

Contents

Contents	xi
1 Introduction	1
1.1 Motivation	1
1.2 Objectives.	5
1.3 Thesis Structure	6
2 Background	7
2.1 Cloud Computing	7
2.2 Containers and Container Orchestrators.	9
2.3 Serverless Computing	11
2.4 Serverless offerings	14
2.5 State of the Art	28
3 Serverless Container-aware Architectures	33
3.1 Generic Architecture.	34
3.2 Framework implementation	36
3.3 Architecture of SCAR.	38

3.4	SCAR usage	41
3.5	On the Lambda function’s ephemeral cache	44
3.6	Study of the AWS Lambda Freeze/Thaw behavior.	47
3.7	Conclusions.	53
4	Event-Driven File-Processing Serverless Programming Model	55
4.1	Highly-scalable HTTP endpoints with API Gateway	58
4.2	S3 file upload/read triggers Lambda Function.	59
4.3	Data management inside the Lambda Function.	61
4.4	Output files trigger new Lambda functions.	64
4.5	Job Processing with AWS Batch.	65
4.6	Cost analysis.	67
4.7	Conclusions.	72
5	Open-source Serverless Computing for Data-Processing Applications	75
5.1	Platform Components.	76
5.2	OSCAR architecture.	80
5.3	Case study: Video Processing Service in On-premises Infrastructure.	86
5.4	Conclusions.	93
6	Use cases	95
6.1	Adding support to programming languages and software in AWS Lambda	96
6.2	Massive image processing service	99
6.3	Video Processing Service in AWS	104
6.4	Plant classification	108
6.5	Multi-cloud workflow for video processing	109
6.6	Air pollution information service	112
6.7	Monetizing Private Algorithm Workflow Executions.	114
6.8	GROMACS in AWS Batch	117
6.9	Scientific diffusion	118

7 Conclusions	121
7.1 Summary and Contributions	121
7.2 Future work	123
Bibliography	125
Index	145
A SCAR client commands	145
B OSCAR Template	153
Acronyms	159