

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

=====

BÁO CÁO THỰC HIỆN ĐỀ TÀI

NGHIÊN CỨU CÁC HỆ THỐNG TÍNH TOÁN HIỆU NĂNG CAO VÀ ỨNG DỤNG MÔ PHỎNG VẬT LIỆU VI MÔ

Trong Khung Khổ Hợp Tác Khoa Học Công Nghệ
Theo Nghị Định Thư Với Ấn Độ Giai Đoạn 2004 – 2005

Chủ Nhiệm Đề Tài: PGS. TS. Nguyễn Thanh Thủy

5957

25/7/2006

Hà Nội, 3-2006

Mục lục

Mục lục	1
Danh mục hình.....	5
1. LỜI MỞ ĐẦU	9
2. NỘI DUNG CHÍNH CỦA BÁO CÁO.....	13
2.1. Tổng quan tình hình nghiên cứu trong và ngoài nước	13
2.1.1. Tổng quan tình hình nghiên cứu về tính toán hiệu năng cao	13
2.1.1.1. Nhu cầu xây dựng hệ thống tính toán hiệu năng cao.....	13
2.1.1.2. Các kiến trúc máy tính song song phổ biến	15
2.1.1.3. Các ứng dụng tính toán song song trong khoa học, công nghệ và môi trường	38

2.1.2.	Chống lỗi trong các hệ thống tính toán hiệu năng cao	40
2.1.3.	Tổng quan tình hình nghiên cứu về tính toán lưới	43
2.1.3.1.	Sự ra đời của tính toán lưới	43
2.1.3.2.	Những hoạt động liên quan đến tính toán lưới.....	48
2.1.3.3.	Các dự án lớn trên thế giới về tính toán lưới	50
2.1.3.4.	Các trung tâm khai thác tính toán lưới	64
2.1.3.5.	Các diễn đàn, hội thảo	68
2.1.3.6.	Tình hình nghiên cứu về tính toán lưới ở Việt Nam	74
2.1.4.	Tổng quan về mô phỏng trong khoa học vật liệu.....	87
2.2.	Những nội dung đã thực hiện	91
2.2.1.	Kết quả nghiên cứu triển khai tính toán hiệu năng cao	91

2.2.1.1.	Kết quả nghiên cứu, thiết lập hệ thống tính toán song song ghép cụm Bkluster	91
2.2.1.2.	Gói phần mềm BKlusware.....	93
2.2.2.	Kết quả nghiên cứu triển khai chống lỗi	114
2.2.3.	Kết quả nghiên cứu triển khai tính toán lưới	122
2.2.3.1.	Tiếp cận dịch vụ trong triển khai lưới	122
2.2.3.2.	Nghiên cứu xây dựng hệ thống BKGrid 2005	131
2.2.3.3.	Sản phẩm phần mềm	144
2.2.4.	Kết quả nghiên cứu triển khai mô phỏng vật liệu	153
2.3.	Tổng kết và đánh giá kết quả thu được.....	159
2.4.	Kết luận và kiến nghị	161
2.4.1.	Kết luận	161

2.4.2. Kiến nghị	162
Danh sách các bài báo	164
Tài liệu tham khảo	168

Danh mục hình

Hình 2-1 Sơ đồ khối của vi xử lý kiểu vector	20
Hình 2-2 Mô hình chung của các máy tính DM-SIMD	22
Hình 2-3 Một số cách kết nối trong các máy tính SM-MIMD	24
Hình 2-4 Một số cách kết nối sử dụng trong các máy DM-MIMD	30
Hình 2-5 Kiến trúc mô hình cluster.....	33
Hình 2-6 Thống kê về hiệu năng của 500 siêu máy tính mạnh nhất Thế Giới tháng 6 – 2005.....	35
Hình 2-7 Tỷ lệ các kiến trúc máy tính trong top500 (06/2005)	38
Hình 2-8 Các lĩnh vực ứng dụng chính của 500 siêu máy tính mạnh nhất Thế Giới (06/2005).....	39

Hình 2-9. Một quan sát về tính toán lưới	46
Hình 2-10. Dự án BioGrid (http://www.biogrid.jp/).....	58
Hình 2-11. Dự án DOE Science Grid (http://doesciencegrid.org/)	61
Hình 2-12. Dự án GridPP (http://www.gridpp.ac.uk/)	63
Hình 2-13. Trung tâm London e-Science (http://www.lesc.ic.ac.uk/).....	66
Hình 2-14. Hội thảo quốc tế lần thứ 7 về tính toán lưới	71
Hình 2-15. Bioinformatics Grid Portal của Phân viện Công nghệ thông tin tại TPHCM	81
Hình 2-16. Sơ đồ hệ thống Grid của Phân viện CNTT tại TPHCM	82
Hình 2-17. Sơ đồ kết nối hệ thống grid với hệ thống grid KIST	84
Hình 2-18 - Các thành phần chính của BKlusware	96
Hình 2-19 - Cửa sổ theo dõi truyền thông điệp giữa các tiến trình	101

Hình 2-20 - Giao diện công cụ đệ trình công việc.....	104
Hình 2-21 Giao diện quản trị thông tin về nút tính toán của hệ thống	108
Hình 2-22 -Triển khai BKlusware trên hệ thống BKluster.....	114
Hình 2-23 Hoạt động của BKluster với môi trường tính toán song song chống lỗi	117
Hình 2-24 Khởi động lại công việc với BKFT	119
Hình 2-25 Giao diện chương trình BKFT - chức năng lấy checkpoint	120
Hình 2-26 Giao diện chương trình BKFT - chức năng restart.....	121
Hình 2-27. Tổng quan phương pháp luận về lưới hóa ứng dụng	126
Hình 2-28. Sáu bước lưới hóa ứng dụng.....	126
Hình 2-29. Kiến trúc của Java Cog Kit.....	135
Hình 2-30. Cổng điện tử Grid Portal	139

Hình 2-31. BKGrid 2005 và nền tảng phát triển	147
Hình 2-32. Kiến trúc BKGrid 2005 và các luồng thông tin.....	149
Hình 2-33. Kịch bản tương tác trong chức năng chạy ứng dụng Weka.....	152
Hình 2-34 Cấu trúc địa phương của mẫu Al_2O_3 lỏng.....	156
Hình 2-35 Phân bố lỗ trống trong hệ xỉ Al_2O_3	158

1. LỜI MỞ ĐẦU

Tính toán hiệu năng cao và tính toán lưới là một trong những lĩnh vực nghiên cứu đang được quan tâm hiện nay trên thế giới. Hiểu một cách đơn giản, tính toán lưới và tính toán hiệu năng cao cho phép tận dụng tối đa tài nguyên tính toán và lưu trữ để tạo ra một hệ thống đáp ứng yêu cầu ứng dụng trong thời gian nhanh hơn, với chi phí thấp hơn. Trong vài năm trở lại đây, tính toán lưới nổi lên như một lĩnh vực nghiên cứu, triển khai ứng dụng nóng hổi, thu hút sự quan tâm và chú ý không chỉ của các viện nghiên cứu, mà cả của các doanh nghiệp lớn, với hi vọng về một môi trường lưới toàn cầu như Internet hiện nay.

Tính toán hiệu năng cao và tính toán lưới góp phần giúp các ngành khoa học như Vật lý, Hoá học, Sinh học, Thiên văn học, Khí tượng thuỷ văn trong việc triển khai các nghiên cứu, thực nghiệm đòi hỏi năng lực tính toán mạnh và đã đạt được những kết quả ban đầu, cả về nghiên cứu cũng như ứng dụng khá thuyết phục. “Dịch vụ theo yêu cầu của doanh nghiệp” (“service on business

demand”) đang là một xu thế được nhiều doanh nghiệp, công ty chấp nhận, do sự hấp dẫn về giải pháp kỹ thuật, không phải đầu tư lớn.

Ở Việt Nam, những nghiên cứu và ứng dụng tính toán hiệu năng cao và tính toán lưới đã bắt đầu được quan tâm ở một số cơ quan nghiên cứu, trường đại học, thậm chí một số công ty tin học. Trong khuôn khổ đề tài nghiên cứu theo Nghị định thư với Trung tâm Nghiên cứu Phát triển các Công nghệ tính toán tiên tiến của Ấn độ, Trung tâm Tính toán Hiệu năng cao, Đại học Bách Khoa Hà Nội đã thực hiện đề tài “Nghiên cứu các Hệ thống Tính toán lưới, Tính toán hiệu năng cao và ứng dụng mô phỏng vật liệu vi mô”, nhằm mục tiêu hướng tới làm chủ các công cụ tính toán hiệu năng cao, tính toán lưới và cung cấp các ứng dụng tính toán hiệu năng cao cho các đơn vị nghiên cứu, triển khai ứng dụng khi có nhu cầu. Đề tài hướng tới các nội dung chính là:

- Làm chủ công nghệ tính toán hiệu năng cao, tính toán lưới nhằm nâng cao chất lượng dịch vụ tính toán: Chia sẻ tài nguyên tính toán và thông

tin; Chống lỗi và khôi phục hệ thống xử lý khi có sự cố; Hỗ trợ người phát triển ứng dụng không chuyên CNTT.

- Tạo lập mô hình hợp tác chuyên môn trong nhóm chuyên gia đa ngành: CNTT -Toán ứng dụng- Công nghệ sử dụng tính toán hiệu năng cao và tính toán lưới giải quyết bài toán ứng dụng.
- Góp phần nâng cao chất lượng đào tạo trình độ cao trong lĩnh vực tính toán hiệu năng cao và tính toán lưới.

Xuất phát từ các mục tiêu cụ thể ở trên, nội dung nghiên cứu của đề tài sẽ tập trung vào những nội dung sau đây:

1. Thiết kế kiến trúc, xây dựng Phần mềm Máy chủ tính toán đáp ứng yêu cầu :

- Nhiều người sử dụng cùng một lúc
- Hỗ trợ người sử dụng không biết sâu về lập trình song song
- Cung cấp dịch vụ tính toán từ xa.

2. Cấu hình cơ sở hạ tầng (cấu hình hệ thống và phần mềm nền) tính toán lưới thử nghiệm:

- Cấu hình hạ tầng thiết bị: nút tính toán và mạng
- Cài đặt phần mềm hệ điều hành cho nút tính toán và cluster
- Cài đặt phần mềm nền tính toán lưới
- Thử nghiệm kết nối

3. Thiết kế và xây dựng môđun chống lỗi đáp ứng yêu cầu:

- Trong suốt với các chương trình ứng dụng
- Ít ảnh hưởng đến hiệu năng hoạt động tổng thể của hệ thống
- Thời gian khôi phục lỗi nhanh.

4. Chương trình mô phỏng vật liệu vi mô

2. NỘI DUNG CHÍNH CỦA BÁO CÁO

2.1. Tổng quan tình hình nghiên cứu trong và ngoài nước

2.1.1. Tổng quan tình hình nghiên cứu về tính toán hiệu năng cao

2.1.1.1. Nhu cầu xây dựng hệ thống tính toán hiệu năng cao

Nhu cầu ứng dụng Công nghệ thông tin trong các hoạt động nghiên cứu khoa học và công nghệ, quản lý kinh tế xã hội ngày càng cao, đòi hỏi phải giải quyết nhiều các bài toán xử lý lớn với khối lượng tính toán khổng lồ đến mức các máy tính tuần tự (sequential computer) không đủ mạnh để có thể đưa ra một phương án lời giải trong giới hạn thời gian cho phép. Ý tưởng xây dựng các máy tính song song tuy đã xuất hiện từ lâu trong các nghiên cứu, nhưng mới chỉ trở thành hiện thực trong vòng vài chục năm gần đây

Để giải quyết các ứng dụng đòi hỏi lượng tính toán lớn, một giải pháp được đưa ra là sử dụng những máy tính song song có nhiều bộ xử lý hoạt động đồng thời. Tuy vậy giải pháp này nhiều khi không khả thi do hạn chế về tài chính.

Các nghiên cứu về kiến trúc máy tính và mạng cho thấy cơ chế song song được sử dụng không chỉ trong các siêu máy tính mà ngay cả trong mạng với các trạm làm việc là máy tính cá nhân. Khi đó, các chương trình ứng dụng tận dụng được không những các bộ xử lý trên cùng một máy tính, mà còn cả các bộ xử lý khác nhau trên các trạm khác nhau trên mạng. Bởi vậy, khả năng chạy đồng thời các tiến trình công việc và đồng bộ chúng trở thành một trong những đòi hỏi cơ bản trong các giải pháp nâng cao hiệu năng của hệ thống. Một ưu điểm cơ bản của tiếp cận này là khả năng mở rộng (scalability) do khả năng cho phép các phần mềm ứng dụng tận dụng tối đa các bộ xử lý trên các nút mạng, một nhân tố quan trọng nhằm chuyển đổi và thu hồi chi phí đầu tư cho các ứng dụng.

2.1.1.2. Các kiến trúc máy tính song song phổ biến

Nguyên tắc phân loại kiến trúc máy tính song song

Máy tính song song là một tập các phần tử tính toán có khả năng truyền thông và kết hợp với nhau để giải quyết các bài toán lớn trong khoảng thời gian chấp nhận được (Almasi and Gottlieb – 1989).

Mặc dù đều dựa trên nguyên lý phối hợp nhiều tài nguyên tính toán để giải quyết bài toán, các máy tính song song có thể có nhiều kiến trúc khác nhau. Nguyên tắc phân loại kiến trúc máy tính song song theo Flynn (1996) dựa trên hai yếu tố cơ bản: lệnh và dữ liệu: Đơn lệnh đơn dữ liệu (SISD - Single Instruction Single Data), đơn lệnh đa dữ liệu (SIMD - Single Instruction Multiple Data), đa lệnh đơn dữ liệu (MISD- Multiple Instruction Single Data) và đa lệnh đa dữ liệu (MIMD-

Multiple Instruction Multiple Data). Trong các kiến trúc trên, kiến trúc SISD tương ứng với máy tính tuần tự. Cho tới thời điểm này, kiến trúc MISD vẫn chưa có một phiên bản chính thức nào được xây dựng hay nghiên cứu phát triển. Các máy tính song song chủ yếu thuộc một trong hai kiến trúc SIMD hoặc MIMD.

Kiến trúc SIMD: Các máy tính SIMD chứa nhiều bộ vi xử lý (các máy tính SIMD tiên tiến có số CPU thay đổi từ 1024 đến 16384). Tại một thời điểm xác định, các bộ vi xử lý thực hiện một lệnh giống nhau tác động lên các đơn vị dữ liệu khác nhau. Sản phẩm tiêu biểu của dòng máy tính SIMD là CPP DAP Gamma hay Quadrics Apemille. Ví dụ khác về kiến trúc SIMD là các bộ xử lý kiểu vector (vectorprocessor). Các bộ vi xử lý kiểu này được thiết kế đặc biệt để xử lý dữ liệu kiểu mảng (tập các phần tử dữ liệu cùng kiểu). Trong chế độ vector, dữ liệu được xử lý song song với tốc độ nhanh hơn nhiều lần so với xử lý dữ liệu kiểu vô hướng. Ví dụ điển hình về máy tính với vi xử lý kiểu vector là các sản phẩm thuộc dòng NEC SX-6i.

Kiến trúc MIMD: Các máy tính song song kiến trúc MIMD có khả năng thực hiện các lệnh khác nhau đối với các dữ liệu khác nhau tại một thời điểm. Do đó,

các máy tính MIMD có khả năng rút ngắn thời gian thực hiện một công việc bằng cách chia công việc đó ra thành những phần việc nhỏ hơn và thực hiện đồng thời các phần việc đó. MIMD có thể xem là kiến trúc máy tính song song phổ biến nhất và được phát triển mạnh nhất, từ những máy tính 4 bộ vi xử lý như NEC SX-6 đến những máy tính với hàng ngàn bộ vi xử lý như IBM p690.

Nhìn từ khía cạnh người phát triển các ứng dụng song song, các máy tính song song có thể chia thành hai loại chính: Máy với kiến trúc bộ nhớ chia sẻ (shared memory) và Máy với kiến trúc bộ nhớ phân tán (distributed memory)

Kiến trúc bộ nhớ chia sẻ: Trong máy tính với bộ nhớ chia sẻ có một vùng nhớ chung để mọi bộ vi xử lý có thể truy cập đến. Người lập trình không phải quan tâm đến việc phân chia dữ liệu.

Các máy tính SIMD, MIMD với bộ nhớ chia sẻ được viết tắt tương ứng là SM-SIMD, SM-MIMD.

Kiến trúc bộ nhớ phân tán: Trong kiến trúc bộ nhớ phân tán, mỗi bộ vi xử lý có một vùng nhớ riêng, bộ vi xử lý này không thể truy cập vào vùng nhớ của các bộ

vi xử lý khác. Các bộ vi xử lý kết nối và truyền dữ liệu cho nhau. Người lập trình phải xác định rõ vị trí của dữ liệu, truyền và nhận dữ liệu một cách tường minh khi viết chương trình. Tương tự, ta dùng các ký hiệu viết tắt DM-SIMD và DM-MIMD.

Trong một số cải tiến gần đây, người ta tạo ra bộ nhớ dùng chung ảo ở mức phần cứng (hệ thống nghiên cứu của Kendall Square) hoặc mô phỏng chương trình (các định hướng – directive trong ngôn ngữ lập trình High Performance Fortran). Các nhà cung cấp hệ thống máy tính đa chip (Massively Parallel Processing – MPP) như HP, SGI cũng phát triển các mô hình lập trình với bộ nhớ dùng chung ảo (ccNUMA). Gói phần mềm TreadMarks cho phép thiết lập bộ nhớ dùng chung ảo cho các máy tính trong một mạng LAN.

Với các hệ thống DM-MIMD, mô hình lập trình truyền thông điệp (Message Passing) với hai thư viện tiêu biểu: PVM (Parallel Virtual Machine) và MPI (Message Passing Interface) thường được sử dụng để phát triển các ứng dụng song song. Mô hình lập trình truyền thông điệp còn được nghiên cứu áp dụng cho các hệ thống có bộ nhớ dùng chung (ví dụ như các hệ thống Nec SX-6, Cray

SV1ex). Hiện nay, PVM đã không còn được tập trung phát triển. Trong khi đó, MPI ngày càng nhận được sự chú ý của các nhà cung cấp với các phiên bản LAM/MPI, MPICH. Trong tương lai không xa, phiên bản thống nhất OpenMPI của cộng đồng mã nguồn mở (MPI Forum) được xem là có nhiều triển vọng.

Với các hệ thống SM-MIMD, chuẩn đang được sử dụng rộng rãi nhất để phát triển các chương trình song song là OpenMP. Quá trình song song hóa chương trình với OpenMP được thực hiện bằng cách chèn vào trong mã nguồn các định hướng biên dịch (directive hoặc pragmas). OpenMP có thể hỗ trợ các ngôn ngữ thuộc dòng Fortran (Fortran 77/90/95) và C, C++.

Máy tính song song đơn lệnh, đa dữ liệu với bộ nhớ chia sẻ (SM-SIMD)

Đa số máy SM-SIMD là các máy tính chỉ có một vi xử lý kiểu vector. Hình vẽ dưới đây mô tả kiến trúc cơ bản của một máy tính vector:

Hình 2-1 Sơ đồ khối của vi xử lý kiểu vector

Một trong những đặc điểm của đa số máy tính vector với bộ xử lý vector (vector processing unit – VPU) là không có bộ nhớ cache. Một thực tế là dung lượng

của bộ nhớ cache không đủ để nạp các dữ liệu kiểu vector, do đó dẫn đến thường xuyên bị tràn bộ nhớ. Một số máy tính vector còn tích hợp thêm bộ xử lý dữ liệu số thực kiểu vô hướng (Scalar floating point unit – FPU), cho phép máy tính hoạt động ở hai chế độ (hệ thống Cray). Đa số máy tính vector thế hệ trước cho phép các toán hạng và kết quả được đọc và ghi trực tiếp trên bộ nhớ (CDC Cyber 205, ETA-10). Tuy nhiên, các máy tính vector hiện nay đều sử dụng thanh ghi vector (vector register).

Máy tính song song đơn lệnh, đa dữ liệu với bộ nhớ phân tán (DM-SIMD)

Các máy tính DM-SIMD còn được gọi là máy tính sử dụng mảng vi xử lý (processor-array). Các bộ vi xử lý trong máy tính đều hoạt động chung một nhịp đồng hồ, thực hiện chung một phép toán tại một thời điểm. Do vậy, không cần đồng bộ hóa, làm cho việc thiết kế các máy tính kiểu này đơn giản hơn. Về cấu trúc, máy tính bao gồm một vi xử lý điều khiển (control processor) và tập các bộ vi xử lý được ghép lại thành mảng. Vi xử lý điều khiển có nhiệm vụ chuyển các câu lệnh cho các vi xử lý trong mảng. Các máy tính DM-SIMD hiện nay thường sử dụng thêm một vi xử lý nữa, đóng vai trò điều khiển các truy nhập đầu – cuối

(front-end processor), nhằm kết nối giữa các đường dữ liệu và vi xử lý điều khiển. Các lệnh không thể thực hiện bởi vi xử lý điều khiển hay mảng các vi xử lý sẽ được chuyển cho vi xử lý điều khiển truy nhập.

Hình 2-2 Mô hình chung của các máy tính DM-SIMD

Các bộ vi xử lý trong mảng có thể ghép với nhau theo kiểu lưới hai chiều (2-D grid), nối vòng, nối chéo, lưới 3 chiều hoặc phức tạp hơn.

Máy tính DM-SIMD thường được chế tạo và sử dụng cho các mục đích đặc biệt, khi bài toán được song song hóa tối đa, không cần trao đổi thông tin giữa các vi xử lý, thường phù hợp với các chương trình xử lý tín hiệu số, xử lý ảnh, mô phỏng Monte-Carlo.

Máy tính song song đa lệnh, đa dữ liệu với bộ nhớ chia sẻ (SM-SIMD)

Một trong những khó khăn chính khi thiết kế máy tính có bộ nhớ dùng chung là giải quyết vấn đề kết nối giữa các vi xử lý và giữa các vi xử lý với bộ nhớ. Bằng thông tổng cộng truy cập đến bộ nhớ tăng tỷ lệ thuận với số lượng vi xử lý. Các vi xử lý được thiết kế tương tác trực tiếp với nhau như: kiểu crossbar, kiểu Ω network, kết nối sử dụng bus dùng chung (central bus). Mỗi giải pháp đều có ưu điểm và nhược điểm riêng.

Hình 2-3 Một số cách kết nối trong các máy tính SM-MIMD

Với các máy MIMD, có thể cùng một lúc nhiều tiến trình khác nhau được thực hiện trên các vi xử lý. Do đó, cần phải đồng bộ hóa. Các bộ vi xử lý đều có các

thanh ghi riêng đóng vai trò truyền thông và trao đổi trực tiếp với nhau thông qua các thanh ghi này. Việc đồng bộ hóa có thể thông qua vùng nhớ dùng chung, tuy rằng chậm hơn trao đổi trực tiếp. Việc truyền thông giữa các vi xử lý và trao đổi dữ liệu được thực hiện trên các đường bus khác nhau nhằm đảm bảo tốc độ cho quá trình đồng bộ hóa.

Một số loại máy tính song song SM-MIMD phổ biến:

Các máy SMP (Symetric Multi Processor): Đây là những máy tính có nhiều bộ vi xử lý. Các vi xử lý truy cập vào vùng nhớ chung với thời gian như nhau.

Các máy tính song song ccNUMA: ccNUMA (Cache Coherent Non-Uniform Memory Access) là tên gọi loại máy tính song song được tạo thành từ một mạng các nút tính toán với bộ nhớ phân tán về mặt vật lý, nhưng theo mô hình chia sẻ về mặt logic (trong hầu hết các máy tính ccNUMA, các nút tính toán là những máy tính SMP). Trong hệ thống ccNUMA, các phần tử dữ liệu được xem như thuộc về bộ nhớ dùng chung, do đó ccNUMA được phân loại thuộc kiến trúc SM-MIMD. Các máy tính song song với kiến trúc ccNUMA được bán trên thị trường bao gồm Bull NovaScale, HP Superdome, and SGI Altix3000.

Về mặt vật lý, các phần tử dữ liệu nằm trong vùng nhớ của các nút tính toán khác nhau, do đó thời gian để các vi xử lý truy cập cùng một phần tử dữ liệu sẽ khác nhau (**Non-Uniform Memory Access**). Một yêu cầu quan trọng khác đó là giá trị của các biến nằm trong bộ nhớ cache của các vi xử lý phải đồng nhất với nhau (**Cache Coherent**). Giải pháp đồng bộ cache được sử dụng phổ biến là dùng giao thức snoopy bus, trong đó các bộ nhớ cache sẽ theo dõi việc truyền các biến giữa các vi xử lý và cập nhật giá trị, nếu trong cache chứa biến tương ứng hoặc sử dụng vùng nhớ để theo dõi và cập nhật các bản sao của biến, gọi là directory memory.

Máy tính song song đa lệnh, đa dữ liệu với bộ nhớ phân tán (DM-SIMD)

DM-MIMD là kiến trúc máy tính chiếm đa phần trong các máy tính song song, mặc dù cấu trúc và vận hành của chúng phức tạp hơn so với các máy tính song song có bộ nhớ chia sẻ hay các máy DM-SIMD. Các máy tính DM-SIMD hiện đại nhất đều có các vi xử lý được ghép thành mảng. Dữ liệu đầu vào có dạng vector hoặc mảng nhiều chiều, được phân chia cho từng bộ vi xử lý dựa vào phần mềm. Sự phát triển của các phần mềm hỗ trợ lập trình, biên dịch, quản lý hệ

thống giúp cho quá trình phát triển ứng dụng, cũng như vận hành các hệ thống DM-MIMD trở lên dễ dàng hơn nhiều.

Các máy tính DM-MIMD giải quyết được nhược điểm cố hữu về băng thông khi truy cập bộ nhớ của các bộ vi xử lý. Một trong những vấn đề đáng quan tâm khi cải thiện hiệu năng của các kiến trúc chia sẻ bộ nhớ là tốc độ của bộ nhớ. Để đạt được tốc độ tính toán tương đương với các máy tính DM-MIMD, các vi xử lý trong các máy tính bộ nhớ chia sẻ phải có tốc độ rất cao và do vậy, đòi hỏi tốc độ truy cập bộ nhớ cũng phải cao tương ứng.

Nhược điểm của các máy tính DM-MIMD so với các máy SM-MIMD là tốc độ truyền thông giữa các bộ vi xử lý và trao đổi dữ liệu giữa các vi xử lý rất chậm. Với các bài toán cần nhiều thao tác truyền thống, hiệu năng thực sự của các máy DM-MIMD rất nhỏ so với hiệu năng cực đại.

Topo kết nối giữa các vi xử lý và tốc độ đường truyền dữ liệu ảnh hưởng rất lớn đến hiệu năng hệ thống. Giải pháp phổ biến nhất là khối đa diện (hypercube). Điểm nổi bật của cấu trúc khối đa diện với 2^d nút là số bước cực đại kết nối hai nút bất kỳ là d . Do đó, độ phức tạp kết nối lưới sẽ tăng lên theo hàm logarit của

số nút. Về lý thuyết, các topo khác như: cây, vòng, mạng 2-D, mạng 3-D đều có thể mô phỏng được với cấu trúc khối đa diện. Tùy mục đích cụ thể sẽ chọn topo kết nối tối ưu. Ví dụ, với bài toán mô phỏng vật lý, thích hợp nhất là kiểu mạng 2-D hoặc mạng 3-D. Trong các máy tính có số bộ vi xử lý nhỏ (ít hơn 64), kiểu kết nối thích hợp nhất là crossbars.

Hình 2-4 Một số cách kết nối sử dụng trong các máy DM-MIMD

Hệ thống tính toán song song ghép cụm (cluster)

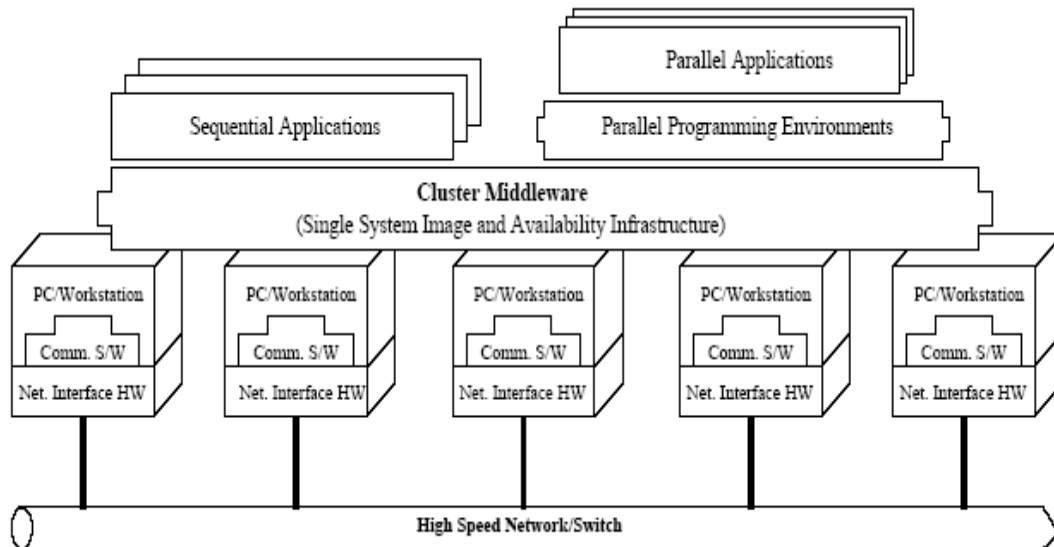
Hệ thống tính toán song song ghép cụm (Cluster) là các hệ thống máy tính song song được xây dựng từ các nút tính toán và thiết bị mạng thông dụng. Mỗi nút tính toán đóng vai trò điều khiển vào/ra là một hệ thống hoàn chỉnh, có khả năng làm việc độc lập. Hệ thống tính toán song song ghép cụm là cũng một máy tính song song, trong đó:

- Các tài nguyên tính toán bao gồm bộ vi xử lý và bộ nhớ trong tại mỗi máy tính,
- Các tài nguyên tính toán này có khả năng truyền thông và kết hợp với nhau thông qua cáp mạng. Thông thường, quy mô của hệ thống chỉ giới hạn trong một mạng cục bộ (LAN), trong đó có một máy tính đóng vai trò máy chủ (server), các máy tính còn lại đóng vai trò nút tính toán (computing node)

Việc thiết lập hệ thống tính toán song song ghép cụm từ những máy tính có cấu trúc đơn giản sử dụng các công nghệ mạng phổ biến đã được bắt đầu từ năm 1994 với mô hình Beowulf Cluster của Thomas Sterling và Donal Becker. Hệ thống tính toán song song phân cụm rõ hơn nhiều so với một siêu máy tính cùng sức mạnh. Điều này làm cho các hệ thống tính toán song song phân cụm ngày càng phổ biến và đặc biệt phù hợp cho các nước đang phát triển, các trường đại học. Tuy nhiên, các hệ thống tính toán song song phân cụm cũng có những hạn chế như: quá trình triển khai, cấu hình hệ thống tương đối phức tạp, hệ thống hoạt động không ổn định bằng các siêu máy tính. Nhược điểm lớn nhất là vấn đề truyền thông giữa các nút tính toán. Việc nghiên cứu nhằm nâng cao khả năng truyền thông là một trong những vấn đề quan trọng hàng đầu trong quá trình phát triển các cấu trúc, mô hình hệ thống phân cụm. Các hệ thống tính toán song song hiện đại đều dùng các kiến trúc mạng tiên tiến như: Myrinet, Gigabit Ethernet,... nhằm nâng cao tốc độ truyền thông giữa các nút tính toán.

Trên thực tế, cơ sở hạ tầng phần cứng mới chỉ là một phần trong toàn thể mô hình chung của tính toán phân cụm. Cùng với sự phát triển của lĩnh vực Công nghệ thông tin trên thế giới, phần mềm đang ngày càng chứng minh được vai trò cốt yếu của mình trong các hệ thống máy tính. Đối với các hệ thống tính toán

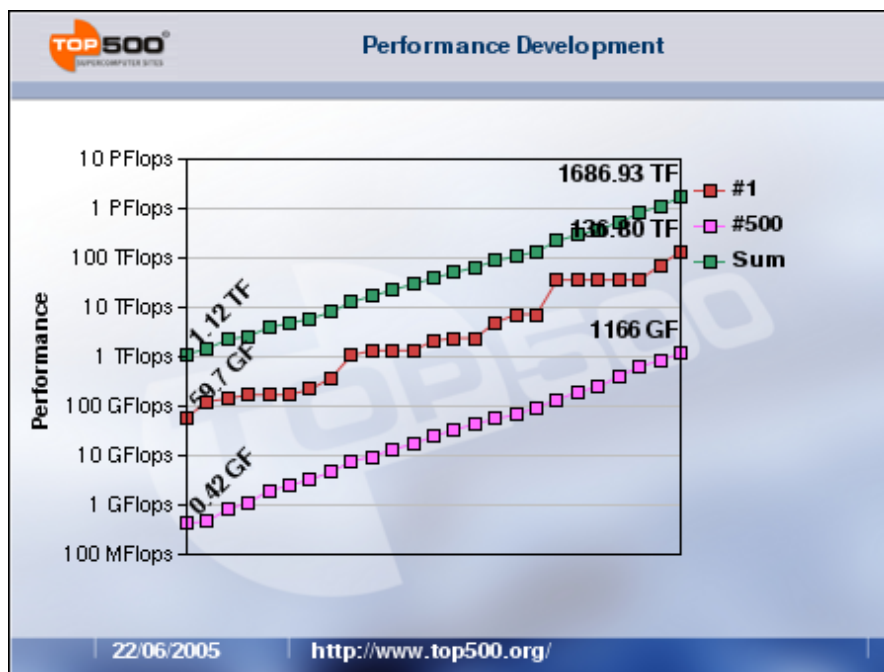
phân cụm, phần mềm càng có vai trò quan trọng hơn. Chính các phần mềm đảm bảo cho hệ thống gồm nhiều máy tính riêng lẻ có thể hoạt động ổn định và cộng tác hiệu quả. Một hệ thống tính toán song song ghép cụm (dựa trên mạng LAN) hoạt động như một hệ thống đơn dưới cái nhìn của người dùng và ứng dụng.



Hình 2-5 Kiến trúc mô hình cluster

Một số thống kê của Top500

Tốc độ tính toán của các hệ thống siêu máy tính ngày càng tăng nhanh. Theo thống kê trong website top500.org, cứ sau khoảng thời gian 6 tháng, tổng năng lực tính toán của 500 máy tính mạnh nhất Thế Giới tăng trung bình 50%, lớn hơn sự tăng tốc độ của vi xử lý đơn theo định luật Moore (2 lần sau mỗi 18 tháng)



Hình 2-6 Thống kê về hiệu năng của 500 siêu máy tính mạnh nhất Thế Giới tháng 6 – 2005

Tại thời điểm tháng 6 năm 2005, siêu máy tính mạnh nhất Thế Giới có hiệu năng tính toán là 137 TFlops (TeraFlops), trong khi máy tính đứng thứ 500 có hiệu năng tính toán là 1.16 Tflops. Năm 2004, năng lực tính toán của máy tính mạnh nhất Thế Giới là 36 TFlops (NEC Earth Simulator). Chỉ còn 5 trong số 10 máy tính mạnh nhất Thế Giới ở thời điểm tháng 11 năm 2004 còn tồn tại trong Top 10 của tháng 5 năm 2005.

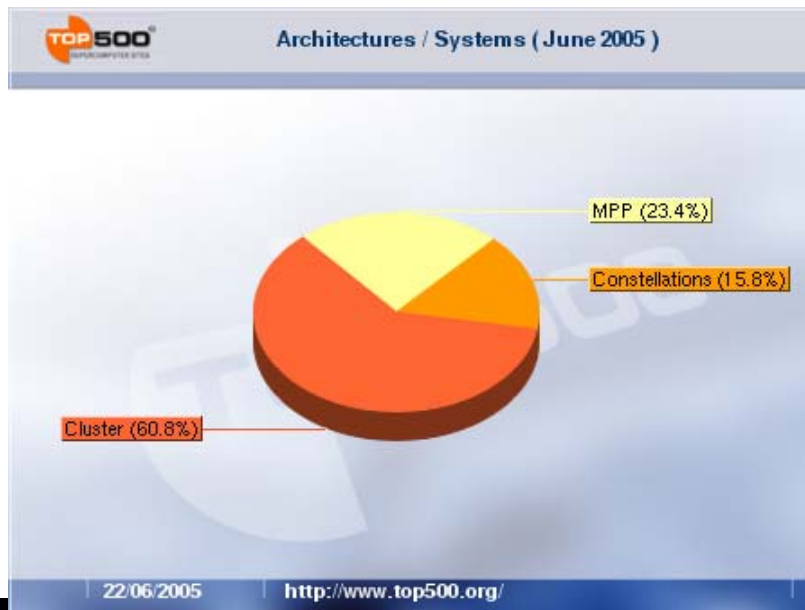
Top500 phân loại các siêu máy tính trên Thế Giới theo 3 loại chính:

Cluster: Trong danh sách thống kê tháng 6/2005, có 304/500 siêu máy tính là cluster với hai cluster mạnh nhất đứng thứ 5 và thứ 7.

Constellation (hay federal): Đây là một dạng cluster đặc biệt với các nút tính toán là các máy tính song song SMP và số lượng vi xử lý trong một nút tính toán nhiều hơn tổng số nút tính toán của toàn hệ thống. Trong danh sách của top500 vào tháng 6/2005 có 79/500 siêu máy tính là constellation, constellation mạnh nhất đứng thứ 41.

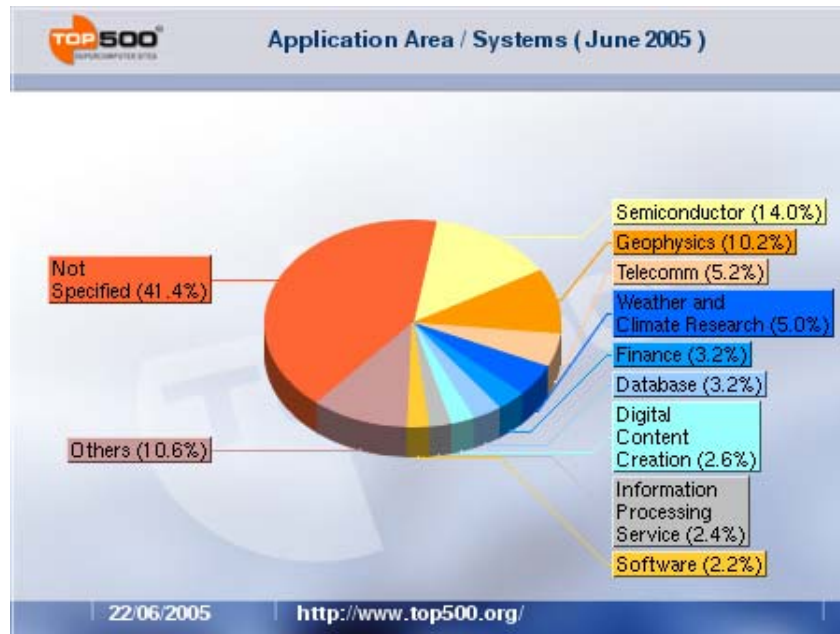
Massively Parallel Processors (MPP): MPP là các máy tính song song được thiết kế và xây dựng một cách chuyên dụng, bao gồm các máy tính vector, DM-

MIMD, SM-MIMD. Các máy tính này được thiết kế, phát triển bởi các nhà cung cấp hàng đầu trong lĩnh vực tính toán hiệu năng cao như: IBM, Cray, SGI, NEC. Có 117 máy tính MPP trong top500 tháng 6/2005 với 8 máy tính trong top10.



Hình 2-7 Tỷ lệ các kiến trúc máy tính trong top500 (06/2005)

2.1.1.3. Các ứng dụng tính toán song song trong khoa học, công nghệ và môi trường



Hình 2-8 Các lĩnh vực ứng dụng chính của 500 siêu máy tính mạnh nhất Thế Giới (06/2005)

2.1.2. Chống lỗi trong các hệ thống tính toán hiệu năng cao

Một trong những đặc điểm của các hệ thống tính toán song song là có thể xảy ra lỗi trên các nút hay trên mạng. Đây được coi là tính chất không ổn định của hệ thống (volatility). Mỗi khi xảy ra lỗi, hệ thống phải chạy lại và mất hết kết quả công việc đã thực hiện trước đó. Vấn đề đặt ra là cần phải tối thiểu hoá các tính toán bị mất nhờ xây dựng một môi trường chuyển giao các thông điệp có khả năng chống lỗi.

Hiện nay, đa số các hệ thống tính toán song song thông dụng đều sử dụng thư viện truyền thông MPI để thực hiện các ứng dụng song song. MPI là một hệ thống truyền thông điệp chuẩn, tuy nhiên nó không cho phép xác định được nên thực hiện việc chống lỗi như thế nào và ở mức nào. Do đó, các chương trình sử dụng MPI đều thiết kế không có phần chống lỗi, mà chỉ cung cấp hai trạng thái công việc: OK hoặc FAILED. Thực tế, các lỗi có thể xảy ra ở một số mức của

chương trình và thường làm cho chương trình buộc phải hủy bỏ. Đối với các chương trình lớn, thời gian chạy tương đối dài. Việc này đòi hỏi nhiều phí tổn, vì các công việc thực hiện được trước khi lỗi xảy ra đều bị mất. Công việc cần làm là lấy lại được trạng thái của hệ thống ngay trước khi lỗi xảy ra và khôi phục được lỗi. Đối với các hệ thống lớn có nhiều nút trạm, cần phải thực hiện sao cho chỉ các trạm liên quan mới phải thực hiện lại công việc, không ảnh hưởng đến các nút trạm khác.

Phương pháp khắc phục lỗi tự động trong MPI, trong suốt với người dùng, hay được sử dụng là sử dụng một bộ điều phối trong việc tạo checkpoint hoặc phương pháp ghi lại thông điệp kết hợp với việc tạo checkpoint không có điều phối. Đã có rất nhiều nghiên cứu về giao thức, việc thực hiện và phương pháp nhằm tối ưu hai cách tiếp cận này.

Phương pháp điều phối các checkpoint có ưu điểm là có chi phí rất thấp trong

quá trình khôi phục lỗi, trong khi việc ghi lại thông điệp đòi hỏi mất thêm một số đáng kể các thông điệp khi phục hồi. Tuy nhiên, phương pháp sử dụng checkpoint có chi phí bộ nhớ cao, vì mỗi khi checkpoint, toàn bộ trạng thái của tiến trình lại được ghi vào bộ nhớ.

Việc sử dụng giao thức nào phụ thuộc vào tần suất xuất hiện của lỗi. Nếu hệ thống có tần suất xuất hiện lỗi càng cao, khoảng thời gian giữa hai lần checkpoint giảm, thì nên sử dụng phương pháp checkpoint kết hợp với ghi lại thông điệp. Ngược lại, nếu tần suất lỗi giảm, phương pháp checkpoint có sự điều phối có thể hiệu quả hơn.

2.1.3. Tổng quan tình hình nghiên cứu về tính toán lưới

2.1.3.1. Sự ra đời của tính toán lưới

Ngày nay, trong các môn khoa học cũng như trong thực tế cuộc sống đang đặt ra những bài toán khó cần tới khả năng tính toán nhanh và chính xác trên cơ sở tận dụng những tài nguyên phần cứng cũng như phần mềm phân tán ở nhiều nơi. Khái niệm tổ chức ảo (Virtual Organization – VO) ra đời nhằm chỉ một tập các cá nhân và/hoặc các tổ chức có các nguồn tài nguyên được *chia sẻ* hay *dùng chung* (sharing) theo các luật đề được ra [1]. Trong khái niệm này, ta cần chú ý hai khía cạnh:

- *các tài nguyên được chia sẻ*: không đơn thuần là chia sẻ file theo quan niệm cũ, tài nguyên mà các tổ chức ảo chia sẻ còn là sự truy cập trực tiếp đến máy tính, phần mềm, dịch vụ, dữ liệu, các trang thiết bị phần cứng và các thiết bị đặc biệt khác;

- *việc chia sẻ tài nguyên* cũng không thể tiến hành tùy tiện mà phải tuân theo các luật được cả người cung cấp cũng như người sử dụng tài nguyên nhất trí. Cần phải định nghĩa rõ ràng và cẩn thận về: cái gì được đem chia sẻ, những ai được quyền sử dụng và điều kiện cho sự chia sẻ là gì?

Sự xuất hiện của các tổ chức ảo cùng với nhu cầu chia sẻ tài nguyên giữa chúng chính là cơ sở để công nghệ tính toán lưới ra đời. Ta có thể hiểu một cách đơn giản: *tính toán lưới chính là bước phát triển tiếp theo của tính toán phân tán, với mục đích cung cấp những dịch vụ tính toán đơn giản cho người dùng, nhưng mang lại sức mạnh tính toán rất lớn bởi tính trong suốt và khả năng kết nối các hệ thống không đồng nhất nhằm chia sẻ các nguồn tài nguyên đa dạng* [4].



Hình 2-9. Một quan sát về tính toán lưới

Để khắc phục những nhược điểm của các công nghệ hiện tại, người ta hướng đến công nghệ tính toán lưới. Trong vòng 6 năm gần đây, những nỗ lực nghiên cứu và phát triển trong cộng đồng tính toán lưới đã đưa ra những giao thức, dịch vụ, và công cụ giải quyết một cách khá triệt để những thách thức nảy sinh khi xây dựng các tổ chức ảo. Những công nghệ này bao gồm:

- Những giải pháp an ninh hỗ trợ việc quản lý giấy ủy nhiệm và các chính sách khi mà sự tính toán được mở rộng ra nhiều tổ chức;
- Các giao thức và dịch vụ quản lý tài nguyên có hỗ trợ các truy nhập từ xa an toàn cho trình tính toán, các tài nguyên dữ liệu và sự định vị của các tài nguyên phức tạp;
- Các giao thức và dịch vụ truy vấn thông tin có cung cấp thông tin cấu hình và trạng thái về các tài nguyên, tổ chức và dịch vụ;

- Các dịch vụ quản lý dữ liệu có nhiệm vụ định vị và chuyển tải các tập dữ liệu giữa các hệ thống lưu trữ và các ứng dụng;

Công nghệ tính toán lưới có thể coi là phần bổ sung cho các công nghệ tính toán phân tán đang tồn tại, chứ không phủ nhận các công nghệ đã có. Ví dụ như:

- Các hệ thống tính toán phân tán enterprise có thể sử dụng công nghệ tính toán lưới để chia sẻ tài nguyên giữa các tổ chức;
- Các ASP, SSP có thể sử dụng công nghệ tính toán lưới để thiết lập thị trường tiêu thụ động các tài nguyên tính toán và lưu trữ, từ đó sẽ vượt qua những giới hạn của các cấu hình tĩnh hiện tại;

Như vậy tính toán lưới hướng đến việc chia sẻ và sử dụng hiệu quả các nguồn tài nguyên thuộc về nhiều tổ chức trên một quy mô rộng lớn (thậm chí là quy mô toàn cầu). Chính các công nghệ mạng và truyền thông phát triển mạnh mẽ trong những năm qua đã biến những khả năng này dần trở thành hiện thực. Các nghiên cứu về tính toán lưới đã và đang được tiến hành là nhằm tạo ra một cơ

sở hạ tầng lưới cho phép dễ dàng chia sẻ và quản lý các tài nguyên đa dạng và phân tán trong môi trường lưới.

2.1.3.2. Những hoạt động liên quan đến tính toán lưới

Các trường đại học, viện nghiên cứu lớn trên thế giới thay vì thành lập các trung tâm tính toán hiệu năng cao, người ta có xu hướng thành lập các trung tâm tính toán lưới (Center for Grid Computing). Những trung tâm này không giới hạn trong phạm vi một trường, một viện nghiên cứu mà nối kết các trung tâm ở nhiều địa điểm lại với nhau. Chẳng hạn, tháng 8/2001 Ủy ban Khoa học Quốc gia Mỹ đã xây dựng một lưới tính toán nối các siêu máy tính trên toàn quốc thành một nguồn xử lý tính toán có tên gọi Distributed Terascale Facility do Trung tâm Siêu máy tính Quốc gia (NCSA) quản lý. Với các trung tâm này có thể triển khai các tính toán cũng như xây dựng cổng thông tin (Information Portal) phục vụ cho nhu cầu truy cập vào hệ thống tính toán lưới.

Các Công ty lớn như IBM từ năm 2001 họ đã đưa vào kế hoạch xây dựng 50 trung tâm tính toán lưới trên toàn thế giới với chi phí tổng cộng trên 4 tỷ dollar. Hãng Sun Microsystems có dự án mã nguồn mở Grid Engine sau khi mua lại Công ty GridWare vào tháng 7/2001.

Các nhà vật lý phân tử ở Anh đã vận hành một hệ thống grid khổng lồ vào 12/2004 ở Nottingham, gọi là GridPP. Mạng máy có tên *Large Hadron Collider Computing Grid* (LCG) được thiết lập từ hơn 6.000 máy tính ở nhiều điểm trên toàn thế giới, trong đó riêng dự án tính toán lưới của các nhà vật lý Anh (GridPP) đóng góp 1.000 chiếc từ 12 địa chỉ khác nhau. Bộ phận *Large Hadron Collider* (LHC) được xây dựng tại trung tâm nghiên cứu CERN của Thụy Sĩ. Các thí nghiệm vật lý hạt nhân mà nó thực hiện sẽ tạo ra những lượng dữ liệu khổng lồ: khoảng 15 petabyte mỗi năm (1 petabyte = 1.024 terabyte, tức là hơn 1.000.000 gigabyte). LCG được coi là một cách xử lý dữ liệu chứ không phải là một cỗ máy cụ thể. Đó là một hướng tiếp cận thay thế cho siêu máy tính tốn kém. Theo Hội đồng nghiên cứu thiên văn và vật lý hạt nhân Anh (PPARC), đến năm 2007,

LCG sẽ có sức mạnh tương đương 100.000 máy tính nhanh nhất hiện nay kết hợp lại và tạo ra một siêu máy tính ảo. Thiết bị này có thể được mở rộng và phát triển thêm nếu cần thiết.

Các kỹ sư tham gia dự án GridPP còn phát triển một bản đồ mô tả những nhiệm vụ tính toán luân chuyển như thế nào quanh LCG. Bản đồ này cũng biểu thị các tác vụ được phân bổ tới những điểm thích hợp nhất trong lưới, vận hành các chương trình và gửi trả kết quả bằng thời gian thực.

2.1.3.3. Các dự án lớn trên thế giới về tính toán lưới

ASTROGRID <http://www.astrogrid.org/>

BIOGRID <http://www.biogrid.jp/>

BIRN <http://www.nbirn.net/>

CONDOR	http://www.cs.wisc.edu/condor/
CROSSGRID	http://www.crossgrid.org/
DAMIEN	http://www.hlr.de/organization/pds/projects/damien/
DATATAG	http://datatag.web.cern.ch/datatag/
DOE SciDAC	http://www.osti.gov/scidac/
DOE SCIENCE GRID	http://doesciencegrid.org/
Earth System Grid II	http://www.earthsystemgrid.org/
ECOGRID	http://www.buyya.com/ecogrid

EGEE	http://public.eu-egee.org/
Esnet	http://www.es.net/
EUROGRID	http://www.eurogrid.org/
FUSIONGRID	http://www.fusiongrid.org/
GÉANT	http://www.dante.net/geant/
GLOBUS	http://www.globus.org/
GRACE	http://www.grace-ist.org/
GRIDBUS	http://www.gridbus.org/
GriDis	http://web.datagrid.cnr.it/GriDis/jsp/index.jsp

GRIDPP	http://www.gridpp.ac.uk/
GRIP	http://www.grid-interopability.org/
GriPhyN	http://www.griphyn.org/index.php
Inca	http://inca.sdsc.edu/
IPG	http://www.ipg.nasa.gov/
jvDgl	http://www.ivdgl.org/
LEAD	http://lead.ou.edu/
LEGION	http://www.cs.virginia.edu/~legion/
MCNC	http://www.mcnc.org/gcns2/

myGrid	http://www.mygrid.org.uk/
NAREGI	http://www.naregi.org/
National Grid Service (UK)	http://www.ngs.ac.uk/
NEES grid	http://www.neesgrid.org/
NextGrid	http://www.nextgrid.org/
NORDUGRID	http://www.nordugrid.org/
NSF MIDDLEWARE INITIATIVE	http://www.nsf-middleware.org/
Open Science Grid Consortium	http://www.opensciencegrid.org/

PPDG	http://www.ppdg.net/
RealityGrid	http://www.realitygrid.org/
SimDat	http://www.simdat.org/
TERAGRID	http://www.teragrid.org/
UNICORE Plus	http://www.fz-juelich.de/unicoreplus/index.html
WestGrid	http://www.westgrid.ca/home.html

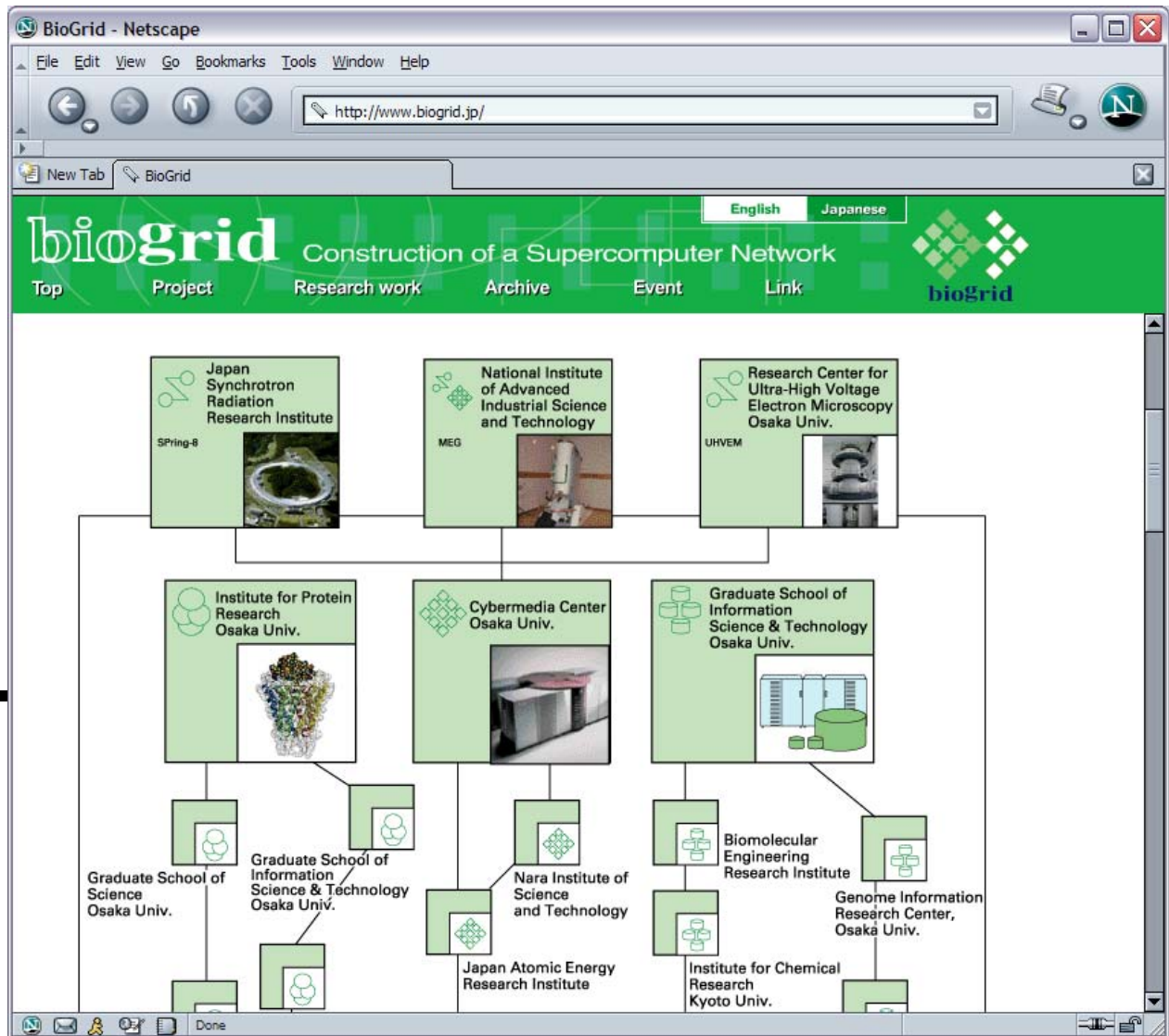
Một số dự án nổi bật:

Dự án BIOGRID

Đây là dự án được chủ trì thực hiện bởi trường Đại học Osaka, nằm trong chương trình công nghệ thông tin của chính phủ Nhật Bản, với sự phối hợp giữa các bộ giáo dục, văn hóa, thể thao, khoa học và công nghệ. Dự án như là một bước khởi đầu để xây dựng mạng siêu máy tính để giải quyết những vấn đề của sinh học và y khoa đặt ra. Dự án bao gồm 3 mục tiêu chính:

- Phân tích để triển khai một mạng siêu máy tính.
- Nghiên cứu về công nghệ lưới dữ liệu (data grid technology) cho phép giải quyết các bài toán với những kiểu dữ liệu khác nhau của các cơ sở dữ liệu lớn.
- Nghiên cứu về công nghệ tính toán lưới (computing grid technology) qua đó tham gia giải quyết những bài toán xử lý dữ liệu đòi hỏi nguồn tài nguyên tính toán rất mạnh (ultra high-speed computing).

Báo cáo tổng kết dự án



Hình 2-10. Dự án BioGrid (<http://www.biogrid.jp/>)

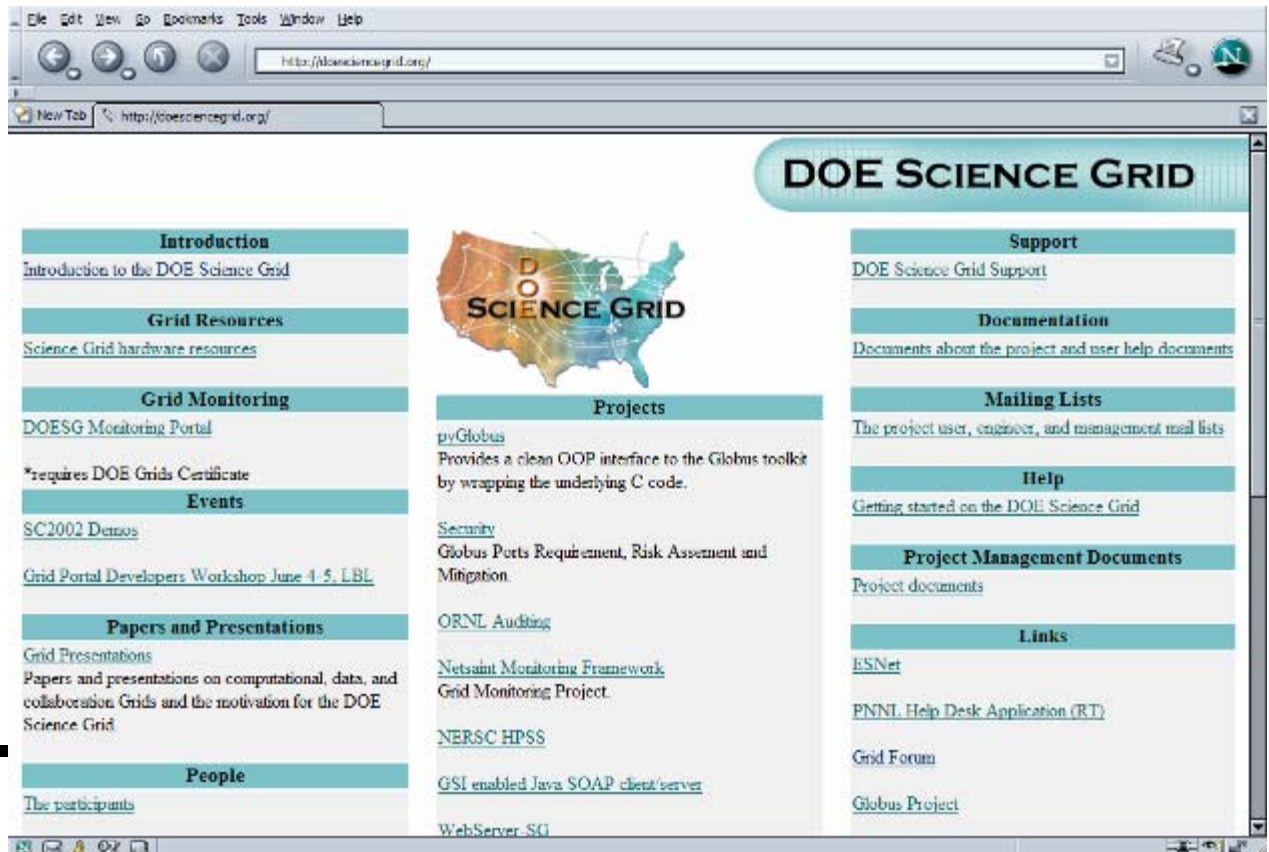
Dự án DOE Science Grid của Bộ Năng lượng, Hoa Kỳ

Đây là dự án được xây dựng quy mô lớn nhằm phối hợp giữa nhiều cơ quan khoa học của Mỹ. Nhằm tạo cơ sở hạ tầng công nghệ thông tin (cyberinfrastructure) phục vụ cho việc phát triển và triển khai các lĩnh vực tính toán phân tán (distributed computing) xử lý dữ liệu và khai thác nguồn tài nguyên công cụ (instrument resources). Mục tiêu dự án bao gồm:

- Cung cấp khả năng xử lý tính toán của các bài toán khoa học đòi hỏi thời gian.
- Cung cấp khả năng xử lý dữ liệu độc lập với vị trí địa lý của nơi yêu cầu, tạo ra một sự quản trị trong suốt từ người yêu cầu đến hệ thống.
- Tạo môi trường truyền thông trên cơ sở những nền tảng đã có để tạo sự trong suốt giữa hệ thống và người sử dụng.

- Tạo môi trường để khai thác và chia sẻ phần mềm, công cụ để tính toán và xử lý dữ liệu.

Báo cáo tổng kết dự án



Hình 2-11. Dự án DOE Science Grid (<http://doesciencegrid.org/>)

Dự án GridPP

Báo cáo tổng kết dự án

File Edit View Go Bookmarks Tools Window Help

http://www.gridpp.ac.uk/

New Tab GridPP - UK Computing for Particle Physics

GridPP
UK Computing for Particle Physics

Welcome page : How to use the Grid : Grid Support : Website Help

Search

About GridPP

- The collaboration
- People
- News
- Meetings
- Demos
- Information for public and schools
- Contact us

Documents

- Talks
- Publications
- Posters
- Abstracts
- Management documents
- Links

Activities

- User Area
- Applications for particle physics
- Middleware
- Deployment & Operations
- Infrastructure
- GridPP in the wider world
- Other projects
- Wiki

Management

- Collaboration Board
- Project Management Board
- User Board
- Dissemination Board

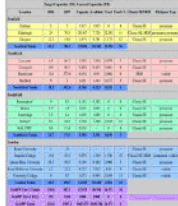
Welcome to GridPP

GridPP is a collaboration of particle physicists and computer scientists from the UK and CERN, who are building a computing Grid for particle physics. GridPP is funded by PPARC as part of its e-Science Programme.

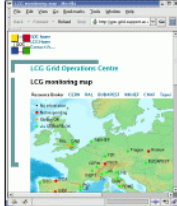
GridPP has built a working prototype grid across 17 UK institutions. Over the next three years, this will be extended to the equivalent of 10,000 PCs.



Status of GridPP infrastructure



GridPP storage status



LHC Computing Grid monitoring

To learn more about GridPP please contact [Sarah Pearce](#) who is the GridPP Dissemination Officer.

For problems or questions about the website, please mail webmaster@gridpp.ac.uk

GridPP news

Wed 21 Dec 2005
[The 12 months of a GridPP year](#)

Thu 8 Dec 2005
[Beyond Einstein Webcast: Honouring the Legend](#)

Fri 2 Dec 2005
[Sleepless in Seattle: GridPP at Supercomputing 05](#)

Fri 18 Nov 2005
[Beyond Einstein: World Wide Webcast 1st December](#)

Meetings

11-12 January 2006
[15th GridPP Collaboration Meeting](#)

24 January 2006
[National Grid Service User Forum](#)

13-16 February 2006
[GGF 16](#)

[Jobs and Training](#)

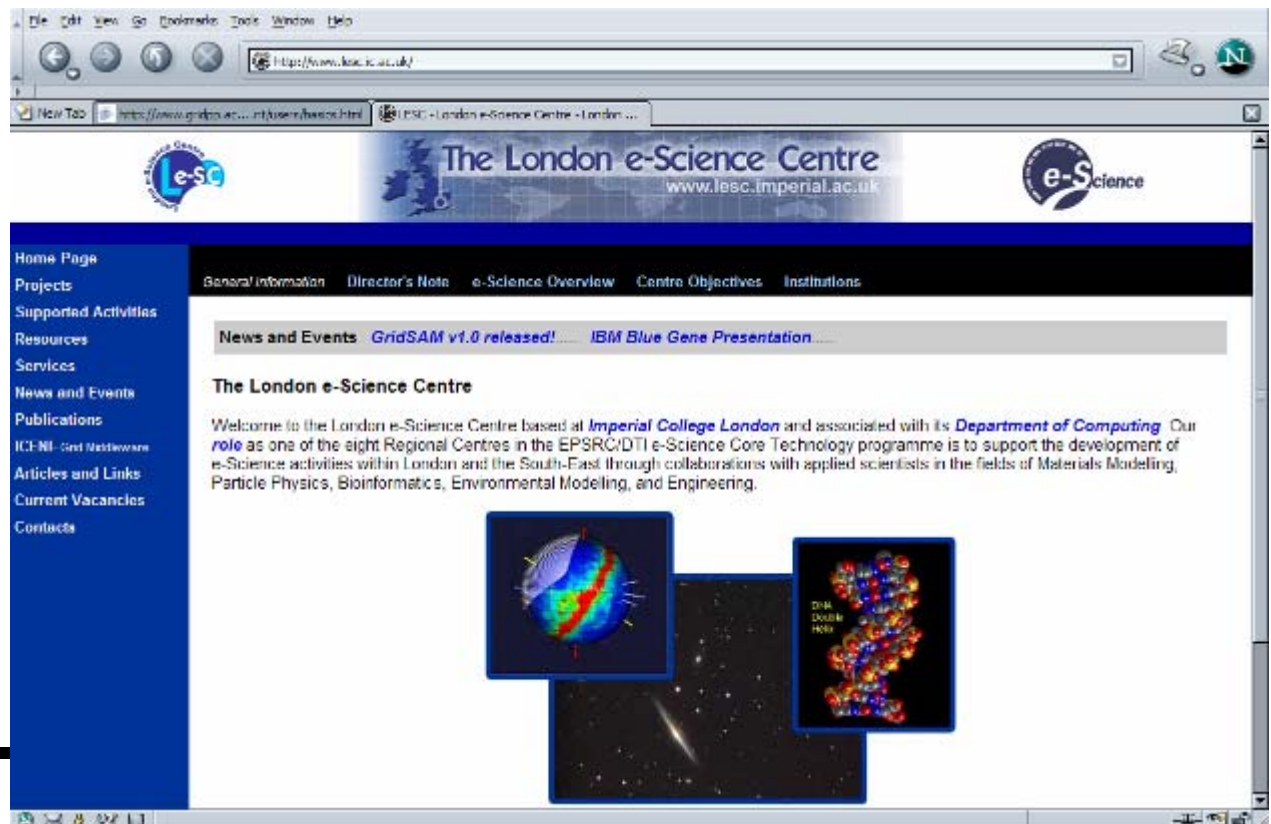
Hình 2-12. Dự án GridPP (<http://www.gridpp.ac.uk/>)

Sự phát triển nhanh chóng của Grid còn đưa đến một lĩnh vực nghiên cứu mới đó là thương mại hóa tài nguyên trên Grid, cụ thể là tài nguyên tính toán, tài nguyên lưu trữ, và các dạng tài nguyên khác như phần mềm, thiết bị chuyên dụng. Trong hướng nghiên cứu này, người ta cần tìm ra các cơ chế để cho người sở hữu tài nguyên và người sử dụng tài nguyên có diễn tả các yêu cầu về chất lượng, khối lượng tài nguyên (người dùng) cũng như giá cả, chính sách quản lý (người sở hữu) trong điều kiện tách biệt về thời gian (múi giờ) và không gian. Để có thể thương mại hóa tài nguyên Grid cần phải có các yếu tố: thị trường để quảng bá thông tin về tài nguyên, mô hình để định giá tài nguyên, giao thức để thương lượng, các cơ chế kế toán, chi trả. Dự án Economy Grid – ECOGRID (<http://www.buyya.com/ecogrid>) nằm trong số các ứng dụng thuộc dạng đó.

2.1.3.4. Các trung tâm khai thác tính toán lưới

Trung tâm London e-Science

Báo cáo tổng kết dự án



Hình 2-13. Trung tâm London e-Science (<http://www.lesc.ic.ac.uk/>)

Trung tâm London e-Science (LeSC), khoa CNTT, Đại học Imperial ra đời và hoạt động từ 9/2001 là một thành phần của chương trình UK e-Science của Vương Quốc Anh. Nhiệm vụ của trung tâm này là nghiên cứu phát triển một số công nghệ hỗ trợ triển khai e-Science cho các trường đại học và viện nghiên cứu ở London và vùng Đông Nam nước Anh. Trung tâm này còn đảm trách hệ thống Access Grid để tổ chức hội thảo truyền hình sử dụng Grid làm nền tảng cho việc chứng thực. Hệ thống Access Grid này kết nối 15 trung tâm nghiên cứu khoa học của Anh và các nước ở Châu Âu, Bắc Mỹ, Châu Á – Thái Bình Dương.

Trung tâm PRGAMA Grid Operation Center, Trung tâm San Diego Super Computer, Đại học California, San Diego (<http://pragm-goc.rocksclusters.org/>) là một tổ chức hỗ trợ đắc lực cho chương trình phát triển các phần mềm trung gian cho tính toán lưới của khu vực Châu Á – Thái Bình Dương (Pacific Rim Applications and Grid Middleware Assembly - PRAGMA).

Chương trình này được sự tài trợ của quỹ nghiên cứu khoa học quốc gia Liên bang Mỹ (NSF), và cùng với sự tham gia của 20 trung tâm nghiên cứu khoa học tính toán thuộc 13 nước trong khu vực như: Mỹ, Nhật Bản, Hàn Quốc, Úc, Trung Quốc, Đài Loan, Singapore, Malaysia, Thái Lan, Ấn Độ, Chi Lê, Switzerland, Mexico. Trung tâm NGO như là một tổ chức hạt nhân của chương trình PGRAMA. Một số kết quả nổi bật đã được thực hiện như sau:

- Ninf-G (<http://ninf.apgrid.org>): đây là dự án được thực hiện bởi các chuyên gia Nhật Bản, nhằm tạo ra công cụ với giao diện trực quan cho phép người sử dụng dễ dàng truy cập đến nguồn tài nguyên mạng lưới, sử dụng các phần mềm và dữ liệu khoa học trên grid. Hiện nay đã có phiên bản 4.1 của hệ thống này được hiện thực như một phần mềm nguồn mở. Người Nhật (và chỉ người Nhật) tổ chức một hội thảo về Ninf-G vào 18/01/2006. Đây là một đóng góp của Nhật vào việc tạo ra các ứng dụng trung gian cho tính toán lưới.

- Nimrod/G (<http://www.csse.monash.edu.au/~davida/nimrod/nimrodg.htm>): cũng là một sản phẩm tương tự Ninf-G, được phát triển bởi Đại học Monash, Úc nhằm

hỗ trợ cho Globus Toolkit để khám phá một cách tự động nguồn tài nguyên cho phép của Grid.

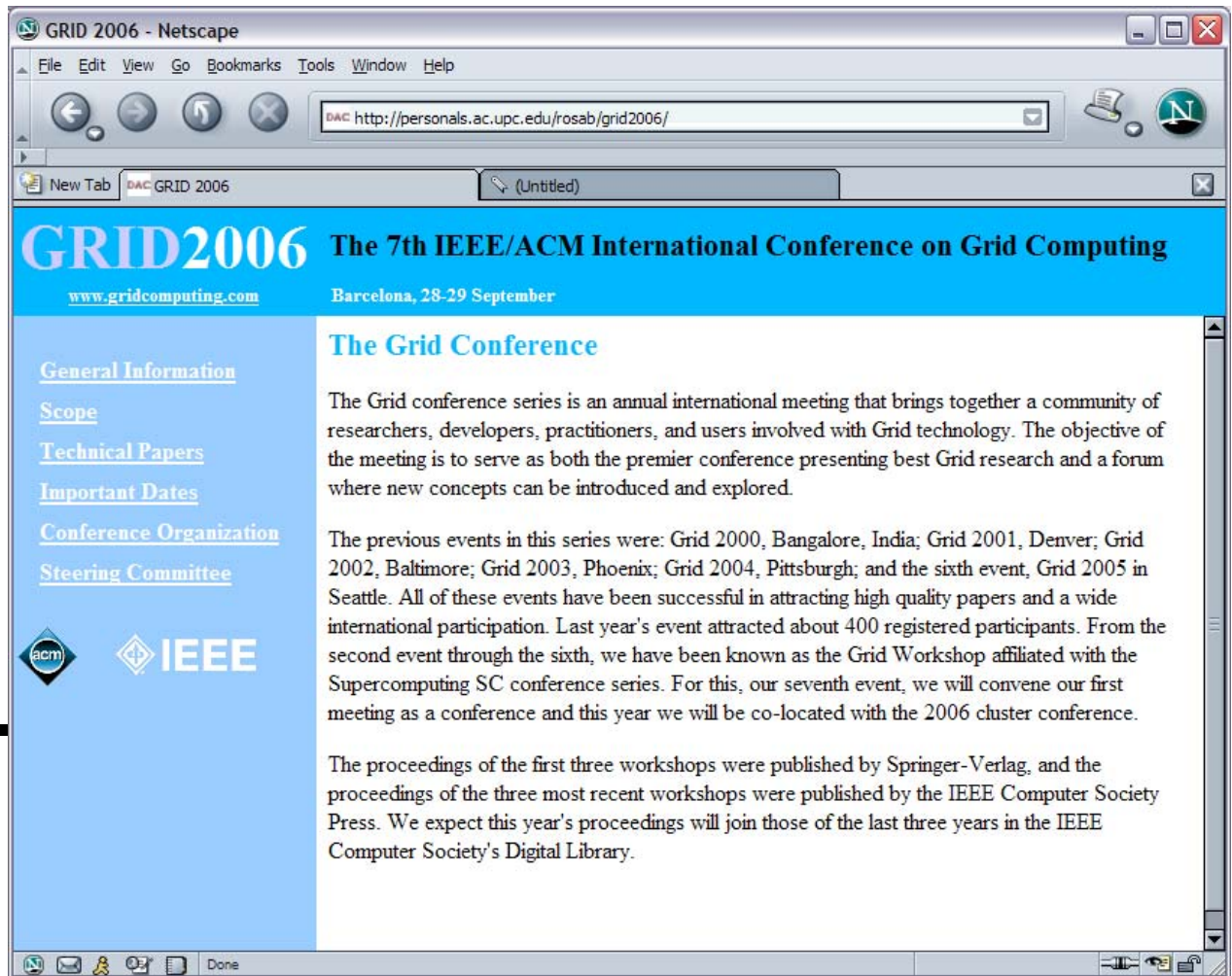
- Dự án BioGrid: nhằm phát triển nền tảng tính toán mạng lưới hỗ trợ phân tích gen, những vấn đề về bệnh. Đây là dự án đang được triển khai.

- Dự án Telescience: nhằm sử dụng mạng toàn cầu và hệ thống tính toán lưới để giúp các thành viên PRAGMA truy cập và sử dụng các thiết bị chẩn đoán y khoa từ xa (máy đo điện tâm đồ não, kính hiển vi điện tử, máy gia tốc, v.v...) của các đại học lớn như San Diego, Osaka, Tokyo, v.v... Qua đó hỗ trợ các nghiên cứu về tế bào não, về thần kinh, về các dịch bệnh SARS, Bird Flu, v.v...

2.1.3.5. Các diễn đàn, hội thảo

Trên thế giới cũng đã có nhiều diễn đàn, hội thảo về tính toán lưới cũng như công nghệ tính toán lưới. Chẳng hạn, diễn đàn về lưới toàn cầu (Global Grid

Forum - GGF) mỗi năm tổ chức 3 lần. Diễn đàn lần thứ 1 tổ chức vào tháng 3/2001, diễn đàn lần thứ 16 được tổ chức tại Athens, Hy Lạp, từ ngày 13 – 16/01/2006 (<http://www.ggf.org/index.php>).



Hình 2-14. Hội thảo quốc tế lần thứ 7 về tính toán lưới

Hội thảo Quốc tế về Tính toán lưới (International Workshop on Grid Computing - <http://personals.ac.upc.edu/rosab/grid2006/>) được tổ chức hàng năm. Hội thảo lần thứ 7 tổ chức vào ngày 28-19/9/2006 ở Barcelona, Tây Ban Nha, (Hình 2-14). Chủ đề của hội thảo bao gồm các vấn đề sau đây:

- Mô hình tính toán Internet (Internet-based Computing Models)
- Ứng dụng e-Science và e-Business (eScience and eBusiness Applications)
- Truy cập và quản trị dữ liệu phân tán và trong phạm vi rộng lớn (Distributed and Large-Scale Data Access and Management)
- Phần mềm trung gian và các công cụ hỗ trợ (Middleware and Toolkits)
- Các công cụ kiểm tra đo đạc, quản trị và tổ chức hệ thống (Monitoring, Management and Organization Tools)

- Quản trị và lập lịch sử dụng tài nguyên (Resource Management and Scheduling)
 - Mạng (Networking)
 - Virtual Instrumentation
 - Metadata, Ontologies, and Provenance
 - Creation and Management of Virtual Enterprises and Organizations
 - Architectures and Fabrics
 - Dịch vụ thông tin (Information Services)
 - Vấn đề an ninh (Security Issues)
 - Mô hình, công cụ và môi trường lập trình (Programming Models, Tools, and Environments)
 - Tiềm lực lưới (Grid Economy)
-

- Tính toán tự trị và tiện ích trên lưới toàn cầu (Autonomic and Utility Computing on Global Grids)
- Đánh giá và mô hình hóa hiệu năng (Performance Evaluation and Modeling)
- Cluster and Grid Integration Issues
- Scientific, Industrial and Social Implications

Khu vực Châu Á – Thái Bình Dương có hội thảo xuất phát từ chương trình PRAGMA như đã đề cập ở trên. Hội thảo lần đầu tiên diễn ra vào tháng 3/2002 tại San Diego, Mỹ. Lần 2 vào tháng 7/2002 tại Seoul, Hàn Quốc. Lần 3 vào tháng 01/2003 tại Fukuoka, Nhật Bản. Lần 4 tại Melbourne, Úc vào tháng 6/2003. Tháng 10/2003 hội thảo lần thứ 5 tổ chức tại Hsinchu, Đài Loan. Lần thứ 6 vào tháng 5/2004 tổ chức tại Bắc Kinh, Trung Quốc và lần thứ 7 vào 15 – 17/9/2004 ở San Diego, Mỹ. Lần thứ 8 tại Singapore vào ngày 02-04/5/2005; lần thứ 9 tổ

chức tại Hyderabad, Ấn Độ vào 20 – 23/10/2005. Hội thảo lần thứ 10 được tổ chức ở Townsville, Queensland, Australia vào 26 – 28/3/2006.

2.1.3.6. Tình hình nghiên cứu về tính toán lưới ở Việt Nam

Nghiên cứu tính toán lưới ở Việt Nam hiện đang trong giai đoạn khởi đầu và đang được xúc tiến khẩn trương.

Sở Bưu chính – Viễn thông, sở Khoa học – Công nghệ thành phố Hồ Chí Minh

Sở Bưu chính – Viễn thông và Sở Khoa học – Công nghệ thành phố Hồ Chí Minh khuyến khích các tổ chức nghiên cứu khoa học tham gia vào tổ chức PRAGMA, từ đó có thể tiếp cận, chuyển giao và làm chủ các công nghệ mới trong lĩnh vực tính toán lưới. Đặc biệt là việc tiếp nhận thông tin, dự án nghiên cứu, khả năng sử dụng tài nguyên tính toán của các nước trong tổ chức PRAGMA. Bên cạnh đó, thành phố Hồ Chí Minh cũng đang xúc tiến thành lập một Viện khoa học và công nghệ tính toán trực thuộc UBND, từ đó có thể tạo ra

các sản phẩm và làm đầu mối để tham gia vào tổ chức PRAGMA này. Tháng 01/2004, Sở Khoa học – Công nghệ được sự hỗ trợ của Bộ Khoa học – Công nghệ đã phối hợp với trường Đại học Khoa học tự nhiên tổ chức hội thảo “Tính toán hiệu năng cao và tính toán mạng lưới”.

Trường Đại học Bách Khoa TP HCM

Trường Đại học Bách khoa, Đại học quốc gia TP HCM thực hiện Dự án vườn ươm tạo công nghệ “Các giải pháp tận dụng kỹ thuật tính toán hiệu năng cao và tính toán lưới để giải các bài toán kỹ thuật thực tế”. Mục tiêu chính của dự án là tìm các giải pháp công nghệ có tính liên thông đa ngành để giải quyết những bài toán ứng dụng lớn xuất hiện trong kỹ thuật. Trong năm đầu tiên, Dự án tập trung tìm hiểu về các chuẩn grid; bộ công cụ Globus Toolkit 4.0; nghiên cứu các hệ thống grid trên thế giới; nghiên cứu phần mềm để liên kết hệ thống máy tính đã có với lưới tính toán; nghiên cứu các vấn đề về quản lý lưới; nghiên cứu quyền sử dụng trên lưới; hiện thực phần mềm tham gia lưới. Trong lĩnh vực ứng dụng, năm đầu tiên đề tài tập trung khảo sát bài toán thực tế; nghiên cứu các phương

pháp phân bố bài toán và dữ liệu; nghiên cứu bảo mật dữ liệu; nghiên cứu bảo vệ thông tin cá nhân trong dữ liệu; nghiên cứu đồng bộ các kết quả của bài toán con. Đối với bài toán tính toán lớn, nhóm nghiên cứu để thực hiện bài toán thiết kế vi mạch.

Trường Đại học Khoa học tự nhiên

Ở trường Đại học Khoa học tự nhiên, Đại học Quốc gia TP HCM, nhóm chuyên môn thuộc khoa Toán – Tin, Công nghệ sinh học đã phối hợp thực hiện một số công việc liên quan đến việc cài đặt gói phần mềm BLAST của Trung tâm thông tin công nghệ sinh học Hoa kỳ (NCBI – National Center for Biotechnology Information) để tìm kiếm các trình tự sinh học và thử nghiệm triển khai hệ thống grid trên mạng cục bộ của khoa.

Phân viện Công nghệ thông tin tại TP HCM

Phân viện Công nghệ thông tin tại TPHCM thực hiện hai đề tài liên quan đến tính toán lưới. Đề tài “Nghiên cứu và triển khai ứng dụng Grid có tính bảo mật cao” thực hiện từ 11/2004 đến 6/2006. Mục tiêu đề tài là ngoài nghiên cứu giải pháp xây dựng hệ thống tính toán mạng lưới có tính bảo mật cao, còn quan tâm đến khai thác năng lực của tính toán mạng lưới để giải quyết một số bài toán lớn như bài toán phân tải hoạt động trong mạng máy tính, bài toán tìm mức độ tương đồng của các trình tự sinh học. Nhóm thực hiện đề tài đã triển khai hệ thống tính toán lưới trên mạng toàn cầu kết nối với viện KISTI – Korea Institute of Science and Technology Information, Hàn Quốc; xây dựng cổng thông tin sinh tin học – <http://biogrid.ioit-hcm.ac.vn> - (Hình 2-15); thử nghiệm bài toán tìm mức độ tương đồng của các trình tự sinh học; xây dựng hệ thống lưu trữ và truy vấn dữ liệu sinh học.

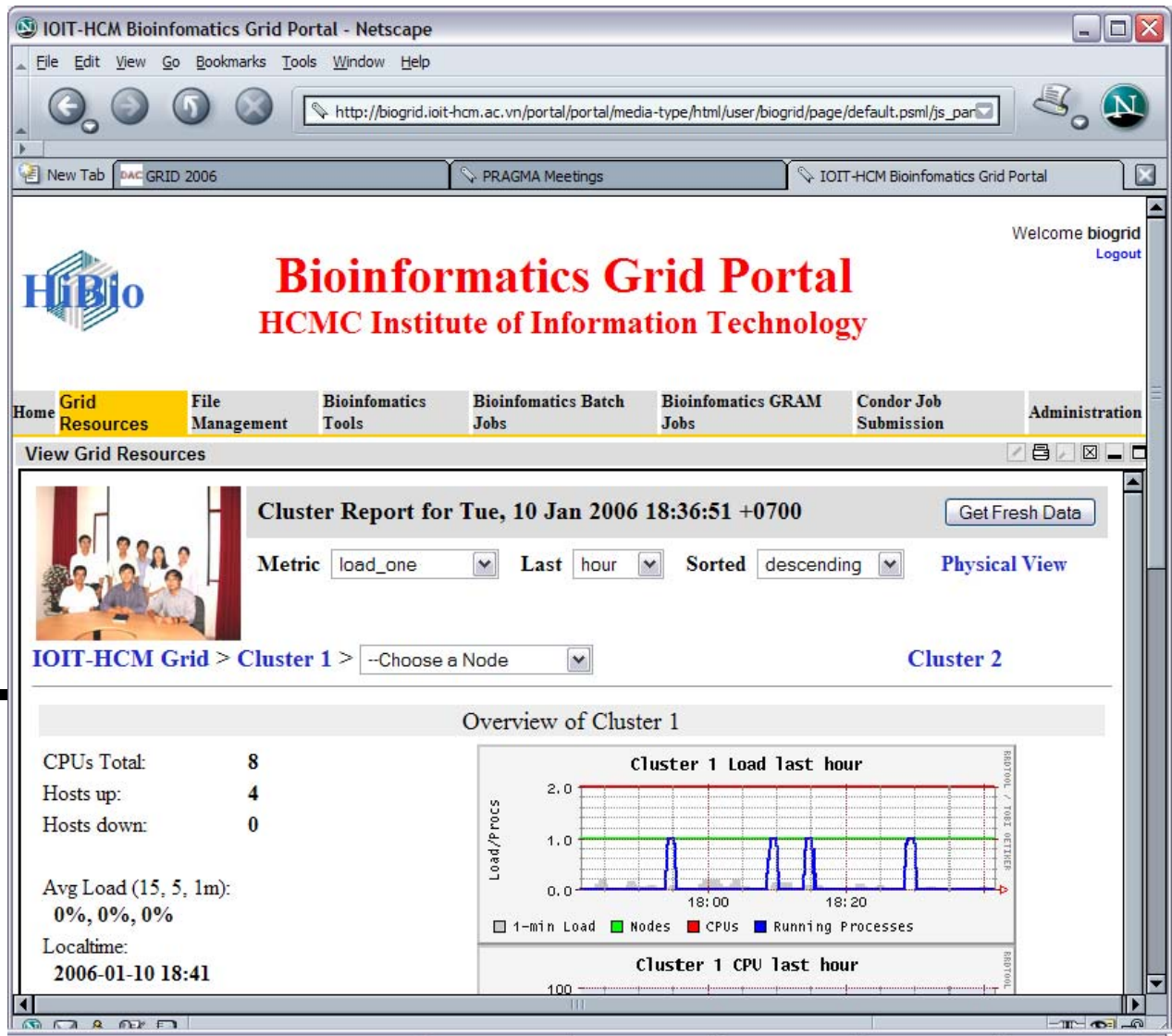
Trong năm 2005 và 2006, Phân viện Công nghệ thông tin tại TPHCM thực hiện đề tài “Tính toán mạng lưới trong việc giải quyết một số vấn đề của tin sinh học”. Nội dung nghiên cứu của đề tài là:

- Xây dựng môi trường tính toán mạng lưới với các máy tính hiện có của Phân viện Công nghệ thông tin tại TPHCM. Sau đó kết nối với máy tính của một số đơn vị khác trong toà nhà 1 Mạc Đĩnh Chi để tạo ra một hệ thống IntraGrid (hay ClusterGrid).
- Thử nghiệm thiết kế mạng lưới tính toán dạng quy mô của nhiều tổ chức (ExtraGrid, CampusGrid) cho Viện Khoa học và Công nghệ Việt Nam, từ đó tạo tiền đề để kết nối với các mạng lưới của khu vực Châu Á (InterGrid, GlobalGrid).
- Môi trường chuyển thông điệp MPICH-G2 cho việc hiện thực giải thuật song song trên hệ thống Grid.
- Các nguyên lý thiết kế phần mềm GridPortal, áp dụng tạo ra GridPortal cho tính toán Sinh học phân tử
- Nghiên cứu các phương pháp xử lý thông tin cho việc chẩn đoán cấu trúc và chức năng của protein; so sánh mức độ tương đồng của nhiều trình tự

Nucleotid, Amino Acid để tìm ra lịch sử tiến hoá. Từ đó tạo ra một số sản phẩm phần mềm phục vụ cho nhu cầu tính toán Tin sinh học.

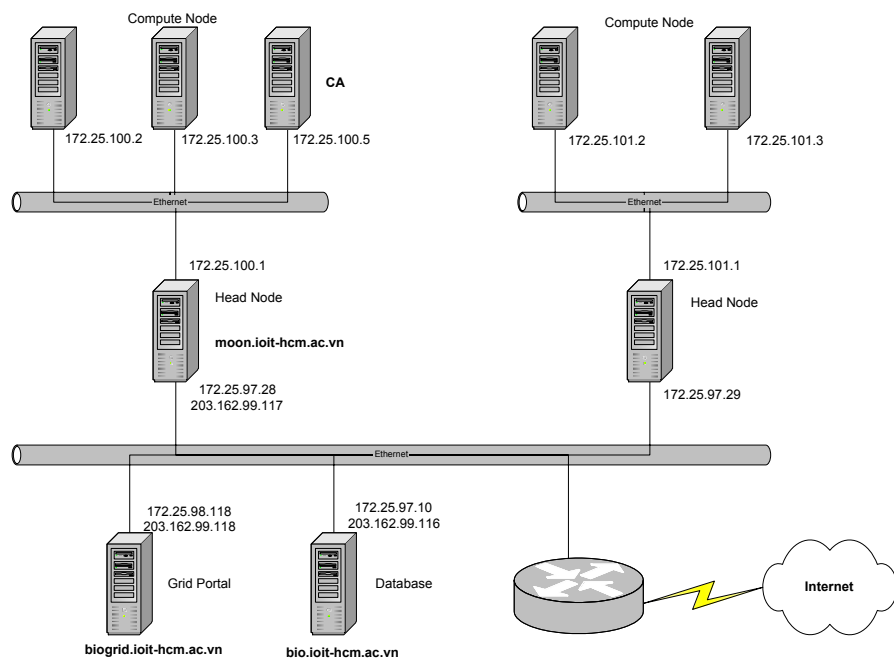
- Cách thức xây dựng phần mềm Tin sinh hoạt động trên grid.
- Cách thức triển khai thực hiện bài toán Tin sinh học trên môi trường tính toán mạng lưới.

Báo cáo tổng kết dự án



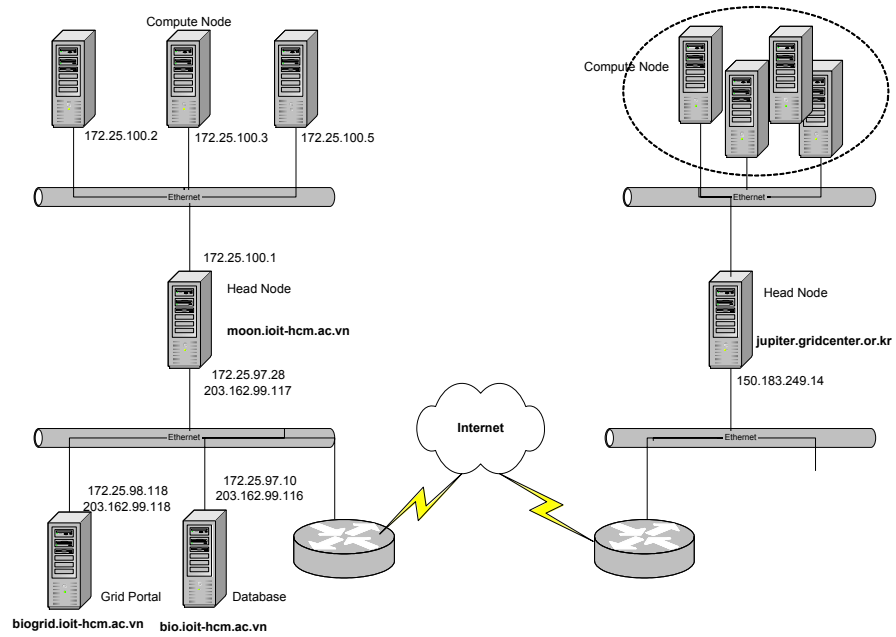
Hình 2-15. Bioinformatics Grid Portal của Phân viện Công nghệ thông tin tại TPHCM

Năng lực các máy trong hệ thống cluster hiện có của Phân viện Công nghệ thông tin tại TPHCM như Hình 2-16.



Hình 2-16. Sơ đồ hệ thống Grid của Phân viện CNTT tại TPHCM

Phía KISTI có 16 nút tham gia hệ thống, với 16 máy PC (Pentium IV 1.7GHz, 1024MB, 500GB HDD). Hệ thống kết nối giữa Phân viện Công nghệ thông tin với KISTI như Hình 2-17.



Hình 2-17. Sơ đồ kết nối hệ thống grid với hệ thống grid KIST

Trung tâm tính toán hiệu năng cao, Đại học Bách khoa Hà Nội

Tại Hà Nội, Trung tâm tính toán hiệu năng cao, Trường Đại Học Bách khoa đã có 4 năm kinh nghiệm nghiên cứu triển khai tính toán song song phân cụm. Các nghiên cứu triển khai về tính toán song song phân cụm dựa trên bộ công cụ quản lý tài nguyên và phân tải Open PBS. Trung tâm đã và đang triển khai một bộ công cụ phần mềm hệ thống trợ giúp người dùng thực hiện các tác vụ : quản trị cluster, theo dõi và đo hiệu năng cluster, thực hiện công việc từ xa. Đặc biệt, công cụ gỡ rối hỗ trợ lập trình song song do trung tâm nghiên cứu và phát triển thu hút được sự quan tâm của nhiều đồng nghiệp tại các hội thảo. Bộ công cụ này hiện tại đang được hoàn thiện và mở rộng thêm các chức năng biên dịch, soạn thảo, chạy chương trình song song.

Trung tâm cũng đã triển khai nghiên cứu và thử nghiệm tính toán lưới từ 2 năm qua và đã thu được một số kết quả bước đầu: (i) làm chủ được các bộ công cụ mã nguồn mở - Globus Toolkit, GridSphere, My Proxy - xây dựng tính toán lưới; (ii) tích hợp thành công các công cụ này để triển khai lưới thử nghiệm BK Grid;

(iii) bước đầu thử nghiệm thành công quy trình lưới hoá ứng dụng; (iv) đặc biệt, trung tâm đã tích hợp thành công công nghệ tác tử trong môi trường lưới, điều này mở ra nhiều triển vọng nghiên cứu các giải pháp quản lý tài nguyên lưới tính toán, quản lý dữ liệu trên lưới dữ liệu, vấn đề an toàn bảo mật dựa trên tác tử. Trung tâm cũng đã bắt đầu quan tâm đến công nghệ lưới dữ liệu, vấn đề hiệu năng trong tính toán lưới. Hiện tại, trung tâm đang triển khai thử nghiệm BKGrid trên 3 cluster theo mô hình Beowulf, mỗi cluster bao gồm 1 server 2 CPU và 6 đến 8 máy PC Pentium III 800/500 MHz.

Các hướng nghiên cứu chính của trung tâm không chỉ dừng lại ở phần hệ thống mà đã bắt đầu mở rộng sang ứng dụng. Trung tâm hiện đang quan tâm đến : (i) ứng dụng công nghệ tính toán song song giải quyết bài toán lượng tử, (ii) ứng dụng công nghệ lưới dữ liệu cho bài toán dự báo thời tiết theo mô hình số trị MM5.

Trung tâm có quan hệ với C-DAC, Đại học San-Jose và nhiều đơn vị có tiềm năng ứng dụng công nghệ tính toán hiệu năng cao trong nước: Học viện Mật

mã, Trung tâm quốc gia Dự báo thời tiết, Viện Khí tượng Thủy văn, khoa Toán-Tin ứng dụng và viện Vật lý kỹ thuật (ĐHBK HN).

Mục 2.2.3 sẽ trình bày tổng quan về các nội dung nghiên cứu về tính toán lưới đã được thực hiện tại trung tâm Tính toán hiệu năng cao – Đại học Bách Khoa Hà Nội trong khuôn khổ đề tài theo nghi định thư.

2.1.4. Tổng quan về mô phỏng trong khoa học vật liệu

Mô phỏng máy tính ra đời và phát triển cùng với sự xuất hiện ngày càng hoàn hảo của máy tính và ngày càng được khẳng định như một môn khoa học. Mô phỏng máy tính, mô hình hóa bằng máy tính ngày càng được áp dụng rộng rãi và thu được rất nhiều thành tựu trong các lĩnh vực nghiên cứu.

Có thể chia các phương pháp mô phỏng trong vật liệu thành: mô hình hóa các môi trường liên tục, mô hình hóa quy mô nguyên tử và mô hình hóa kết hợp cả hai phương pháp trên. Trong mô hình hóa các môi trường liên tục, vật liệu được coi như một mô hình liên tục và thường quá trình mô phỏng là giải phương trình

đạo hàm riêng bằng phương pháp phần tử hữu hạn hoặc sai phân hữu hạn. Trong khi đó, ở mô phỏng quy mô nguyên tử, vật liệu được xem như tập hợp của các nguyên tử riêng biệt có quy luật vận động riêng. Kỹ thuật mô phỏng quy mô nguyên tử đòi hỏi phải có máy tính càng mạnh càng tốt, và tùy theo từng trường hợp cụ thể mà có rất nhiều phương pháp, kỹ thuật được nghiên cứu, phát triển.

Các nhà hóa lý lượng tử chia các phương pháp mô phỏng quy mô nguyên tử thành hai lĩnh vực chính: Cơ học lượng tử và phương pháp cấu trúc điện tử.

Cấu trúc điện tử (còn gọi là phương pháp *ab initio*) dựa trên nền tảng là cơ học lượng tử, xuất phát từ việc giải phương trình Schrödinger với chỉ phép gần đúng đoạn nhiệt và gần đúng một điện tử, cho phép tính toán chính xác tính chất của vật liệu. Tuy nhiên, cũng vì vậy mà khối lượng tính toán của nó rất đồ sộ, thậm chí chỉ với hệ một vài nguyên tử, phân tử đã thường được giải trên các siêu máy tính, hơn nữa, hiện nay cũng chỉ giới hạn ở nghiên cứu các tính chất tĩnh.

Bên cạnh lý thuyết ab initio truyền thống ở trên, với sự ra đời của lý thuyết hàm mật độ, các kỹ thuật tính toán cấu trúc điện tử ngày càng được nghiên cứu, hoàn thiện, nhất là với sự xuất hiện của hướng tiếp cận hợp nhất động lực học phân tử và hàm mật độ đã phần nào có thể cho phép nghiên cứu hệ thống lớn hơn, cỡ vài chục nguyên tử và thường được sử dụng làm tiêu chuẩn so sánh với các phương pháp khác, đặc biệt là trong nhiều trường hợp không có dữ liệu thực nghiệm tương ứng.

Mặt khác, mặc dù thường kém chính xác hơn phương pháp ab initio, cơ học phân tử (phương pháp trường lực) lại có thể áp dụng cho hệ lớn hơn và cho phép nghiên cứu cả tính chất tĩnh và động. Trung tâm của phương pháp này là mô tả tương tác của các hạt trong hệ bởi một thế năng hiệu dụng là hàm của tọa độ các hạt nhân nguyên tử:

$$U(r_1, r_2, \dots, r_N) = \sum U^{(1)}(r_i) + \sum U^{(2)}(r_i, r_j) + \sum U^{(3)}(r_i, r_j, r_k) + \sum U^{(N)}(r_i, r_j, \dots, r_N)$$

Trong đó:

- U(1): thể 1 thành phần (tương ứng với thể năng trong trường ngoài)
- U(2): thể tương tác cặp
- U(3): thể 3 thành phần

Việc tìm biểu thức chính xác cho thể U là công việc hết sức khó khăn, có thể nói là khó có thể đạt được. Vì vậy, hiện nay người ta vẫn phải sử dụng các mô hình tương tác gần đúng như: Giả thể, Thể bán thực nghiệm và Thể thực nghiệm. Cũng cần phải nhấn mạnh là các thể bán thực nghiệm và thực nghiệm thường được xây dựng từ kết quả so sánh và làm khớp với cơ sở dữ liệu thực nghiệm, có thể một phần tính toán bằng *ab initio*, thậm chí bằng tay và quan trọng nhất là tùy thuộc vào đặc điểm của từng hệ vật liệu mà có mô hình thể tương tác hợp lý được phát triển ứng dụng.

Hiện nay, một số phương pháp mô phỏng quy mô nguyên tử được sử dụng rộng rãi để xây dựng mô hình và nghiên cứu tính chất của vật liệu là: Động lực học phân tử, Monte-Carlo và Cực tiểu hoá (còn gọi là hồi phục hoá). Riêng với vật

liệu vô định hình còn có phương pháp Random Network. Một điểm đáng chú ý là khi xây dựng mô hình vật liệu vô định hình có thể coi phương pháp hồi phục hoá (thống kê hồi phục) như một giới hạn của động lực học phân tử khi $T \rightarrow 0K$.

2.2. Những nội dung đã thực hiện

2.2.1. Kết quả nghiên cứu triển khai tính toán hiệu năng cao

2.2.1.1. Kết quả nghiên cứu, thiết lập hệ thống tính toán song song ghép cụm Bkluster

Nhu cầu về các hệ thống tính toán lớn đã thúc đẩy các hướng nghiên cứu trên nhiều lĩnh vực kiến trúc máy tính, phần cứng và phần mềm. Trong thập niên cuối cùng của thế kỷ trước, cách tiếp cận ghép các hệ thống nhỏ thành một hệ thống lớn hơn được xem là khả thi, đặc biệt phù hợp với điều kiện kinh tế, công nghệ của các nước đang phát triển. *Hệ thống tính toán song song ghép cụm dựa trên kiến trúc Beowulf và mô hình lập trình truyền thông điệp được triển khai tại Trung*

tâm Tính Toán Hiệu Năng Cao (ĐHBKHN) có tên gọi BCluster.

Trên thực tế, với một tập các máy tính được nối mạng và cài đặt một môi trường truyền thông giữa các tiến trình như LAM/MPI, ta đã có thể triển khai các chương trình tính toán song song dựa trên mô hình truyền thông điệp. Tuy nhiên, để hệ thống đó thực sự trở thành một dịch vụ tính toán, nó còn phải cho phép nhiều người dùng cùng khai thác một lúc cũng như phục vụ cho nhiều người lớp người dùng. Thực tế cho thấy, để khai thác được các hệ thống phân cụm ở mức thô sơ (chỉ được trang bị môi trường truyền thông giữa các tiến trình), người dùng phải có những kiến thức về hệ thống, hệ điều hành ở mức độ nhất định. Nói cách khác, những người có nhu cầu tính toán thực sự lại gần như không có khả năng sử dụng hệ thống. Hơn nữa, hầu hết việc triển khai của hệ thống không thực sự khai thác hết hiệu năng của hệ thống, gây lãng phí.

Xuất phát từ quan sát đó, nhóm phát triển mong muốn xây dựng hệ thống tính toán phân cụm BCluster thành một dịch vụ tính toán, thậm chí là một hệ thống tính toán thực sự hỗ trợ tối đa nhiều lớp người dùng khai thác hệ thống một cách hiệu quả:

- Xét từ phương diện một hệ thống tính toán, trước hết BKluster cho phép người dùng thực hiện một tác vụ tính toán trên hệ thống BKluster. Hơn thế, BKluster còn phải là một hệ đa nhiệm, đa phiên làm việc, nghĩa là cho phép nhiều người cùng khai thác hệ thống cùng một lúc một cách độc lập.
- Từ góc độ người phát triển ứng dụng, Bkluster bao gồm các công cụ hỗ trợ việc phát triển một chương trình tính toán được thực thi. Các công cụ này không chỉ hữu ích với những người chưa thực sự quen với việc phát triển chương trình song song, mà sẽ trở thành một công cụ chuẩn trong việc xây dựng các chương trình tính toán theo chuẩn MPI.
- Từ góc độ quản trị, Bkluster bao gồm các công cụ giúp những người quản trị hệ thống có khả năng nắm bắt toàn bộ thông tin hệ thống cũng như thiết lập các tham số giúp hệ thống hoạt động một cách hiệu quả nhất.

2.2.1.2. Gói phần mềm BKlusware

Tập hợp các mô đun phần mềm giúp BKluster trở thành một hệ thống tính toán được đặt tên là BKlusware.

Nhóm phát triển xây dựng BKlware dựa trên nguyên tắc phần mềm mã nguồn mở, theo hai hướng chính sau đây:

- Tận dụng gói phần mềm mã nguồn mở phù hợp với yêu cầu. Dựa trên đó, phát triển tinh chỉnh và bổ sung các module cần thiết để có thể tích hợp vào hệ thống. Đi theo hướng này, có thể kể đến gói phần mềm PBS trong việc xây dựng một bộ quản lý tác vụ của BKluster, GDB trong bộ công cụ hỗ trợ tìm lỗi chương trình BKPD hay Ganglia trong phần việc giám sát hệ thống.
- Ngoài việc tinh chỉnh và bổ sung cho những phần mềm sẵn có, để được một phần mềm hoàn thiện phù hợp với yêu cầu cụ thể, phải xây dựng các gói phần mềm khác như: công cụ quản lý các gói phần mềm trên một cluster, bộ dịch ngôn ngữ PCS (ngôn ngữ mô tả đơn giản cho phép mô tả các thao tác cơ bản của đại số tuyến tính).

Các chức năng chính của BKlware

Gói phần mềm BKlware bao gồm 3 thành phần chính:

- o Hệ thống phát triển chương trình
- o Hệ thống thực thi
- o Bộ công cụ quản trị và giám sát hệ thống

BKlaware hỗ trợ hai dạng kết nối giữa các dịch vụ hệ thống và dịch vụ người dùng: Kết nối cục bộ và Kết nối từ xa:

- Dưới hình thức kết nối cục bộ, tầng dịch vụ người dùng đóng vai trò là shell của hệ thống.
- Dưới hình thức kết nối từ xa theo mô hình client- server, tầng dịch vụ người dùng thực chất là các client của nhiều dịch vụ tích hợp.



Hình 2-18 - Các thành phần chính của BKlware

Hệ thống phát triển chương trình

Hệ thống phát triển chương trình bao gồm một tập các công cụ hỗ trợ người sử dụng trong tất cả các pha của quá trình viết mã cho một chương trình bao gồm: soạn thảo và quản lý mã nguồn, biên dịch tập mã nguồn thành chương trình thực thi và tìm lỗi chương trình.

Hệ thống phát triển chương trình cũng giúp người dùng có hai hướng tiếp cận để xây dựng chương trình tính toán:

- Xây dựng chương trình tính toán dựa theo mô hình lập trình thông điệp .
- Xây dựng chương trình thông qua đặc tả các thao tác tính toán dựa trên các phép toán đại số bằng ngôn ngữ PCS.

Theo hướng tiếp cận thứ nhất người dùng phải có kiến thức nhất định về lập trình song song đặc biệt là mô hình lập trình truyền thông điệp. Với những người lập trình chuyên nghiệp như vậy, các công cụ của hệ thống trợ giúp họ trong việc

~~soạn thảo mã lệnh cũng của, quản lý các file mã nguồn trong một dự án và biên~~
Trung tâm Tính Toán Hiệu Năng Cao - HUT

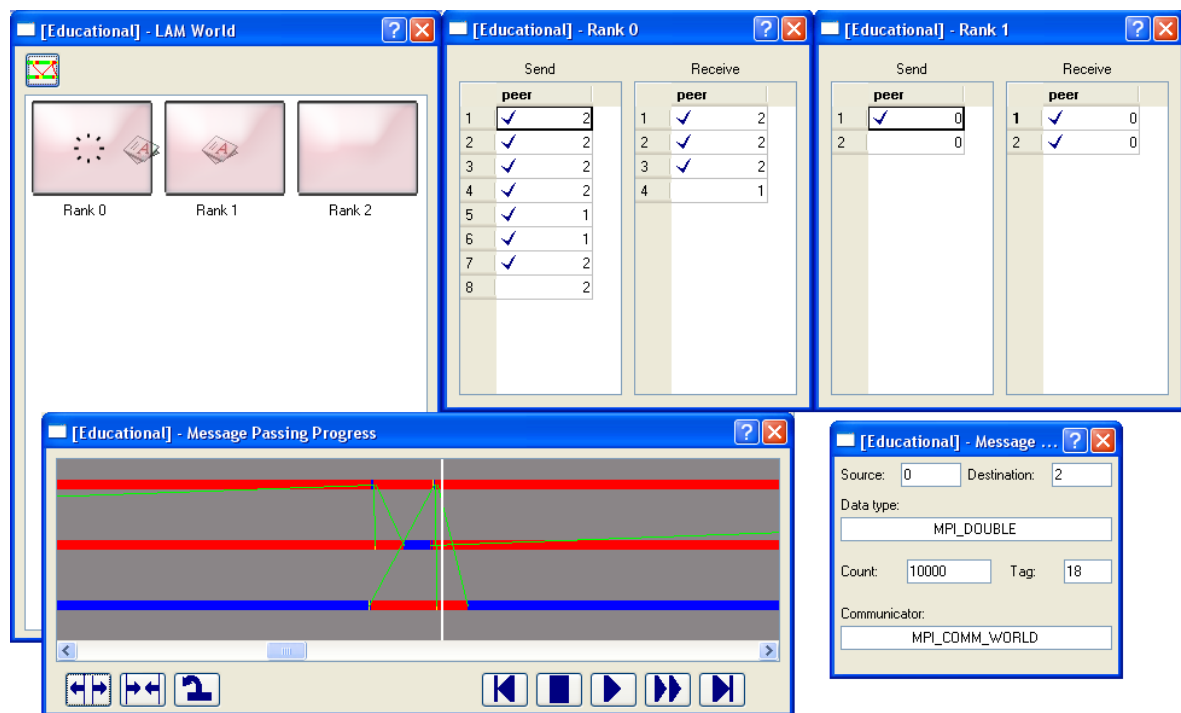
dịch chương trình bằng một lệnh duy nhất. Hướng tiếp cận thứ 2 dành cho những người không có nhiều kiến thức về lập trình song song nói riêng và tin học nói chung. Họ sẽ mô tả bài toán của mình thông qua các ma trận và các phép toán trên đó. Thứ tự của các phép toán cũng được mô tả thông qua một số lệnh lặp và điều kiện đơn giản. Việc biên dịch hệ thống gắn liền với việc “song song hóa bài toán”.

Hiện tại, module gỡ rối chương trình mới chỉ hỗ trợ những người lập trình chuyên nghiệp. Nói cách khác, module này mới chỉ áp dụng cho các chương trình được xây dựng theo hướng tiếp cận đầu tiên. Với module này, người dùng có thể:

- Xác định trạng thái hiện tại của mỗi tiến trình
- Xem nội dung các thông điệp đã trao đổi
- Điều khiển luồng chương trình

Việc xác định trạng thái hiện tại của từng tiến trình cục bộ được thực hiện bằng cách xem giá trị của các biến trong các chương trình đó tại thời điểm mong

muốn. Thời điểm được xác định sẽ được đặt các break point thông qua giao diện của cửa sổ soạn thảo mã nguồn. Người dùng cũng có thể xem xét nội dung thông điệp thông qua các biến vùng đệm được tạo ra trước và sau mỗi khi gửi và nhận thông điệp trên các tiến trình. Cuối cùng, người dùng có theo dõi luồng chương trình bằng cách sử dụng các break point hợp lý và cơ chế chạy, dừng chương trình tại các break point đã định.



Hình 2-19 - Cửa sổ theo dõi truyền thông điệp giữa các tiến trình

Trong hệ thống phát triển chương trình, nhóm nghiên cứu cũng đề xuất môi trường kiểm thử cho phép kiểm thử các chương trình trong quá trình xây dựng. Ở mức độ đơn giản nhất, môi trường này cho phép thực thi trên hệ thống và cho phép thu thập kết quả trả lại cho người dùng. Về mặt chức năng, môi trường kiểm thử chương trình gần giống môi trường thực thi tính toán. Tuy nhiên, việc thu thập kết quả được thực hiện ở mức thấp và thô hơn. Nghĩa là kết quả chạy trên từng nút được gửi về mà không có sự tổng hợp.

Hệ thống thực thi

Hệ thống thực thi gồm các công cụ và ứng dụng giúp người dùng khai thác khả năng tính toán của hệ thống. Người dùng khai thác hệ thống dưới hai hình thức:

- Đề trình yêu cầu tính toán bằng công cụ đề trình công việc
- Sử dụng các ứng dụng đã cài đặt sẵn

Với một số ứng dụng có sẵn, người dùng có thể tận dụng khả năng tính toán để ~~giải quyết một số bài toán với kích thước lớn trong một số lĩnh vực như tối ưu phi~~

tuyến, phân tích dữ liệu và mô hình hóa. Đây là những ứng dụng được xây dựng chuyên biệt cho hệ thống và phục vụ cho các mục đích tính toán chung trong một phạm vi rộng. Các lĩnh vực được chọn mang tính phổ biến nhưng không đảm bảo phù hợp cho tất cả các bài toán của người dùng. Điều này đòi hỏi phải có những bước tiền xử lý đầu vào cũng như xử lý kết quả cho phù hợp với các ràng buộc mà ứng dụng yêu cầu.

Ngoài việc sử dụng các ứng dụng cụ thể, người dùng có thể đề đạt một yêu cầu tính toán của riêng mình thông qua công cụ để trình công việc. Một yêu cầu tính toán được mô tả bởi 2 phần chính:

- Tập các chương trình thực thi
- File đặc tả các tham số cấu hình tính toán

Tập các chương trình mô tả quá trình tính toán, trong đó, mỗi tiến trình cụ thể được tạo ra là một thể hiện của một chương trình trong tập này. Các chương trình được biên dịch ở dạng nhị phân có khả năng thực thi trên hệ thống. Trong trường hợp, người dùng có chương trình mã nguồn, họ có thể sử dụng Hệ thống

phát triển chương trình để tạo ra các chương trình thực thi. Hướng tiếp cận được khuyến cáo vì nó đảm bảo chương trình có thể chạy tốt trên hệ thống.

File đặc tả tham số cấu hình tính toán chứa các thông tin để thực thi chương trình trong đó quan trọng là số thể hiện cụ thể của mỗi chương trình và các thể hiện này được thực thi trên bao nhiêu nút tính toán. Bộ các tham số được đề xuất dựa trên các tham số yêu cầu cho một công việc của PBS (một số tham số của PBS đã được cụ thể hóa theo cấu hình của hệ thống).

Các yêu cầu tính toán được đề xuất thông qua một giao diện, trong đó tập các chương trình thực thi có thể được lựa chọn trên thư mục làm việc hiện hành của người dùng trên BKluster hoặc được upload từ hệ thống cục bộ. Các tham số sẽ được điền vào bởi người dùng hoặc do hệ thống tự xác định nếu người dùng chọn cấu hình mặc định

Submit Job

File ma nguon

	Ten file ma nguoc
1	thu
2	thu1

Ten file kích ban

Mac dinh

Lua chon

Thuoc tinh lua chon

So nut tinh toan 1 So luong Cpus 20

Ten hang doi #workq@may22

Dung luong bo nho can su dung 10mb

Walltime

Gio 0 Phut 0 Giay 0

De tring cong viec

Hình 2-20 - Giao diện công cụ đệ trình công việc

Bộ công cụ quản trị và giám sát hệ thống

Bộ công cụ quản trị và giám sát hệ thống giúp người quản trị nắm bắt toàn bộ các thông tin bên trong hệ thống. Đây là công cụ giành cho người quản trị, nó đưa lại một cái nhìn về hệ thống khác hẳn hai hệ thống thực thi và phát triển chương trình. Hai hệ thống trên cố gắng che lấp sự phức tạp phía dưới của hệ thống sao cho tính phân tán và các yếu tố cấu thành hệ thống phân cụm là trong suốt với người dùng. Người dùng nhìn nhận hệ thống như một “máy tính” thông thường. Ngược lại, để có thể khai thác hiệu quả bộ công cụ hiểu rõ các thông tin được cung cấp, người quản trị phải có những hiểu biết về hệ thống phân cụm. Bộ công cụ được đề xuất để đơn giản hóa các thao tác của người quản trị và hiện thị thông tin một cách trực quan.

Các thông tin về hệ thống được chia làm hai lớp

- Các thông tin tĩnh
- Các thông tin động

Các thông tin tĩnh chủ yếu mô tả cấu hình hệ thống bao gồm số lượng nút, cấu hình từng nút, các tham số hệ thống và người dùng hệ thống. Các thông tin này

có tính ổn định cao. Cùng với khả năng thu thập các thông tin mang tính quản lý, thông tin về hiệu năng hệ thống cũng được cung cấp thông qua một bộ đánh giá hiệu năng của hệ thống.

Các thông tin động phục vụ việc giám sát hệ thống, các thông tin này biến thiên liên tục trong khoảng thời gian nhỏ. Các thông tin này bao gồm, số công việc đang được thực thi, số tiến trình trên mỗi nút và tài nguyên hiện tại được sử dụng. Tài nguyên ở đây để cập chủ yếu tới phần trăm sử dụng CPU, bộ nhớ trong và thông lượng mạng.

Ngoài việc cung cấp một cách tương đối đầy đủ và toàn diện các thông tin liên quan tới hệ thống, bộ công cụ cũng cho phép người quản trị thao tác trên các thông tin này, đặc biệt là các thông tin quản lý. Các thao tác chủ yếu là thêm, bớt, tìm kiếm một nút tính toán, xác lập các tham số hệ thống.

Báo cáo tổng kết dự án

Bkcluster

File Edit Help

Quan Tri Server

Choose Object: Node&Queue management

Update Time: 30 seconds

Queue Management

Queues

- Execution Queues
 - EXE - deque
 - EXE - exeque
- Other Queues
 - NOR - run_que
 - NOR - test
- Routing Queues
 - ROUT - newque

Property	Value	Explanation
1 queue_type	Execution	
2 Priority	80	
3 total_jobs	0	
4 state_count	Transit:0 Queued:0 Held:0 Waiting:0 Running:0 Exiting:0	
5 acl_group_enable	True	

Change Queue Properties

Name: mem Value: Unset OK

Delete Queue Create Queue

Node Management

Nodes Status

- Free Nodes
 - localhost
- Down Nodes
 - Unknown
 - sun

Property	Value	Explanation
1 state	free	
2 np	5	
3 properties	red,blue	
4 ntype	cluster	
5 loadave	1.11	
6 physmem	262080kb	

Update Node Properties

Name: Value: OK

CPU Usage: 86%

Memory Usage: 73%

Add Node Delete Node

Updating

Hình 2-21 Giao diện quản trị thông tin về nút tính toán của hệ thống

Những đặc điểm mới, sáng tạo của BKlusware

BKlusware là gói phần mềm mang **đầy đủ tính năng, hỗ trợ nhiều mức người sử dụng khác nhau** trong việc xây dựng, vận hành, quản trị hệ thống tính toán song song phân cụm và **phù hợp với điều kiện vận hành thực tế trong nước.**

- Hiện nay, ở Việt Nam chưa có những gói phần mềm chuyên dụng, phục vụ cho việc cấu hình, vận hành, theo dõi và sử dụng các hệ thống tính toán song song ghép cụm. BKlusware được phân tích, thiết kế và xây dựng dựa trên những kinh nghiệm thực tế của quá trình nghiên cứu, triển khai và vận hành hệ thống tính toán song song theo mô hình ghép cụm tại Trung tâm Tính Toán Hiệu Năng Cao. Với phương châm phát triển dựa trên những gói phần mềm mã nguồn mở có sẵn và bổ sung, xây dựng thêm các tính năng còn thiếu, BKlusware kế thừa những kết quả nghiên cứu, phát triển đã có trên Thế Giới, đồng thời thể hiện được những đóng góp của tập thể nhóm nghiên

cứu tại trường Đại Học Bách Khoa Hà Nội trong lĩnh vực tính toán hiệu năng cao.

- BKlusware là một gói phần mềm trọn vẹn, đầy đủ các tính năng phục vụ cho việc xây dựng, vận hành và quản trị một hệ thống tính toán song song phân cụm. Đứng trên phương diện người dùng, gói phần mềm BKlusware hỗ trợ sử dụng nhiều mức: người dùng thuần tuý, người phát triển chuyên nghiệp và người quản trị hệ thống:
 - o Người dùng thuần tuý khai thác hệ thống dựa trên các phần mềm tính toán có sẵn hoặc đưa ra yêu cầu tính toán dựa trên một ngôn ngữ đặc tả của hệ thống đề ra. Đối với nhóm người dùng này, khái niệm song song là trong suốt.
 - o Người phát triển chuyên nghiệp có thể dùng các chức năng của BKlusware để viết các phần mềm, các chương trình tính toán nhằm tối ưu hoá dựa trên đặc trưng của từng bài toán cụ thể. Các chương trình được phát triển theo mô hình lập trình truyền thông điệp. Tuy nhiên, ở mức độ này người dùng cũng có thể bỏ qua một số vấn đề phức tạp trong việc

- tính toán song song như việc phân tải, sắp đặt các chương trình song song trên các nút tính toán. Nói cách khác, với những người phát triển chuyên nghiệp, BKlware đã che lấp sự phức tạp trong tổ chức và các thành phần hệ thống, giúp người dùng nhìn BKluster như một máy tính nhiều bộ vi xử lý.
- o Lớp người dùng cuối cùng – người quản trị mạng buộc phải nhìn hệ thống một cách chân thực – một tập các máy tính được “liên kết” dựa trên môi trường truyền thông. Các công cụ quản trị và theo dõi được xây dựng không phải để che đậy sự phức tạp của hệ thống. Nó chỉ giúp người dùng giảm bớt khối lượng công việc phải làm của người quản trị cũng như hiện thị một cách trực quan về hệ thống, giúp người quản trị có những quyết định đúng đắn.
 - BKlware phù hợp với điều kiện vận hành và thực tế sản xuất tại Việt Nam: ngày nay công tác nghiên cứu, mô phỏng thường đòi hỏi phải đưa ra kết quả xử lý một khối lượng dữ liệu rất lớn trong khoảng thời gian chấp nhận được. Việc xây dựng các hệ thống tính toán song song ghép cụm (cluster) từ những

máy tính có cấu hình đơn giản (ví dụ: các PC) tỏ ra phù hợp và hiệu quả với những trung tâm, viện nghiên cứu không có đủ điều kiện trang bị các siêu máy tính. Gói phần mềm BKlusware sẽ là một giải pháp cho việc cấu hình, cài đặt, vận hành một tập các máy tính có cấu hình đơn giản như một siêu máy tính duy nhất.

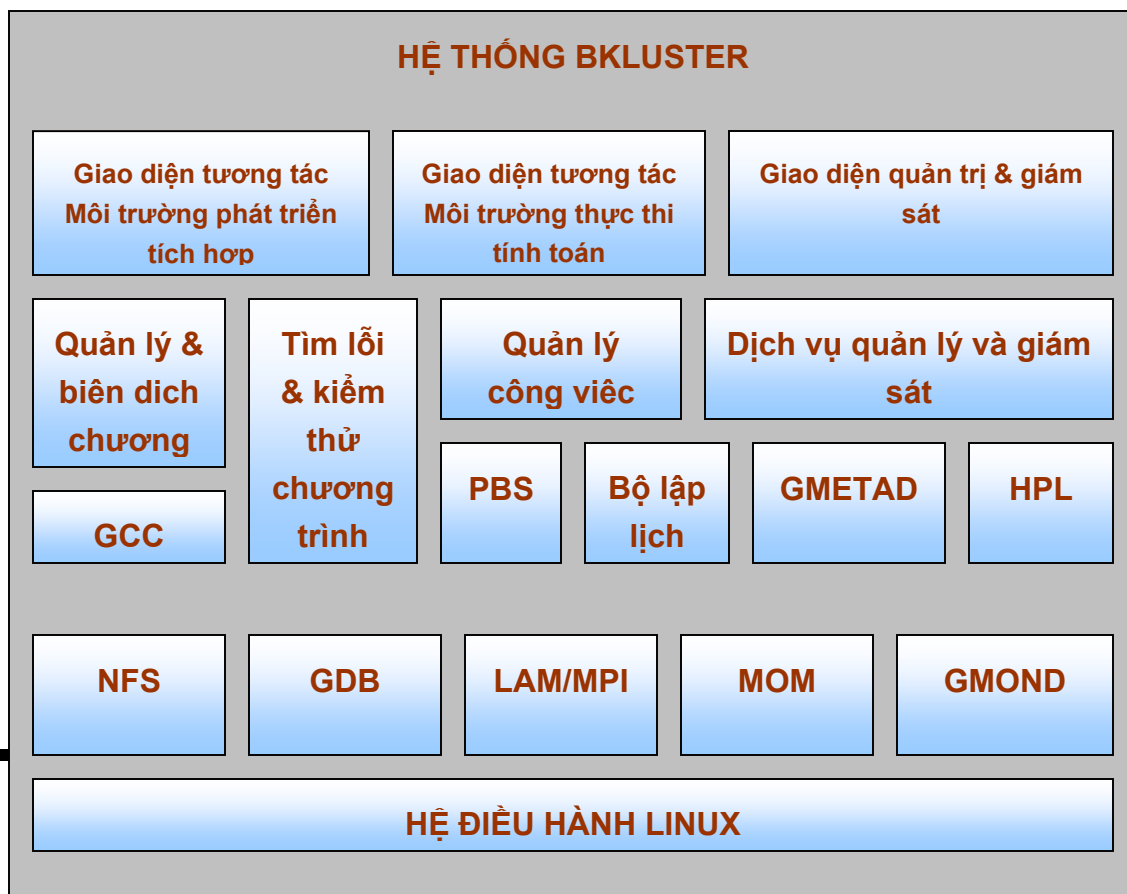
Các kết quả đạt được và khả năng ứng dụng

Việc phát triển BKlusware về cơ bản đã hoàn thành. BKlusware đang được triển khai, đánh giá, kiểm thử trên hệ thống BKluster tại Trung tâm Tính Toán Hiệu Năng Cao trường Đại Học Bách Khoa Hà Nội.

Lĩnh vực ứng dụng của sản phẩm: BKlusware có thể được sử dụng như một giải pháp hoàn chỉnh trong việc cài đặt, cấu hình, vận hành, quản trị các hệ thống tính toán song song ghép cụm từ các máy tính có cấu trúc đơn giản (PC) tại các trung tâm, viện nghiên cứu.

Dự kiến quy mô ứng dụng: BKlusware có thể triển khai trên các hệ thống tính toán song song theo mô hình ghép cụm, không phụ thuộc vào số nút tính toán.

Một hệ thống tính toán sử dụng BKlusware hoàn toàn có thể tham gia vào Lưới Tính Toán (Grid Computing) – một công nghệ mới trong lĩnh vực tính toán phân tán nhằm tận dụng năng lực tính toán các đối tác.



Hình 2-22 -Triển khai BKlware trên hệ thống BKluster

2.2.2. Kết quả nghiên cứu triển khai chống lỗi

Khả năng chống lỗi trong hệ thống BKluster được tạo thành từ sự kết hợp hoạt động của 3 gói phần mềm mã nguồn mở LAM/MPI, PBS và BLCR với môđun chống lỗi do nhóm nghiên cứu phát triển BKFT (Bach Khoa Fault Tolerance). Với môi trường tính toán song song được thiết lập bởi LAM/MPI, khi một tiến trình con của một chương trình song song gặp lỗi sẽ khiến cho toàn bộ chương trình phải thực hiện lại từ đầu, điều này gây ra sự lãng phí rất lớn về thời gian cũng như tài nguyên tính toán vì đặc điểm của các ứng dụng song song là cần thời gian chạy lâu và số lượng tài nguyên tính toán rất lớn. Phần mềm BLCR là kết quả nghiên cứu của trường Đại học Berkeley, Hoa Kỳ cho phép tạo ra khả năng chống lỗi bằng phương pháp checkpoint cho LAM/MPI. BLCR cung cấp cho người sử dụng các câu lệnh để có thể ghi lại checkpoint của một chương trình

song song bất kỳ cũng như khởi động lại chương trình song song từ những checkpoint đó. Khi hoạt động kết hợp với LAM/MPI, BLCR đã thiết lập được một môi trường tính toán song song có khả năng checkpoint/restart các ứng dụng.

Tuy nhiên, khi triển khai trong mô hình chung của hệ thống BKluster, môi trường song song chống lỗi LAM/MPI – BLCR đã có những hạn chế sau:

- Không tương tác được với hệ thống quản lý tài nguyên và phân tải PBS để có thể lấy được danh sách các ứng dụng song song đang chạy trên hệ thống.
- Khi một chương trình song song được khởi động lại bằng BLCR, PBS không biết và không quản lý được công việc này

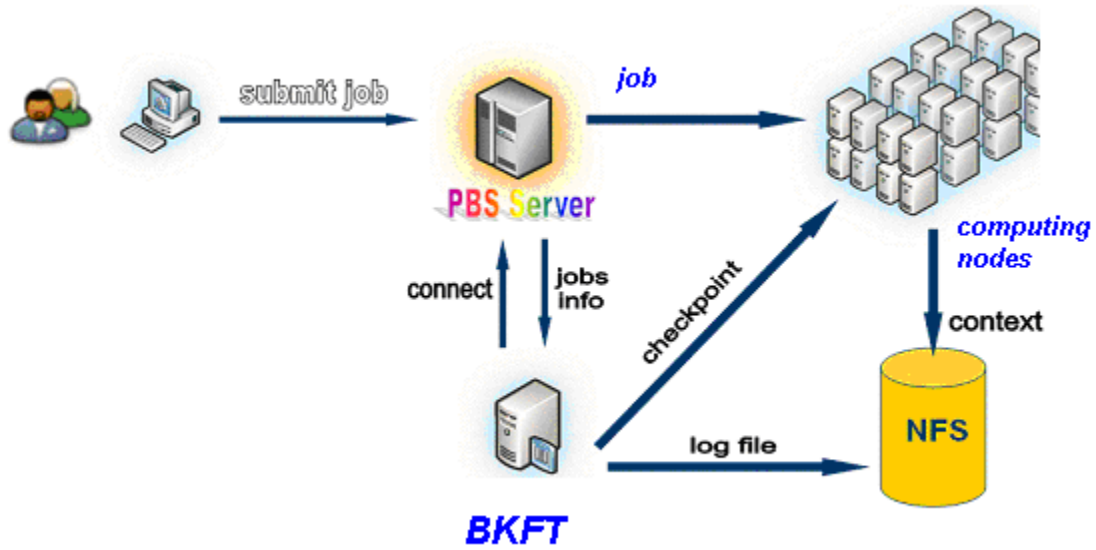
Để khắc phục những nhược điểm trên, nhóm nghiên cứu đã phát triển mô đun chống lỗi BKFT với những chức năng sau:

- Tương tác với PBS để lấy được danh sách các công việc đang thực hiện trong hệ thống.

- Gọi các lệnh của BLCR nhằm lấy checkpoint của một hoặc một số công việc bất kỳ do PBS quản lý
- Quản lý các checkpoint trong 1 cơ sở dữ liệu
- Đề trình lại công việc song song từ các checkpoint với PBS

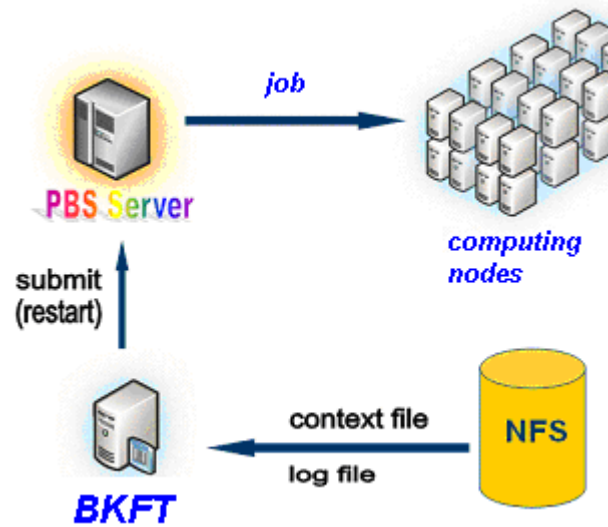
Kịch bản thực hiện công việc của hệ thống BKluster với khả năng chống lỗi như sau:

- Công việc song song được đề trình đến PBS server qua Hệ thống thực thi
- BKFT định kỳ truy vấn đến PBS server để cập nhật danh sách công việc
- BKFT định kỳ lấy checkpoint các công việc đang thực hiện trên hệ thống. BKFT cũng cho phép người quản trị lấy checkpoint của một công việc vào một thời điểm bất kỳ
- Các checkpoint được cập nhật vào cơ sở dữ liệu trên hệ thống file mạng NFS



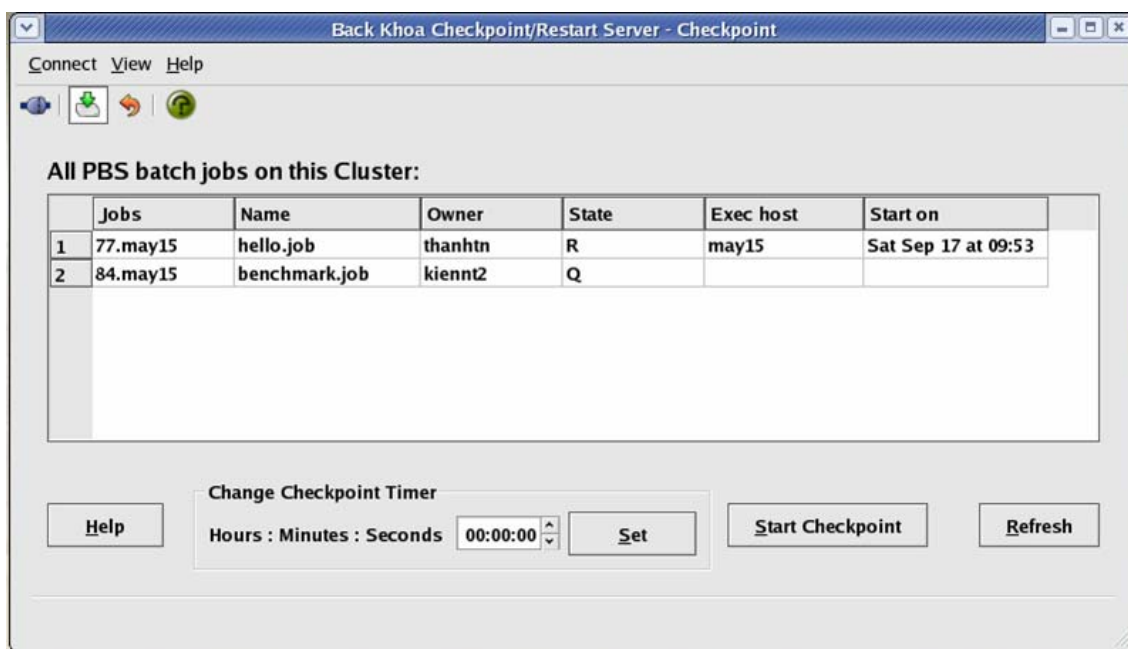
Hình 2-23 Hoạt động của BKluster với môi trường tính toán song song chống lỗi
Khi một hoặc một số tiến trình gặp lỗi, nó sẽ được BKFT khởi động lại như sau:

- Đọc lại checkpoint từ cơ sở dữ liệu
- Tự động sinh ra script, khởi động lại công việc từ thời điểm lấy checkpoint dưới dạng một công việc được quản lý bởi PBS

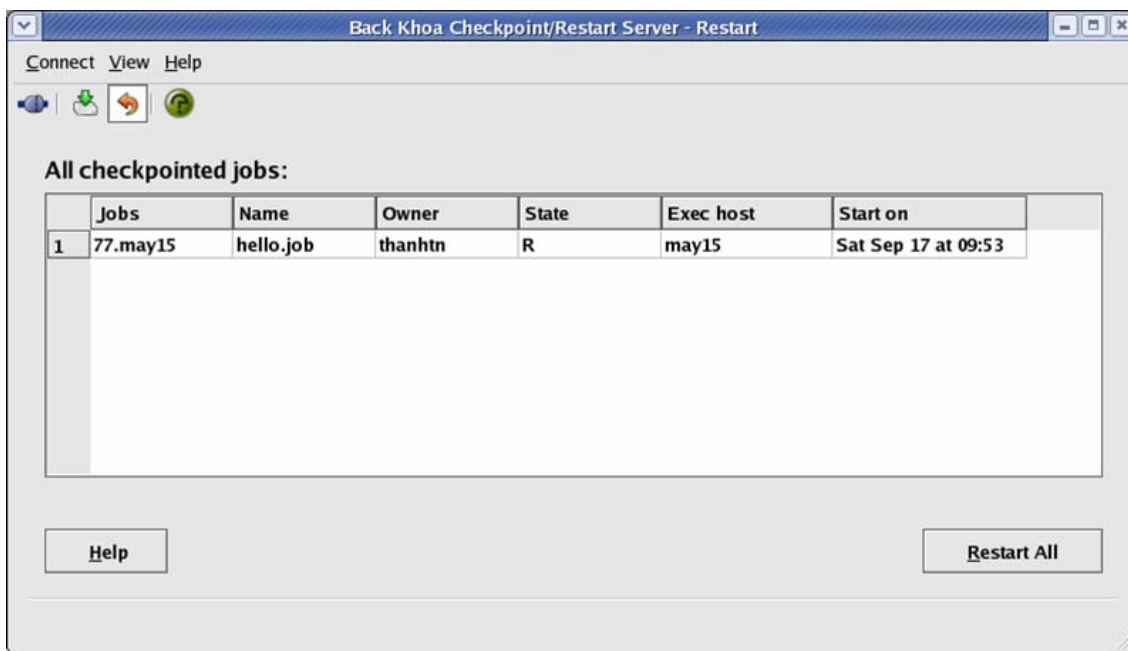


Hình 2-24 Khởi động lại công việc với BKFT

Môđun BKFT đã được hoàn thành và tích hợp hoạt động tốt cùng với gói phần mềm BKlusware. Với khả năng chống lỗi, BKluster có thể phục hồi lại toàn bộ công việc từ thời điểm gần nhất trước khi xảy ra lỗi hệ thống.



Hình 2-25 Giao diện chương trình BKFT - chức năng lấy checkpoint



Hình 2-26 Giao diện chương trình BKFT - chức năng restart

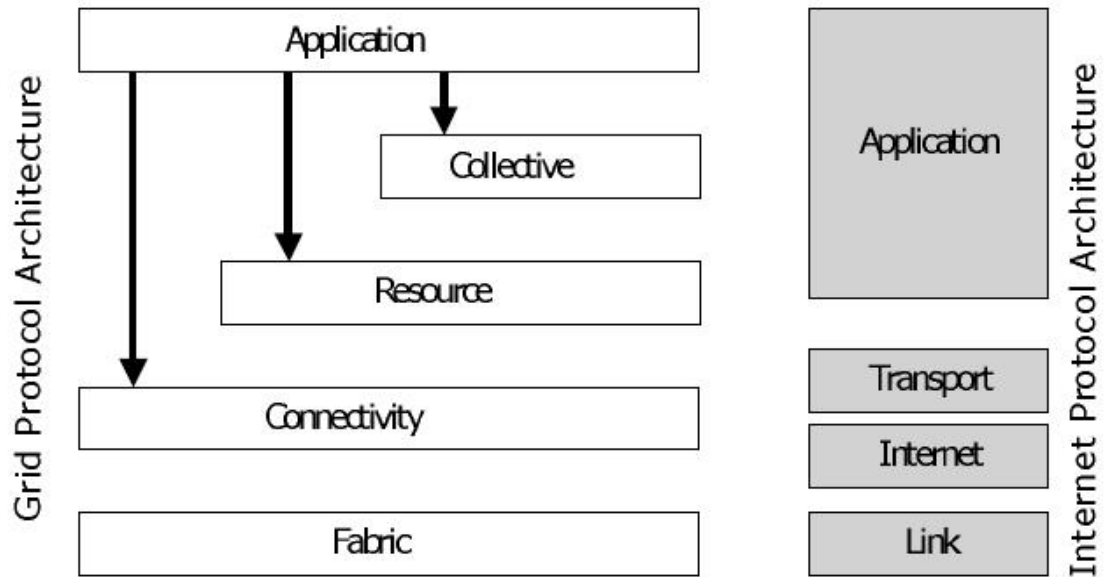
2.2.3. Kết quả nghiên cứu triển khai tính toán lưới

2.2.3.1. Tiếp cận dịch vụ trong triển khai lưới

Kết quả nghiên cứu cũng chỉ ra các thành phần của một hệ thống tính toán lưới bao gồm: thành phần quản lý, phần mềm donor, phần mềm đệ trình, thành phần quản lý phân tán, bộ lập lịch, các thành phần truyền thông, các thành phần quản lý, theo dõi và đo lường.

Việc sử dụng lưới cũng được đề cập dưới góc nhìn của người phát triển hệ thống, với các thao tác: lập kế hoạch, cài đặt, quản trị người dùng và tài nguyên và chia sẻ dữ liệu. Đối với người sử dụng, các thao tác được quan tâm là: đăng ký và cài đặt phần mềm lưới, đăng nhập hệ thống, truy vấn và đệ trình các công việc, cấu hình dữ liệu, đặt trước tài nguyên.

Kiến trúc chung của lưới:



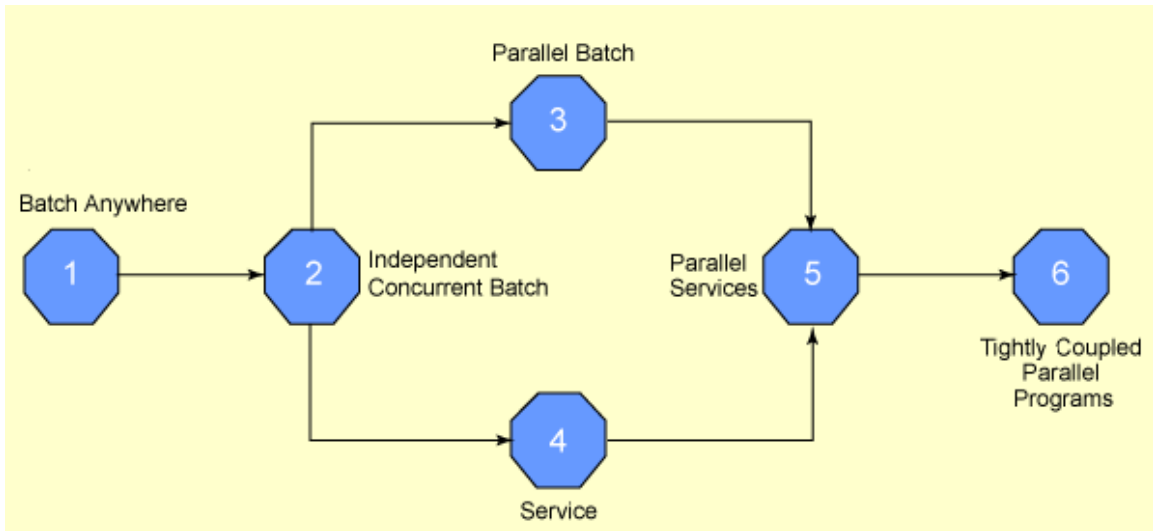
Kiến trúc giao thức lưới được nghiên cứu trong sự so sánh với kiến trúc giao thức Internet. Với bản chất bao gồm khả năng liên tác, các giao thức và các dịch vụ chuẩn cho để các APIs, SDKs, kiến trúc lưới được phân thành các tầng bao

gồm: tầng Fabric, tầng Connectivity, tầng Resource, tầng Collective và tầng Application. Các thành phần trong một tầng có một số đặc điểm chung và được xây dựng trên khả năng của các tầng thấp hơn.

Phương pháp luận về việc lưới hóa ứng dụng

Một trong những đặc điểm nổi bật của tính toán lưới so với những công nghệ trước đây là tính hướng dịch vụ của nó. Bởi vậy, phương pháp luận về việc lưới hóa ứng dụng cũng như các kỹ thuật liên quan đến lập trình dịch vụ lưới đã được đầu tư nghiên cứu khá kĩ càng.

Hình 2-27. Tổng quan phương pháp luận về lưới hóa ứng dụng



Hình 2-28. Sáu bước lưới hóa ứng dụng

Hình 2-27 và Hình 2-28 mô tả tổng quan phương pháp luận về lưới hóa ứng dụng và các bước thực hiện. Theo phương pháp luận này, bước đầu tiên của Trung tâm Tính Toán Hiệu Năng Cao - HUT

quá trình lưới hoá là khảo sát quá trình thực hiện đơn giản của một ứng dụng trong một lưới. Sau đó là tìm hiểu cách các ứng dụng cần lưới hoá sử dụng hai bước đầu tiên để có thể thực thi như là các công việc theo lô đơn lẻ hoặc song song. Tiếp theo, các bước ba và bốn chỉ ra cách chia nhỏ một ứng dụng theo lô và dịch vụ hoá ứng dụng đó để phục vụ người sử dụng thông qua phần đệm. Hai bước cuối cùng sẽ đưa ra một số dạng thức (paradigm) khả thi để triển khai một ứng dụng sử dụng dịch vụ lưới theo cách song song. Mặc dù các bước được đề cập dưới đây là cần thiết trong quá trình lưới hoá ứng dụng nhưng không bắt buộc phải thực hiện đủ cả năm bước. Rất nhiều ứng dụng chỉ đạt được bước thứ năm và rất ít có thể đạt được bước sáu. Bước thứ sáu được chuyên biệt hoá cho một số ứng dụng và được thiết kế từ ban đầu để hỗ trợ cho mô hình xử lý song song kết nối chặt (tightly coupled) cho các chương trình mà các chương trình này tạo nên ứng dụng hoàn chỉnh.

Vấn đề bảo mật trong môi trường lưới

Do đặc điểm hỗn tạp và không đồng nhất của các tổ chức và tài nguyên trong lưới, vấn đề bảo mật trong lưới là một trong những vấn đề được quan tâm hàng đầu. Có những vấn đề *bảo mật* mới chưa từng gặp trong các công nghệ bảo mật hiện tại cho hệ thống tính toán phân tán truyền thống. Ví dụ, các tính toán song song đòi hỏi nhiều tài nguyên tính toán, dẫn tới nhu cầu phải thiết lập các mối quan hệ bảo mật, không đơn giản chỉ là với client và server, mà giữa hàng trăm tiến trình thực hiện trong không gian tập hợp nhiều miền quản trị. Ngoài ra, cần phải có các chính sách bảo mật liên miền cho lưới, các công nghệ điều khiển truy nhập giữa các miền khác nhau cũng phải được hỗ trợ.

Các ứng dụng và hệ thống lưới có thể đòi hỏi bất cứ chức năng nào trong các chức năng cơ bản của bảo mật như là: chứng thực, điều khiển truy nhập, toàn vẹn, bí mật. Khi phát triển kiến trúc bảo mật lưới, cũng cần phải lựa chọn giải pháp để đáp ứng được đòi hỏi của các đặc tính rất riêng của lưới:

Đăng nhập một lần: Khi bắt đầu một tính toán đòi hỏi sử dụng tài nguyên, cho thuê tài nguyên hay truyền thông nội bộ, người dùng có thể được chứng thực, và sẽ không phải chứng thực trong các tính toán tiếp theo.

Giấy ủy nhiệm người dùng (mật khẩu, khóa bí mật, ...): phải được bảo vệ bằng các chính sách như mã hóa, hệ thống file bảo mật, phân quyền, ...

Tích hợp các giải pháp bảo mật địa phương: các giải pháp bảo mật liên miền phải tích hợp với các giải pháp bảo mật địa phương để đảm bảo độc lập của các thành viên lưới.

Cơ sở hạ tầng giấy ủy nhiệm, giấy chứng nhận thống nhất: Truy nhập liên miền đòi hỏi phải có một quy ước thống nhất để biểu diễn định danh của các thực thể lưới như là người dùng, tài nguyên,... Vì thế, cần có một chuẩn để mã hóa các giấy chứng nhận cho mục đích bảo mật. Hiện tại, X509 là chuẩn cho các giấy chứng nhận phổ biến trong môi trường lưới.

Hỗ trợ bảo mật nhóm truyền thông: Một tính toán có thể đòi hỏi một số các tiến trình, cùng cộng tác các hoạt động của chúng với nhau như là một nhóm. Tổ hợp các nhóm tiến trình sẽ thay đổi trong vòng đời của một tính toán. Vì thế, cần cung cấp bảo mật truyền thông nhóm động. Không có giải pháp bảo mật nào hiện tại hỗ trợ tính năng này, thậm chí là thư viện lập trình bảo mật GSS-API còn không cung cấp bảo mật nhóm.

Độc lập công nghệ: các chính sách bảo mật không phục vụ cho một công nghệ phát triển ứng dụng cụ thể nào. Hơn nữa, có thể cài đặt các chính sách bảo mật trong một phạm vi các công nghệ bảo mật, dựa trên cả kĩ thuật mã hóa công khai và phân phối khóa công khai.

Ứng dụng công nghệ Agent trong môi trường lưới

Do tính hỗn tạp của môi trường lưới, có thể áp dụng công nghệ Agent vào lưới. Agent với các đặc điểm như tính tự trị, khả năng cộng tác, khả năng phản ứng, khả năng di động và tính cá nhân tỏ ra rất phù hợp cho việc quản lý tài nguyên và lập lịch trong lưới.

Các nghiên cứu trong đề tài tập trung vào vấn đề tích hợp công nghệ Agent vào các dịch vụ trong môi trường lưới, theo nghĩa xây dựng một dịch vụ lai giữa lưới và Agent. Dịch vụ này có khả năng tồn tại trong cả 2 container: globus container ứng với môi trường lưới và jade container ứng với môi trường agent. Cách tiếp cận được đề xuất đơn giản hơn so với tiếp cận của một số nhóm nghiên cứu cùng mục tiêu trên thế giới như nhóm SoFAR của đại học Southampton (Anh). Trong khuôn khổ đề tài đã xây dựng được dịch vụ lai giữa Agent và lưới, gọi là HAGS (Hybrid Agent Grid Service), sử dụng kỹ thuật thừa kế trong lập trình hướng đối tượng và kỹ thuật “xưởng chế tác” (Factory) - một trong các kỹ thuật cơ bản của lập trình dịch vụ lưới [5].

2.2.3.2. Nghiên cứu xây dựng hệ thống BKGrid 2005

BKGrid 2005 là phiên bản hệ thống lưới thử nghiệm triển khai tại trung tâm Tính toán Hiệu năng cao - Đại học Bách Khoa Hà Nội. Tất cả các thành phần của BKGrid 2005 đều được phát triển trên cơ sở các bộ công cụ mã nguồn mở mạnh

trên thế giới hiện nay. Các công cụ này được ứng dụng rộng rãi tại nhiều trung tâm tính toán hiệu năng cao, nhiều trung tâm nghiên cứu về tính toán lưới trên thế giới, đang tiếp tục được hoàn thiện bởi cộng đồng mã nguồn mở. Các bộ công cụ mã nguồn mở đã làm chủ được và ứng dụng trong triển khai BKGrid 2005 bao gồm:

GlobusToolkit (GT)

GT là bộ công cụ mã nguồn mở hàng đầu hiện nay được dùng để xây dựng môi trường lưới. Nó được phát triển bởi Globus Alliance, một tổ chức lớn bao gồm nhiều viện nghiên cứu và công ty tại Mỹ. GT cho phép người dùng trong lưới chia sẻ năng lực tính toán của các tài nguyên, dữ liệu và các công cụ khác một cách an toàn mà không phải quan tâm đến các hạn chế về khoảng cách địa lý cũng như các chính sách tự trị tại mỗi nút lưới. GT bao gồm các dịch vụ phần mềm, các thư viện phục vụ cho các mục đích như: theo dõi, khám phá và quản lý tài nguyên; quản lý dữ liệu; truyền thông và an toàn bảo mật. Nó được đóng gói thành một tập các thành phần có thể được dùng độc lập hoặc kết hợp với

nhau để phát triển ứng dụng. Mục tiêu của GT là loại bỏ mọi trở ngại do tính không đồng nhất giữa các thành viên lưới có thể ngăn trở khả năng hợp tác, tạo nên tính trong suốt đối với người sử dụng, khiến cho người sử dụng khi ngồi trước máy tính cá nhân của mình, khi tham gia vào lưới có cảm tưởng như đang được sử dụng một máy tính ảo với năng lực khổng lồ.

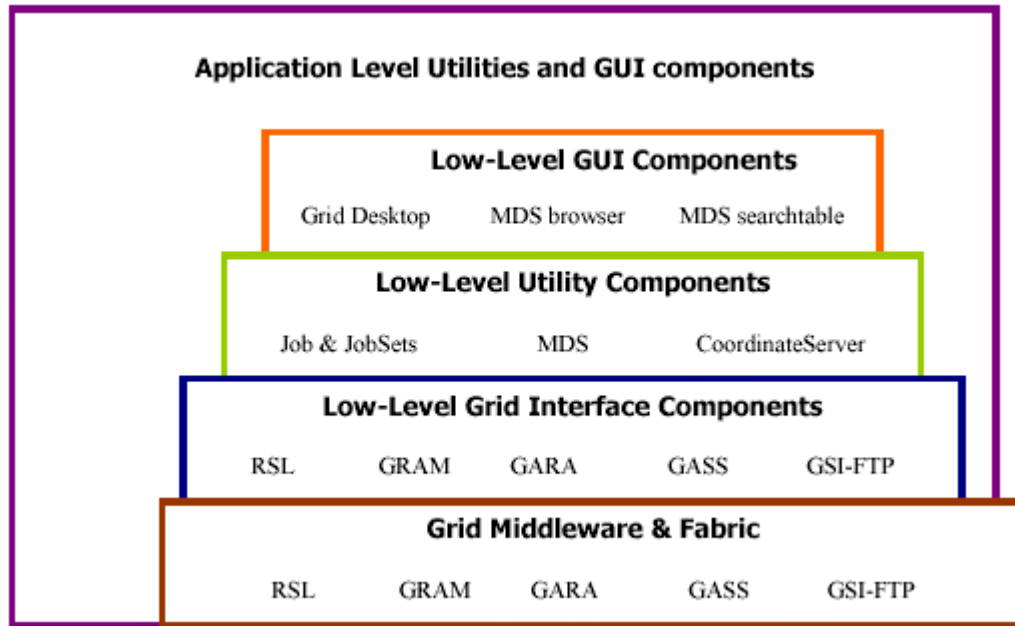
Trong hệ thống BKGrid 2005 sử dụng phiên bản GT3.2.1. Điểm mới tại phiên bản này là các thành phần được dựa trên các chuẩn OGSI và OGSA, với ý tưởng cơ bản: mọi thứ đều là dịch vụ lưới - một bước phát triển mở rộng hơn so với dịch vụ Web. Trong đề tài đã nghiên cứu làm chủ được các quy trình: Cài đặt GT3.2.1, xin giấy chứng nhận, thiết lập các cấu hình người dùng lưới, và đặc biệt triển khai công cụ cài đặt giúp người sử dụng tiện lợi hơn trong việc cài đặt. Hiện tại GT4, phiên bản nâng cao so với GT3.2.1, đang trong quá trình nghiên cứu và thử nghiệm.

Java Cog Kit

Java Cog Kit là bộ cung cụ phát triển lưới, ánh xạ giữa công nghệ Java và Globus Toolkit, cho phép các nhà phát triển xây dựng các ứng dụng lưới bằng ngôn ngữ Java. Java Cog Kit được lựa chọn để phát triển lưới bởi một số lý do sau đây:

- Tối đa hóa sự năng động, khả năng mở rộng và tái sử dụng phần mềm.
- Ngôn ngữ Java là độc lập về hệ điều hành và các nền tảng phát triển, do vậy nó tương thích với môi trường không đồng nhất như lưới. Ngoài ra nó cung cấp nhiều thư viện cho phép phát triển các ứng dụng mạng, các kỹ thuật liên tác JAAS, JINI, CORBA, IIOP, ...
- Sử dụng cơ sở hạ tầng bảo mật lưới GSI từ Java.

Dưới đây là kiến trúc của Java Cog Kit



Hình 2-29. Kiến trúc của Java Cog Kit

Grid Middleware & Fabric: Đây là tầng thấp nhất trong kiến trúc, chứa tất cả các gói để giao tiếp với các dịch vụ và các tài nguyên. Hiện tại Java Cog Kit cung cấp một số dịch vụ cơ bản sau đây:

- An toàn và bảo mật (GSI).
- Đề trình và theo dõi các công việc thực thi từ xa (GRAM).
- Đánh giá chất lượng dịch vụ (GARA).
- Dịch vụ truyền file bảo mật (GSIFTP)
- Dịch vụ tra cứu thông tin (MDS)
- Dịch vụ lưu trữ giấy ủy nhiệm (MyProxy)

Các thành phần giao diện lưới mức thấp (Low-Level Grid Interface Components): ánh xạ tới các dịch vụ lưới hay sử dụng, được xây dựng bằng công nghệ Java Bean và Enterprise Java Bean.

Các thành phần tiện ích mức thấp (Low-Level Utility Components): cung cấp các tiện ích mức thấp cho phép tăng hiệu quả trong việc liên kết các thành phần Globus Toolkit được cài đặt bằng ngôn ngữ C.

Các thành phần giao diện mức thấp (Low-Level GUI Components): cung cấp tập các thành phần giao diện mức thấp có khả năng tái sử dụng như: bộ soạn thảo LDAP, RSL, trình duyệt LDAP, ...

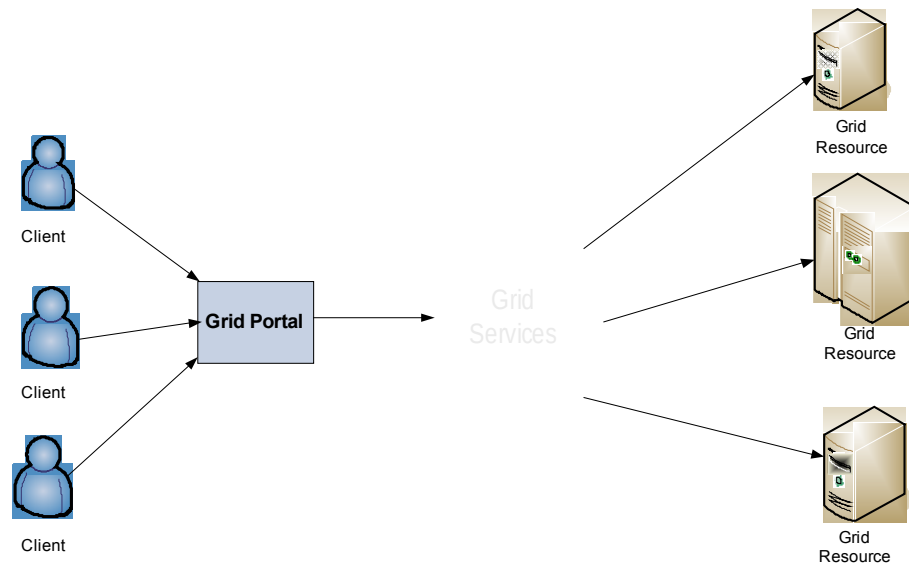
Các thành phần giao diện và tiện ích mức ứng dụng (Application Level Utility & GUI components): là cầu nối giữa Java Cog với các ứng dụng thực tế.

Với kiến trúc trên, Java Cog Kit cho phép

- Các nhà phát triển tạo ra các thành phần trung gian (middleware), liên kết giữa nhà cung cấp dịch vụ và các nhà phát triển ứng dụng phía trên.
- Các nhà phát triển ứng dụng GridPortal dễ dàng truy cập tài nguyên lưới và phát triển dịch vụ lưới như một dịch vụ của Portal.

GridPortal là sự kết hợp giữa công nghệ lưới và Portal, cung cấp một giao diện thống nhất cho các dịch vụ lưới, môi trường bảo mật và khả năng truy nhập các tài nguyên lưới thông qua trình duyệt Web.

Hiện nay trên thế giới phát triển rất nhiều Grid Portal phục vụ cho nhiều mục đích khác nhau, có thể tạm chia ra hai loại Grid Portal chính. Loại thứ nhất là Science Portal, cung cấp chức năng tính toán lưới cho người dùng hoàn toàn thông qua các khái niệm khoa học. Do đó, người dùng hoàn toàn không cần phải quan tâm nhiều đến các giao thức tính toán phân tán phức tạp mà chỉ tập trung vào các vấn đề chuyên môn của mình. Ví dụ các hệ thống tính toán lưới phục vụ trong các lĩnh vực Y tế, Vật lý hạt nhân, nguyên tử... Loại thứ hai là User Portal, cung cấp cho người dùng các tài nguyên lưới (Grid Resource) và các dịch vụ lưới (Grid Services). Người dùng có thể sử dụng để tạo ra các công cụ ứng dụng riêng biệt. Kiểu Portal này yêu cầu người dùng phải có hiểu biết sâu hơn về các khái niệm và nền tảng tính toán lưới.



Hình 2-30. Cổng điện tử Grid Portal

Hệ thống BKGrid 2005 sử dụng phần mềm Gridsphere Portal, được phát triển bởi các tác giả Jason Novotny, Michael Russell và Oliver Wehrens tại học viện

Albert Einstein, Đức. Gridsphere hoạt động như một ứng dụng Web, đòi hỏi môi trường trình chủ như Jakarta Tomcat

Được xây dựng để hỗ trợ phát triển các ứng dụng chạy trên Web, đặc biệt là các ứng dụng lưới, Gridsphere có các đặc điểm sau đây:

- Sử dụng phần mềm trung gian Hibernate, hỗ trợ hầu hết các hệ CSDL chính bao gồm MySQL, Postgres, DB2, HsqlDB, ...
- Hỗ trợ các chuẩn Portlet mới nhất hiện nay: JSR 168 Portlet API, Portlet API IBM's WebSphere 4.2, đồng thời dễ dàng phát triển và tương tác với "third-party portlets".
- Sử dụng các visual beans và thư viện thẻ giao diện người dùng của Gridsphere (UI) để phát triển các portlet phức tạp, đặc tả các xếp đặt hiển thị thông qua XML một cách linh hoạt
- Cơ chế điều khiển truy nhập dựa trên phân quyền (RBAC) với các loại người dùng: `quests`, `users`, `admins` and `super users`.

- Mô hình dịch vụ portlet (portlet service) cung cấp các chức năng có thể tái sử dụng cho nhiều portlet.
- GridSphere core portlets bao gồm các tính năng như: login, logout, người dùng và điều khiển truy nhập

Các công nghệ portal mà Gridsphere Portal sử dụng là portlet và dịch vụ portlet, xin được giới thiệu dưới đây:

Portlet & Grid Porlet

Portlet là một chuẩn mở được đưa ra cho các bộ công cụ để phát triển Web Portal và Grid Portal, với mục đích để đóng gói, triển khai và chuyên nghiệp hóa các thành phần nghiệp vụ web. Portlets đưa ra API cho việc xây dựng các thành phần giao diện ảo trong nội dung trang Web và các nhà cung cấp dịch vụ. Mỗi portlet cung cấp một cửa sổ nhỏ trong một trang portal, và có thể có nhiều portlet trong một trang portal.

Grid Portlet hoàn toàn giống như các Portlet, nhưng thể hiện giao diện cho các dịch vụ lưới thông qua Portal. Chính vì đặc trưng này, khi phát triển các Grid Portlet, đòi hỏi phải có một số cấu hình ở máy chủ Web để hiểu được các đặc tả giao diện dịch vụ theo chuẩn GWSDL khi triệu gọi các dịch vụ lưới từ xa.

Dịch vụ Portlet (Portlet Service)

Dịch vụ portlet là kiến trúc mới do Gridsphere đưa ra, mục đích là đóng gói các thao tác nghiệp vụ logic để có thể tái sử dụng qua nhiều portlet. Hiện tại, các dịch vụ Portlet khởi tạo và hủy bởi PortletServiceFactory, có các phương thức khởi tạo, hủy và quản lý vòng đời trong dịch vụ. Ngoài ra, còn có thêm một số tính năng bổ xung như các dịch vụ hướng người dùng (User services) cung cấp các mức điều khiển truy nhập dựa trên thông tin điều khiển truy nhập của người dùng.

JADE (Java Agent DEvelopment Framework)

JADE là một framework được phát triển hoàn toàn trên ngôn ngữ Java. Nó cho phép triển khai hệ đa agent một cách dễ dàng thông qua một middle-ware tuân theo chuẩn của FIPA, tên ban đầu là Middleware thuộc dự án FACTS do nhóm các nhà khoa học trong bộ phận nghiên cứu công nghệ Obj và Agent thuộc khoa công nghệ thông tin trường ĐH Parma, Italia và tổ chức CSELT phát triển. Hiện nay, Jade là phần mềm miễn phí được cung cấp bởi TILAB (giữ bản quyền). Phiên bản mới nhất là JADE 3.3 (ngày 2/3/2005), trong hệ thống BKGrid 2005, chúng tôi sử dụng phiên bản JADE 3.1.

Như đã đề cập trong phần trước về việc xây dựng dịch vụ lai giữa Agent và lưới và khả năng ứng dụng agent vào việc quản lý tài nguyên trong lưới, với các thư viện và tính mở của JADE, đề tài chọn JADE làm môi trường phát triển Agent và đã tìm ra cơ chế lai giữa 2 môi trường JADE và GT.

2.2.3.3. Sản phẩm phần mềm

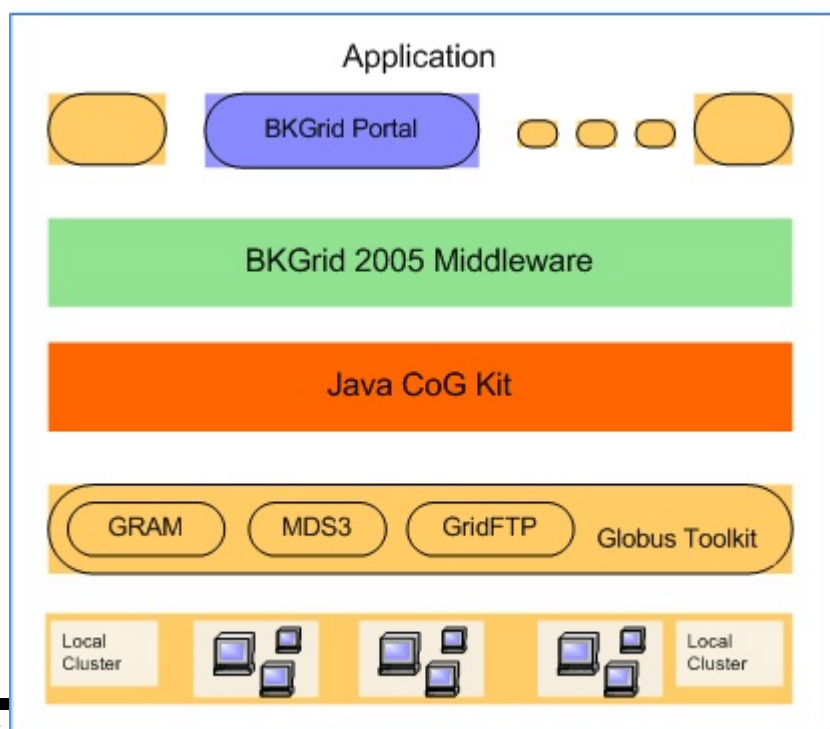
Về mặt sản phẩm, kết quả nghiên cứu được trình bày ở đây là hệ thống BKGrid 2005, hệ thống lưới thử nghiệm. Sản phẩm đã được triển khai thử nghiệm tại Trung tâm Tính toán Hiệu năng cao, Bộ môn Hệ thống Thông tin (khoa Công nghệ Thông tin), Đại học Bách Khoa Hà Nội, được thử nghiệm cài đặt tại một số đơn vị khác. Việc triển khai hệ thống ngày càng trở nên dễ dàng hơn với bộ đĩa cài đã và đang được đóng gói để ngày càng thân thiện hơn với người sử dụng.

Phần này xin giới thiệu chung về hệ thống và kịch bản sử dụng. Chi tiết các thành phần sẽ được giới thiệu trong báo cáo của đề tài nhánh.

Hệ thống BKGrid 2005 được xây dựng dựa trên kiến trúc phân tầng, các thành phần đều hướng dịch vụ, phục vụ những yêu cầu cụ thể, riêng biệt của người dùng. Một trong những điểm mạnh của hệ thống là có tính khả mở, có giao diện tiếng Việt thân thiện người dùng, và cung cấp một số tính năng cơ bản, đặc trưng của tính toán lưới, như một môi trường bảo mật an toàn cho người dùng

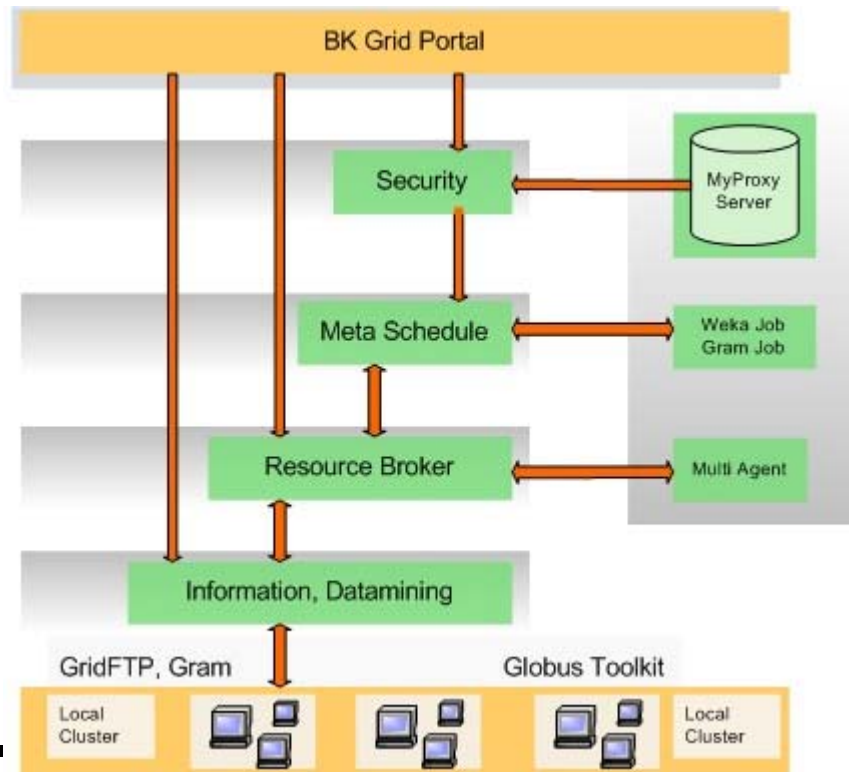
lưới thông qua cổng điện tử portal, khả năng đệ trình các công việc từ xa để tận dụng hiệu năng tính toán của hệ cluster trong lưới, khả năng quản trị tri thức và dữ liệu, cụ thể là giải quyết một lưới các bài toán về khai phá dữ liệu, và đặc biệt là kiến trúc agent đa cấp lai với dịch vụ lưới cho phép mở rộng các tài nguyên lưới trong môi trường phân tán.

BKGrid 2005 được xây dựng như một phần mềm trung gian (middleware), bao gồm một tập các dịch vụ cơ sở ở tầng trên của bộ công cụ phát triển lưới Globus Toolkit 3.2.



Hình 2-31. BKGrid 2005 và nền tảng phát triển

Dưới đây là kiến trúc của BKGrid 2005 được thể hiện theo sự phân bố của các luồng thông tin dữ liệu:



Hình 2-32. Kiến trúc BKGrid 2005 và các luồng thông tin

Dựa trên quan điểm người dùng, các luồng dữ liệu trong hệ thống được phân tầng, thể hiện sự chuyên nghiệp hóa về chức năng của từng thành phần nghiệp vụ. Cụ thể là:

- **Tầng dưới cùng:** là các hệ cluster cục bộ, với hiệu năng tính toán cao, chính là các tài nguyên tính toán của lưới. Ngay phía trên là bộ công cụ phát triển lưới Globus Toolkit.

- **Tầng thông tin và khai phá dữ liệu:** cung cấp các khả năng thông tin và khai phá dữ liệu trong các tài nguyên cục bộ, có thể cung cấp trực tiếp cho người dùng qua portal hoặc một số thông tin cơ bản về dịch vụ và hệ thống cho các tầng phía trên.

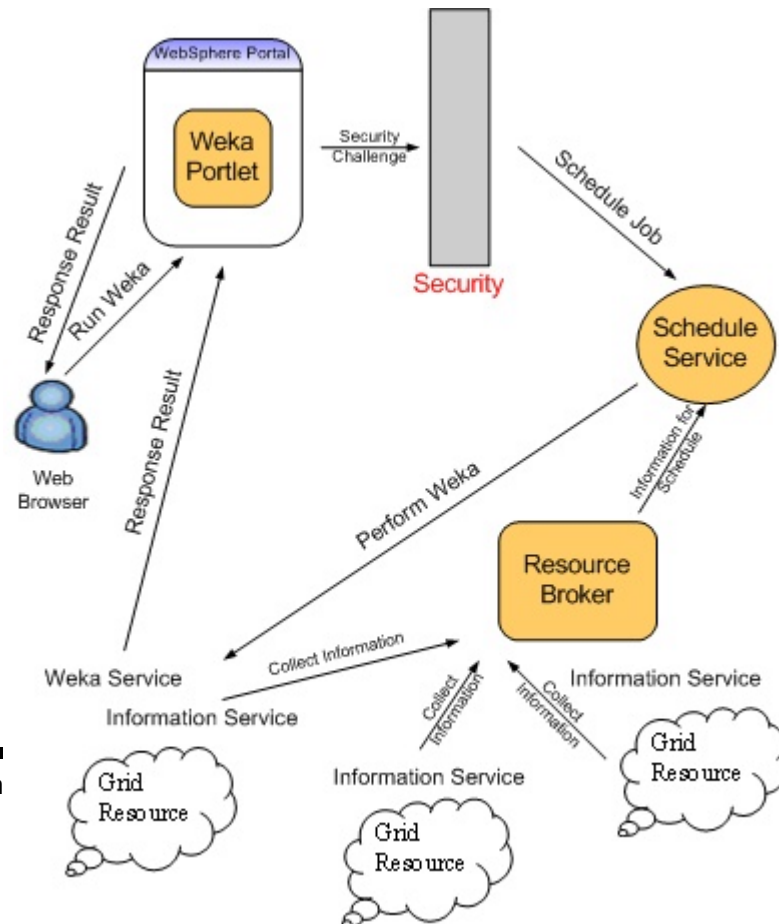
- **Tầng môi giới tài nguyên:** sử dụng công nghệ agent với kiến trúc của các hệ đa agent, thương lượng với các tài nguyên trong lưới để lấy các thông tin về hệ thống, dịch vụ mà phía dưới cung cấp

- **Tầng lập lịch mức cao:** tiếp nhận yêu cầu người dùng, sử dụng các thông tin do tầng môi giới tài nguyên cung cấp, lựa chọn tài nguyên phù hợp để thực hiện công việc. Công việc thực hiện có thể là về yêu cầu khai phá dữ liệu (weka job), hay đệ trình các công việc từ xa (Gram job), tầng lập lịch phải đảm nhận theo dõi công việc, phục hồi chống lỗi trước khi thông báo kết quả cho người dùng.

- **Tầng bảo mật:** trực tiếp hỗ trợ bảo mật cho tầng lập lịch, truy xuất các giấy ủy nhiệm người dùng qua máy chủ MyProxy, kiểm tra tính hợp lệ của người dùng trước khi cho phép người dùng đệ trình công việc xuống dưới tầng lập lịch.

- **BKGrid Portal:** phía trên cùng là cổng dịch vụ lưới BKGrid Portal, cung cấp giao diện đồ họa cho người dùng sử dụng các dịch vụ mà hệ thống cung cấp.

Để hiểu rõ hơn về tương tác giữa các thành phần trong hệ thống, sau đây là một kịch bản hoạt động trong chức năng chạy ứng dụng Weka:



Hình 2-33. Kịch bản tương tác trong chức năng chạy ứng dụng Weka

Đây là một trong những chức năng chính của hệ thống, cung cấp cho người dùng lưới một ứng dụng về DataMining. Chính vì thế mà kịch bản hoạt động của thành phần này là khá phức tạp, liên quan đến hầu hết các thành phần khác trong hệ thống. Trên hình vẽ, hoạt động của chức năng này như sau:

- Khi người dùng có nhu cầu chạy ứng dụng Weka, thông qua một trình duyệt Web, họ sẽ đăng nhập vào portal, cung cấp các tham số cần thiết để thực hiện ứng dụng .
- Portal sẽ kiểm tra tính hợp lệ của người dùng trong lưới, sau đó chuyển yêu cầu đến bộ lập lịch.
- Bộ lập lịch sẽ lấy các thông tin cần thiết lập lịch từ bộ môi giới tài nguyên, thông tin tài nguyên được bộ môi giới tập hợp từ rất nhiều tài nguyên khác nhau, thông qua dịch vụ thông tin.

- Sau khi đã có đủ thông tin cần thiết, bộ lập lịch tiến hành lập lịch, lựa chọn một tài nguyên phù hợp nhất để thực hiện ứng dụng Weka thông qua dịch vụ Weka trong tài nguyên.
- Cuối cùng, dịch vụ Weka trả về kết quả thực hiện cho portal, rồi trả cho người dùng qua trình duyệt Web.

2.2.4. Kết quả nghiên cứu triển khai mô phỏng vật liệu

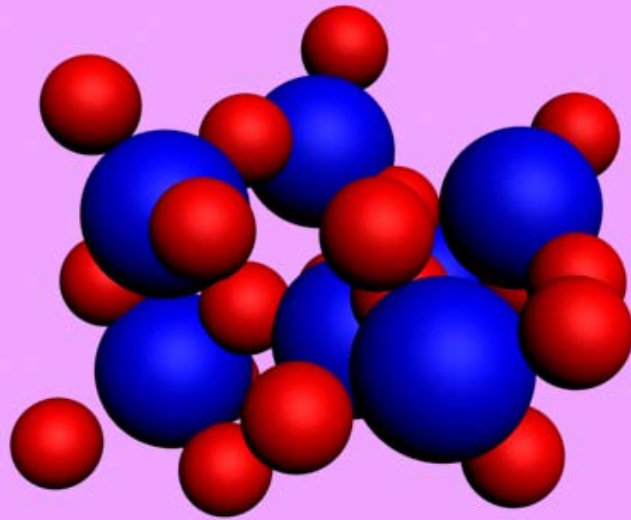
Từ cuối năm 2003, Bộ môn Vật Lý Tin Học, trường Đại Học Bách Khoa Hà Nội đã tiến hành nghiên cứu và song song hóa các chương trình mô phỏng vật liệu nano với hai nội dung sau:

- Xây dựng chương trình mô phỏng song song cho hệ x³ 5 nguyên
- Mô phỏng chấm lượng tử, dây lượng tử và giếng lượng tử

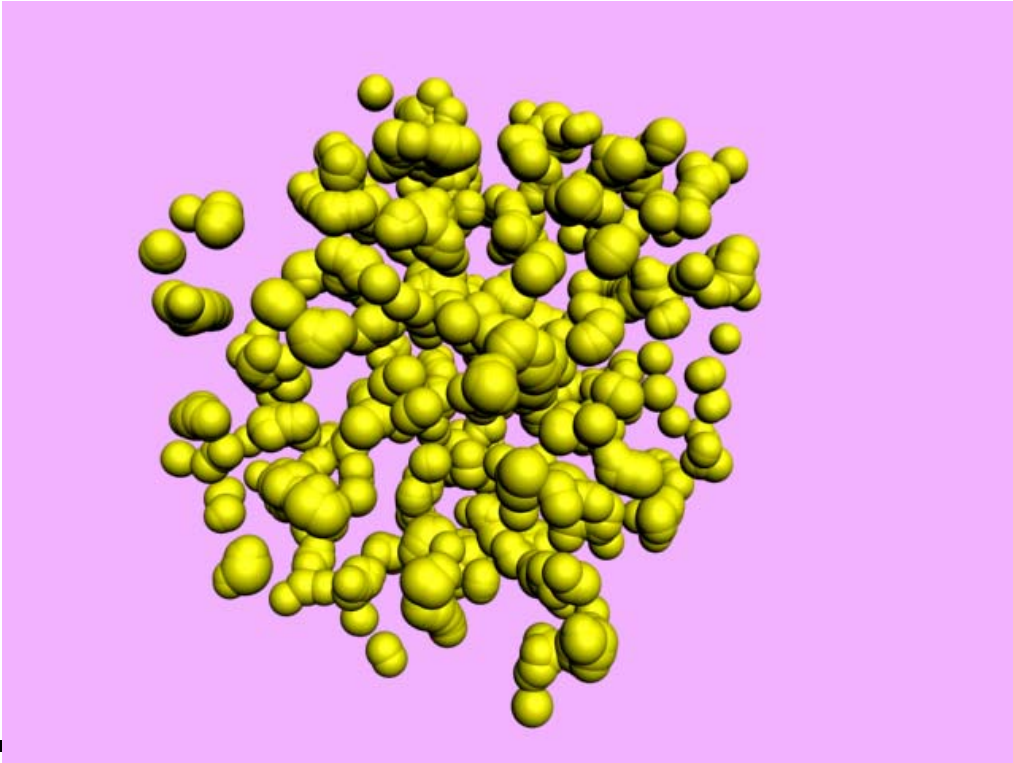
Chương trình động lực học phân tử song song PMD (Parallel Molecular Dynamic) là chương trình động lực học phân tử song song đầu tiên ở Việt

Trung tâm Tính Toán Hiệu Năng Cao - HUT

Nam) được phát triển trong môi trường MPI. Kết quả thực hiện chương trình trên mạng LAN gồm 8 máy cho thấy thời gian tính toán giảm xuống 4 lần so với mô phỏng trên một máy đơn. PMD cho phép xây dựng mô hình Al_2O_3 lỏng gồm 3000 nguyên tử mô phỏng, kết quả cho thấy hàm phân bố xuyên tâm phù hợp tốt với thực nghiệm nhiễu xạ tia X. PMD cũng đã khảo sát cấu trúc, Topology của các cụm AlO_3 , AlO_4 , AlO_5 , Al_2O , Al_3O , Al_4O , liên kết $\text{Al} - \text{O} - \text{Al}$, phân bố các lỗ hổng trong mô hình.



Hình 2-34 Cấu trúc địa phương của mẫu Al_2O_3 lỏng



Hình 2-35 Phân bố lỗ trống trong hệ xỉ Al_2O_3

Nhóm nghiên cứu cũng đã sử dụng phương pháp mô phỏng Semi-empirical Pseudopotential (SEPM) tính toán cấu trúc điện tử của các vật liệu $\text{In}_x\text{Ga}_{1-x}\text{N}$. Những kết quả thu được cho phép khảo sát giếng lượng tử, dây lượng tử và chấm lượng tử đối với các hợp kim thuộc họ InGaN.

Danh sách các luận văn bảo vệ theo hướng mô phỏng:

1	Mô phỏng tính chất khuếch tán và cấu trúc của vật liệu vô định hình.	Luận văn tiến sĩ
2	Mô phỏng động lực học phân tử song song cho vật liệu Quart.	Luận văn thạc sĩ
3	Mô phỏng hệ Silica vô định hình.	Luận văn thạc sĩ
4	Xây dựng chương trình song song mô phỏng	Luận văn tốt nghiệp đại học

	động lực học phân tử, ứng dụng cho mẫu Al_2O_3 lỏng.	
--	--	--

2.3. Tổng kết và đánh giá kết quả thu được

Đề tài “Nghiên cứu các hệ thống tính toán lưới, tính toán hiệu năng cao và ứng dụng mô phỏng vật liệu vi mô” là một trong những đề tài nghiên cứu đầu tiên về tính toán hiệu năng cao và tính toán lưới ở Việt Nam, đóng vai trò quan trọng cho việc xây dựng nền tảng bước đầu đội ngũ chuyên môn và học tập kinh nghiệm của các chuyên gia thuộc Trung tâm Phát triển các Công nghệ tính toán tiên tiến của Ấn độ (C-DAC).

Tập thể cán bộ tham gia đề tài đã thực hiện đầy đủ các nhiệm vụ đặt ra trong thuyết minh đề cương ban đầu cũng như trong hợp đồng. Trong hai năm Đề tài đã tập trung nghiên cứu vào những vấn đề cơ bản nhất trong các nội dung nghiên cứu đã đăng ký là:

- Nghiên cứu, thiết kế kiến trúc, xây dựng Phần mềm Máy chủ tính toán
- Nghiên cứu cấu hình hạ tầng hệ thống và xây dựng Phần mềm nền tính toán lưới
- Nghiên cứu, thiết kế xây dựng Môđun chống lỗi
- Nghiên cứu thiết kế và xây dựng Chương trình mô phỏng vật liệu vi mô

Đề tài đã tạo ra một số sản phẩm phần mềm có tiềm năng ứng dụng, sau khi cài đặt thử nghiệm, đã sẵn sàng chuyển giao cho các cơ quan nghiên cứu, triển khai ứng dụng. Do thời gian thực hiện trong vòng hai năm, việc triển khai các sản phẩm này còn cần thời gian để xem xét và thử nghiệm để đạt độ ổn định cao. Cùng với kết quả đó, đề tài đã hoàn thành 14 bài báo, báo cáo khoa học. Các kết quả này là nền tảng cho việc tiếp tục triển khai các nghiên cứu tiếp theo, nhất là đối với tính toán lưới, một lĩnh vực còn mới mẻ đối với Việt Nam cũng như trên thế giới.

Một trong những kết quả quan trọng của đề tài là đã tạo ra được đội ngũ nghiên

cứu có năng lực, có kinh nghiệm, nhiệt tình, và đi đúng hướng. Đề tài cũng đã góp phần tạo ra sự hợp tác có hiệu với một số tập thể nghiên cứu trong nước và một số trung tâm nghiên cứu ở nước ngoài, đặc biệt Trung tâm Phát triển các Công nghệ tính toán tiên tiến của Ấn độ.

2.4. Kết luận và kiến nghị

2.4.1. Kết luận

Đề tài thực hiện đầy đủ những nội dung và kết quả đã đăng ký, đảm bảo đúng tiến độ.

Một số kết quả dưới hình thức phần mềm sẵn sàng có thể chuyển giao:

- Thiết kế và phát triển Phần mềm Máy chủ tính toán BKlusware
- Cấu hình hạ tầng hệ thống và xây dựng Phần mềm tính toán lưới BKGrid
- Thiết kế và phát triển môđun chống lỗi BKFT

- Chương trình mô phỏng vật liệu vi mô MD

2.4.2. Kiến nghị

Những nghiên cứu và tìm hiểu của đề tài mới chỉ là những kết quả bước đầu, đặt nền móng cho các nghiên cứu tiếp theo. Hiện nay trên thế giới, mặc dầu các hướng nghiên cứu này được đầu tư tập trung và kéo dài trong các dự án khá đồ sộ, nhưng vẫn còn rất nhiều vấn đề mở, có khả năng phát triển tiếp tục.

Sau đây là một số hướng phát triển của đề tài:

- a. Triển khai và xây dựng lưới cho toàn Việt nam trên cơ sở thiết kế một kiến trúc hạ tầng tính toán lưới chuẩn cho Việt nam nhằm tích hợp các Trung tâm tính toán hiệu năng cao trên phạm vi toàn quốc
- b. Đóng gói phần mềm thành sản phẩm hoàn thiện với các chức năng đầy đủ sẵn sàng cho việc cài đặt trong các môi trường hệ thống có cấu hình không thuần nhất.

c. Nghiên cứu và phát triển một số dịch vụ trong hệ thống tính toán lưới:

- Dịch vụ lưới dữ liệu
- Công nghệ tác tử trong việc xây dựng dịch vụ

d. Phát triển các ứng dụng lưới trong các lĩnh vực:

- Sinh tin học
- Dự báo khí tượng thủy văn

Danh sách các bài báo

1. H. Ledang, T.-T Nguyen et al, **Towards an Agent-based grid environment**, *Proceedings of FAIR'2005. HoChiMinh September 23-27, 2005. 10 pages.*
2. T.-M Dao, T.-T Nguyen and H. Ledang, **An economic-based scheduling algorithm for grid – application in developing BKGrid**, *Proceedings of 2nd National Symposium on Information & Communication Technology : research, development and application (ICT.rda 2004), Hanoi September 24-25, 2004. 10 pages.*
3. H. Ledang, T.-T Nguyen et al, **Grid-Weka**, *Presented at National Conference on selective issues in Information Technology, Haiphong August 11-13, 2005. 10 pages.*

4. Minh DT, Thuy NT, Hung LD et al. **BKGrid – hệ thống lưới thử nghiệm.** *Presented at National Conference on selective issues in Information Technology, Danang August 11-13, 2004. 10 pages.*
5. Minh DT, Thuy NT and Hung LD. **Vấn đề lập lịch mang tính kinh tế trong tính toán lưới – áp dụng cho hệ thống BKGrid.** *Presented at the 2nd National Symposium on Information & Communication Technology : research, development and application (ICT.rda 2004), Hanoi September 24-25, 2004. 10 pages.*
6. TuTT, ThuyNT, QuanDM et al. **Cải tiến bộ lập lịch và cơ chế cân bằng tải của gói phần mềm PBS trong hệ thống tính toán song song ghép cụm,** *Presented at the 2nd National Symposium on Information & Communication Technology : research, development and application (ICT.rda 2004), Hanoi September 24-25, 2004. 10 pages.*
7. VinhLV, HungPK, **Study of electronic band structures, effective-mass parameters, and gap bowing of $\text{In}_x\text{Ga}_{1-x}\text{N}$ materials,** *Proceeding of Seventh German-Vietnamese Seminar on Physics and Engineering, Halong City, April 2004*

8. VinhLV, AnhLH, HungPK, **The Study of material parameters of a ternary compound $\text{In}_x\text{Ga}_{1-x}\text{N}$** . *The second international workshop on Nanophysics and Nanotechnology (IWONN'04)*
 9. HueNV, ThuanNT, HungPK, **Nghiên cứu hệ Al_2O_3 bằng phương pháp động lực học phân tử**. Báo cáo tại hội nghị Vật Lý Toàn Quốc lần thứ VI, Hà Nội 23-25/11, 2005
 10. VinhLT, **Mô hình hóa quá trình chuyển pha cấu trúc của oxit Silic vô định hình**, Báo cáo tại hội nghị Vật Lý Toàn Quốc lần thứ VI, Hà Nội 23-25/11, 2005
 11. HungPK, NghiepDM, VinhLT, HongNV, HueHV, *Simulation on study on microstructure of liquid and amorphous Al_2O_3*
 12. HungPK, TrangNT, NhanNT. **Computer simulation of local microstructure and thermodynamic properties in amorphous silica**. *The ninth Asia Pacific Physic Conference, Hanoi, Vietnam – October 25-31, 2004*
 13. HueHV, HungPK, HeNM, HongNV. **Nghiên cứu thể tích tự do và lỗ trống trong kim loại vô định hình Coban**. Báo cáo hội nghị toàn quốc, 5-7/11/2003
-

14. HungPK, VinhLT, HueHV, Simulation Study of Liquid SiO₂ under pressure

Tài liệu tham khảo

- [1] **Ian Foster, Carl Kesselman, Steven Tuecke**, *The Anatomy of the Grid*
- [2] **Ian Foster, Carl Kesselman, Jeffrey M. Nick, Steven Tuecke**, *The Physiology of the Grid*
- [3] **Ian Foster**, *What is the Grid? A Three Point Checklist*
- [4] **IBM Red Books**, *Introduction to Grid Computing with Globus*
- [5] **Borja Sotomayor**, *The Globus Toolkit 3 Programmer's Tutorial*
- [6] **Thomas Sterling**, "An Introduction to PC Clusters for High Performance Computing", California Institute of Technology and NASA Jet Propulsion Laboratory, USA

[7] **Albeaus Bayucan Robert L. Henderson Casimir Lesiak Bhroam Mann Tom Proett Dave Tweten**, “Portable Batch System External Reference Specification”. *MRJ Technology Solutions*

[8] **Albeaus Bayucan Robert L. Henderson Casimir Lesiak Bhroam Mann Tom Proett Dave Tweten**, “Portable Batch System Internal Design Specification”. *MRJ Technology Solutions*

[9] **Albeaus Bayucan Robert L. Henderson Casimir Lesiak Bhroam Mann Tom Proett Dave Tweten**, “Portable Batch System Administrator Guide”. *MRJ Technology Solutions*

[10] **Albeaus Bayucan Robert L. Henderson Casimir Lesiak Bhroam, Mann Tom, Proett Dave Tweten**, “Portable Batch System user guide”. *MRJ Technology Solutions*

[11] **PACS Training Group**, “Introduction to MPI”, *University of Illinois*.

[12] LAM/MPI Forum, “www.lam-mpi.org”

[13] **The LAM/MPI Team**, “LAM/MPI Documents”, *Open Source Lab*

[14] **Nicolas J. Nevin**, “The XMPI API and trace file format”, *January 29, 1997*.

“UC Berkeley Grid” <http://www.millennium.berkeley.edu/ganglia/>