Instituto de
computação

Developing Checkpointing and Recovery Procedures with the Storage Services of Amazon Web Services

Luan Teylo[1], Rafaela C. Brum[1], Luciana Arantes[2], Pierre Sens[2] and Lúcia M. A. Drummond[1]

[1] *Federal Fluminense University - IC/UFF*
[2] *Sorbonne Université - LIP6*

Introduction

Clouds, usually, offer VMs in different markets, with different guarantees in terms of availability and prices

   On-demand VMs:
   - High availability
   - Cannot be interrupted by the provider

   Spot VMs:
   - Offer up to 90% discount compared with on-demand prices
   - Low availability
   - Interrupted by the provider when it needs the resources back

Introduction

As the VMs in the spot market are subject to revocation by the provider, the adoption of fault tolerance techniques is essential to minimize possible job losses

Introduction
Checkpoint/Recovery

Checkpoint/Recovery is a common technique for imbuing a program or system with fault tolerant qualities. It allows tasks to recover after some interruption, failure or task abortion.

Introduction

When using a checkpoint, it is essential to ensure the files availability for the task recovery. They have to be ALWAYS available.
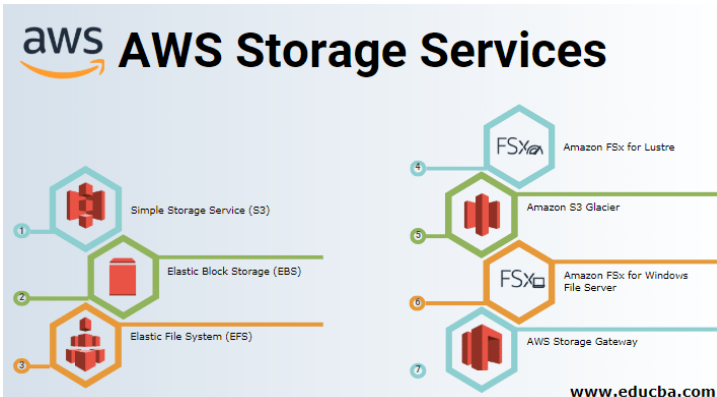
FAILURE-FREE

Non-volatile memory

Introduction

In cloud environments, different storage options can be hired and used along with the VMs. Amazon Web Services (AWS), for example, offers several storage services[a] with different features and purposes.
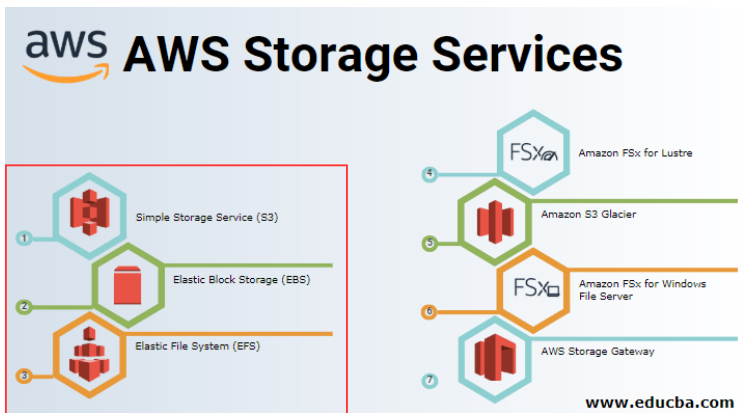
---

[a] https://aws.amazon.com/pt/products/storage/

Introduction

In this work, we are interested in general-purpose storage options that can be used to store and recover checkpoints files during the execution of applications in spot VMs

Introduction
Amazon Simple Storage Service (S3)

▶ Can be utilized to store and recover any amount of data

▶ Provides storage to a wild range of objects sizes, from 0 Bytes to 5TB

▶ The price of each stored GB per month is US\$0.023[a] and for every 1000 requests of PUT, COPY, POST or LIST type it is charged US\$0.005

---

[a]price of standard class in region us-east-1 (April 2020)

Amazon S3

Introduction
Amazon Elastic Block Store (EBS)

▶ Offers local storage volumes with capacity vary from 1GB to 16TB

▶ EBS volumes are persistent and can be kept even without any VM associated with it

▶ Price of US$0.10 per GB per month[a]

_____

[a]price in region us-east-1 (April 2020)

Amazon **EBS**

Introduction
Amazon Elastic File System (EFS)

▶ Provides a simple and scalable file system

▶ Compatible with the Network File System version 4 (NFSv4.0 or NFSv4.1).

▶ Charges $0.30 per GB stored per month[a]

---
[a]price of frequently access class in region us-east-1 (April 2020)

Amazon EFS

In this work, we evaluate checkpoint and recovery procedures by adopting those storage services. Those procedures were included in a previously proposed framework, called HADS (Hibernation Aware Dynamic Scheduling).

The main contributions are the following:

- ▶ Extension of HADS with new checkpoint and recovery procedures;

- ▶ Evaluation of the scalability and impact of the proposed strategies in terms of execution and monetary costs, in different storage services

HADS-CheckRec

The proposed procedures of checkpointing and rollback recovery were included in the framework HADS in a new module called HADS-CheckRec

The module executes the following actions:

► Contract and configure the storage service chosen by the user

► Coordinate the checkpoint records

► Perform the recovery procedure

Experimental Tests

The experimental tests were performed by using VMs of type c3.2xlarge. To emulate the workload we used a synthetic application[a]

Checkpoints are taken by using the Checkpoint Restore In Userspace tool (CRIU)[b]. A widely used checkpointing tool that can record the state of individual applications

---

[a] *Maicon Melo Alves and Lúcia Maria de Assumpção Drummond. A multivariate and quantitative model for predicting cross-application interference in virtual environments. Journal of Systems and Software (2017)*
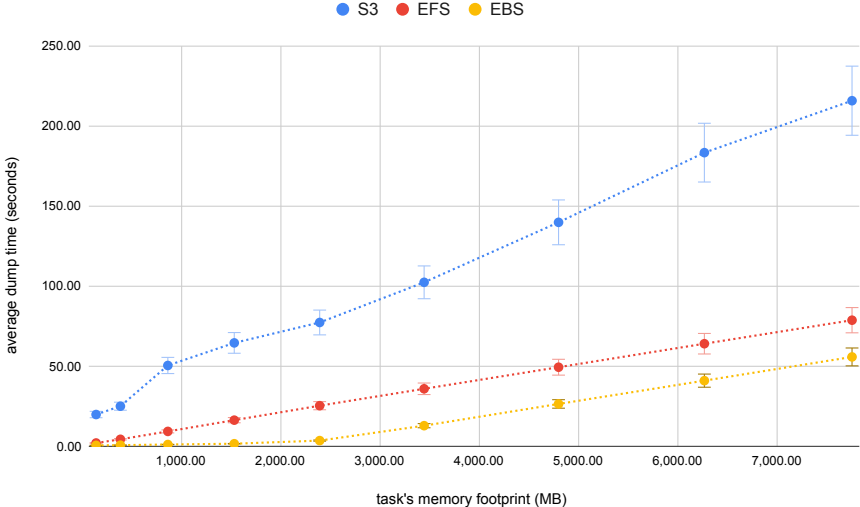
[b] https://criu.org/

Experimental Tests
Dump Time

The dump time is the overhead spent writing out the checkpoint files. To characterize that overhead we create a set of synthetic tasks with memory footprint varying from 140 MB to 7,750 MB (one task by memory size).

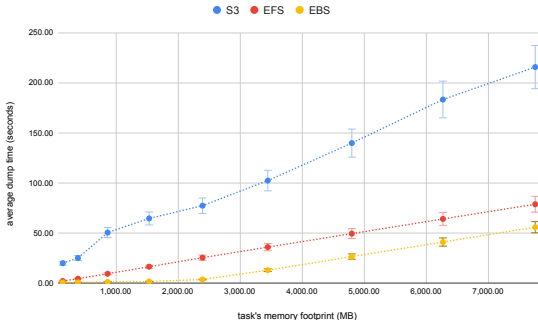# Experimental Tests
Dump Time Without Concurrence

Experimental Tests
Dump Time Without Concurrence

The dump time with S3 presented an increment of **72.57%** and **89.37%** on average when compared to EFS and EBS, respectively

EBS presented the best results, with dump time varying from 0.65 to 55.82 seconds, followed by EFS (2.12 to 78.73 seconds)

Experimental Tests

Dump Time With Concurrence

Task with the biggest memory footprint (7,750 MB) was executed considering scenarios where one, two, four, and six VMs shared the same file system. To avoid concurrency with other resources, we considered only one task per VM
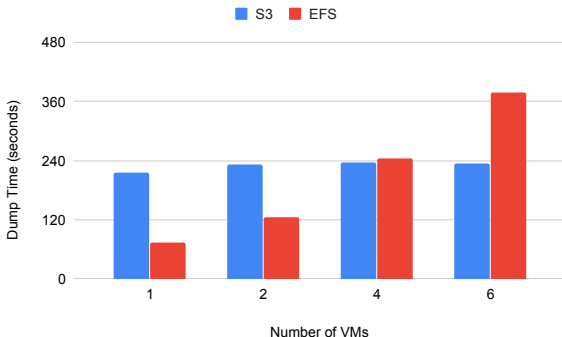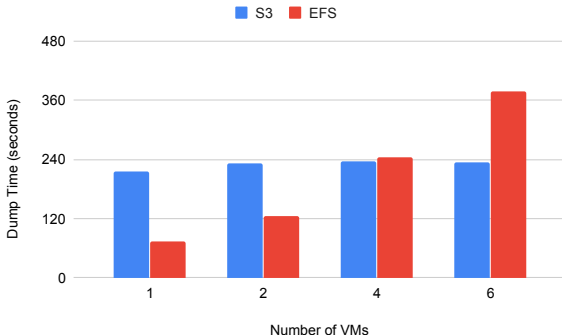
Experimental Tests
Dump Time With Concurrence

The average dump time with S3 was 65.92% greater than EFS with one VM. That difference drops to 46.31% with two VMs. at the four VMs scenario, the time already becomes bigger in EFS then S3 (3.03% of increment)
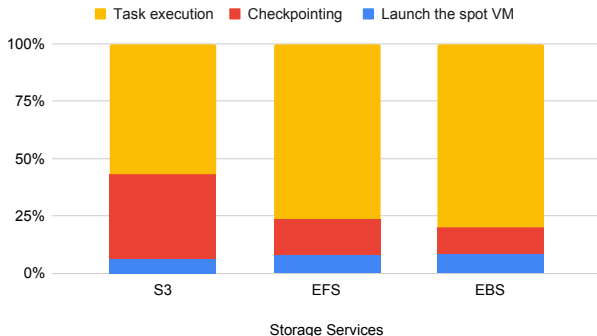
In the six VMs scenario, the dump time with concurrent checkpoint recording increased **37.89%** with EFS in comparison to S3.

Experimental Tests
Overall Overhead Analysis

The bar chart shows the average percentages of time spent by HADS-CheckRec operations in the scenario where there are no spot revocations
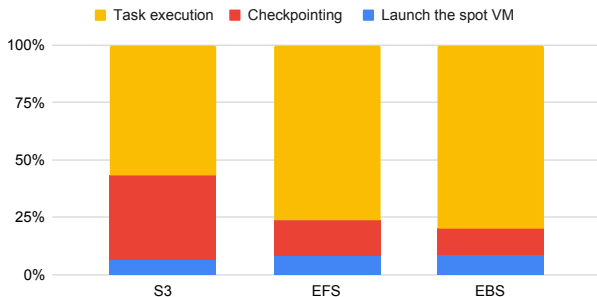
Experimental Tests

Overall Overhead Analysis

In the case of S3 the overhead of launching a spot VM was 6% of the total execution time, while in EFS and EBS it was 8%

The checkpoint time represented 37.1%, 15.4% and 11.4% of the total execution time using the services S3, EFS and EBS, respectively

The useful work accomplished by the VM represents 56.7% of the total execution time using S3, 76.5% in the case of EFS and 79.8% using EBS
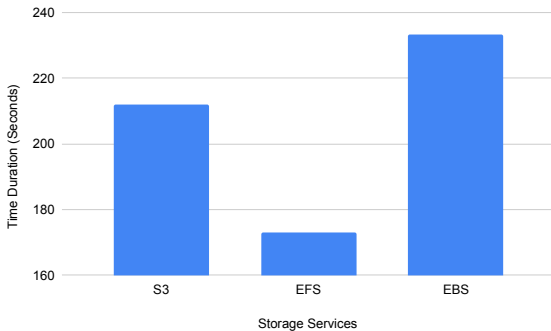
Experimental Tests

Recovery Procedure Evaluation

We consider the execution of a 20 minutes task and a 5 minutes checkpoint interval. The VM revocation was emulated by terminating the VM after 10 minutes of execution. Thus, in this test, only one checkpoint was recorded before the revocation (saving the first 5 minutes of execution)

The time of EBS is **9.14%** higher than S3 and **25.86%** higher than EFS

Experimental Tests
Monetary Cost for Long-Running Tasks

We considered an application with only one task executing for 30 days without any interruption or revocation. In terms of storage, the users are charged at 30 days based price, and we assumed that 30 GBs of data were kept in the storage service, including the checkpoint files, along those days.

Table: Monetary Costs of Services S3, EBS and EFS in a Long-running Application

Checkpoint Interval (h)	# of Checkpoints	Total Execution Time (h)			Total Monetary Cost (US$)		
		S3	EBS	EFS	S3	EBS	EFS
1	720	763.14	731.16	735.75	$23.13	$24.50	$30.64
5	144	728.63	722.23	723.15	$22.11	$24.24	$30.27
10	72	724.31	721.12	721.57	$21.99	$24.20	$30.22
15	48	722.88	720.74	721.05	$21.94	$24.19	$30.20
20	36	722.16	720.56	720.79	$21.92	$24.19	$30.20
25	28	721.68	720.43	720.61	$21.91	$24.18	$30.19

Experimental Tests
Monetary Cost for Long-Running Tasks

While the user pays **US$0.69** for the 30 GBs stored for 30 days in S3, in EBS and EFS those costs are **US$3.0** and **US$9.01**, respectively
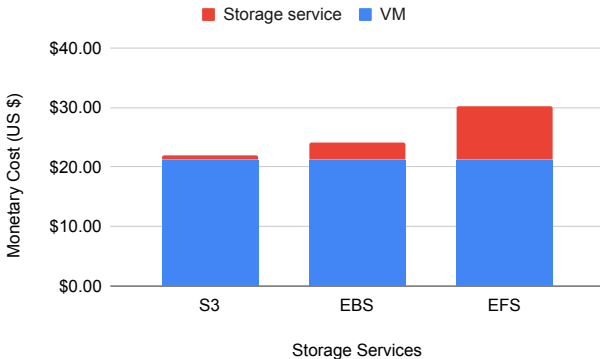
Conclusion

Our results showed that EBS outperformed the other approaches in terms of time spent on recording a checkpoint. But it required more time in the recovery procedure

Conclusion

Our results showed that EBS outperformed the other approaches in terms of time spent on recording a checkpoint. But it required more time in the recovery procedure

EFS presented checkpointing and recovery times close to EBS but with higher monetary costs than the other services.

Conclusion

Our results showed that EBS outperformed the other approaches in terms of time spent on recording a checkpoint. But it required more time in the recovery procedure

EFS presented checkpointing and recovery times close to EBS but with higher monetary costs than the other services.

S3 proved to be the best option in terms of monetary cost but required a longer time for recording a checkpoint, individually. However, when concurrent checkpoints were analyzed, which can occur in a real application with lots of tasks, in our tests, S3 outperformed EFS in terms of execution time also

Next Steps

▶ We intend to evaluate other checkpoint approaches, including the two-step asynchronous recording

▶ The impact of the used checkpoint interval on the monetary cost and execution time

Thank You

email: luanteylo@id.uff.br