ASK THE EXPERT: PERFORM AT YOUR BEST WITH POSTGRES

19th July, 2022

EDB

OUR SPEAKERS

HOST

Kevin Li
Director Sales
Engineer, EMEA

SPEAKER

Piotr Kolodziej
Senior Sales Engineer,
EMEA

EDB™

AGENDA

- Good Performance: What Does It Mean?
- What are the key challenges?
- What are the best practices?
- How can EDB help with the tooling
- Demo (PEM)
- Q&A

GOOD PERFORMANCE: WHAT DOES IT MEAN?

Shorter and shorter response times?	=>	Speed
More and more operations per unit of time?	=>	Throughput
More and more users to be served? More and more data to be handled?	=>	Scale

Is this possible to satisfy all of them at the same time?

CAN WE SPEED UP?

	Rough estimation as of 2022
RAM access time	Few nanoseconds
SQL query execution time (small table, single row via primary key, prepared statement, data pages in database shared memory)	Few microseconds
LAN TCP/IP round-trip latency (eg. app server <-> database)	Hundred(s) of microseconds
Single disk access time (flash, SSD)	Tens of microseconds or more
Single disk read (magnetic disk)	Few milliseconds
Insert to the table with 1 index and commit	Hundreds of microseconds
WAN TCP/IP round-trip latency (eg. sync the replica)	From milliseconds to hundreds of milliseconds
Access to the data on the magnetic tape	Many minutes

EDB

Image source: https://openclipart.org/detail/314141/slow-rabbit-fast-hare

WHERE WE MAY SPEND THE TIME

Operation	Estimated comparison factor
SQL query execution time (small table, single row via primary key, prepared statement, data pages in database shared memory)	1
LAN TCP/IP round-trip latency (eg. app server <-> database)	x 100
Single disk access time (flash, SSD)	x 10-100
Single disk read (magnetic disk)	x 1 000
Insert to the table with 1 index and commit	x 100
WAN TCP/IP round-trip latency (eg. sync the replica)	x 1 000 - 1 000 000
Access to the data on the magnetic tape	x 100 000 000 or much more

EDB

Image source: https://openclipart.org/detail/314141/slow-rabbit-fast-hare

POLL

There is a small query that executes in 5-6 microseconds on average.

Can we expect 99.999% query execution times will fit 10 ms response time window?

1) Yes, we are on the safe side
2) No, we are not
3) We can’t evaluate it properly

EDB™

CAN WE BELIEVE AVERAGES WHEN FOCUSED ON RESPONSE TIME?

Imagine the following situation:

- Average query execution time
  - 5.65 microseconds
  - Sample: 645 535 executions
- The goal (Expectation)
  - All queries have to complete it time less than 10 milliseconds
  - Acceptable miss rate:
  - 10 queries per 1 million
- Are we on the safe side?

It can be a product of the response times:

Range boundaries (microseconds)		Count
0	8	632473
9	32	12765
33	128	1164
129	512	43
513	2 048	0
2 049	8 192	0
8 193	32 768	0
32 769	131 072	0
131 073	524 288	0
524 289	2 097 152	0

OK

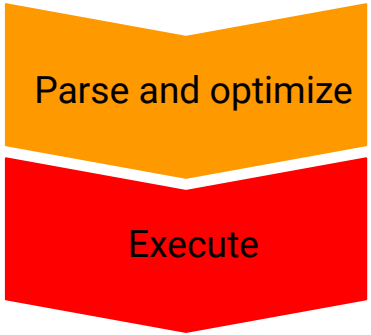
Range boundaries (microseconds)		Count
0	8	632452
9	32	12342
33	128	321
129	512	232
513	2 048	32
2 049	8 192	0
8 193	32 768	34
32 769	131 072	1032
131 073	524 288	0
524 289	2 097 152	0

Problem

KEY CHALLENGES

WHAT CAN AFFECT PROCESSING TIME OF THE SIMPLEST QUERY?

```
select * from pk.test_pk2
where id= 12005
and ts >= timestamp
'2022-07-07 00:00:00'
```




```
  id   |              ts               |      txt
-------+-------------------------------+---------------
 12005 | 2022-07-09 21:39:04.729068+02 | Test data 111
(1 row)
```

- Non-optimal SQL query plan, eg.
  - Stale statistics used by optimizer
  - Database schema design issues
- Execution costs above the expectations, eg.
  - Data pages swept from shared buffers, conflicting workload
  - Access costs to frequently changed data (MVCC)
  - Performance degradation by heavy UPDATEs or massive INSERT/DELETE statements
    - Less effective sequential reads
    - Will persist if not vacuumed
- Generic platform performance topics, eg.
  - OS scheduler: waiting for CPU on busy systems
  - SMP/NUMA: CPU cache efficiency, memory access latency
  - Virtual memory management and memory shortages
  - Conflict when an aggressive OS file system caching is set up
  - I/O affected by concurrent activity

EDB™

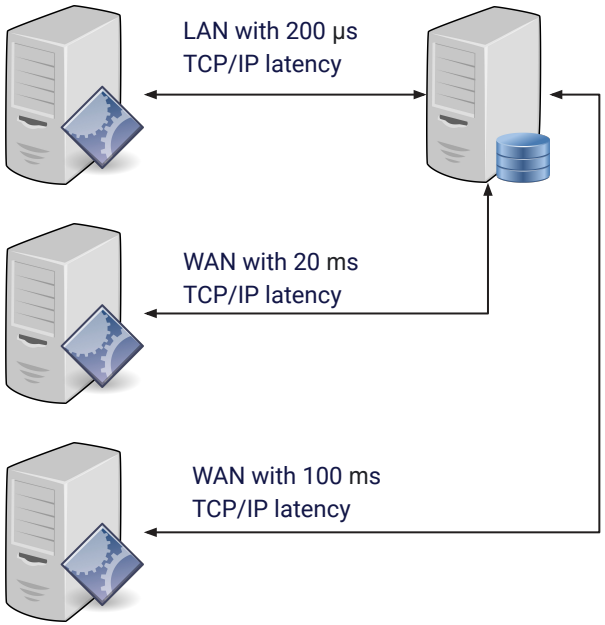
POLL

What CPU usage should be the maximum when we focus on stable short response times for small queries?

1) > 95%
2) 80%
3) 60-70%
4) < 50%

EDB™

LATENCY BETWEEN APPLICATION AND THE DATABASE



- May be the significant part of overall response time
  - Chatty applications - many small database requests
  - Fetching row by row in separate database calls
  - Not processing arrays of rows with SELECT or DML statements
  - Not buffering the data of frequent use and very slowly changing pattern (eg. configuration)
  - Not leveraging database procedural language to group a set of small SQL statements into one unit of work
- Let's assume the application server calls the database 40 times to render the response
  - LAN with 200 µs TCP/IP latency: 8ms spent in the network
  - **WAN with 20 ms latency: 800ms spent in the network**
  - **WAN with 100 ms latency: 4s spent in the network**
  - **More round-trips, more exposure to network lags above average latency**

EDB

DATABASE THROUGHPUT GOALS

Handle the workload in a unit of time

- Representing the business activity
- Defined by an application(s)
- Set of operations (queries, transactions, etc.)

On a specific configuration

- Hardware & network resources
- Software components

Run at the certain scale

- Database size
- Concurrency level
- Relevant to the business

Things to measure

- Achieved number of operations
- Achieved response times
- System and database load

EDB

My database cannot handle more workload, should I add more CPUs?

WHY ADDING CPUS MAY NOT SOLVE THE PROBLEM

Where is the the bottleneck?
What is the root cause?
What are the factors to consider?

- System architecture
- Infrastructure capacity and capabilities
- Database design and maintenance practices
- Data volume
- User population
- Non-human devices
- Automated interfaces
- Load profile, business activity
- Availability overhead
- Integrity overhead
- Confidentiality overhead

- Data retention
- Data skew
- Code quality
- Others…

EDB

BEST PRACTICES

CHOOSE RIGHT STORAGE FOR YOUR DATABASE

- Revolutionary change between magnetic disks (HDDs) and modern SSDs (and Flash, NVM)
- Disk I/O pains could be reduced if not even removed in some cases
- Select the right disks and consider to store:
  - **Hot data** with frequent random access: prefer SSDs
  - **Warm data** with infrequent random access: SSDs or a set of “high performance” HDDs*
  - **Cold data** with rare random access: may use a set of high capacity HDDs
  - Archive data with almost no random access: candidate to use high capacity HDD
  - Disk-based backup: may use a set of high capacity HDDs optimized for throughput
- Workload profiles may differ
  - OLTP: Significant size of “Hot data”, frequently changed
  - Data Warehouse: Significant amount of “Warm data”, may have “Cold data” and “Archive data”
    - Data warehouses may also contain some “Hot data”

EDB™

*High performance HDD: Much lower capacity and higher rpm rate than high capacity HDD

VERIFY DATABASE PERFORMANCE BEFORE GOING LIVE OR BEFORE THE CHANGE

Pre-defined database load generators

- Eg. HammerDB, pgbench
- TPC-C, TPC-H, other benchmarks

Good for

- Stress test the infrastructure
- Tune OS & database to some extent
- Compare the infrastructures
- Learn the impact of an infrastructure change

Limitations

- Will not verify your application performance and scalability

Application testing tools

- Define and run with tools like Apache Jmeter, LoadRunner, WebLOAD, etc.

Good for

- End-to-end testing of application, database and the infrastructure
- Scalability pattern of the application
- Identify and resolve system bottlenecks before going live
- Change impact analysis

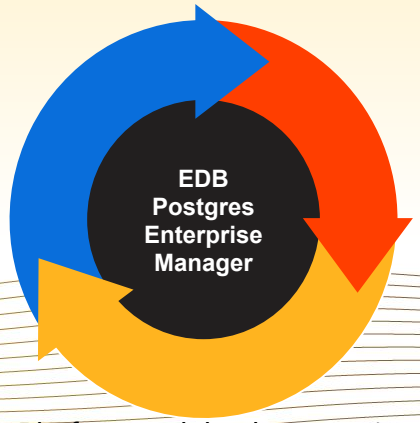
Limitations

- Prepare and maintain the adequate workload definition and test data set

ANALYZE THE DATABASE LOAD (OBSERVE, ORIENT, RECOMMEND ACTIONS)

- **Platform level**
  - CPU usage
  - Memory usage
  - I/O, network
- **Postgres kernel**
  - Configuration
  - Schema metadata
  - Server level performance metadata
  - Session level metadata
- **EDB Postgres kernel extensions**
  - Wait States
  - SQL Profiler
  - Dynamic Runtime Instrumentation Tools Architecture (DRITA)

Tune infrastructure, database server, SQL and database structure



Static analysis, recommended practices

Platform and database runtime metrics, events and alerts

WHAT DO YOU NEED TO PRACTICE GOOD DATABASE PERFORMANCE

- Team of people supported by effective tools
- Resources to run the tests and analysis
- Time, time and time: it is a continuous process
- Ability and empowerment to introduce the changes
- Design and coding hygiene

EDB™

PROACTIVE APPROACH (HYGIENE OF THE DATABASE PERFORMANCE)

- Logical database design
- Application code quality: queries, DMLs, etc.
- Database server configuration and tuning
- Application architecture, topology, database connection management
- Application - database traffic nature
- Database calls pragmatics: how do they relate to the business activity?
- Workload management (queuing, scheduling)

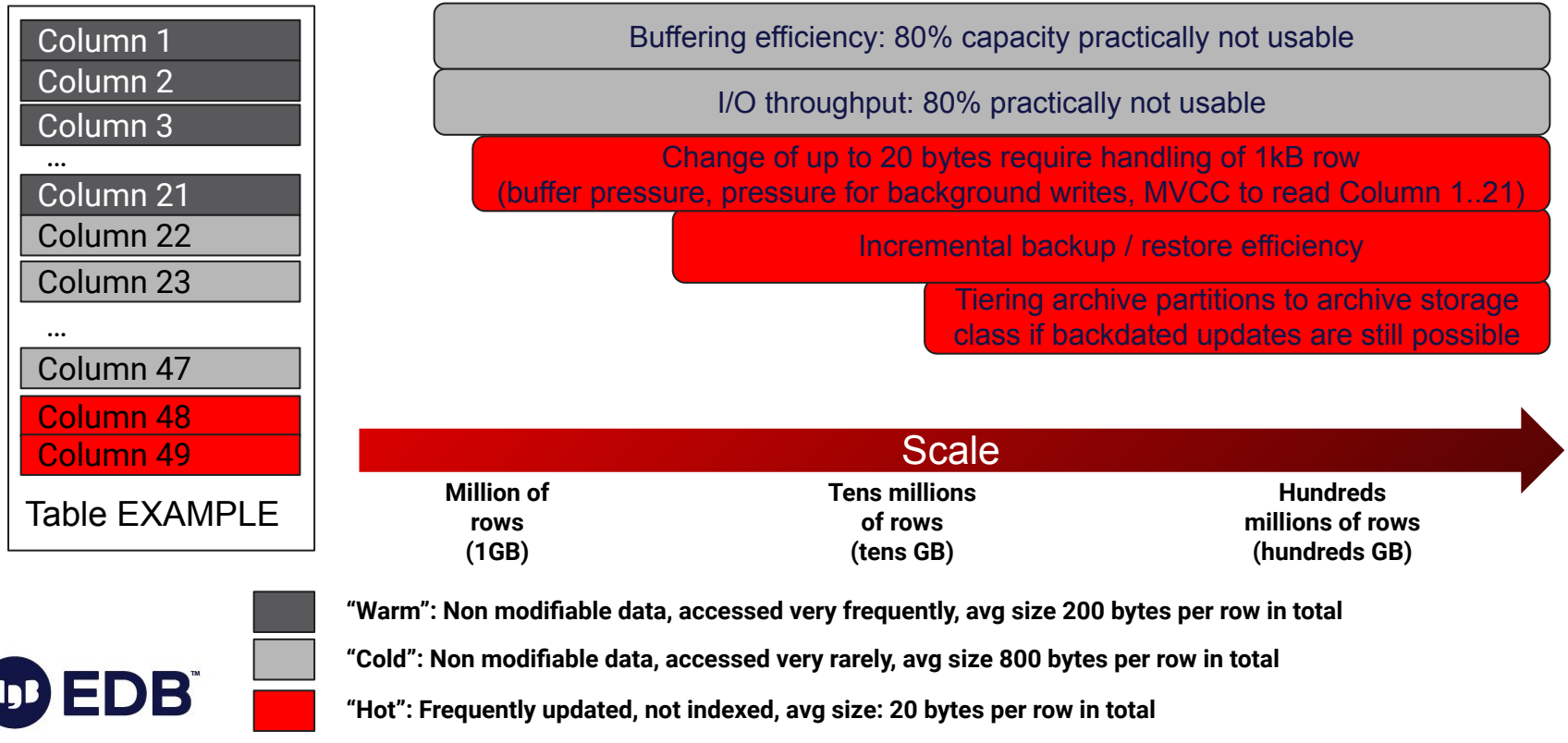
EDB

PROACTIVE APPROACH (HYGIENE OF THE DATABASE PERFORMANCE)

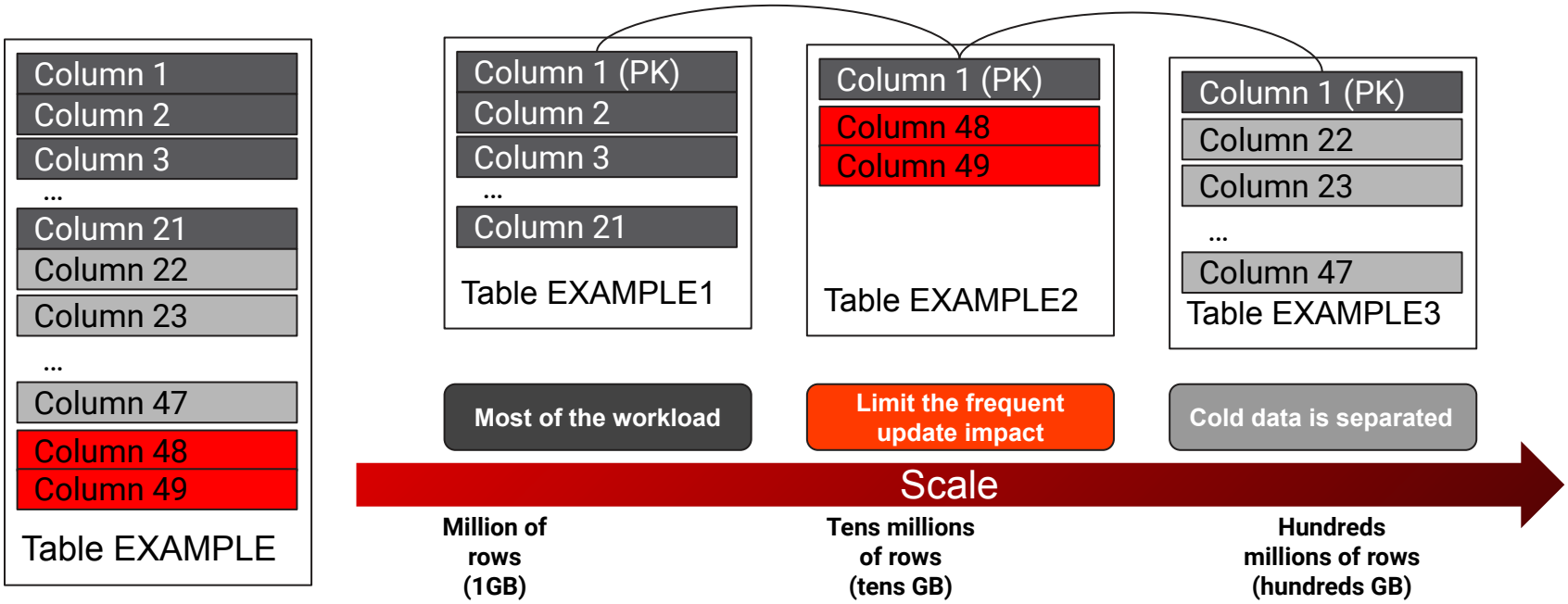
- Logical database design: To scale you may need to go out of the box
- Application code quality: queries, DMLs, etc.
- Database server configuration and tuning
- Application architecture, topology, database connection management
- Application - database traffic nature
- Database calls pragmatics: how do they relate to the business activity?
- Workload management (queuing, scheduling)

EDB

LOGICAL DESIGN: HOW WE CAN SCALE A TABLE WITH VARIOUS DATA "TEMPERATURES"?



WHEN 1:1 RELATIONSHIP MAY MAKE SENSE



"Warm": Non modifiable data, accessed very frequently, avg size 200 bytes per row in total

"Cold": Non modifiable data, accessed very rarely, avg size 800 bytes per row in total

"Hot": Frequently updated, not indexed, avg size: 20 bytes per row in total

HOW EDB CAN HELP

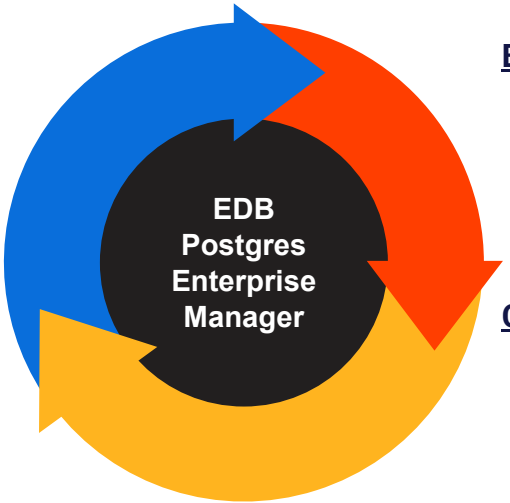
HOW EDB MAY HELP TO DIAGNOSE AND OPTIMIZE THE DATABASE PERFORMANCE

Optimize structure and SQL

- SQL Profiler
  - Workload filter
  - Comprehensive analysis
- Index Advisor

Relevant for:

- Application development, unit tests, performance tests with PL/pgSQL and SPL debugger
- Production troubleshooting and tuning



Base hygiene support

- Static analysis and recommended practices
  - Tuning Wizard
  - Postgres Expert
- Database configuration and schema design

Continuous database runtime diagnostics

- Monitoring facilities
  - Performance metrics: Capacity Manager
  - Charts, probes, alerts, dashboards
  - Postgres Log Analytics Expert
  - Performance Diagnostics tool
- Database and infrastructure level
- Persistence and historical data access

EDB

EDB Professional Services
Supporting You with all important tasks


DEMO

THANK YOU FOR ATTENDING!

EDB™