

Benchmark PostgreSQL

SaveinCloud x AWS



Fernando Franquini
DragonBD - Partner SaveinCloud

Benchmark

Este projeto tem como objetivo realizar um **benchmark de desempenho** entre ambientes PostgreSQL **hospedados na AWS (RDS + EC2)** e na plataforma **SaveinCloud (SiC - Virtuozzo + VPS)**.

O foco principal da análise será medir e comparar métricas de **desempenho de instruções SQL**, garantindo que as configurações dos ambientes sejam default de cada um para ser um teste justo.

Para a realização dos testes, utilizaremos a ferramenta [Acronis DB Bench](#), executando diferentes workloads para medir **desempenho de leitura, escrita e atualização de dados**.

Os testes serão conduzidos com **diferentes cargas de trabalho** para avaliar a escalabilidade e eficiência de cada ambiente, permitindo a escolha da melhor opção com base em **desempenho e custo-benefício**.

Resultados

Comparação de Desempenho - Operações por Segundo (TPS)							
Operação	SiC-2 (SSD 2000)	SiC-1 (NVMe 2000)	SiC-2 (SSD 5000)	SiC-1 (NVMe 5000)	AWS GP3 (12000)	AWS IO1 (2000)	AWS IO1 (5000)
Insert (1 Worker)	692	1308	754	1609	255	266	270
Insert (16 Workers)	9026	16885	10145	17234	3322	3566	3510
Update (1 Worker)	522	946	607	1021	198	216	208
Update (16 Workers)	6719	10113	7293	11847	2747	2919	2935
Select (1 Worker)	1976	3622	2156	4182	954	964	976
Select (16 Workers)	21739	48733	25530	49879	13983	14198	13911

Nota: Evoluindo a **quantidade de IOPS** e também o **throughput**, notamos que o **ambiente performa melhor**.

Custos

Comparação Geral de Configuração das Instâncias					
Instância	Disco	IOPS	Throughput	Custo mensal (aprox.)	Tempo
SiC-2 (SSD - 2000 IOPS)	SSD/NVMe	2000	400 MBps	R\$1.131,68	700h 12GB - 30h 12GB
SiC-2 (SSD - 5000 IOPS)	SSD/NVMe	5000	600 MBps	R\$899,36	365h 8GB - 365h 16GB
AWS m6i.xlarge (GP3 - 12000 IOPS)	GP3 (400GB)	12.000	156 MBps	R\$2.130,00	Full
AWS m6i.xlarge (IO1 2000)	IO1 (400GB)	2000	156 MBps	R\$3.562,00	Full
AWS m6i.xlarge (IO1 5000)	IO1 (400GB)	5000	156 MBps	R\$5.660,00	Full

Das Execuções

Alguns exemplos da forma de execução:

```
./acronis-db-bench --connection-string  
"postgresql://capin:ru4FDQ6CaZuVo9D091MP@172.31.33.90:5432/postgres?sslmode=require" -t all  
-U 500000 -u 100000 > bench-m6i-xlarge.log
```

```
./acronis-db-bench --connection-string  
"postgresql://postgres:ZGsmPJH6zpZ6BWnjJA7q@172.31.16.207:5432/postgres?sslmode=require" -t  
all -U 500000 -u 100000 > bench-r6i-xlarge.log
```

```
./acronis-db-bench --connection-string  
"postgresql://webadmin:GiiNOVSRCE@10.100.29.6:5432/postgres?sslmode=disable" -t all -U  
500000 -u 100000 > bench-sic-2.log
```

```
./acronis-db-bench --connection-string  
"postgresql://webadmin:NIBnaz71566@10.100.33.51:5432/postgres?sslmode=disable" -t all -U  
500000 -u 100000 > bench-sic-1.log
```

Dos Ambientes

AWS (Amazon Web Services)

Primeira instância foi provisionado um RDS AWS com disco **EBS gp3 (400GB garantindo 12.000 IOPS)**:

- **RDS AWS m6i.xlarge**: 4 vCPUs, 16GB de RAM

No segundo momento, foram criadas outras duas instâncias RDS AWS com disco **EBS io1**, com IOPS provisionado manualmente, tamanho do disco **400GB**:

- **RDS AWS m6i.xlarge**: 4 vCPUs, 16GB de RAM, 2000 IOPS
- **RDS AWS m6i.xlarge**: 4 vCPUs, 16GB de RAM, 5000 IOPS

OBS.: As duas instâncias RDS na AWS possuem um throughput de 156,25 MBps, podendo atingir até 1250 MBps por um período de 30 minutos consecutivos, como limite máximo.

SaveinCloud - SiC (Virtuozzo)

Utilizamos também duas instâncias, que foram criadas na plataforma SiC, utilizando **armazenamento SSD e/ou NVMe** para maior desempenho:

- **SiC-1:** 19 vCPUs, 16GB de RAM, **NVMe**
- **SiC-2:** 19 vCPUs, 16GB de RAM, **SSD**

Dentro dos nossos testes, na SiC, utilizamos o padrão atual de **IOPS e Throughput**, que são **2000 IOPS e 400 MBps**.

Em um **segundo teste**, utilizamos os **mesmos discos** mas uma **nova configuração** que em breve estará disponível de **IOPS e Throughput**, que são **5000 IOPS e 600 MBps**.

Com isso, teremos **duas partes** da nossa análise dos resultados.

Dos Resultados com Configuração Padrão

Nos **primeiros quadros** comparamos cada **ambiente de 16Gb de RAM**, as instâncias **SiC 1 e SiC 2** com a instância **m6i.xlarge** da AWS, depois temos um quadro geral com as três instâncias juntas.

Comparação de Resultados

AWS (m6i.xlarge) x SaveinCloud SSD (sic-2)

Configurações dos Ambientes

Parâmetro	SiC - bench-sic-2	AWS - bench-m6i-xlarge
Banco	PostgreSQL 16.6	PostgreSQL 16.6
CPU	19 vCPUs	4 vCPUs (m6i.xlarge)
Memória	16 GB	16 GB
Disco	SSD (2000 IOPS)	EBS gp3 (400GB, 12000 IOPS)
SO	Linux (Red Hat 11.5)	Linux (Red Hat 7.3)

Testes de Inserção

Teste	SiC - bench-sic-2 (rows/sec)	AWS - bench-m6i-xlarge (rows/sec)
insert-tenant	375.3	162.6
insert-cti	689.3	251.8
insert-light	723.1	255.3
insert-medium	682.3	258.1
insert-heavy	510.0	243.9
insert-json	634.0	251.8

SiC foi consideravelmente superior em todas as inserções, com aproximadamente **2.5x mais throughput** que a AWS.

Testes de Atualização

Teste	SiC - bench-sic-2 (rows/sec)	AWS - bench-m6i-xlarge (rows/sec)
update-medium	510.0	198.4
update-heavy	478.0	195.6
update-heavy-partial	528.4	198.8
update-heavy-sameval	532.8	198.7

SiC foi 2.5x mais rápido nas atualizações. Isso pode estar relacionado ao SSD utilizado na SiC, enquanto a AWS usa **EBS gp3**, que tem latência maior.

Testes de Seleção

Teste	SiC - bench-sic-2 (rows/sec)	AWS - bench-m6i-xlarge (rows/sec)
select-medium-rand	1651	954
select-heavy-rand	1666	946
select-medium-last	1927	975
select-heavy-last	1791	962
select-ts-sql	4975707 values/sec	3494864 values/sec

SiC apresentou desempenho melhor em todos os testes de seleção, sendo cerca de **1.7x mais rápido** que a AWS.

Comparação de Resultados

AWS (m6i-xlarge) x SaveinCloud NVMe (sic-1)

Configurações dos Ambientes

Parâmetro	SiC - bench-sic-1	AWS - bench-m6i-xlarge
Banco	PostgreSQL 16.6	PostgreSQL 16.6
CPU	19 vCPUs	4 vCPUs (m6i.xlarge)
Memória	16 GB	16 GB
Disco	NVMe (2000 IOPS)	EBS gp3 (400GB, 12000 IOPS)
SO	Linux (Red Hat 11.5)	Linux (Red Hat 7.3)

Testes de Inserção

Teste	SiC - bench-sic-1 (rows/sec)	AWS - bench-m6i-xlarge (rows/sec)
insert-tenant	738.2	162.6
insert-cti	1388.0	251.8
insert-light	1451.0	255.3
insert-medium	1411.0	258.1
insert-heavy	949.2	243.9
insert-json	1059.0	251.8

SiC foi significativamente mais rápido, sendo em média ~5 vezes mais eficiente nas inserções.

Testes de Atualização

Teste	SiC - bench-sic-1 (rows/sec)	AWS - bench-m6i-xlarge (rows/sec)
update-medium	1093.0	198.4
update-heavy	974.9	195.6
update-heavy-partial	1058.0	198.8
update-heavy-sameval	1105.0	198.7

SiC foi aproximadamente 5 vezes mais rápido nas atualizações.

Testes de Seleção

Teste	SiC - bench-sic-1 (rows/sec)	AWS - bench-m6i-xlarge (rows/sec)
select-medium-rand	3804.0	954.4
select-heavy-rand	3935.0	946.7
select-medium-last	4284.0	975.1
select-heavy-last	3989.0	962.8
select-ts-sql	576277 values/sec	238627 values/sec

SiC teve um desempenho cerca de 4 vezes maior nas consultas de leitura.

Testes com vários Workers (16 Threads)

Teste	SiC - bench-sic-1 (rows/sec)	AWS - bench-m6i-xlarge (rows/sec)
insert-light	16181.0	3322.0
insert-medium	15653.0	3299.0
insert-json	13856.0	3187.0
update-medium	10968.0	2747.0
update-heavy	11618.0	2702.0
select-medium-rand	44250.0	13983.0
select-heavy-rand	36657.0	13954.0
select-medium-last	48693.0	14407.0
select-heavy-last	47659.0	14076.0

SiC manteve uma vantagem expressiva em todas as operações quando testado com múltiplos workers.

Comparação Geral Entre as Três Instâncias

AWS (m6i.xlarge) x SaveinCloud (SiC-1) x SaveinCloud (SiC-2)

Configuração das Instâncias

Parâmetro	SiC-1	SiC-2	AWS - m6i.xlarge
Banco	PostgreSQL 16.6	PostgreSQL 16.6	PostgreSQL 16.6
CPU	19 vCPUs	19 vCPUs	vCPUs
Memória	16 GB	16 GB	16 GB
Disco	NVMe (2000 IOPS)	SSD (2000 IOPS)	EBS gp3 (400GB, 12000 IOPS)
Sistema Operacional	Linux (Red Hat 11.5)	Linux (Red Hat 11.5)	Linux (Red Hat 7.3)
Custo mensal	R\$1.413,00	R\$1.413,00	~R\$2.130,00 (USD 427,78)

Testes de Inserção (1 worker x 16 workers)

Teste	SiC-1 (rows/sec)	SiC-2 (rows/sec)	AWS - m6i.xlarge (rows/sec)
insert-light (1W)	1489	723.1	255.3
insert-medium (1W)	1123	682.3	258.1
insert-heavy (1W)	921.3	510.0	243.9
insert-json (1W)	1025	634.00	251.8
insert-light (16W)	16089	7553	3322
insert-medium (16W)	15810	7442	3299
insert-heavy (16W)	10145	6849	3187

SiC-1 tem a melhor taxa de inserção, sendo até 5x mais rápido que AWS.
SiC-2 ainda supera AWS, mas tem desempenho inferior ao SiC-1.

Testes de Atualização (1 worker x 16 workers)

Teste	SiC-1 (rows/sec)	SiC-2 (rows/sec)	AWS - m6i.xlarge (rows/sec)
update-medium (1W)	907.1	510.0	198.4
update-heavy (1W)	937.1	478.0	195.6
update-heavy-partial (1W)	1034	528.4	198.8
update-heavy-sameval (1W)	1053	532.8	198.7
update-medium (16W)	9677	5274	2747
update-heavy (16W)	11293	6380	2702

SiC-1 tem o melhor desempenho em atualizações, sendo até 4x mais rápido que AWS.
SiC-2 também é mais rápido que AWS, mas abaixo do SiC-1.

Testes de Seleção (1 worker x 16 workers)

Teste	SiC-1 (rows/sec)	SiC-2 (rows/sec)	AWS - m6i.xlarge (rows/sec)
select-medium-rand (1W)	3886	1651	954
select-heavy-rand (1W)	3184	1666	946
select-medium-last (1W)	4164	1927	975
select-heavy-last (1W)	2752	1791	962
select-medium-rand (16W)	39232	23128	13983
select-heavy-rand (16W)	39028	22743	13954
select-medium-last (16W)	31126	25330	14407
select-heavy-last (16W)	31759	24656	14076

SiC-1 teve o melhor desempenho em leitura, sendo 2.5x mais rápido que AWS.
SiC-2 ainda supera AWS, mas é mais lento que SiC-1.

Testes de Workers (1 worker x 16 workers)

Workers	SiC-1 - TPS	SiC-2 - TPS	AWS - TPS
1 Worker	1053 TPS	510 TPS	198 TPS
16 Workers	11811 TPS	5274 TPS	2746 TPS

SiC-1 escalou melhor com mais workers, enquanto AWS teve ganhos limitados.
AWS parece estar limitada pelo EBS gp3, enquanto **SiC-1 e SiC-2 conseguem aproveitar melhor seus discos SSD**.

Comparação de Custo-Benefício

Parâmetro	SiC-1	SiC-2	AWS - m6i.xlarge
Desempenho geral	★★★★★	★★★★	★★★
Custo mensal	R\$1.413,00	R\$1.413,00	~R\$2.130,00
Melhor para	Workloads intensas	Workloads médias	Cargas leves

SiC-1 é o melhor custo-benefício: desempenho superior por um preço menor.
SiC-2 ainda é uma boa opção, mas com **desempenho inferior ao SiC-1**.
AWS tem o pior custo-benefício, sendo **a mais cara e mais lenta**.

Conclusão Final para 16Gb RAM

- **SiC-1 é a melhor escolha:** Desempenho **superior** em **todos os testes**, melhor escalabilidade e **menor custo**.
- **SiC-2 ainda é melhor que AWS**, mas com **desempenho inferior ao SiC-1**.
- **AWS tem o pior desempenho geral** e é a **opção mais cara**, limitada pela **latência do EBS gp3**.

Novos Ambientes para SiC

AWS (Amazon Web Services)

Duas instâncias foram provisionadas na RDS AWS com discos **IO1 (400GB com IOPS provisionado)**:

- **RDS AWS m6i.xlarge**: 4 vCPUs, 16GB de RAM com **2000 IOPS**
- **RDS AWS m6i.xlarge**: 4 vCPUs, 16GB de RAM com **5000 IOPS**

Saveincloud - SiC (Virtuozzo)

As mesmas duas instâncias que foram alteradas na plataforma SaveinCloud, por padrão eram de **2000 IOPS** e **400Mbps** de **throughput** e, que utilizam **armazenamento SSD e/ou NVMe** porém com **600Mbps de throughput** e com **5000 IOPS**:

- **SiC-1**: ~19 vCPUs, 12Gb de RAM (reservados) até 16GB de RAM, **NVMe**
- **SiC-2**: ~19 vCPUs, 12Gb de RAM (reservados) até 16GB de RAM, **SSD**

Comparação Geral 2000 IOPS:

AWS m6i.xlarge (IO1 2000 IOPS) x SiC-1 (NVMe) x SiC-2 (SSD)

Configuração das Instâncias

Parâmetro	SiC-1 (NVMe)	SiC-2 (SSD)	AWS - m6i.xlarge
Banco de dados	PostgreSQL 16.6	PostgreSQL 16.6	PostgreSQL 16.6
CPU	~19 vCPUs	~19 vCPUs	4 vCPUs
Memória	12 - 16 GB	12 - 16 GB	16 GB
Disco	NVMe	SSD	EBS IO1 - 2000 IOPS
Sistema Operacional	Linux (Red Hat 11.5)	Linux (Red Hat 11.5)	Linux (Red Hat 7.3)
Custo mensal	R\$1.413,00	R\$1.413,00	~R\$ 3.562,00 (USD 715.38)
IOPS	2000	2000	2000

Testes de Inserção (1 worker x 16 workers)

Teste	SiC-1 (NVMe)	SiC-2 (SSD)	AWS - m6i.xlarge (IO1 - 2000 IOPS)
insert-light (1W)	1308	692	266
insert-medium (1W)	1283	667	265
insert-heavy (1W)	978	538	252
insert-json (1W)	1118	627	260
insert-light (16W)	12.618	8.556	3.566
insert-medium (16W)	14.218	8.455	3.519
insert-heavy (16W)	13.995	6.893	3.432

SiC-1 teve desempenho até 5x superior ao AWS IO1 (2000 IOPS).
SiC-2 teve melhor desempenho que AWS, mas inferior ao SiC-1.

Testes de Atualização (1 worker x 16 workers)

Teste	SiC-1 (NVMe)	SiC-2 (SSD)	AWS - m6i.xlarge (IO1 - 2000 IOPS)
update-medium (1W)	946	522	217
update-heavy (1W)	889	467	217
update-heavy-sameval (1W)	745	511	224
update-medium (16W)	10.113	7.009	2.919
update-heavy (16W)	11.345	6.895	2.825

SiC-1 foi 3.5x superior ao AWS IO1 em atualizações sob carga alta.
SiC-2 teve desempenho intermediário, mas ainda superior ao AWS.

Testes de Seleção (1 worker vs 16 workers)

Teste	SiC-1 (NVMe)	SiC-2 (SSD)	AWS - m6i.xlarge (IO1 - 2000 IOPS)
select-medium-rand (1W)	3.194	1.822	965
select-heavy-rand (1W)	2.255	1.798	953
select-medium-last (1W)	3.488	1.976	979
select-heavy-last (1W)	3.113	1.431	964
select-medium-rand (16W)	36.498	22.770	14.198
select-heavy-rand (16W)	36.335	23.104	13.927
select-medium-last (16W)	48.733	21.739	14.405
select-heavy-last (16W)	48.586	24.624	14.187

SiC-1 foi **até 3.5x mais rápido** em seleções sob carga alta comparado à AWS IO1.
SiC-2 teve **desempenho sólido**, ainda **melhor que AWS**, mas inferior ao NVMe.

Testes de Workers

Workers	SiC-1 - TPS	SiC-2 - TPS	AWS - TPS
1 Worker	946 TPS	522 TPS	217 TPS
16 Workers	10.113 TPS	7.009 TPS	2.919 TPS

SiC-1 escalou muito melhor com mais workers, enquanto **AWS** teve crescimento limitado.
O EBS IO1 de 2000 IOPS não conseguiu entregar desempenho competitivo.

Comparação de Custo-Benefício

Parâmetro	SiC-1 (NVMe)	SiC-2 (SSD)	AWS - m6i.xlarge (IO1 - 2000 IOPS)
Desempenho Geral	★★★★★	★★★★★	★★★★
Custo Mensal	R\$ 1.413,00	R\$ 1.413,00	~R\$ 3.562,00 (USD 715.38)
Melhor para	Workloads intensas	Workloads médias	Cargas leves

SiC-1 é a melhor opção, com menor custo e maior desempenho.
SiC-2 ainda supera AWS, mas com desempenho inferior ao NVMe.
AWS tem o pior custo-benefício, sendo mais cara e menos eficiente.

Conclusão Final

- **SiC-1 (NVMe)** é a **melhor escolha**, apresentando **desempenho até 5x superior à AWS IO1**, com **custo 60% menor**.
- **SiC-2 (SSD)** ainda é **melhor que AWS**, mas inferior ao NVMe.
- **AWS m6i.xlarge com IO1 2000 IOPS** teve **desempenho fraco**, não justificando o alto custo.

Recomendação Final

Se busca o **melhor desempenho e custo-benefício** → Escolha **SiC-1 (NVMe)**.

AWS m6i.xlarge IO1 2000 IOPS tem o **pior custo-benefício** e **deve ser evitada**.

Comparação Geral 5000 IOPS:

AWS m6i.xlarge (IO1 5000 IOPS) x SiC-1 (NVMe) x SiC-2 (SSD)

Configuração das Instâncias

Parâmetro	SiC-1 (NVMe)	SiC-2 (SSD)	AWS - m6i.xlarge (IO1 - 5000 IOPS)
Banco de Dados	PostgreSQL 16.6	PostgreSQL 16.6	PostgreSQL 16.6
CPU	~19 vCPUs	~19 vCPUs	4 vCPUs
Memória	12 - 16 GB	12 - 16 GB	16 GB
Disco	NVMe	SSD	EBS IO1 - 5000 IOPS
Sistema Operacional	Linux (Red Hat 11.5)	Linux (Red Hat 11.5)	Linux (Red Hat 7.3)
Custo Mensal	R\$ 1.413,00	R\$ 1.413,00	~R\$ 5.660,00 (USD 1135.38)
IOPS	5000	5000	5000

Testes de Inserção (1 worker vs 16 workers)

Teste	SiC-1 (NVMe)	SiC-2 (SSD)	AWS - m6i.xlarge (IO1 - 5000 IOPS)
insert-light (1W)	1308	692	270
insert-medium (1W)	1283	667	269
insert-heavy (1W)	978	538	255
insert-json (1W)	1118	627	262
insert-light (16W)	16.885	9.026	3.530
insert-medium (16W)	13.185	8.630	3.473
insert-heavy (16W)	13.551	7.808	3.407

SiC-1 teve **desempenho até 5x superior** ao AWS IO1 (5000 IOPS).
SiC-2 teve **melhor desempenho que AWS**, mas **inferior ao SiC-1**.

Testes de Atualização (1 worker vs 16 workers)

Teste	SiC-1 (NVMe)	SiC-2 (SSD)	AWS - m6i.xlarge (IO1 - 5000 IOPS)
update-medium (1W)	914	522	206
update-heavy (1W)	870	467	202
update-heavy-sameval (1W)	984	511	209
update-medium (16W)	13.046	6.719	2.908
update-heavy (16W)	10.913	6.186	2.881

SiC-1 foi **3.5x superior ao AWS IO1** em atualizações sob carga alta.

SiC-2 teve **desempenho intermediário**, mas **ainda superior ao AWS**.

Testes de Seleção (1 worker vs 16 workers)

Teste	SiC-1 (NVMe)	SiC-2 (SSD)	AWS - m6i.xlarge (IO1 - 5000 IOPS)
select-medium-rand (1W)	3.422	1.904	976
select-heavy-rand (1W)	3.343	1.778	968
select-medium-last (1W)	3.904	1.975	992
select-heavy-last (1W)	3.612	1.850	983
select-medium-rand (16W)	45.172	24.102	13.938
select-heavy-rand (16W)	47.263	22.955	14.035
select-medium-last (16W)	49.814	25.385	14.365
select-heavy-last (16W)	48.998	24.426	13.814

SiC-1 foi **até 3.5x mais rápido** em seleções **sob carga alta** comparado à AWS IO1.

Testes de Workers

Workers	SiC-1 - TPS	SiC-2 - TPS	AWS - TPS
1 Worker	914 TPS	522 TPS	206 TPS
16 Workers	13.046 TPS	6.719 TPS	2.908 TPS

SiC-1 escalou muito melhor com mais workers, enquanto AWS teve crescimento limitado. O EBS IO1 de 5000 IOPS não conseguiu entregar desempenho competitivo.

Comparação de Custo-Benefício

Parâmetro	SiC-1 (NVMe)	SiC-2 (SSD)	AWS - m6i.xlarge (IO1 - 5000 IOPS)
Desempenho Geral	★★★★★	★★★★★	★★★
Custo Mensal	R\$ 1.413,00	R\$ 1.413,00	~R\$ 5.660,00 (USD 1135.38)
Melhor para	Workloads intensas	Workloads médias	Cargas leves

SiC-1 é a melhor opção, com menor custo e maior desempenho.
SiC-2 ainda supera AWS, mas com desempenho inferior ao NVMe.
AWS tem o pior custo-benefício, sendo mais cara e menos eficiente.

Conclusão Final

- SiC-1 (NVMe) é a melhor escolha, apresentando desempenho até 5x superior à AWS IO1, com custo 75% menor.
- SiC-2 (SSD) ainda é melhor que AWS, mas inferior ao NVMe.
- AWS m6i.xlarge com IO1 5000 IOPS teve desempenho fraco, não justificando o alto custo.

Recomendação Final

Se busca o melhor desempenho e custo-benefício → Escolha SiC-1 (NVMe).
Se quiser um equilíbrio entre custo e desempenho → SiC-2 (SSD) ainda supera AWS.
AWS m6i.xlarge IO1 5000 IOPS tem o pior custo-benefício e deve ser evitada.