的にしたこの活動は、高い空間的解像度で次の4つの非常に複雑なモデルを開発し合体させる。大気大循環、海洋大循環、大気化学、海洋化学。
- (e) 極微重力環境における Rayleigh-Benard-Marangoni 問題のスケラブル並列有限要素計算。Texas 大との共同研究。低重力環境での流体流のモデリングによって、宇宙における高品質薄膜の製造とコーティングプロセスの効果や、宇宙ステーションの生命維持システムと安全システムが正常に機能するかどうかテストされる。
- (f) 星の内部の乱流対流とダイナモ。Chicago 大との共同研究。
この研究は、太陽のような星の内部のほとんど理解されていない基本的な乱流プロセスを調べる。
- (g) 太陽の活動と太陽圏力学 (heliospheric dynamics) の理解。Naval Research Laboratory との共同研究。NASA の観測結果を使って、太陽のコロナ (一番外側の層) の磁場に現れるもつれた3次元構造をモデリングする。
- (h) 太陽圏のマルチスケールモデリングのための並列適応方法、相対論的天体物理学用の多目的3次元コード、重力波天文学。Michigan 大との共同研究。コロナから自由に流れる恒星間媒体まで、コンピュータを使った研究によって、太陽風と銀河ガス (galactic gas) およびプラズマとの間の相互作用や、太陽風と太陽系内の磁化された天体や磁化されていない天体との間の相互作用を理解する。

- (i) 相対論的天体物理学と重力波天文学用の多目的 3 次元コード — 中性子星の合体 (すなわち融合) への適用。Illinois 大 (Urbana-Champaign) との共同研究。

このプロジェクトは流体力学と一般相対性原理を組み合わせて、2 つの中性子星の融合をコンピュータを使って調べる方法を開発する。この融合は相対論的天体物理学の多くの側面を包含するプロセスであるため、ブラックホールや超新星などの類似現象の研究の基礎となる。

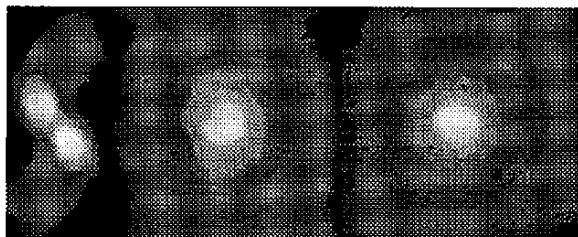


図 4 中性子星の融合のシミュレーション

中性子星の融合のシミュレーションを示す図 4 では、明るい色ほど物質密度は高く、暗い色ほど物質密度は低い。重力の影響を一部取り入れたシミュレーションは、重力放射の損失が融合の際の不安定な状態に影響することを示している。NASA が支援する研究開発で現在進行中の完全な相対性理論計算だけが、重力波信号を予測し、結果として起こるブラックホールの可能性を調べることができる。

(2) CAS (Computational Aerosciences)

CAS プロジェクトは、Grand Challenge 問題を解決するための、コンピュータを使った手法とアプリケーションを開発している。一例として、高速民間輸送と高性能航空機に関する進行中の研究がある。図 5 のような高性能コンピューティング技術によって生成される画像から、航空機のエンジンの各ブレードに関係するウェークシステムが明らかになる。

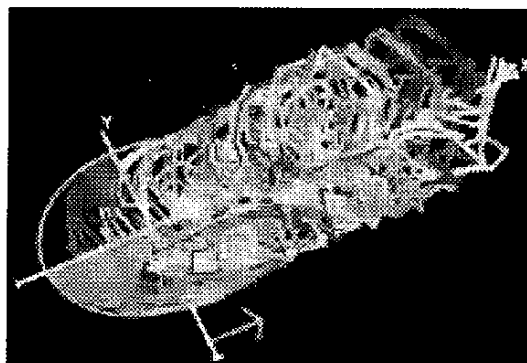


図 5 航空機エンジンのコンピュータ画像

このような画像は、ブレードと渦の間の相互作用に関する理解を研究者が深めるのに役立つ。

CAS は NAS (Numerical Aerodynamic Simulation) 計画および NREN (NASA Research and Education Network) と協力して、航空宇宙分野の研究者が NREN と CAS のテストベッドに確実にアクセスできるようにする。CAS プロジェクトはまた、入手可能性 (affordability) とサイクルタイムを対象とする新しいメトリクス (計測尺度) の共同開発を奨励している。高速民間輸送 (機体とエンジン) と高性能航空機 (機体と単純なエンジン) に関する研究が進行中である。

3.4 DOE

(1) DOE の Grand Challenge アプリケーション

Grand Challenge アプリケーションとしての研究開発は次に挙げる。

- ・量子色力学ソフトウェアの生の処理効率
- ・相対論的重イオン粒子衝突実験
- ・ASCI (Accelerated Strategic Computing Initiative) アプリケーション
- ・地球規模の変化のモデリングの視覚化。DOE はまた、コラボレーションツールと情報保証 (information surety) 手法に関する研究開発も実施する。1998 年度に DOE は、DOE 2000、Grand Challenge、ASCI、ACTS の R&D の活動を調整する。

3.5 NIH

(1) 分子構造予測とシミュレーション

1998 年度に、NIH の組織の NLM (National Library of Medicine)、NCRR (National Center for Research Resources)、DCRT (Division of Computer Research and Technology)、NCI (National Cancer Institute)、NIGMS (National Institute of General Medical Sciences) は医薬品業界での使用を対象にしたアブイニシオ構造予測の新方法を改良する。NIH は、水環境中のタンパク質、DNA、細胞膜複合体 (complex) のより大規模なシミュレーションを実施するための計算技術を改善する。

- (a) NCRR は、生化学、分子生物学、細胞生物学の分野の並列コンピューティングアプリケーションが、Web ベースのブラウザを通してスーパーコンピューティングシステムや大型データベースなどのリソースにアクセスできるようにする。したがって、レセプタサイト (receptor site) や他の医薬品設計 R&D に関係する複雑なシミュレーションは、インターネットを通して実行できる。
- (b) NCRR は、3 次元グラフィックスソフトウェア、磁気共鳴分光学データ解析と分子構造決定用のソフトウェアツール、他のソフトウェアツールを統合して、構造ベースの

医薬品設計に対し新しい機能を提供する。

- (c) CIT はタンパク質薬品の結合エネルギーを予測する方法を改善し、生物医学のアプリケーション用に高性能コンピューティング方法を開発する。
- (d) NCI は完全統合された高性能コンピューティングセンタに最新の機能を用意し、高性能の並列コンピューティング方法と通信方法を生物医学のアプリケーションに適用し、生物医学のアプリケーション用の新しいスケーラブル並列アーキテクチャを評価する。

(2) 3 次元 PET 画像の再構成

PET (Positron Emission Tomography: 陽電子放射断層撮影) では、インバース・プロBLEM (inverse problem) として知られている 1 組の投影測定から画像が再構成される。再構成される画像の品質の改善は、走査プロセスの統計的モデルを再構成アルゴリズムに取り入れ改善するが、ML (Maximum Likelihood: 最尤) 基準をベースにすることにより、ノイズの低減と解像度の改善が行える。

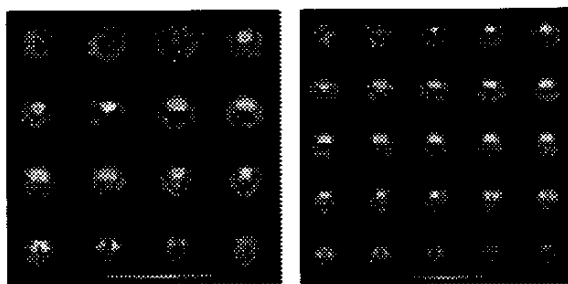


図 6 PET 画像の横断面

図 6 は、最尤アルゴリズムを使って再構成された 3 次元陽電子放射断層撮影 画像の横断面である。NIH によるこの研究では、グルコースのフッ素 18 類似体 (^{18}F -2-deoxy-D-glucose) を注射されたラットに対し、視覚的刺激が与えられた。高分解の皮質と皮質下の活性化が見られる。

PET は、マウスやラットを使ったゲノム操作の形質の決定に使われている。

NIH の Nuclear Medicine Department は、高解像度の小動物 PET 画像システムを、新しいシンチレーション結晶技術を使って開発している。画像の感度は、3 次元データの収集と ML ベースの再構成の使用の両方により改善されている。

CIT は、これらのスキャナのための 3 次元アルゴリズムを開発した。これらのアルゴリズムは、スキャナをモデリングする行列の内部の固有な希薄性 (sparsity) と対称性を利用する。再構成ソフトウェアはまた、問題を並列に分解し、CIT の IBM SP2 並列コンピューティングシステムの高バンド幅 I/O を利用する。初期の視覚的刺激の研究によって、前例のない皮質と皮質下の活動がマウスとラットの脳に見出された。

高性能コンピューティングシステムを視覚化ツールと組み合わせると、複雑な生物医学組織の理論的研究が可能になる。CIT は、この活動を 1998 年度に完了する計画である。Nuclear Medicine Department は、この新世代の小動物走査システムのために、かつては達成不能だった解像度 0.5 mm の目標を変化率 10%以内で達成することを試みる。1999 年度に CIT は、処理速度の増加の恩恵を受ける可能性がある生物医学分野のアプリケーションを対象に、方法とアルゴリズムの開発を継続する。これらのアプリケーションには、電子顕微鏡写真の画像処理、放射線療法計画作成、医療画像、タンパク質と核酸の配列解析 (sequence analysis) などがある。

図 7 はリポタンパク質の複合体モデルで、中央の脂質 (緑の部分) はタンパク質 (青と赤の部分) で取り囲まれている。HDL (高密度リポタンパク質) 粒子の構造に関する実験結果とタンパク質構造予測を組み合わせ、HDL の予備的モデルを作成した。1 次元の配列 (sequence) データからの 3 次元のタンパク質構造の予測には、高性能並列コンピューティング手法を使う。複雑な生物機能の理解には、大型分子複合体の構造と特性を決定することが必要である。

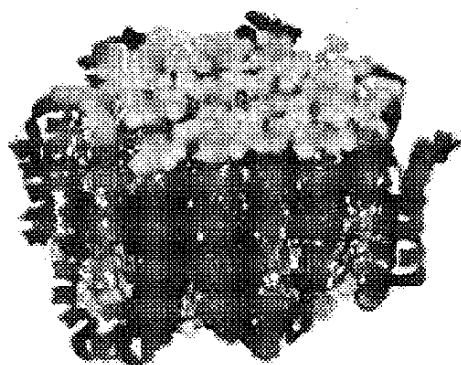


図 7 リポタンパク質の複合体モデル

図 8 は、磁気共鳴分光法 (magnetic resonance spectroscopy) によって決定されたプリオン・タンパク質の形態である。プリオンは新しいクラスの「感染性」病原体で、ウイルスとの違いはその構造と、それが引き起こす神経変成疾患 (neurodegenerative disease) である。プリオン疾患は、散発性・遺伝性・感染性の疾患として現れ、動物のスクレイピー、ウシ海綿状脳症 (狂牛病) や、クールー、クロイツフェルト・ヤコブ病 (CJD)、人間がかかる致命的な家族性不眠症などが含まれる。

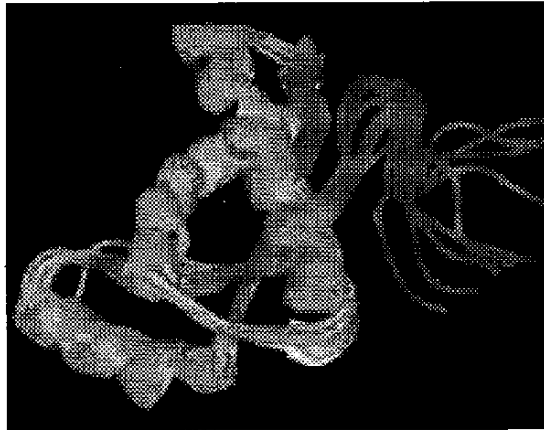


図8 プリオン・タンパク質の形態表示

(3) 画像管理と通信システム

CIT、NCI、NIH Clinical Center は、医療研究・教育用のマルチメディア通信をサポートするために、高速遠隔医療ネットワークを試作している。プロトタイプは、それぞれ 16 ポートの非同期通信モード (ATM、Asynchronous Transfer Mode) スイッチを 2 個含み、155 Mb/s のマルチメディア通信をサポートする。St. Louis の Washington 大は、プロトタイプ医療ワークステーションを含む ATM ベースのマルチメディア診察環境を開発した。このネットワーク環境は、当初は高性能放射線療法計画作成をサポートする。

(4) RCWS (Radiology Consultation Workstation)

RCWS は、リアルタイム遠隔医療診察用と医療教育用に特化された画像ワークステーションである。医療画像を高解像度表示するように設計された RCWS を使うと、X 線医師と他の医療専門家が前述の ATM ネットワークを介して遠隔診察を行えるようになる。専門家たちはカーソルにより臓器や病変を識別し、識別された部位は、診察に参加している他のすべての RCWS サイトにリアルタイムで送信され、放射線療法の計画が作成される。膨大な処理が必要な放射線療法計画作成の計算には、CIT の IBM SP2 並列コンピューティングシステムが使われる。

RCWS の目標の 1 つは、NIH のプロトタイプ ATM 遠隔医療ネットワークを、Washington, DC 地区の軍の医療センタと St. Louis の Washington 大に接続することである。この目標のために、NIH は、Washington, DC 地区を一周する ATDNet (Advanced Technology Development Network) に接続されている。NIH は、NASA の ACTS (Advanced Communication Technology Satellite) を今後利用する。ACTS は、NASA の GSFC (Goddard Space Flight Center) で ATDNet と接続し、Kansas City (MO) との接続を提供する。

1998 年度には、追加の RCWS ノードが Walter Reed Army Medical Center と National Naval

Medical Center に設置されて、この 2 つの施設と NCI との間でマルチメディア遠隔医療診察が可能になる。画像通信ソフトウェアが開発されて、RCWS と特定の商用放射線療法計画作成システムとの間でコンピュータ断層撮影と核磁気共鳴映像法 (MRI) の画像を交換できるようになる。1998 年度には、心臓病学のアプリケーションが 2 番目の医療特別問題として RCWS の適用対象になる。

3.6 EPA

(1) 環境保護

EPA の計画の目標は、環境保護ミッションの目標を達成するために HECC 技術の進歩を加速することである。多角的な環境モデリングのための問題解決環境に関する EPA の R&D は、大気の質のモデリングの枠組みに関する技術開発・統合の初期の成果を足掛かりにしている。

EPA の HECC R&D はスケーラブルな計算アルゴリズムを開発し、他の政府機関が開発した HECC 技術を統合して、地域問題解決環境と、多角的な複数汚染源・マルチスケールの環境曝露評価のための新しいパラダイムの実現可能性を調べる。EPA は、複雑な高性能環境管理ツールの基盤を構築するために、HECC 技術開発とテストに関する政府機関間の協力を促進する。

(2) EPA のスケーラブルソフトウェアライブラリ

EPA が支援するプロジェクトは、複雑で多角的な生態系モデルと人間曝露 (human exposure) モデルを地域において開発と使用が容易になるために、スケーラブルソフトウェアライブラリ、クロスプラットフォーム分散データ管理・処理方法、高性能地理空間解析を含む多変数視覚化手法、統合された問題解決の枠組みのためのモジュール式ソフトウェアを開発する。この枠組み技術によって、学際的な活動調整、データとコードの共有、統合された環境問題解決ツールへの新しい科学技術の迅速な導入が促進される。

1998 年度の EPA の計画を次に示す。

- (a) 並列ソルバと再利用可能な環境プロセス・モジュールのための 3 次元流体力学モデルと水質モデルをモジュール化する。
- (b) 並列処理に適した木/森林の成長・更新のコンポーネントベースのモデルの構築について調べる。
- (c) ラスタベースデータ (raster-based data) をクラスタ化するためと、クラスタに対し算術演算を実行するために、並列アルゴリズムを開発する。
- (d) レガシー環境モデリングコードをサポートするために、並列化エージェントを発表する。

(3)環境モデリング (Models-3)

Models-3 は、環境評価・意思決定支援ツールの開発・使用のためのソフトウェアシステムである。Community Multiscale Air Quality モデルにより、地表レベルのオゾン、酸性物の堆積 (acid deposition)、視界、微粒子について都市から地方までの規模で大気の水質のシミュレーションを実施する。Models-3 のコンポーネントは、大気の水質のモデリングのための各種の機能と互換性のあるソース放出リスト (source emission inventories) の設計と作成を支援する。

Models-3 の枠組みは、ユーザと運用モデルの間、科学者と開発モデルの間、ハードウェアとソフトウェアの間のインタフェースとして働くため、規制や政策の分析、大気化学と大気物理学の相互作用の理解などの各種の環境管理作業を実行しやすくする。この枠組みは、特化されたオブジェクトライブラリと、重要なシステムコンポーネントを分離する階層化設計を採用しており、ハードウェア/ソフトウェアのアップグレードの影響を最小にできる。クライアント/サーバ・アーキテクチャを、標準化されたデータインタフェースや、メタデータを格納するオブジェクト指向データベースと一緒に使うことで、スケーラブルコンピューティングプラットフォームのトランスペアレントな使用と、ネットワークを通じたデータへのアクセスを可能としている。このオブジェクトデータベースは、モデル領域、投影 (map projection)、グリッド解像度、化学薬品の種類などのアプリケーション固有のグローバルに共有されるデータも格納するため、モデルの詳細特性に対するユーザの管理を維持しながら科学コードを交換することが可能である。ライブラリベースのグラフィカル・ユーザインタフェースによって、モデルの実行が容易になり、各種の視覚化パッケージや解析パッケージへのアクセスも促進される。

1999年度のEPAの計画を次に示す。

- (a)空間的・時間的解像度が異なる並列タスクの間の命令とデータの交換の同期を取るためにフレームワークスーパーバイザを開発する。
- (b)マルチスペクトル/マルチソース衛星データとモデル予測データを視覚的に比較する手法を開発する。
- (c)空間/時間環境データの多次元マルチスケール表現のために時間的地理情報システムを試作する。
- (d)適応グリッド (adaptive grid) 手法に関係する交差を処理するための臨時的並列アルゴリズムの使用のデモンストレーションを実施する。
- (e)気象学モデルと水文学モデルをリンクする並列データ同化を開発する。

(4)帯水層のモデリング

1998年度は、解析モデリングの進歩により、多孔質媒体帯水層の地域規模の表現が拡張された。この研究は、井戸用のポイントシンク、川用のラインシンク、可変涵養 (variable recharge) 用の地域的要素などの水理学的特徴を表す多くの閉じた形式の解析解の重ね合

わせをベースにしている。これらのソリューションには、境界条件が特定の内部の境界に沿って満足されるように選択し組み合わせることが可能な自由度がある。並列処理コンピュータで3次元井戸要素を実用とする Indiana 大との共同研究により、スケーラブル並列コンピュータやパーソナルコンピュータで利用可能なパブリックドメイン解析要素研究コード (ModAEM) が開発された。Minnesota 大は、可変密度流を沿岸の帯水層の解析要素表現に実用化し、事例研究の一つとして、Upper Chesapeake の帯水層での淡水から塩水への変化をシミュレートしている。

図9のモデルは、層状の帯水層 (stratified aquifer) 中へ部分的に浸透している井戸への線対称の流れを示している。

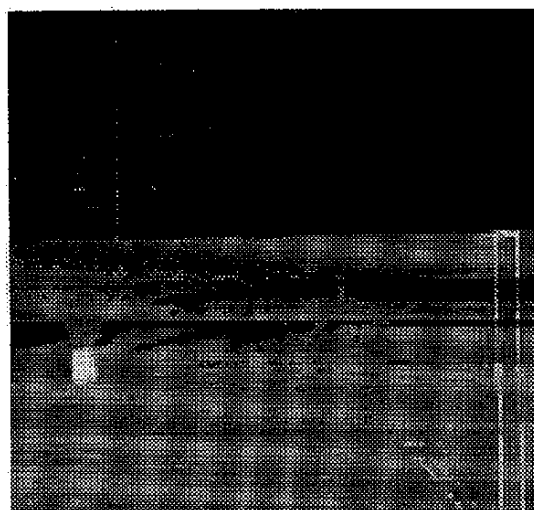


図9 帯水層に掘られた井戸への線対称の流れ

(5) 大規模環境流と輸送

Kentucky 大では、最小2乗有限要素法 (LSFEM, least-squares finite element method) と LES (Large Eddy Simulation) を組み合わせて乱流をシミュレートした。このデモは、3次元の剪断流問題と自然対流問題について実施され、最大 100 万個の有限要素と 700 万個以上の未知要素を含む大きな剪断流問題について、単一プロセッサの計算負荷を 75% 減らした。この堅牢な方法によって、乱流の予測と制御、気象学、気候の変動、大気汚染、水汚染、土壌汚染の分野でのより大きな流れと輸送の問題をより効率的に解決することが可能になる。

3.7 NIST

(1) 計算科学のためのオンライン参照データ

1998 年度は、数学ソフトウェアリポジトリを問題解決環境とし、基礎線形代数用のオブ

ジェクト指向ライブラリと関連機能を完成させ、分散システムの性能のデモを実施する。

さらに、科学プロセスをオンライン化するネットワークベースのリソースを継続して開発する。たとえば、Matrix Market (<http://math.nist.gov/MatrixMarket/>) は、線形代数のアルゴリズムとソフトウェアの比較研究に使える産業用アプリケーションから収集された大型の疎行列を、研究者とソフトウェア開発者に提供する。現在、数学参照データの対話型オンラインライブラリの基礎になるように、よく知られた NBS (National Bureau of Standards、NIST の前身) Handbook of Mathematical Functions の改訂・更新作業が行われている。改訂版には、コンピュータの代数システムやワードプロセッサへのダウンロードに適した、豊富な公式、グラフィックス、表が格納される。

(2) 高性能線形代数ソフトウェア

中核的な数値線形代数演算について、性能だけでなく有用性と保守性も高めるソフトウェア設計戦略に関する研究が継続して行われる。たとえば、産官学のコンソーシアムである BLAS (Basic Linear Algebra Subprograms) Technical Forum と協力し、疎行列カーネルとの標準インタフェースの仕様を定めている。このような標準があれば、メーカーやソフトウェア開発者は、移植可能なインタフェースを持つ特別にチューニングされた高性能カーネルを開発できる。また NIST は、C++ と Java による数値ツールキットの開発におけるオブジェクト指向ソフトウェア設計戦略の使用に関するデモも実施している。

3.8 NOAA

(1) NOAA におけるアルゴリズム開発

1998 年度、中規模の大気モデルで 5~10 キロメートルの解像度を達成するために、スケーラブルシステム上でのアルゴリズム開発を継続することを計画している。NOAA は、将来の予報システムの最適化のためのデータ同化のニーズをテストするために、次世代環境観測システムの設計を調査するとともに、従来の共有メモリマシンからスケーラブルシステムへの変換を促進するためのソフトウェアツールを開発する。NOAA は、Princeton (NJ) の GFDL (Geophysical Fluid Dynamics Laboratory) の高性能コンピューティングシステムを使って実施されている科学実験を強化し、さまざまなグリッド解像度での ETA モデルの性能評価と、実用的な予報での ETA モデルの可能性の評価を実施する。

(2) 天気予報の科学の推進

米国の気象観測データは、10 億ドルのリソースである。観測システムをシステム全体の観点から設計することは可能でなかったため、新技術の利用可能性が、観測システムの発展を推進してきた。NOAA の NAOS (North American Atmospheric Observing System) 計画の目標は、既存の観測リソースをもっと十分に利用するとともに、将来の観測システムの

発展をもっと意図的にコントロールすることである。

NAOS 計画で使われるモデルの開発を迅速化できるように、Boulder(CO)にある NOAA の FSL(Forecast Systems Laboratory) は SMS(Scalable Modeling System) ソフトウェアツールセットを開発した。SMS は、超並列コンピューティングシステムへの既存の数値天気予報モデルの転送が容易になるように設計されている。主な特長は、SMS を使って並列化されたモデルは、各種の高性能コンピューティングシステムへ移植可能なことである。1999 年度の FSL の計画は、高性能コンピューティングシステムを獲得すること、この新しいハードウェアを利用するために NWP(national weather prediction) モデルの強化を継続すること、そしてこの新しいハードウェアと解像度のより高いモデルを使って NAOS の実験を実施することである。

(3)GFDL (Geophysical Fluid Dynamics Laboratory)

大気中の温室効果ガス量の増加により地球が温暖化され、来世紀には未曾有の猛烈なハリケーンに見舞われる可能性がある。ほとんどのハリケーンは、陸上や温度の低い海域上で弱まる以前に、潜在的な最大勢力に達しない。しかし、最大勢力に達する一部のハリケーンは、海面の温度上昇に起因する温暖化が起こるとより勢力を増すことが予測される。

GFDL で実施した新しいシミュレーションによる見積もりでは、熱帯地方の海面の温度が 2℃ 余り上昇すると、最大勢力のハリケーンの風速は 5~12% 増加する。

地球温暖化に起因するハリケーンの最大勢力のこのような増加は理論的には 10 年余り前に提案されていたが、現実的なハリケーンの構造をシミュレートするために実際に使われているハリケーン予報モデルを使ってこの問題を調べたのは、この調査が初めてである。

GFDL は、世界的気候モデルからの情報を高解像度ハリケーン予報モデルとリンクさせることにより、現在の気候と温室効果ガスにより温暖化された気候の両方から選んだハリケーンの標本をシミュレートした(図 10)。

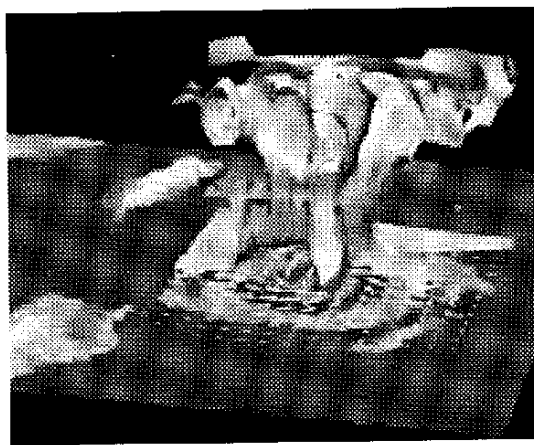


図 10 ハリケーンのシミュレーション

GFDL の高解像度ハリケーン予報モデルは、NOAA の National Centers for Environmental Prediction で過去数回のハリケーンシーズンに熱帯性嵐の進路予報に使われ、成果を上げている。また GFDL の気候モデルは、将来の気候に対する温室効果ガスの影響を見積もるために気候研究者によって使われている主要モデルである。

これらの成果は、高性能コンピューティングを必要とする 2 つの異なる分野 -- 気候変動とハリケーン予報 -- での研究をどのように組み合わせれば、世界的な気候変動が将来の気象状況に与える恐れのある影響についての新しい情報が得られるか、を示している。

4. インフラストラクチャ

4.1 はじめに

米国では、連邦政府機関により多数の CIC 研究施設が運営されており、その活動は成功していると考えられている。その成功要因として、つぎの理由が挙げられている。

- (1) 多くの可能化技術 (例、高速ネットワーク、スーパーコンピューティングシステム、並列アーキテクチャ、大容量データ記憶、仮想現実表示デバイス) 開発
- (2) 研究者や研究施設のスタッフ、ハードウェア/ソフトウェア・ベンダ、業界の外郭団体の献身

また、全研究施設が、幼稚園から大学の学部までを対象にした広範な教育機会と、研究者、大学院生、学部生のための訓練を提供し、またシステムとアプリケーションの両方について最適ベンチマーキングを実施している。

4.2 研究施設の要件

これらの施設は、次の作業を行う能力を有しているとされている。

- (1) 初期のプロトタイプシステムを評価しフィードバックを開発者に返す
- (2) 視覚化システムと仮想現実システムの既存高性能システムへの導入
- (3) 他の施設では利用できないシステムによるフルスケールアプリケーションの実行
- (4) スケールダウンされたシステムによる並列ソフトウェアの開発
- (5) 大規模アプリケーションのリモート接続による実行とネットワーク接続分野の革新的技術へのアクセス

4.3 研究施設の活動資金

これらの研究施設はその資金を主に CIC 研究開発政府機関から提供されているが、ハードウェア/ソフトウェア・ベンダから機器と人材の面で多くの支援を受け、また特定専門分野の政府機関からの資金、州や地元からの資金、業界の外郭団体からの寄付も得ている。

これらの施設は、CIC 技術の調査と最終的にはその利用のための低リスクの協力環境を提供している。

4.4 研究施設が行う活動例

(1) NSF

(a) PACI (Partnerships for Advanced Computational Infrastructure)

1998 年度に NSF は、標準的な主要大学で利用できる性能レベルより 2 桁優れた高性能コンピューティングを大学の研究者たちが利用できるようにするために、PACI を開始した。(PACI に関しては付属資料 5 を参照。)

(2) DOE

(a) HPCRP (High Performance Computing Resource Provider)

NERSC (National Energy Research Scientific Computing Center) は、DOE のエネルギー科学分野での計算ニーズを満足させるため、Argonne 国立研究所、Los Alamos 国立研究所、Oak Ridge 国立研究所の HPCRP を推進する。これらのサイトのシステムは、Intel の Paragon、IBM の SP2、Cray の T3E と C90、SGI の Origin 2000 などである。DOE の ESnet (Energy Sciences Network) は、初期の全国計算網の一部である。1998 年度に DOE は、生産性や有用性などの HPCRP の成果を評価する。

(b) ACTS (Advanced Computational Testing and Simulation Research) ツールキット

1998 年度、Sandia 国立研究所の計算生物学の研究などを行う。そこでは、ゲノムシーケンシングとタンパク質折り畳み (protein folding) のためのアルゴリズムの開発・提供に重点を置いている。1999 年度の計画は、ACTS インタフェース定義を中心に設計されたソフトウェアツールの試験的な使用などである。

(c) ASAP (Academic Strategic Alliances Program)

ASCI 計画の一部として ASAP があり、共通した重要な分野で CIC の R&D 計画と協力している。現在、ASAP は米国の 5 大学と共同で、高性能シミュレーション性能向上のための活動を展開しており、その詳細は本編 2.6.3 で述べた。

4.5 CIC R&D 研究施設一覧

ここでは、研究施設を主な資金源である政府機関別にまとめた。

(1) NSF

(a) Supercomputing Centers

1985 年度から 1998 年度まで NSF による資金により運営された。

- ・ Cornell Theory Center (CTC), Ithaca, NY
- ・ National Center for Supercomputing Applications (NCSA), Urbana-Champaign, IL
- ・ Pittsburgh Supercomputing Center (PSC), Pittsburgh, PA
- ・ San Diego Supercomputing Center (SDSC), San Diego, CA
- ・ Computing research facilities at the National Center for Astrophysical Research (NCAR), Boulder, CO

(b) PACI Centers

1998 年度から始まる。PACI は NSF Supercomputing Centers を引き継ぎ、HPCC 分野における新規課題をミッションとする。

- ・ National Computational Science Alliance (NCSA), Urbana-Champaign, IL
- ・ National Partnership for Advanced Computational Infrastructure (NPACI), San Diego, CA

(c) Science & Technology Centers

- ・ Center for Cognitive Science, Pennsylvania 大
- ・ Center for Computer Graphics and Scientific Visualization, Utah 大
- ・ Center for Research in Parallel Computation (CRPC), Rice 大

(2) NASA

(a) Testbeds

- ・ Ames Research Center, Moffett Field, CA
- ・ Goddard Space Flight Center, Greenbelt, MD
- ・ Jet Propulsion Laboratory, Pasadena, CA
- ・ Langley Research Center, Langley, VA
- ・ Lewis Research Center, Cleveland, OH

(3) DOE

(a) Laboratories

- ・ Argonne 国立研究所, Chicago, IL

- Los Alamos 国立研究所, Los Alamos, NM
- National Energy Research Supercomputer Center at Lawrence Berkeley 国立研究所, Berkeley, CA
- Oak Ridge 国立研究所, Oak Ridge, TN

(4) NIH

(a) Systems

- Frederick Biomedical Supercomputing Center at the National Cancer Institute
- Supercomputing resources at the Division of Computer Research and Technology (now part of the CIT)

(b) National Center for Research Resources' High Performance Computing Resource Centers

- Biomedical Computation Resource, California 大, San Diego
- Parallel Computing Resource for Structural Biology, North Carolina 大, Chapel Hill
- Parallel Processing Resource for Biomedical Scientists, Cornell Theory Center, Cornell 大
- Resource for Concurrent Biological Computing, Beckman Institute, Illinois 大
- Supercomputing for Biomedical Research, Pittsburgh Supercomputing Center
- Theoretical Simulation of Biological Systems, Columbia 大

(c) National Center for Research Resources' Scientific Visualization Resource Centers

- Interactive Graphics for Molecular Studies, North Carolina 大, Chapel Hill
- Special Research Resource for Biomolecular Graphics, California 大, San Francisco

(5) NOAA

(a) Laboratories

- Forecast Systems Laboratory, Boulder, CO
- Geophysical Fluid Dynamics Laboratory, Princeton, NJ
- National Centers for Environmental Prediction, Camp Springs, MD

(6) EPA

(a) Systems

- National Environmental Supercomputing Center, Bay City, MI

付属資料5 PACI (Partnerships for Advanced Computational Infrastructures)

1999 年度 Blue Book に掲載されている PACI の記述を紹介する。

1. 概 観

PACI は、NSF が推進するプログラムであり、計算科学分野で米国の優位を保つことを目的としている。実現手段として、すべての研究分野で、コンピューティングシステムの使用を促進するとともに、幼稚園から大学院までの生徒・学生を対象に、新しい教育機会を提供する。そのリーダー役として NCSA (本部は Urbana-Champaign の Illinois 大) と NPACI (UCSD (California 大-San Diego) 内) がある。

1.1 NCSA (National Computational Science Alliance)

NCSA が実現をめざす分散環境は、米国における最高の計算関連研究を可能にする全国的な情報インフラであり、それを試作することが目標である。NCSA は、以下の4つの主要グループから構成されている。

- (a) Application Technologies Teams : 技術開発を推進する
- (b) Enabling Technologies Teams : コンピュータ科学の研究成果を便利なツールやインフラに変える
- (c) Regional Partners : 全米各地のサイトへの技術の配付を支援する先進レベルと中間レベルのコンピューティングリソースを備えている
- (d) Education, Outreach, and Training Teams : それに米国民を教育し社会のさまざまな部門での技術の利用を促進する

Illinois 大にある最先端のサイトがサポートする各種のハイエンドマシンとアーキテクチャにより、全国の科学者と技術者がハイエンド計算を実行することが可能になる。

1.2 NPACI

(National Partnership for Advanced Computational Infrastructure)

NPACI は、多様なハードウェアサイトとハイエンド・サイトを持つ全国規模のメタコンピューティング環境を有している。NPACI は、UCSD (California 大-San Diego) にある各種の最先端マシンおよびアーキテクチャを通して、全米各地のハイエンドユー

ザである科学者と技術者の計算ニーズに対応する他、これらのハイエンドユーザが利用できるように、アプリケーションやコンピュータ科学者が開発した技術とツールの移転を促進する。主な重点分野は、データ中心型のコンピューティング、デジタルライブラリ、工学と社会科学を含む複数の分野にまたがる大型データ・セット操作で、全国のパートナーからの支援を受ける。

2. NCSA (National Computational Science Alliance)の構成

NCSA には、50 以上の米国の大学と研究機関の計算科学者、コンピュータ科学者、教育・訓練、現場出張サービスの専門家が参加している。NCSA の国家的技術 grid-- 21 世紀の計算/情報インフラのプロトタイプ -- は、Urbana-Champaign の Illinois 大や NCSA 内部の他の最先端施設に置かれたさまざまなハイエンド並列コンピューティングシステムから構成される。

2.1 NCSA のパートナー

NCSA のパートナーのほとんどは、大学と政府の研究所である。これらの機関の研究者たちは、grid 用の技術を開発し、それを米国各地のユーザ団体に普及する。連邦政府と業界のパートナーや戦略的ベンダは、新しい技術をテストし、工業面と商業面の可能性をフィードバックする。

2.1.1 Research partners

(1) 大学

Arizona 州立大、Boston 大、California 州立大 - San Diego、California 工科大
Georgia 工科大、Indiana 大、Massachusetts 工科大、Montana 州立大
North Carolina 州立大、Northwestern 大、Ohio State 大、Old Dominion 大
Oregon 州立大、Princeton 大、Purdue 大、Rice 大、Rutgers 大、South Dakota 州立大、Southeastern 大 Research Association、Stanford 大、
New York 州立大 - Albany、Syracuse 大、Alabama 大 - Huntsville、Arizona 大
California 大 - Berkeley - Davis - Irvine - Los Angeles - Santa Barbara - Santa Cruz - San Diego - San Francisco
Houston 大、Illinois 大 - Chicago - Urbana-Champaign、Iowa 大、Kansas 大
Kentucky 大、Maryland 大、Massachusetts 大、Michigan 大、Minnesota 大

New Mexico 大/Sevilleleta Long-Term Ecological Research Program/
 North Carolina 大- Chapel Hill、Oregon 大、Pennsylvania 大、Pittsburgh 大
 Southern California 大、Tennessee 大、Texas 大 - Austin、Utah 大、Virginia 大
 Washington 大、Wisconsin 大- Madison、Virginia 技術大、Washington 大-St. Louis

(2) 研究所

Argonne 国立研究所

Lawrence Berkeley 国立研究所 (LBNL)/National Energy Research Supercomputer Center、Lawrence Livermore 国立研究所 (LLNL)、Los Alamos 国立研究所 (LANL)
 North Central Regional Education Laboratory、National Radio Astronomy Observatory、Pacific Northwest National Laboratory (PNL)/Environmental Molecular Sciences Laboratory、Salk Institute、Shodor Education Foundation、The Scripps Research Institute、American Indian 高等教育コンソーシアム、BBN Systems and Technologies、Center for Advanced Research in Biotechnology (CARB)、Committee on Institutional Cooperation、Computing Research Association、EPSCoR Foundation、Jet Propulsion Laboratories、Kitt Peak National Observatory、Krell Institute、Maryland Virtual High School of Science and Math、Maui High Performance Computing Center、North Carolina Supercomputing Center、Ohio Supercomputer Center、Alabama Supercomputer Authority

2.1.2 Industrial Partners

フォーチュン 500 にのる 15 社が開発技術を見学するため、NCSA とパートナーシップを組んでいる。

企業名	Fortune 500 Industrial Sector
Allstate Insurance Company*	Insurance
Boeing Company*	Aerospace
Caterpillar Inc. *	Industrial and farm equipment
Eastman Kodak Company*	Scientific and photographic equipment
Eli Lilly and Company*	Pharmaceuticals
FMC Corporation	Agricultural chemicals/diversified
Ford Motor Company	Automobiles
J.P. Morgan*	Banking/finance
Motorola, Inc.*	Electronics
Phillips Petroleum Co.	Petroleum and support
SABRE Group, Inc.	Transportation

Schlumberger Ltd.*	Petroleum and support
Sears, Roebuck and Co.*	General merchandise
Shell Oil Company*	Petroleum and support
Tribune Company*	Information/entertainment

* Have signed letters of intent to join the Alliance industrial partnership

2.1.3 戦略的ベンダ

情報、通信、ソフトウェアの主要な企業を含む Commercial vendors は、アライアンスパートナーとして密接に作業しており、ハード/ソフト、サービスの初期のプロトタイプの開発と評価を行っている。協力は、アライアンスというより個々の機関間でなされている。

Ameritech、Computer Associates、Hewlett-Packard Company/Convex Technology Center
IBM Corporation、Microsoft Corporation、Platform Computing Corporation
The Portland Group, Inc.、Pyramid Systems, Inc.、Silicon Graphics, Inc./Cray
Research、Sun Microsystems

2.1.4 連邦政府のパートナー

NCSA と連邦政府機関は、米国政府内で先進技術を開発し適用している。連邦政府のパートナーを次に示す。

- (1) DoD (国防総省) の High Performance Computing Modernization Program
- (2) World Wide Web Federal Consortium
- (3) DOE (エネルギー省) の ASCI (Accelerated Strategic Computing Initiative)

3. NPACI (National Partnership for Advanced Computational Infrastructure) の構成

NPACI は、科学と経済の発展における世界的な指導力を米国が維持していく助けになる米国の団体による学際的研究を支援するために、くまなく普及する連続的な計算インフラを構築している。NPACI の目的は、徐々に高性能化するシステムになるようにリンクされる異種システムから構成されるこのインフラを、トランスペアレントで、使いやすく、どこでも利用可能なものにすることである。

UCSD (California 大-San Diego) をリーダとし、SDSC (San Diego Supercomputing Center) を基盤として利用する NPACI には、米国の 37 の大学機関と研究機関 (18 の州に存在) が参加している。アプリケーションニーズによって推進される NPACI は、処理パワーを一つにまとめるために、高性能コンピュータやデータ・サーバ、アーカイバル記憶システムをリンクするソフトウェア・インフラを開発している。開発作業の重点は、生み出されるインフラが堅牢なものとなり迅速に配置されるように、個別に資金提供を受ける研究プロジェクトを利用する HECC 分野 (スラスト) に置かれている。この開発作業は、教育・訓練や現場出張サービスの計画や、業界との協力によって補完される。

NPACI は、計算資源、研究、教育のパートナーから構成され、準会員と業界のパートナーも含まれる。

3.1 Resource partners

UCSD/SDSC、California 工科大

California 大 - Berkeley - Davis - Los Angeles - Santa Barbara

Houston 大/Keck (with Baylor and Rice 大)、Maryland 大、Michigan 大、Texas 大

Washington 大- St. Louis

3.2 Research and education partners

California 州立大、Center for Research on Parallel Computation

LTER/New Mexico 大、Oregon 州立大、Rutgers 大、Salk Institute、Stanford 大

The Scripps Research Institute、California 大 - Santa Cruz、Kansas 大、Southern

California 大、Tennessee 大、Virginia 大、Wisconsin 大

3.3 準パートナー

NPACI 準パートナーには、付加的な連関とインフラの共有資格が与えられる。

CARB、EPSCoR Foundation、JPL、Kitt Peak National Observatory (KPNO)、LANL、LBL/NERSC、LLNL、Montana 州立大、PNL/Environmental Molecular Sciences Laboratory (EMSL)、California 大 - Irvine - San Francisco、Massachusetts 大、Pennsylvania 大

3.4 業界パートナー

NPACI の業界パートナーは、NPACI の研究課題の決定を支援するとともに、かなりの額の経費を分担する。これらのパートナーは、次のような協力を通して NPACI に参加している。

(1) SGI/Cray は UCSD と一緒に、Cray の GigaRing 技術を異種コンピューティング環境に組み入れている。

(2) Hewlett-Packard/Convex と Caltech は、大きなキャッシュコヒーレント (cache-coherent) 領域をサポートするために Convex のシステムソフトウェアを強化している。

(3) IBM は、オブジェクト・リレーショナルデータベースシステムを大容量記憶システムと統合するためと、Massive Data Analysis Testbed を開発するために、UCSD との協力を継続している。

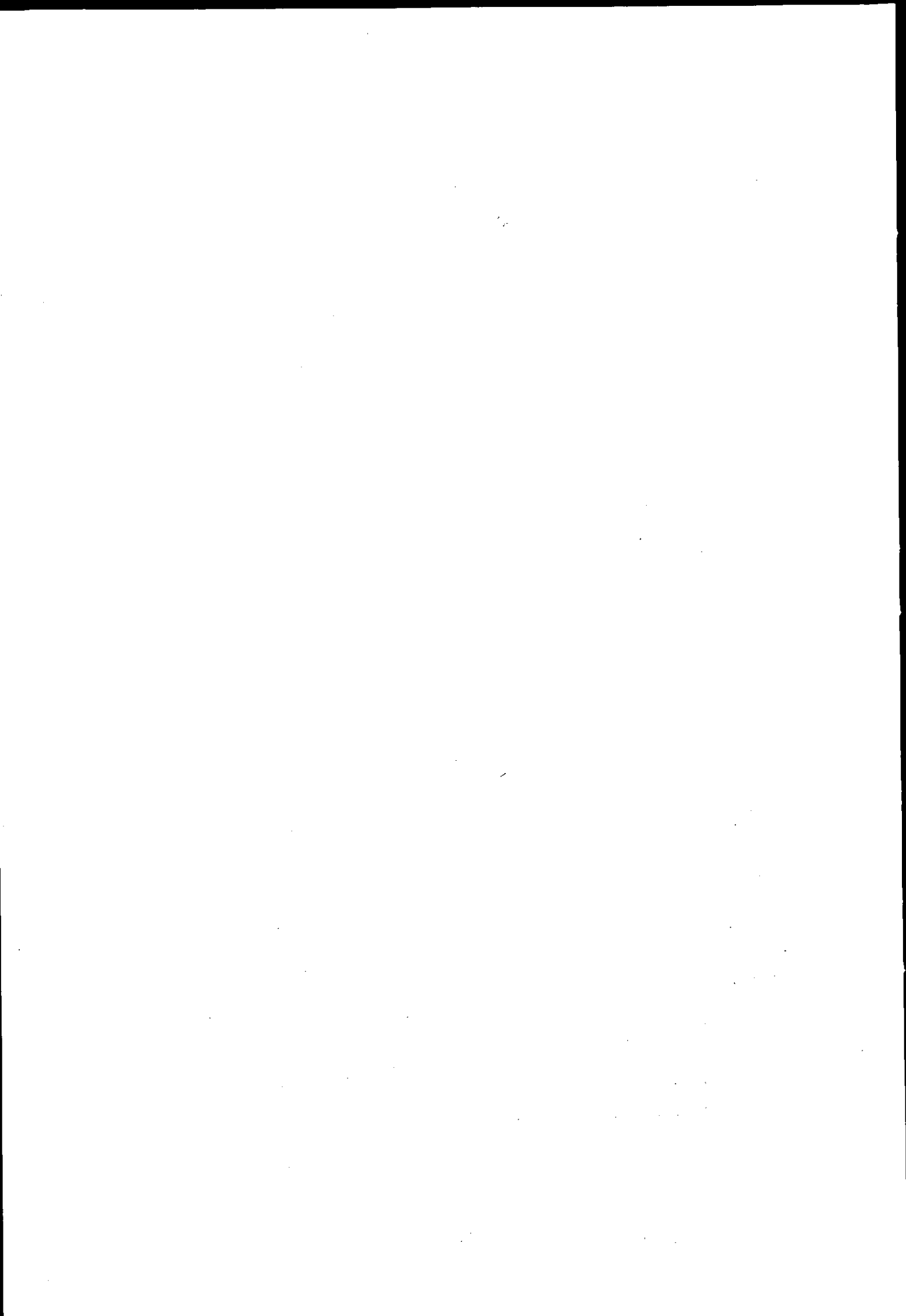
(4) Sun Microsystems は、NOW (Network of Workstation) に関する UC Berkeley との協力と、セキュリティに関する UCSD との協力を継続している。Sun は、データサービング (data serving) に関する Sun の技術の有効性を実証するために、UCSD との新しい協力を開始した。

(5) DEC は、科学的環境における Unix の代替としての NT オペレーティングシステムの使用に関する UCSD との共同調査を継続する。

(6) Seagate などの大手ディスクメーカーは UCSD と一緒に、UCSD の数 TB (テラバイト) の記憶リソースをテストベッドとして使ってディスクの信頼性を調べる。

付属資料6 ワーキンググループ資料リスト

資料番号	資料タイトル	配付日	備考
PFM-WG-2001	ペタフロップスマシン技術調査 WG 資料	98.10.20	山口主査運営方針他
PFM-WG-2002	TI の 0.07 μ m CMOS 量産化技術報道について	98.10.20	AITEC 提出資料
PFM-WG-2003	各階層における重要課題は何か	98.10.20	AITEC 提出資料
PFM-WG-2004	スーパーコンパイラシステムに関する調査研究	98.10.20	武田氏 OHP コピー
PFM-WG-2005	最新の半導体デバイス開発進捗状況について	98.11.27	坂本氏 OHP コピー
PFM-WG-2006	デカナノレベル U L S I 技術開発の現状と問題点	98.11.27	鳥海氏 OHP コピー
PFM-WG-2007	ディクテーション技術の概要と動向	98.12.25	電総研伊藤氏配付資料
PFM-WG-2008	音声処理技術研究の現在及び将来	99.1.18	伊藤氏 OHP コピー
PFM-WG-2009	ローバルコンピューティング技術の現在と将来	98.1.18	高木氏 OHP コピー
PFM-WG-2010	PITAC 中間報告 エグゼクティブサマリ日本語訳	98.1.18	AITEC 提出資料
PFM-WG-2011	PITAC 中間報告 オリジナル全文	98.1.18	AITEC 提出資料
PFM-WG-2012	自動音声翻訳技術の現状について	98.1.18	ATR 隅田氏 OHP コピー
PFM-WG-2013	ATR-MATRIX spontaneous speech translation system	98.1.18	同上
PFM-WG-2014	情報処理 11、1998	98.1.18	東大喜連川氏配布資料
PFM-WG-2015	PITAC 中間報告 全文日本語訳	98.2.5	AITEC 提出資料
PFM-WG-2016	平成 10 年度報告書「題名と概要」予稿	98.2.15	AITEC 提出資料
PFM-WG-2017	超大容量・超高性 データベース	98.2.15	喜連川氏 OHP コピー



本書の全部あるいは一部を断りなく転載または複写（コピー）することは、
著作権・出版権の侵害となる場合がありますのでご注意下さい。

ペタフロップスマシン技術に関する調査研究 Ⅲ

©平成11年3月発行

発行所 財団法人 日本情報処理開発協会

先端情報技術研究所

東京都港区芝2丁目3番3号

芝東京海上ビルディング4階

TEL(03)3456-2511

