Craig A. Lee (Ed.)

LNCS 2242

Grid Computing – GRID 2001

Second International Workshop
Denver, CO, USA, November 2001
Proceedings

IEEE

acm

GRID 2001

TFCC
IEEE Task Force on Cluster Computing

SC2001

Springer

Craig A. Lee (Ed.)

Grid Computing – GRID 2001

Second International Workshop
Denver, CO, USA, November 12, 2001
Proceedings

Springer

Series Editors

Gerhard Goos, Karlsruhe University, Germany
Juris Hartmanis, Cornell University, NY, USA
Jan van Leeuwen, Utrecht University, The Netherlands

Volume Editor

Craig A. Lee
The Aerospace Corp., High Performance Computing
2350 East El Segundo Blvd., El Segundo, CA 90245, USA
E-mail: lee@aero.org

Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Grid computing : GRID 2001 ; second international workshop, Denver, CO, USA, November 12, 2001 ; proceedings / Craig A. Lee (ed.). - Berlin ; Heidelberg ; New York ; Barcelona ; Hong Kong ; London ; Milan ; Paris ; Tokyo : Springer, 2001
(Lecture notes in computer science ; Vol. 2242)
ISBN 3-540-42949-2

CR Subject Classification (1998): C.2, D.1-4

ISSN 0302-9743
ISBN 3-540-42949-2 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag Berlin Heidelberg New York
a member of BertelsmannSpringer Science+Business Media GmbH

http://www.springer.de

Printed in Germany

Typesetting: Camera-ready by author
Printed on acid-free paper SPIN: 10845876 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 2242
Edited by G. Goos, J. Hartmanis, and J. van Leeuwen

Preface

The term "grid computing" is based on an analogy with the electrical power grid: computing capabilities should be ubiquitous and easy to use. While the development of what we now call grid computing is, in many ways, part of a natural progression of work done in the last decade, what's special about it is that all of its enabling technologies are converging at once: (1) a widely deployed, network infrastructure will connect virtually every device in the world, (2) an interface technology is widely understood and embraced by virtually every segment of science, technology, commerce, and society, and (3) there is a widespread, and growing, understanding of the properties, capabilities, and services that are necessary and *possible* to utilize this infrastructure. Information services and resource brokers will allow the dynamic sharing of resources for applications large and small and enable virtual organizations. These properties, capabilities, and services will be used in different contexts to enable different styles of computing such as Internet computing and Peer-to-Peer computing. To facilitate the adoption of standard practices, the Global Grid Forum (www.gridforum.org) was formed to identify common requirements and push for eventual standardization.

The phenomenal growth of grid computing and related topics has created the need for this workshop as a venue to present the latest research. This year's workshop builds on the success of last year's. Grid 2000, chaired by Rajkumar Buyya and Mark Baker, was held in conjunction with HiPC 2000, in Bangalore, India, and attracted participants from 15 countries. This year's workshop was held in conjunction with Supercomputing 2001, the world's premier meeting for high-performance computing. We sincerely thank Sally Haerer, David Culler, and Ian Foster for making this happen.

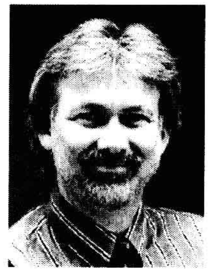
This year's Program Committee represented 12 countries on 4 continents and authors submitted papers from 7 countries on 3 continents. This certainly attests to the wide-spread, international importance of grid computing. We heartily thank all of the authors and the members of the Program Committee. It is the contribution of their valuable time and effort that has made this workshop a success. A very special thanks is extended to Dennis Gannon for his stimulating keynote address. Dennis has a long history of identifying the important issues and clearly elucidating them.

We thank our sponsors, the ACM, the IEEE, the IEEE Computer Society, the IEEE Task Force on Cluster Computing, and also Supercomputing 2001 for making the workshop and these proceedings possible. We are very grateful to Prof. Cauligi Raghavendra and Prof. Viktor Prasanna for allowing Grid 2001 to host its web site at the University of Southern California. Using WIMPE from Dartmouth College for managing the workshop through the web site from wherever I was proved to be invaluable. (Just make sure no other project fills up the disk partition with `/tmp`. ;-) We also wish to thank Jan van Leeuwen of Utrecht University (LNCS Series Editor) and Alfred Hofmann of Springer-Verlag

(Executive Editor) for publishing the proceedings. A special thanks goes to Anna Kramer of Springer-Verlag (Computer Science Editorial Assistant). Her prompt help made prefecting these proceedings as easy as `\addtocounter{one}{2+3}`.

Finally we wish to thank all who attended Grid 2001 in Denver. We now invite you to study these proceedings and their contribution to the further development of grid computing.

August 2001



Craig A. Lee
Grid 2001 Program Chair
www.gridcomputing.org

Grid 2001 Sponsoring Institutions

Association for Computing Machinery (ACM SIGARCH)
http://www.acm.org

Institute of Electrical and Electronics Engineers (IEEE)
http://www.ieee.org

IEEE Computer Society
http://www.computer.org

Supercomputing 2001
http://www.sc2001.org

IEEE Task Force on Cluster Computing (TFCC)
http://www.ieeetfcc.org

Grid 2001 Organization

Workshop Chair

Craig A. Lee
The Aerospace Corporation, El Segundo, California, USA

Program Committee

David Abramson, Monash University, Australia
Ishfaq Ahmad, The Hong Kong University of Science and Technology, China
Giovanni Aloisio, University of Lecce, Italy
Ruth Aydt, University of Illinois, Urbana-Champaign, USA
David Bader, University of New Mexico, USA
Mark Baker, University of Portsmouth, UK
Rajkumar Buyya, Monash University, Australia
Henri Casanova, University of California, San Diego, USA
Steve Chapin, Syracuse University, USA
Frederica Darema, National Science Foundation, USA
Jack Dongarra, University of Tennessee/ORNL, USA
Wolfgang Gentzsch, Sun Microsystems, USA
Jonathan Giddy, Monash University, Australia
Sergi Girona, Polytechnic University of Catalunya, Spain
Ken Hawick, University of Wales, UK
Hai Jin, Huazhong University of Science and Technology, China
William Johnston, Lawrence Berkeley National Laboratory, USA
Domenico Laforenza, Institute of the Italian National Research Council, Italy
Gregor von Laszewski, Argonne National Laboratory, USA
Miron Livny, University of Wisconsin, USA
Satoshi Matsuoka, Tokyo Institute of Technology, Japan
Jarek Nabrzyski, Poznań Supercomputing and Networking Center, Poland
Lalit Patnaik, Indian Institute of Science, India
Thierry Priol, IRISA/INRIA, France
Alexander Reinefeld, ZIB, Germany
Mitsuhisa Sato, Real World Computing Partnership, Japan
Peter Sloot, University of Amsterdam, The Netherlands
Alan Sussman, University of Maryland, USA
Domenico Talia, ISI-CNR, Italy
Yoshio Tanaka, NIAIST, Japan
Mary Thomas, San Diego Supercomputing Center, USA
Brian Tierney, Lawrence Berkeley National Laboratory, USA

Additional Referees

Vishwanath P. Baligar, Indian Institute of Science, India
Christian Pérez, INRIA, France
Alexandre Denis, INRIA, France
Dave DiNucci, Elepar, USA
Jason R. Mastaler, Albuquerque High Performance Computing Center, USA
Florian Schintke, ZIB, Germany
Otto Sievert, University of California, San Diego, USA
James Stepanek, The Aerospace Corporation, USA

Lecture Notes in Computer Science

For information about Vols. 1–2151
please contact your bookseller or Springer-Verlag

Vol. 2152: R.J. Boulton, P.B. Jackson (Eds.), Theore Proving in Higher Order Logics. Proceedings, 2001. X,395 pages. 2001.

Vol. 2153: A.L. Buchsbaum, J. Snoeyink (Eds.), Algorithm Engineering and Experimentation. Proceedings, 2001. VIII, 231 pages. 2001.

Vol. 2154: K.G. Larsen, M. Nielsen (Eds.), CONCUR 2001 – Concurrency Theory. Proceedings, 2001. XI, 583 pages. 2001.

Vol. 2155: H. Bunt, R.-J. Beun (Eds.), Cooperative Multimodal Communication. Proceedings, 1998. VIII, 251 pages. 2001. (Subseries LNAI).

Vol. 2156: M.I. Smirnov, J. Crowcroft, J. Roberts, F.Boavida (Eds.), Quality of Future Internet Services. Proceedings, 2001. XI, 333 pages. 2001.

Vol. 2157: C. Rouveirol, M. Sebag (Eds.), Inductive Logic Programming. Proceedings, 2001. X, 261 pages. 2001. (Subseries LNAI).

Vol. 2158: D. Shepherd, J. Finney, L. Mathy, N. Race (Eds.), Interactive Distributed Multimedia Systems. Proceedings, 2001. XIII, 258 pages. 2001.

Vol. 2159: J. Kelemen, P. Sosík (Eds.), Advances in Artificial Life. Proceedings, 2001. XIX, 724 pages. 2001. (Subseries LNAI).

Vol. 2161: F. Meyer auf der Heide (Ed.), Algorithms – ESA 2001. Proceedings, 2001. XII, 538 pages. 2001.

Vol. 2162: Ç. K. Koç, D. Naccache, C. Paar (Eds.), Cryptographic Hardware and Embedded Systems – CHES 2001. Proceedings, 2001. XIV, 411 pages. 2001.

Vol. 2163: P. Constantopoulos, I.T. Sølvberg (Eds.), Research and Advanced Technology for Digital Libraries. Proceedings, 2001. XII, 462 pages. 2001.

Vol. 2164: S. Pierre, R. Glitho (Eds.), Mobile Agents for Telecommunication Applications. Proceedings, 2001. XI, 292 pages. 2001.

Vol. 2165: L. de Alfaro, S. Gilmore (Eds.), Process Algebra and Probabilistic Methods. Proceedings, 2001. XII, 217 pages. 2001.

Vol. 2166: V. Matoušek, P. Mautner, R. Mouček, K. Taušer (Eds.), Text, Speech and Dialogue. Proceedings, 2001. XIII, 452 pages. 2001. (Subseries LNAI).

Vol. 2167: L. De Raedt, P. Flach (Eds.), Machine Learning: ECML 2001. Proceedings, 2001. XVII, 618 pages. 2001. (Subseries LNAI).

Vol. 2168: L. De Raedt, A. Siebes (Eds.), Principles of Data Mining and Knowledge Discovery. Proceedings, 2001. XVII, 510 pages. 2001. (Subseries LNAI).

Vol. 2169: M. Jaedicke, New Concepts for Parallel Object-Relational Query Processing. XI, 161 pages. 2001.

Vol. 2170: S. Palazzo (Ed.), Evolutionary Trends of the Internet. Proceedings, 2001. XIII, 722 pages. 2001.

Vol. 2171: R. Focardi, R. Gorrieri (Eds.), Foundations of Security Analysis and Design. VII, 397 pages. 2001.

Vol. 2172: C. Batini, F. Giunchiglia, P. Giorgini, M. Mecella (Eds.), Cooperative Information Systems. Proceedings, 2001. XI, 450 pages. 2001.

Vol. 2173: T. Eiter, W. Faber, M. Truszczynski (Eds.), Logic Programming and Nonmonotonic Reasoning. Proceedings, 2001. XI, 444 pages. 2001. (Subseries LNAI).

Vol. 2174: F. Baader, G. Brewka, T. Eiter (Eds.), KI 2001: Advances in Artificial Intelligence. Proceedings, 2001. XIII, 471 pages. 2001. (Subseries LNAI).

Vol. 2175: F. Esposito (Ed.), AI*IA 2001: Advances in Artificial Intelligence. Proceedings, 2001. XII, 396 pages. 2001. (Subseries LNAI).

Vol. 2176: K.-D. Althoff, R.L. Feldmann, W. Müller (Eds.), Advances in Learning Software Organizations. Proceedings, 2001. XI, 241 pages. 2001.

Vol. 2177: G. Butler, S. Jarzabek (Eds.), Generative and Component-Based Software Engineering. Proceedings, 2001. X, 203 pages. 2001.

Vol. 2178: R. Moreno-Díaz, B. Buchberger, J.-L. Freire (Eds.), Computer Aided Systems Theory – EUROCAST 2001. Proceedings, 2001. XI, 670 pages. 2001.

Vol. 2180: J. Welch (Ed.), Distributed Computing. Proceedings, 2001. X, 343 pages. 2001.

Vol. 2181: C. Y. Westort (Ed.), Digital Earth Moving. Proceedings, 2001. XII, 117 pages. 2001.

Vol. 2182: M. Klusch, F. Zambonelli (Eds.), Cooperative Information Agents V. Proceedings, 2001. XII, 288 pages. 2001. (Subseries LNAI).

Vol. 2183: R. Kahle, P. Schroeder-Heister, R. Stärk (Eds.), Proof Theory in Computer Science. Proceedings, 2001. IX, 239 pages. 2001.

Vol. 2184: M. Tucci (Ed.), Multimedia Databases and Image Communication. Proceedings, 2001. X, 225 pages. 2001.

Vol. 2185: M. Gogolla, C. Kobryn (Eds.), «UML» 2001 – The Unified Modeling Language. Proceedings, 2001. XIV, 510 pages. 2001.

Vol. 2186: J. Bosch (Ed.), Generative and Component-Based Software Engineering. Proceedings, 2001. VIII, 177 pages. 2001.

Vol. 2187: U. Voges (Ed.), Computer Safety, Reliability and Security. Proceedings, 2001. XVI, 249 pages. 2001.

Vol. 2188: F. Bomarius, S. Komi-Sirviö (Eds.), Product Focused Software Process Improvement. Proceedings, 2001. XI, 382 pages. 2001.

Vol. 2189: F. Hoffmann, D.J. Hand, N. Adams, D. Fisher, G. Guimaraes (Eds.), Advances in Intelligent Data Analysis. Proceedings, 2001. XII, 384 pages. 2001.

Vol. 2190: A. de Antonio, R. Aylett, D. Ballin (Eds.), Intelligent Virtual Agents. Proceedings, 2001. VIII, 245 pages. 2001. (Subseries LNAI).

Vol. 2191: B. Radig, S. Florczyk (Eds.), Pattern Recognition. Proceedings, 2001. XVI, 452 pages. 2001.

Vol. 2192: A. Yonezawa, S. Matsuoka (Eds.), Metalevel Architectures and Separation of Crosscutting Concerns. Proceedings, 2001. XI, 283 pages. 2001.

Vol. 2193: F. Casati, D. Georgakopoulos, M.-C. Shan (Eds.), Technologies for E-Services. Proceedings, 2001. X, 213 pages. 2001.

Vol. 2194: A.K. Datta, T. Herman (Eds.), Self-Stabilizing Systems. Proceedings, 2001. VII, 229 pages. 2001.

Vol. 2195: H.-Y. Shum, M. Liao, S.-F. Chang (Eds.), Advances in Multimedia Information Processing – PCM 2001. Proceedings, 2001. XX, 1149 pages. 2001.

Vol. 2196: W. Taha (Ed.), Semantics, Applications, and Implementation of Program Generation. Proceedings, 2001. X, 219 pages. 2001.

Vol. 2197: O. Balet, G. Subsol, P. Torguet (Eds.), Virtual Storytelling. Proceedings, 2001. XI, 213 pages. 2001.

Vol. 2198: N. Zhong, Y. Yao, J. Liu, S. Ohsuga (Eds.), Web Intelligence: Research and Development. Proceedings, 2001. XVI, 615 pages. 2001. (Subseries LNAI).

Vol. 2199: J. Crespo, V. Maojo, F. Martin (Eds.), Medical Data Analysis. Proceedings, 2001. X, 311 pages. 2001.

Vol. 2200: G.I. Davida, Y. Frankel (Eds.), Information Security. Proceedings, 2001. XIII, 554 pages. 2001.

Vol. 2201: G.D. Abowd, B. Brumitt, S. Shafer (Eds.), Ubicomp 2001: Ubiquitous Computing. Proceedings, 2001. XIII, 372 pages. 2001.

Vol. 2202: A. Restivo, S. Ronchi Della Rocca, L. Roversi (Eds.), Theoretical Computer Science. Proceedings, 2001. XI, 440 pages. 2001.

Vol. 2204: A. Brandstädt, V.B. Le (Eds.), Graph-Theoretic Concepts in Computer Science. Proceedings, 2001. X, 329 pages. 2001.

Vol. 2205: D.R. Montello (Ed.), Spatial Information Theory. Proceedings, 2001. XIV, 503 pages. 2001.

Vol. 2206: B. Reusch (Ed.), Computational Intelligence. Proceedings, 2001. XVII, 1003 pages. 2001.

Vol. 2207: I.W. Marshall, S. Nettles, N. Wakamiya (Eds.), Active Networks. Proceedings, 2001. IX, 165 pages. 2001.

Vol. 2208: W.J. Niessen, M.A. Viergever (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2001. Proceedings, 2001. XXXV, 1446 pages. 2001.

Vol. 2209: W. Jonker (Ed.), Databases in Telecommunications II. Proceedings, 2001. VII, 179 pages. 2001.

Vol. 2210: Y. Liu, K. Tanaka, M. Iwata, T. Higuchi, M. Yasunaga (Eds.), Evolvable Systems: From Biology to Hardware. Proceedings, 2001. XI, 341 pages. 2001.

Vol. 2211: T.A. Henzinger, C.M. Kirsch (Eds.), Embedded Software. Proceedings, 2001. IX, 504 pages. 2001.

Vol. 2212: W. Lee, L. Mé, A. Wespi (Eds.), Recent Advances in Intrusion Detection. Proceedings, 2001. X, 205 pages. 2001.

Vol. 2213: M.J. van Sinderen, L.J.M. Nieuwenhuis (Eds.), Protocols for Multimedia Systems. Proceedings, 2001. XII, 239 pages. 2001.

Vol. 2214: O. Boldt, H. Jürgensen (Eds.), Automata Implementation. Proceedings, 1999. VIII, 183 pages. 2001.

Vol. 2215: N. Kobayashi, B.C. Pierce (Eds.), Theoretical Aspects of Computer Software. Proceedings, 2001. XV, 561 pages. 2001.

Vol. 2216: E.S. Al-Shaer, G. Pacifici (Eds.), Management of Multimedia on the Internet. Proceedings, 2001. XIV, 373 pages. 2001.

Vol. 2217: T. Gomi (Ed.), Evolutionary Robotics. Proceedings, 2001. XI, 139 pages. 2001.

Vol. 2218: R. Guerraoui (Ed.), Middleware 2001. Proceedings, 2001. XIII, 395 pages. 2001.

Vol. 2220: C. Johnson (Ed.), Interactive Systems. Proceedings, 2001. XII, 219 pages. 2001.

Vol. 2221: D.G. Feitelson, L. Rudolph (Eds.), Job Scheduling Strategies for Parallel Processing. Proceedings, 2001. VII, 207 pages. 2001.

Vol. 2224: H.S. Kunii, S. Jajodia, A. Sølvberg (Eds.), Conceptual Modeling – ER 2001. Proceedings, 2001. XIX, 614 pages. 2001.

Vol. 2225: N. Abe, R. Khardon, T. Zeugmann (Eds.), Algorithmic Learning Theory. Proceedings, 2001. XI, 379 pages. 2001. (Subseries LNAI).

Vol. 2226: K.P. Jantke, A. Shinohara (Eds.), Discovery Science. Proceedings, 2001. XII, 494 pages. 2001. (Subseries LNAI).

Vol. 2227: S. Boztaş, I.E. Shparlinski (Eds.), Applied Algebra, Algebraic Algorithms and Error-Correcting Codes. Proceedings, 2001. XII, 398 pages. 2001.

Vol. 2229: S. Qing, T. Okamoto, J. Zhou (Eds.), Information and Communications Security. Proceedings, 2001. XIV, 504 pages. 2001.

Vol. 2230: T. Katila, I.E. Magnin, P. Clarysse, J. Montagnat, J. Nenonen (Eds.), Functional Imaging and Modeling of the Heart. Proceedings, 2001. XI, 158 pages. 2001.

Vol. 2232: L. Fiege, G. Mühl, U. Wilhelm (Eds.), Electronic Commerce. Proceedings, 2001. X, 233 pages. 2001.

Vol. 2233: J. Crowcroft, M. Hofmann (Eds.), Networked Group Communication. Proceedings, 2001. X, 205 pages. 2001.

Vol. 2234: L. Pacholski, P. Ružička (Eds.), SOFSEM 2001: Theory and Practice of Informatics. Proceedings, 2001. XI, 347 pages. 2001.

Vol. 2237: P. Codognet (Ed.), Logic Programming. Proceedings, 2001. XI, 365 pages. 2001.

Vol. 2239: T. Walsh (Ed.), Principles and Practice of Constraint Programming – CP 2001. Proceedings, 2001. XIV, 788 pages. 2001.

Vol. 2240: G.P. Picco (Ed.), Mobile Agents. Proceedings, 2001. XIII, 277 pages. 2001.

Vol. 2241: M. Jünger, D. Naddef (Eds.), Computational Combinatorial Optimization. IX, 305 pages. 2001.

Vol. 2242: C.A. Lee (Ed.), Grid Computing – GRID 2001. Proceedings, 2001. XII, 185 pages. 2001.

Table of Contents

Grid Architecture and Policies

Performance and Practice

Author Index 185

Grid Application Design Using Software Components and Web Services

Dennis Gannon

Indiana University
Bloomington, Indiana

Abstract. Software Component systems are widely used in the commercial world for designing desktop applications and multi-tier business systems. They have not been widely used in large scale scientific computation. However, as our computing platform has evolved into Grid systems, a distributed component architecture is gaining support as a programming model for building heterogeneous, wide-area application. There are several interesting Grid component architectures that are currently being used. Some are derived from the Corba component model and others are based on EJB or other object systems. The DOE CCTTSS organization has developed a model that supports parallel applications as well as distributed computation called CCA. Components in CCA are defined by public interfaces called "ports" which define the endpoints in the communication channels that link an application's components together.

Over the past year a number of companies have defined another approach to designing distributed application based on a concept called "Web Services." A Web Service is a process that provides a network-accessible interface of "services" that is described by an XML specification called WSDL (Web Services Description Language). The Grid Forum is currently working on several projects which allow Grid services to be defined and accessed as webservices.

In this talk we will describe the simple duality that exists between component-based programming and web services. We will also discuss the difficult problems of integrating web services protocols like SOAP into high performance distributed systems.

Design and Implementation of a CORBA Commodity Grid Kit

Snigdha Verma[1], Manish Parashar[1], Jarek Gawor[2] and Gregor von Laszewski[2]

[1] The Applied Software Systems Laboratory,
Department of Electrical and Computer Engineering,
Rutgers University, 94 Brett Road, Piscataway, NJ 08854-8058, U.S.A
{snigdha,parashar}@caip.rutgers.edu
http://www.caip.rutgers.edu/TASSL/CorbaCoG/
[2]Mathematics and Computer Science Division
Argonne National Laboratory, 9700 S. Cass Ave, Argonne, Il, 60440, U.S.A.
{gawor,gregor}@mcs.anl.gov
http://www.globus.org/cog/

Abstract. This paper reports on an ongoing research project aimed at designing and deploying a CORBA Commodity Grid (CoG) Kit. The overall goal of this project is to explore how commodity distributed computing technologies and state-of-the-art software engineering practices can be used for the development of advanced Grid applications. As part of this activity, we are investigating how CORBA can be integrated with the existing Grid infrastructure. In this paper, we present the design of a CORBA Commodity Grid Kit that provides a software development framework for building a CORBA "Grid domain." We then present our experiences in developing a prototype CORBA CoG Kit that supports the development and deployment of CORBA applications on the Grid by providing them access to the Grid services provided by the Globus toolkit.

1. Introduction

The past decade has seen the emergence of computational Grids aimed at enabling programmers and application developers to aggregate resources[1] scattered around the globe. However, developing applications that can effectively utilize the Grid still remains a difficult task. Although, there exist Grid services that enable application developers to authenticate, access, discover, manage, and schedule remote Grid resources, these services are often incompatible with commodity technologies. As a result, it is difficult to integrate these services into the software engineering processes and technologies that are currently used by application developers. Recently, a number of research groups have started to investigate Commodity Grid Kits (CoG Kits) to address this problem. Developers of CoG Kits have the common goal of developing mappings and interfaces between Grid services and a particular

[1] In this paper we use resources to collectively refer to computers, data stores, services and applications.

commodity technology (such as Java platform [1] [2], Java Server Pages [3], Python [4], and Perl [5]). We believe that CoG Kits will encourage and facilitate the use of the Grid, while at the same time leveraging the benefits of the commodity technology.
Recent years have also seen significant advances in commodity distributed technologies aimed at easing application development in distributed environments. One such technology is the **C**ommon **O**bject **R**equest **B**roker **A**rchitecture (CORBA) [6] defined by the Object Management Group (OMG). CORBA specifies an open, vendor independent and language independent architecture for distributed application development and integration. Furthermore, CORBA defines a standard interoperability protocol (i.e. GIOP and IIOP) that enables different CORBA implementations and applications to interoperate and be portable across vendors. CORBA has emerged as a popular distributed computing standard and meets the necessary requirements to be considered by application developers as part of the Grid infrastructure. It is therefore natural to investigate the development of a CoG Kit that integrates CORBA with the Grid such that CORBA applications can access (and provide) services on the Grid. Such an integration would provide a powerful application development environment for high-end users and create a CORBA "Grid domain".
This paper presents the design and implementation of a CORBA CoG Kit that provides CORBA application with access to Grid services provided by the Globus toolkit [7]. In this paper we first give a brief overview of the Grid and its architecture and introduce the services and protocols that we intend to integrate within the CORBA CoG Kit. We then briefly outline requirements, advantages and disadvantages of CORBA technologies from the point of view of Grid application developers. Next, we present the architecture of the CORBA CoG Kit, and describe the design, implementation, and application of a prototype. Finally, we conclude our paper and identify the directions of ongoing and future activities.

2. The Grid

The term "Grid" has emerged in the last decade to denote an integrated distributed computing infrastructure for advanced science and engineering applications. The Grid concept is based on coordinated resource sharing and problem solving in dynamic multi-institutional virtual organizations [8]. Grid computing not only provides access to a diverse set of remote resources distributed across different organizations, but also facilitates highly flexible sharing relationships among these resources, ranging from client-server to peer-to-peer. An example of a typical client-server relationship is the classical model where a remote client submits jobs to batch queues for resources at a supercomputer center. An example of peer-to-peer relationship is the collaborative online interaction and steering of remote (distributed) high-end applications and advanced instruments [9].

Grids must support different levels of control ranging from fine-grained access control to delegation and from single user to multi user, and different services such as scheduling, co-allocation and accounting. These requirements are not sufficiently addressed by the current commodity technologies, including CORBA. Although sharing of information and communication between resources is allowed, it is not easy to coordinate the use of distributed resources spanning multiple institutions and organizations. The Grid community has developed protocols, services and tools, which address the issues arising from sharing resources in peer communities. This community is also addressing security solutions that support management of credentials and policies when computations span multiple institutions, secure remote access to resources, information query protocols that provide services for obtaining the configuration and status information about the resources. Because of the diversity of the Grid it is difficult to develop an all-encompassing Grid architecture. Recently, a layered Grid architecture representation has been proposed [8] that distinguishes a

- ***fabric layer***, that interfaces to local control including physical and logical resources such as files, or even a distributed file system,
- ***connectivity layer***, that defines core communication and authentication protocols supporting Grid-specific network transactions,
- ***resource layer***, that allows the sharing of a single resource while using a
- ***collective layer*** that allows to view resources as collection,
- and an ***application layer*** that uses the appropriate components of each layer to support applications.

Each of these layers may contain protocols, APIs, and SDKs to support the development of Grid applications. This general layered architecture of the Grid is shown in the left part of Fig. 1.

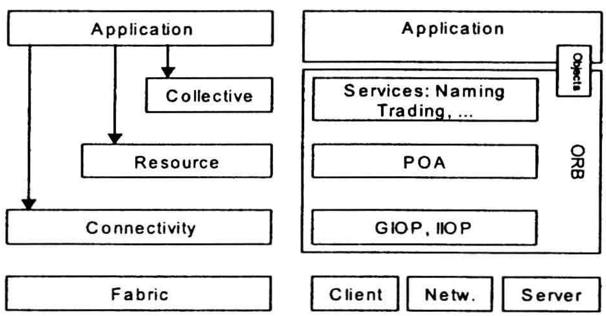


Fig. 1. The Grid Architecture and CORBA (The figure on the left shows the Grid architecture. The figure on the right shows how CORBA fits into the Grid Architecture).

3. CORBA and Grid Computing

CORBA provides advanced capabilities and services for distributed computing and can support the Grid architecture as shown in Fig. 1. Features of CORBA that makes it a suitable candidate for a CoG Kit include its high-level modular programming model, availability of advanced services (e.g. security, naming, trading, event, transaction, etc.) and readymade solutions, interoperability, language independence, location