PostgreSQL Cloud Performance

Oskari Saarenmaa

pgconf.eu 2017, Warsaw

Speaker

- CEO, Co-founder @ Aiven, a cloud DBaaS company
- Previously: database consultant, software architect
- PostgreSQL user since 1999 (rel 6.4)
  - Contributed some bug fixes and minor features to core
  - Worked on extensions and tooling in the PG ecosystem

@OskariSaarenmaa

Agenda

Aiven

Why cloud – why not cloud?

Operating in clouds, self-managed or database as a service

Operation modes, performance, data durability

Storage systems, network, provisioned iops, volume types

Methodology

The results

Aiven

Independent Database as a service provider in all major clouds

Launched in the Vienna pgconf.eu (2015) with a managed PostgreSQL service in AWS & GCP

7 DB products now available in 63 regions

PostgreSQL 10 available!

amazon web services™

Google Cloud Platform

Microsoft Azure

DigitalOcean

UpCloud

Why cloud?

Because it is “someone else’s computer”, so someone else:

- Buys the hardware and covers capital costs
- Installs new and replaces broken hardware
- Resources available on-demand, no need to wait for procurement

When using DBaaS “someone else” also:

- Installs and maintains the software, takes care of backups
- Integrated monitoring and metrics
- Backups, replication and other tooling running in minutes

Why not cloud?

Because it is “someone else’s computer”, so you have:

- Less control over details
- Operational concerns
  - Will there be someone to fix issues in case of problems?
- Compliance concerns
  - Someone else has physical access to the data
- Potentially much higher operational costs
  - Especially when only looking at infrastructure costs
  - Assuming you can plan your hardware use well in advance

Roll your own

Maintain your own databases:

- + Lift & shift an existing production on-prem DB to cloud
- + Superuser access
- + All custom extensions

- - Manage backups, plan for scaling
- - Slower provisioning
- - No built-in monitoring

Database as a Service

Use a DB as a service provider:

- + Automatic provisioning and maintenance of systems
- + New clusters available in minutes
- + Integrated monitoring systems
- + Point-in-time recovery built in

- - Limited PL/language support
- - No superuser access (usually)

Performance considerations

Hardware: CPU, storage IO, network

Software: tuning for my workload?

Network: plan to access the database from the same network, typically fast access to data from the same region and availability zone – some differences in the top end

CPU: much the same across all clouds

Storage: not the same across clouds

Data access (latency)

CPU caches < RAM < Local disk < Network disk (usually)

Local disks ("instance storage") in the cloud only available for the lifetime of a single VM instance – data durability must be guaranteed across node faults using other means:

- Replication
- Incremental backup of data as it's written

Turns out we can do both reliably with PostgreSQL

Network disks

+ Persistent past node lifetime
+ Almost infinitely scalable

- Really slow, or
- Quite expensive (PrIOPS)
- Compete with others over limited IO bandwidth
- Not free of faults

Local disks

+ Fast
+ Potentially *really* fast
+ Cheap

- Available in limited sizes
  - (or not at all)
- Ephemeral
  - Node shuts down: data is gone

What to measure

Number of different things affect performance

None of the comparisons match your production workload

Use benchmarks to measure changes in the execution environment over time

Benchmarks

5 Infrastructure clouds

2 Database sizes

PostgreSQL 10

PGBench

Methodology

1. Provision a benchmark host in the target cloud
2. Provision a DB instance from a DBaaS provider
    a. 16 GB RAM instances
    b. 64 GB RAM instances
3. Initialize with a large dataset
    a. Roughly 2x memory size
4. Run PGBench with a varying number of clients for 1 hour

16 GB RAM instances, with network disks

AWS	GCP	Azure	DigitalOcean	UpCloud
m4.xlarge	n1-standard-4	Standard D3v2	16GB	4CPUx16GB
4 vCPU	4 vCPU	4 vCPU	4 vCPU	4 vCPU
16 GB	15 GB	14 GB	16 GB	16 GB
350 GB EBS	350 GB PD-SSD	350 GB P20	350 GB block storage	350 GB MAXIOPS

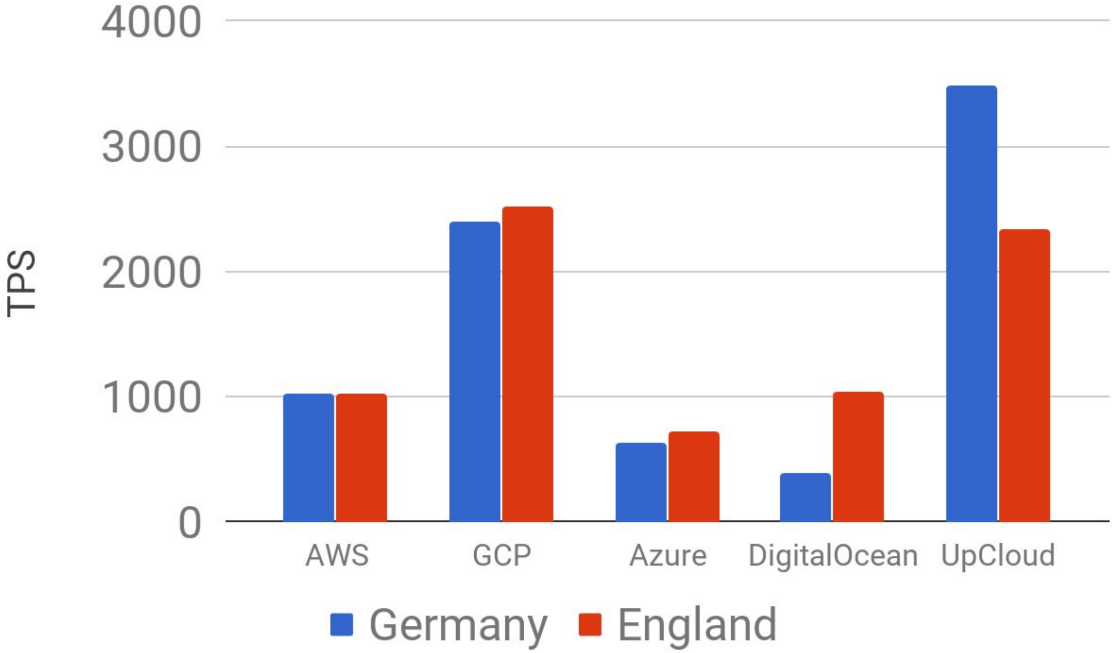
PostgreSQL 10.0
Linux 4.13.5

Data encrypted on disk
SSL required for clients
WAL archiving enabled

```
work_mem = 12MB
shared_buffers = 3GB
max_wal_size = 16GB
wal_level = replica
```

```
pgbench --initialize --scale=2000
pgbench --jobs=4 --client=16 --time=3600
```

16 GB RAM instances, with network disks



64 GB RAM instances, with network disks

AWS	GCP	Azure	DigitalOcean	UpCloud
m4.4xlarge	n1-standard-16	Standard D5v2	64GB	16CPUx60GB
16 vCPU	16 vCPU	16 CPU	16 CPU	16 CPU
64 GB	60 GB	56 GB	64 GB	60 GB
1 TB EBS	1 TB PD-SSD	1 TB P30	1 TB block storage	1 TB MAXIOPS

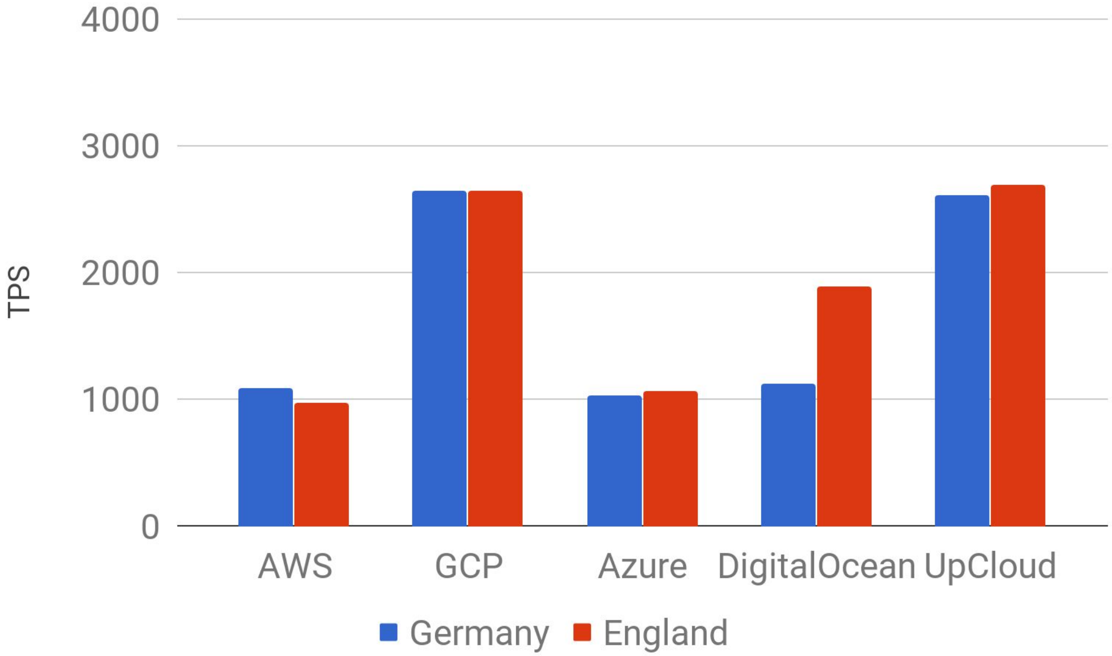
PostgreSQL 10.0
Linux 4.13.5

Data encrypted on disk
SSL required for clients
WAL archiving enabled

```
work_mem = 32MB
shared_buffers = 12GB
max_wal_size = 50GB
wal_level = replica
```

```
pgbench --initialize --scale=8000
pgbench --jobs=4 --client=64 --time=3600
```

64 GB RAM instances, with network disks



aiven

Benchmarks: local vs network disks

Google Cloud: Up to 8 local NVMe disks attached to any instance type

AWS: Fixed NVMe disks with i3.* instance types

Other clouds: *nothing applicable*

16 GB RAM instances, with local disks

AWS

i3.large
2 vCPU
15 GB
350 GB NVMe (max 475 GB)

GCP

n1-standard-4
4 vCPU
15 GB
350 GB NVMe (max 3 TB)

Azure, DigitalOcean, UpCloud

Not applicable

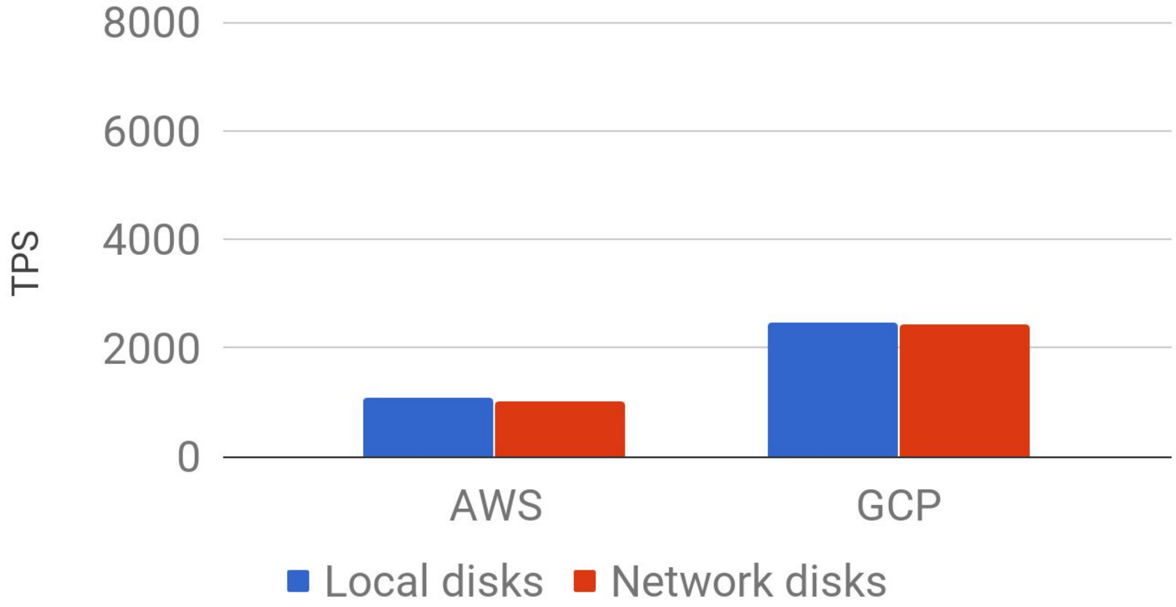
PostgreSQL 10.0
Linux 4.13.5

Data encrypted on disk
SSL required for clients
WAL archiving enabled

```
work_mem = 12MB
shared_buffers = 3GB
max_wal_size = 16GB
wal_level = replica
```

```
pgbench --initialize --scale=2000
pgbench --jobs=4 --client=16 --time=3600
```

16 GB RAM instances, local vs network disks



64 GB RAM instances, with local disks

AWS

i3.2xlarge
8 vCPU
61 GB
1000 GB NVMe (max 1900 GB)

GCP

n1-standard-16
16 vCPU
60 GB
1000 GB NVMe (scale up to 3 TB)

Azure, DigitalOcean, UpCloud

Not applicable

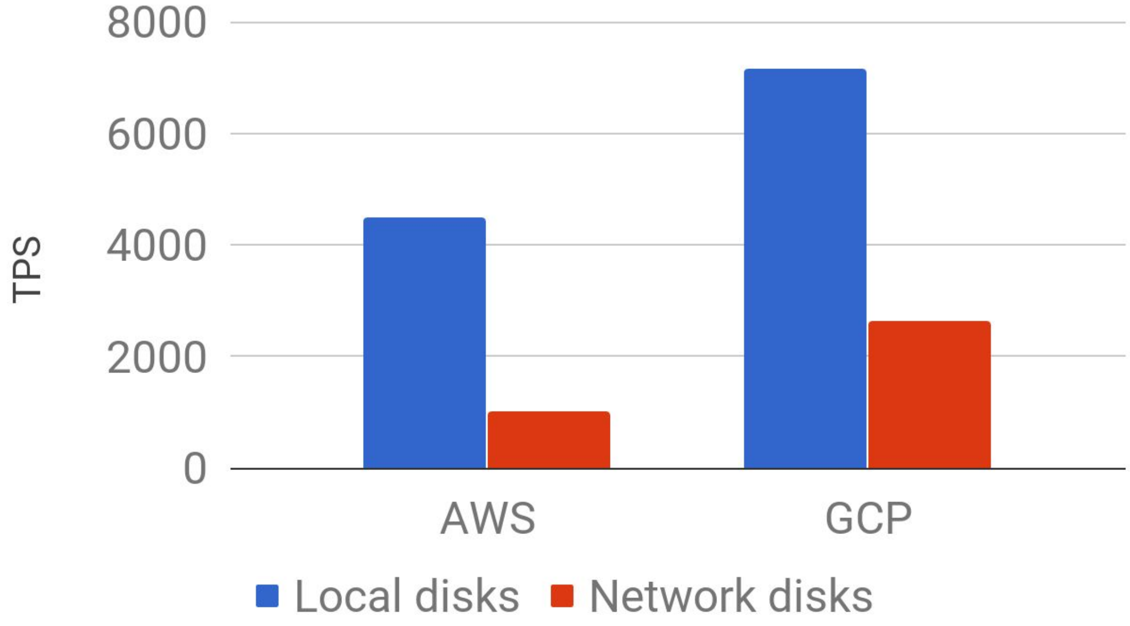
PostgreSQL 10.0
Linux 4.13.5

Data encrypted on disk
SSL required for clients
WAL archiving enabled

```
work_mem = 32MB
shared_buffers = 12GB
max_wal_size = 50GB
wal_level = replica
```

```
pgbench --initialize --scale=8000
pgbench --jobs=4 --client=64 --time=3600
```

64 GB RAM instances, local vs network disks



aiven

DBaaS comparison in AWS

Aiven PostgreSQL in AWS (10.0)
Amazon RDS for PostgreSQL (9.6.3)
Amazon Aurora with PostgreSQL (9.6.3)

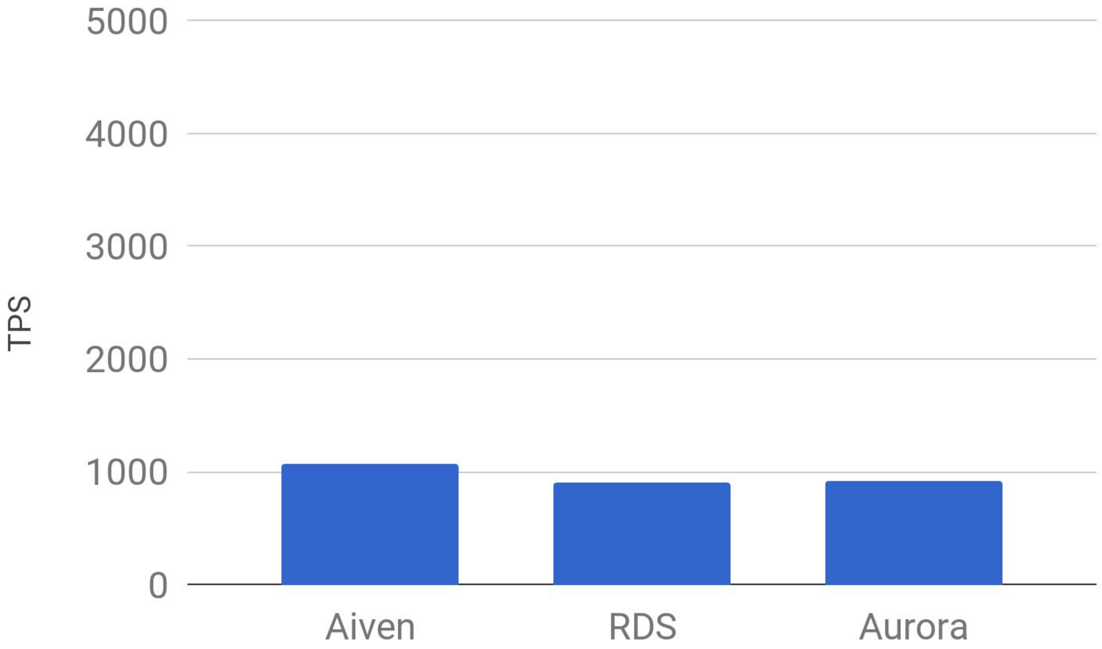
(eu-west-1, Ireland)

AWS DBaaS 16 GB RAM services

Aiven	RDS	Aurora
i3.large	db.m4.xlarge	db.r4.large
2 vCPU	4 vCPU	2 vCPU
15 GB	16 GB	15 GB
350 GB NVMe	350 GB EBS	transparently scalable storage
PostgreSQL 10.0	PostgreSQL 9.6.3	PostgreSQL 9.6.3
Data encrypted	Encryption disabled	Encryption disabled

```
pgbench --initialize --scale=2000
pgbench --jobs=4 --client=16 --time=3600
```

AWS DBaaS 16 GB RAM services

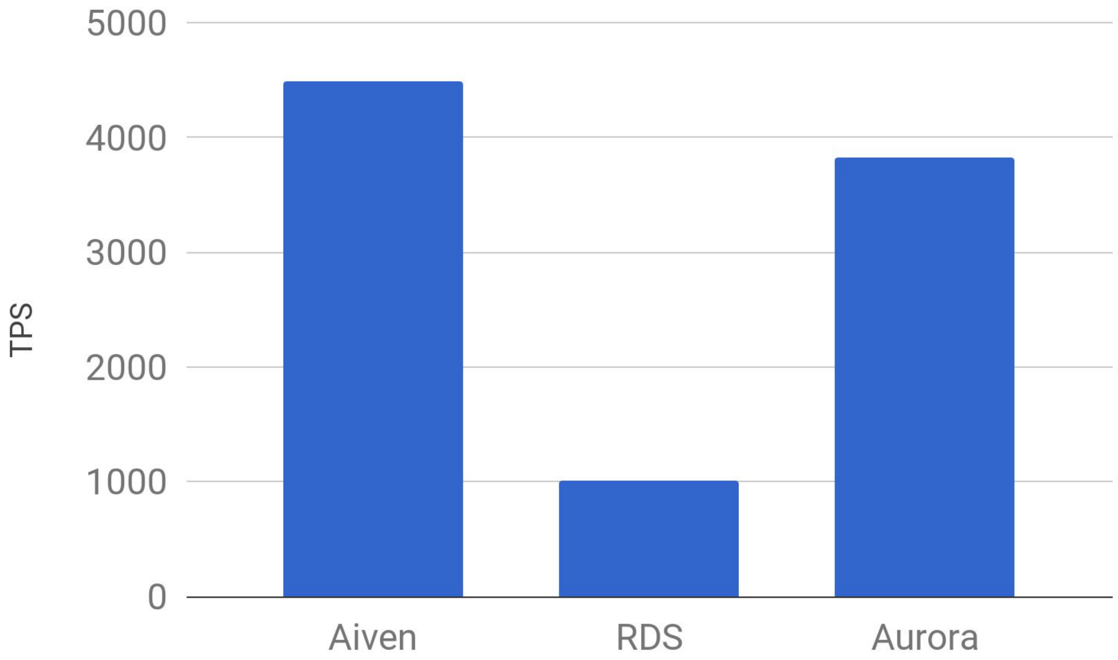


AWS DBaaS 64 GB RAM services

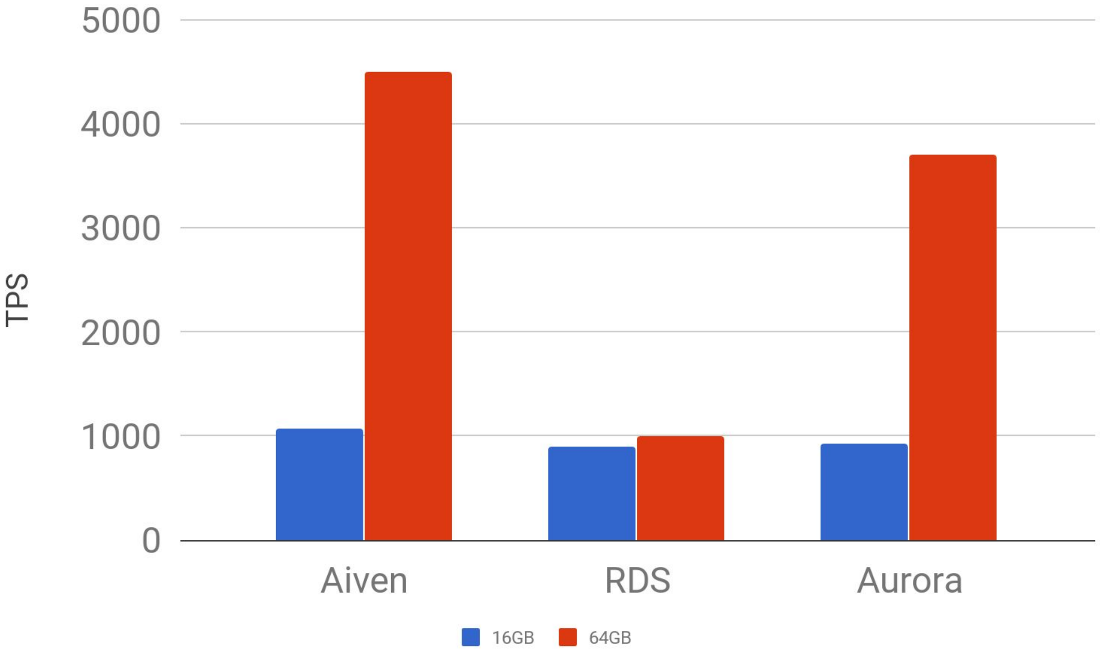
Aiven	RDS	Aurora
i3.2xlarge	db.m4.4xlarge	db.r4.2xlarge
8 vCPU	16 vCPU	8 vCPU
61 GB	60 GB	61 GB
1000 GB NVMe	1000 GB EBS	transparently scalable storage
PostgreSQL 10.0	PostgreSQL 9.6.3	PostgreSQL 9.6.3
Data encrypted	Encryption disabled	Encryption disabled

```
pgbench --initialize --scale=8000
pgbench --jobs=4 --client=64 --time=3600
```

AWS DBaaS 64 GB RAM services



AWS DBaaS 16 vs 64 GB RAM services



Questions?

Cool t-shirts for the first ones to ask a question!

(try out Aiven and get a cool t-shirt even if you didn’t ask a question)

aiven

Thanks!

@OskariSaarenmaa

@aiven_io