
Contents

LIST OF ABBREVIATIONS	xi
INTRODUCTION	xiii
PART 1. MANAGEMENT OF DISTRIBUTED INFRASTRUCTURES	1
CHAPTER 1. DISTRIBUTED INFRASTRUCTURES BEFORE THE RISE OF VIRTUALIZATION	3
1.1. Overview of distributed infrastructures	3
1.1.1. Cluster	3
1.1.2. Data center	4
1.1.3. Grid	4
1.1.4. Volunteer computing platforms	5
1.2. Distributed infrastructure management from the software point of view	6
1.2.1. Secured connection to the infrastructure and identification of users	6
1.2.2. Submission of tasks	7
1.2.3. Scheduling of tasks	8
1.2.4. Deployment of tasks	9
1.2.5. Monitoring the infrastructure	9
1.2.6. Termination of tasks	10

1.3. Frameworks traditionally used to manage distributed infrastructures	10
1.3.1. User-space frameworks	10
1.3.2. Distributed operating systems	11
1.4. Conclusion	12
CHAPTER 2. CONTRIBUTIONS OF VIRTUALIZATION . . .	13
2.1. Introduction to virtualization	13
2.1.1. System and application virtualization	13
2.1.2. Abstractions created by hypervisors	16
2.1.3. Virtualization techniques used by hypervisors	17
2.1.4. Main functionalities provided by hypervisors	19
2.2. Virtualization and management of distributed infrastructures	22
2.2.1. Contributions of virtualization to the management of distributed infrastructures	22
2.2.2. Virtualization and cloud computing	24
2.3. Conclusion	25
CHAPTER 3. VIRTUAL INFRASTRUCTURE MANAGERS USED IN PRODUCTION	27
3.1. Overview of virtual infrastructure managers	27
3.1.1. Generalities	27
3.1.2. Classification	28
3.2. Resource organization	28
3.2.1. Computing resources	28
3.2.2. Storage resources	30
3.3. Scheduling	31
3.3.1. Scheduler architecture	31
3.3.2. Factors triggering scheduling	33
3.3.3. Scheduling policies	35
3.4. Advantages	37
3.4.1. Application programming interfaces and user interfaces	39
3.4.2. Isolation between users	39
3.4.3. Scalability	40
3.4.4. High availability and fault-tolerance	42

3.5. Limits	45
3.5.1. Scheduling	45
3.5.2. Interfaces	46
3.6. Conclusion	46
PART 2. TOWARD A COOPERATIVE AND DECENTRALIZED FRAMEWORK TO MANAGE VIRTUAL INFRASTRUCTURES	49
CHAPTER 4. COMPARATIVE STUDY BETWEEN VIRTUAL INFRASTRUCTURE MANAGERS AND DISTRIBUTED OPERATING SYSTEMS	51
4.1. Comparison in the context of a single node	52
4.1.1. Task lifecycle	52
4.1.2. Scheduling	53
4.1.3. Memory management	56
4.1.4. Summary	58
4.2. Comparison in a distributed context	59
4.2.1. Task lifecycle	59
4.2.2. Scheduling	61
4.2.3. Memory management	62
4.2.4. Summary	64
4.3. Conclusion	64
CHAPTER 5. DYNAMIC SCHEDULING OF VIRTUAL MACHINES	67
5.1. Scheduler architectures	67
5.1.1. Monitoring	68
5.1.2. Decision-making	68
5.2. Limits of a centralized approach	69
5.3. Presentation of a hierarchical approach: Snooze	70
5.3.1. Presentation	70
5.3.2. Discussion	71
5.4. Presentation of multiagent approaches	72
5.4.1. A bio-inspired algorithm for energy optimization in a self-organizing data center	72
5.4.2. Dynamic resource allocation in computing clouds through distributed multiple criteria decision analysis	73

5.4.3. Server consolidation in clouds through gossiping . . .	74
5.4.4. Self-economy in cloud data centers – statistical assignment and migration of virtual machines	75
5.4.5. A distributed and collaborative dynamic load balancer for virtual machine	76
5.4.6. A case for fully decentralized dynamic virtual machine consolidation in clouds	77
5.5. Conclusion	78

PART 3. DVMS, A COOPERATIVE AND DECENTRALIZED FRAMEWORK TO DYNAMICALLY SCHEDULE VIRTUAL MACHINES 83

CHAPTER 6. DVMS: A PROPOSAL TO SCHEDULE VIRTUAL MACHINES IN A COOPERATIVE AND REACTIVE WAY 85

6.1. DVMS fundamentals	86
6.1.1. Working hypotheses	86
6.1.2. Presentation of the event processing procedure . . .	87
6.1.3. Acceleration of the ring traversal	90
6.1.4. Guarantee that a solution will be found if it exists	90
6.2. Implementation	95
6.2.1. Architecture of an agent	96
6.2.2. Leveraging the scheduling algorithms designed for entropy	98
6.3. Conclusion	99

CHAPTER 7. EXPERIMENTAL PROTOCOL AND TESTING ENVIRONMENT 101

7.1. Experimental protocol	101
7.1.1. Choosing a testing platform	101
7.1.2. Defining the experimental parameters	101
7.1.3. Initializing the experiment	102
7.1.4. Injecting a workload	102
7.1.5. Processing results	103
7.2. Testing framework	103

7.2.1. Configuration	103
7.2.2. Components	104
7.3. Grid'5000 test bed	106
7.3.1. Presentation	107
7.3.2. Simulations	107
7.3.3. Real experiments	108
7.4. SimGrid simulation toolkit	109
7.4.1. Presentation	109
7.4.2. Port of DVMS to SimGrid	109
7.4.3. Advantages compared to the simulations on Grid'5000	110
7.4.4. Simulations	110
7.5. Conclusion	111

CHAPTER 8. EXPERIMENTAL RESULTS AND VALIDATION OF DVMS 113

8.1. Simulations on Grid'5000	113
8.1.1. Consolidation	114
8.1.2. Infrastructure repair	119
8.2. Real experiments on Grid'5000	123
8.2.1. Experimental parameters	123
8.2.2. Results	125
8.3. Simulations with SimGrid	128
8.3.1. Experimental parameters	128
8.3.2. Results	129
8.4. Conclusion	130

CHAPTER 9. PERSPECTIVES AROUND DVMS 133

9.1. Completing the evaluations	133
9.1.1. Evaluating the amount of resources consumed by DVMS	133
9.1.2. Using real traces	134
9.1.3. Comparing DVMS with other decentralized approaches	135
9.2. Correcting the limitations	135
9.2.1. Implementing fault-tolerance	135
9.2.2. Improving event management	136

9.2.3. Taking account of links between virtual machines	138
9.3. Extending DVMS	138
9.3.1. Managing virtual machine disk images	139
9.3.2. Managing infrastructures composed of several data centers connected by means of a wide area network	139
9.3.3. Integrating DVMS into a full virtual infrastructure manager	140
9.4. Conclusion	140
CONCLUSION	141
BIBLIOGRAPHY	145
LIST OF TABLES	157
LIST OF FIGURES	159
INDEX	163