

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Alfred Kobsa

University of California, Irvine, CA, USA

Friedemann Mattern

ETH Zurich, Switzerland

John C. Mitchell

Stanford University, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz

University of Bern, Switzerland

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

TU Dortmund University, Germany

Madhu Sudan

Microsoft Research, Cambridge, MA, USA

Demetri Terzopoulos

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Gerhard Weikum

Max-Planck Institute of Computer Science, Saarbruecken, Germany

Mumtaz Siddiqui Thomas Fahringer

Grid Resource Management

On-demand Provisioning, Advance Reservation,
and Capacity Planning of Grid Resources



Springer

Authors

Mumtaz Siddiqui

Institute for Computer Science, University of Innsbruck

Technikerstraße 21a, 6020 Innsbruck, Austria

E-mail: Mumtaz.Siddiqui@uibk.ac.at

Thomas Fahringer

Distributed and Parallel Systems Group

Institute of Computer Science, University of Innsbruck

Technikerstraße 21a, 6020 Innsbruck, Austria

E-mail: tf@dps.uibk.ac.at

Library of Congress Control Number: 2009942714

CR Subject Classification (1998): C.2, F.2, D.2, H.3, I.2.11, C.4, I.6, K.6.5

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

ISSN 0302-9743

ISBN-10 3-642-11578-0 Springer Berlin Heidelberg New York

ISBN-13 978-3-642-11578-3 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

springer.com

© Springer-Verlag Berlin Heidelberg 2010

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper SPIN: 12833689 06/3180 5 4 3 2 1 0

dedicated to our families
for their love, encouragement, and steadfastness

Preface

The Grid computing paradigm has evolved tremendously since the beginning of this millennium. It enables a flexible, secure and coordinated resource sharing among dynamic collections of individuals and institutions. The evolution has occurred in various sectors – different Grid infrastructures such as computational Grid, data Grid and knowledge Grid are available to fulfill the diverse requirements of scientific communities. Today various application development and execution environments are present to create new Grid-enabled applications and to execute them onto the Grid. During this timeframe, the major focus of the research community remained on the development of an effective Grid middleware that could lead to a robust operating and management system. The middleware provides services for security, scheduling, information dissemination and resource provisioning.

Despite all these efforts, resource allocation remained a challenge for building an effective Grid resource management, as Grid applications are continuously evolving over time. In the Grid, resource management intends to make high-performance computational resources available on-demand to anyone from anywhere at anytime without undermining the resource autonomy. This is still an art due to non-dedicated heterogeneous resources distributed under multiple trust domains spanning across the Internet under the dynamic Grid environments.

In this monograph, we address the challenges of providing an effective Grid resource management by applying various techniques such as automatic brokerage, dynamic allocation, on-demand resource synthesis, advance reservation and capacity planning. In contrast to the conventional computing environments, applying these techniques in the Grid is all but straightforward. Nevertheless, in the Grid, application of such techniques is of paramount importance for resource management with effective provisioning, better utilization and an optimal allocation.

First of all, we identify the requirement of automatic resource brokerage in the Grid. The development of the Grid with diversified applications competing for scarce resources accentuates the need for an automatic resource

brokerage – this is required to enable the Grid to shield its middleware complexities and to lead toward an invisible, intuitive, and robust runtime environment. Conventional research in the area of Grid resource management mainly focuses on job scheduling with manual or semi-manual resource allocation where resources must be prepared in advance with required applications and system software. This results in non-portable applications hard-coded to a specific Grid environment. Furthermore, it undermines the possible availability of a wide range of logical resources such as software services and tools, thus restricting Grid users to a limited set of physical resources. However, the importance of logical resources is accentuating with the evolution of Grid applications. Some advanced Grid programming environments allow application developers to specify Grid application components at a higher level of abstraction, which then require a dynamic mapping between high-level resource descriptions and actual deployments.

Advance reservation has been largely ignored in the Grid. A Grid application is mostly distributed in nature and runs in a workflow paradigm where dynamic mappings of application components may require advance allocation of resources so that resources could be available as the workflow execution progresses. Advance reservation improves behavioral predictability and on-demand provisioning QoS. A service-level agreement is required to ensure terms and conditions to be agreed upon during negotiation. In the Grid, advance reservation of resources is a challenging task due to the dynamic behavior of the Grid, multi-constrained contending applications, under utilization concerns, and lack of support for agreement enforcement. The under-utilization of resources is a major issue with advance reservation. For a workflow application, resources must be available even under the worst possible conditions for a successful execution. As a result, resource allocation time has to be much longer than average execution time, resulting in computing power wastage.

Nowadays a huge collections of logical resources such as software services and tools are independently and freely available in the Grid, distributed among different physical resources. In order to utilize these logical resources a manual composition is required to build a Grid-enabled application. The manual composition is not only a time-consuming process but it also requires a domain-specific knowledge. Thus, a large collection of logical resources are left unused.

This monograph renders boundaries of Grid resource management, identifies research challenges, proposes new solutions and introduces new techniques with implementations in the form of a Grid resource management system called *GridARM*. GridARM is part of Askalon—a Grid application development and execution environment. The system is designed and developed as a scalable distributed resource manager that delivers resources on-demand, works for resource providers, and optimizes resource utilization with better load distribution and capacity-planning strategies. Optimal load distribution among resources is done according to their proportional share in the Grid.

Novel techniques are introduced for on-demand provisioning, advance reservation, and capacity planning. On-demand provisioning becomes possible with automatic deployment, resource synthesis, and advance reservation. In contrast to existing resource managers, *GridARM* covers logical resources as well by enabling advanced Grid programming environments to specify application components at higher level of abstractions and mapping them to actual deployments dynamically. Henceforth, it simplifies abstract descriptions and separates them from concrete deployments. We further exploit Semantic Web technologies for the Grid to specify explicit definitions and unambiguous machine-interpretable resource descriptions for intelligent resource matching and automatic resource synthesis capabilities.

We introduce a smart negotiation protocol to make optimal resource allocation for a better resource utilization. Advance reservation is supported with a practical solution for agreement enforcement. In order to address under-utilization concerns, we introduce capacity planning with multi-constrained optimized resource allocations. We model resource allocation as an *on-line strip packing problem* and introduce a new mechanism that optimizes resource utilization and other QoS parameters while generating contention-free solutions.

Furthermore, this monograph introduces a new mechanism for automatic synthesis of logical resources by applying ontology rules. The synthesis process generates new compound resources that can be provisioned as new or alternative options for negotiation and advance reservation. This is a major advantage compared to other approaches that only focus on resource matching. The newly generated compound activities provide aggregated capabilities that otherwise may not be possible; this leads toward an automatic generation of complex workflow applications. In addition, we introduce semantics in capacity planning for improving optimization in resource allocation.

The newly introduced techniques and proposed solutions are already integrated in Askalon Grid runtime environment and deployed in the Austrian Grid. The book also demonstrates the effectiveness of the system through well-performed experiments.

June 2009

Mumtaz Siddiqui
Thomas Fahringer

Contents

Part I Overview

1	Introduction	3
1.1	Motivation	5
1.1.1	Collaboration Instead of Isolation	6
1.1.2	Discovery and Selection	6
1.1.3	Lifecycle Management	7
1.1.4	On-Demand Provisioning	7
1.1.5	Role of Planning	8
1.1.6	Service-Level Agreement	8
1.1.7	Optimized Resource Allocation	9
1.1.8	Synthesis and Aggregation	9
1.1.9	Grid Enablement	9
1.1.10	Portability	10
1.1.11	Semantics in the Grid	10
1.2	Research Goals	11
1.2.1	Automatic Resource Brokerage	11
1.2.2	Dynamic Registration and Automatic Deployment	12
1.2.3	Advance Reservation and Co-allocation	12
1.2.4	Capacity Management and Planning	13
1.2.5	Semantics in the Grid	14
1.2.6	Standard Adaptation	14
1.3	Organization	15
1.3.1	Part 1: Overview	15
1.3.2	Part 2: Brokerage	15
1.3.3	Part 3: Planning	16
1.3.4	Part 4: Semantics	16
1.3.5	Part 5: Conclusion	16
1.3.6	Appendices	16

2	Model	17
2.1	The Grid	17
2.1.1	Characteristics	18
2.1.2	Layers	19
2.1.3	Architectures	21
2.2	Resources and Applications	23
2.2.1	Activities	24
2.2.2	Workflows	25
2.2.3	Grid Node	26
2.3	Grid Operating Environment	27
2.3.1	Open Grid Services Architecture	27
2.3.2	WS-Resource Framework	28
2.3.3	Globus Toolkit	29
2.4	Askalon: A Grid Runtime Environment	32
2.4.1	Workflow Composition	33
2.4.2	Resource Management	34
2.4.3	Workflow Scheduling	34
2.4.4	Workflow Enactment	35
2.4.5	Performance Prediction and Analysis	35
2.5	Semantic Grid	35
2.5.1	Ontology	36
2.5.2	Web Ontology Language	36
2.5.3	Ontology Query Language	37
2.6	Resource Management	37
2.6.1	Provisioning	40
2.6.2	Allocation Negotiation	40
2.6.3	Capacity Planning	41
2.6.4	Manageability Models	42
2.7	Summary	44

Part II Brokerage

3	Grid Resource Management and Brokerage System	47
3.1	Introduction	47
3.2	Architectural Overview	50
3.2.1	Resource Management	50
3.2.2	Node Management	51
3.2.3	Activity Management	51
3.2.4	Allocation Management	51
3.3	System Model	51
3.3.1	Resource Discoverer	52
3.3.2	Candidate Set Generator	54
3.3.3	Resource Synthesizer	57
3.3.4	Resource Selector	57

3.3.5	A Steady System with Proportional Distribution	61
3.4	Implementation	61
3.4.1	Customization	62
3.4.2	Superpeer	63
3.4.3	Standard Adaptation	67
3.5	Experiments and Evaluation	67
3.6	Related Work	75
3.7	Summary	78
4	Grid Activity Registration, Deployment and Provisioning Framework	79
4.1	Introduction	79
4.2	On-Demand Provisioning Motivation	81
4.2.1	An Example Using Bare Grid	81
4.2.2	<i>GLARE</i> -Based Solution	84
4.3	System Model	88
4.3.1	Activity Manager	90
4.3.2	Deployment Manager	93
4.3.3	Activity Type Registry	93
4.3.4	Activity Deployment Registry	93
4.4	Implementation	94
4.4.1	Automatic Deployment Using <i>Expect</i>	95
4.4.2	Static and Dynamic Registration	96
4.4.3	On-Demand Provisioning	97
4.4.4	Self-Management and Fault Tolerance	99
4.5	Experiments and Evaluation	100
4.6	Related Work	103
4.7	Summary	105

Part III Planning

5	Allocation Management with Advance Reservation and Service-Level Agreement	109
5.1	Introduction	109
5.2	Model	111
5.2.1	Agreement	111
5.2.2	Agreement Lifecycle	114
5.3	Negotiation	115
5.3.1	Attentive Allocation	116
5.3.2	Progressive Allocation	118
5.3.3	Share-Based Allocation	119
5.4	Implementation	119
5.4.1	Allocator	120
5.4.2	Co-allocator	120

5.4.3	Agreement Enforcement	121
5.4.4	Priority Provision	122
5.4.5	Standards Adaptation	123
5.5	Experiments and Evaluation	123
5.6	Related Work	125
5.7	Summary	126
6	Optimizing Multi-Constrained Allocations with Capacity Planning	127
6.1	Introduction	127
6.2	System Model	129
6.2.1	Allocation Problem	129
6.2.2	Multi-Constrained Optimization	131
6.3	Negotiation Protocol	134
6.3.1	Allocation Offer Generation	135
6.3.2	Co-allocation Offer Generation	141
6.3.3	Contention Elimination	143
6.3.4	Cost Model	143
6.4	Experiments and Evaluation	144
6.5	Related Work	151
6.6	Summary	153
Part IV Semantics		
7	Semantics in the Grid: Towards Ontology-Based Resource Provisioning	157
7.1	Introduction	157
7.2	Describing Resources with Semantics	158
7.2.1	Concept Description	160
7.3	Architectural Extension	163
7.4	Resource Ontologies	164
7.4.1	Physical Resource Ontology	165
7.4.2	Resource Ensembles	166
7.4.3	Logical Resource Ontology	167
7.5	Discovering Resources with Semantics	167
7.6	Subsumption-Based Resource Matching	170
7.7	Evaluation	171
7.7.1	Subsumption: An Example	172
7.8	Related Work	175
7.9	Summary	177

8	Semantics-Based Activity Synthesis: Improving On-Demand Provisioning and Planning	179
8.1	Introduction	179
8.2	Motivation	181
8.3	Synthesis Model	184
8.3.1	Ontology Rules	184
8.3.2	Activity Synthesis Problem	185
8.4	Applying Patterns for Activity Synthesis	186
8.4.1	Sequential Flow Patterns	186
8.4.2	Parallel Flow Patterns	187
8.5	On-Demand Provisioning	189
8.5.1	Built-Ins and Constraints	190
8.5.2	Assumptions and Effects	191
8.6	Improving Capacity Planning	191
8.7	Discussion and Experiments	192
8.8	Related Work	196
8.9	Summary	197

Part V Conclusion

9	Conclusion	201
9.1	Resource Management Model	201
9.2	Towards Automatic Resource Management	202
9.3	Dynamic Registration and Automatic Deployment	203
9.4	Negotiation for Service-Level Agreement (SLA)	203
9.5	Multi-Constrained Optimization and Capacity Planning	204
9.6	Semantics in the Grid	205
9.7	Future Research	206
A	Notations	207
	References	211
	Index	223