ORACLE

Is PostgreSQL Catching Up to Oracle Database?

Francisco Munoz Alvarez

Distinguished Product Manager

Oracle Database High Availability (HA), Scalability and Maximum Availability Architecture (MAA) Team

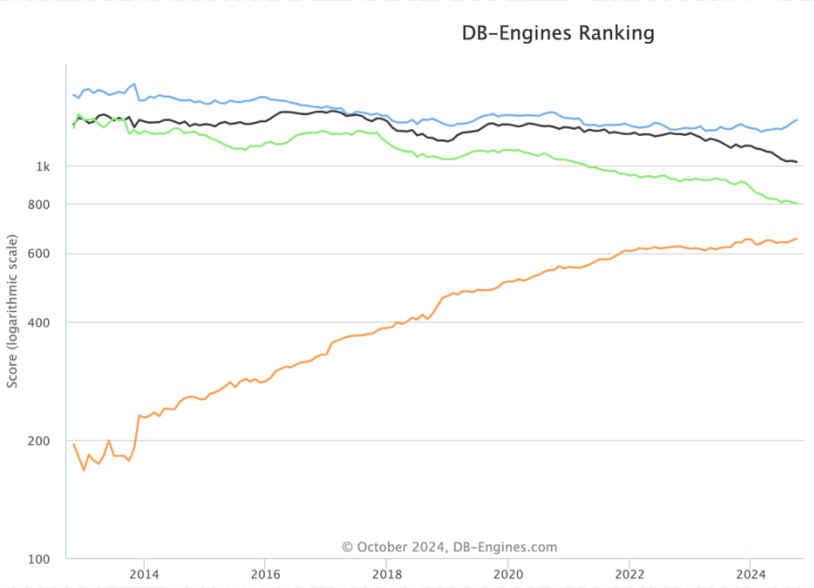
 @fcomunoz

 http://www.linkedin.com/in/franciscomunozalvarez

 www.oraclemaa.com

Ecosystem

The PostgreSQL Momentum



Source: db-engines.com

The PostgreSQL Momentum

Popularity increased with public clouds offering PostgreSQL services.

Why is PostgreSQL so popular?

- **Open source**
  Free to use, modify, and distribute.
- **No upfront costs**
  No licensing cost, no complex contracts.
- **Broad offers from service providers**
  On-premises and cloud services.
- **SQL features**
  Good SQL support, multi-model (JSON, spatial data, etc.)
- **Community support**
  A large community of developers.
- **Ease of use**
  Small footprint, simple to deploy.

What about:

- **Ecosystem**
- **Extensions**
- **Architecture**
- **Data Protection**
- **Security**
- **Availability**
- **Compatibility**
- **Observability**
- **Performance**
- **Scalability**
- **Innovation**
- **Total cost of ownership**

What our Customers Say About PostgreSQL

Technology Sprawl

Integrating too many extensions and specialized databases increases security risks and compatibility issues. The resulting IT architecture is significantly more complex.

Inferior Performance and Observability

Applications ported from Oracle to PostgreSQL often do not meet performance requirements and require application re-architecture.

Lack of Enterprise-Grade capabilities

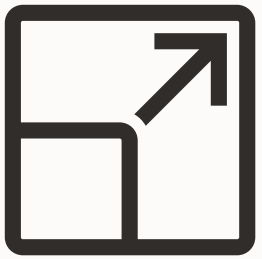
The lack of advanced security, data lifecycle, protection, and high availability features forces customers to buy additional software or build non-business-related solutions in-house.

Higher Operational Costs

Upgrades and application deployments require more downtime, while administrative tasks are heavily affected by the PostgreSQL architecture.

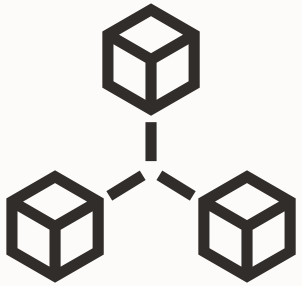
Be Ready for the Unexpected

Can you face unexpected growth?



Meeting performance requirements at scale can be extremely expensive on-premises and in the Cloud. Only Oracle Exadata and Exadata Cloud Services offer peace of mind with predictable performance and cost.

Do you know the transactional semantics of PostgreSQL variants?



Scalability limits in PostgreSQL require scale-out solutions offered by Postgres-compatible databases. However, data distribution has intrinsic complexity and comes with trade-offs.

https://videohub.oracle.com/media/+Database+Development+Champions+-+Are+Are+Cloud+Databases+fast+enough+for+Hardcore+OLTP/1_o5aus0zy

Ecosystem

The Complexity of the PostgreSQL Ecosystem

Growing ecosystem resulting in a *Complex Landscape* and *Complex Governance*

Distributors – Service Providers

- EDB
- Cybertec
- PostgreSQL Professional
- Fujitsu
- Percona
- SplendidData
- StackGres
- Pigsty
- …

Forks

- EDB
- YvorySQL
- Yugabyte
- CockroachDB
- Greenplum
- FerretDB
- Oriole
- PowerGres
- …

Cloud Variants

- OCI
- Azure
- AWS RDS
- GCP
- Heroku
- Neon
- Aiven
- Tembo
- Exoscale
- Clever Cloud
- Timescale
- StackIT
- …

Extensions

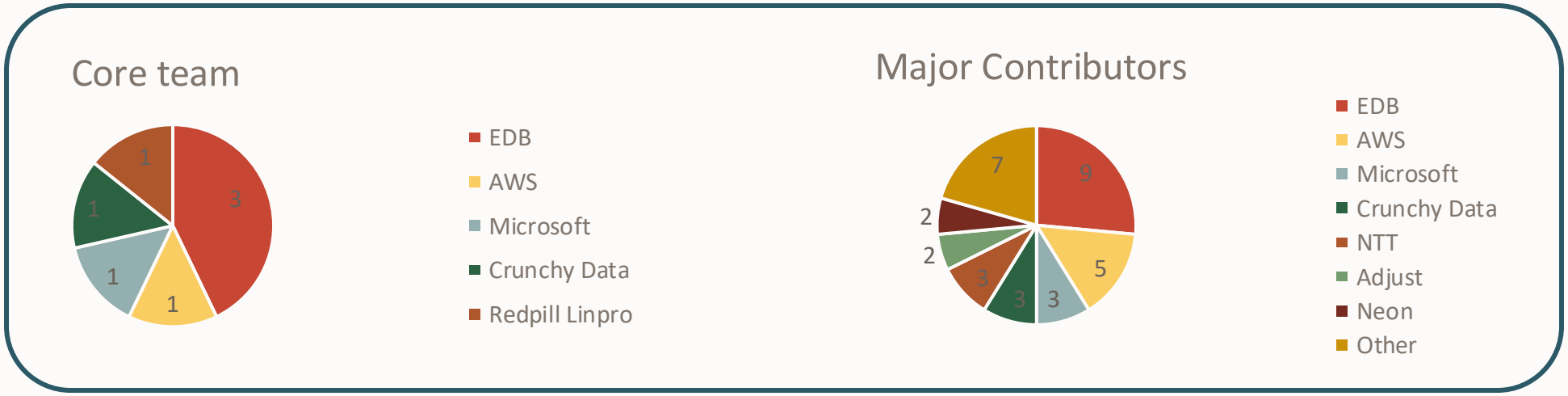
- PostGIS
- Citus
- TimescaleDB
- Greenplum
- WarehousePG
- ParadeDB
- Hydra
- pgvector
- pg_quack
- …
- **1000+ extensions!**

PostgreSQL Core Product

Or the “Vanilla” PostgreSQL

Lightweight engine

- Appealing for containerization.
- Minimal enterprise features.
- How easy is it to integrate new code?
- Who decides what goes into the core product?
- Often pushing back suggestions (see hints).
  https://www.postgresql.org/message-id/m3d6avxqmm.fsf%40wolfe.cbbrowne.com



Extensions

PostgreSQL Needs Extensions

To solve many trivial problems

- Verify the password complexity (passwordcheck)
- Get real-time statement statistics (pg_stat_statements)
- Get basic database metadata (db_info)
- Run REST calls (http)
- Use optimizer hints (pg_hint_plan)
- …

To mimic some Oracle features

- Interval partitioning (pg_partman)
- Table reorganization (pg_repack)
- Job scheduling (pg_cron)
- Temporal validity (periods)
- Flashback Data Archive (pgmemento)
- Active Session History (pgsentinel)
- Message queueing (Qgres)
- …

A common argument from the PostgreSQL community is:
"These capabilities do not belong to a database engine".

Many PostgreSQL extensions might not have the required maturity to run production databases.

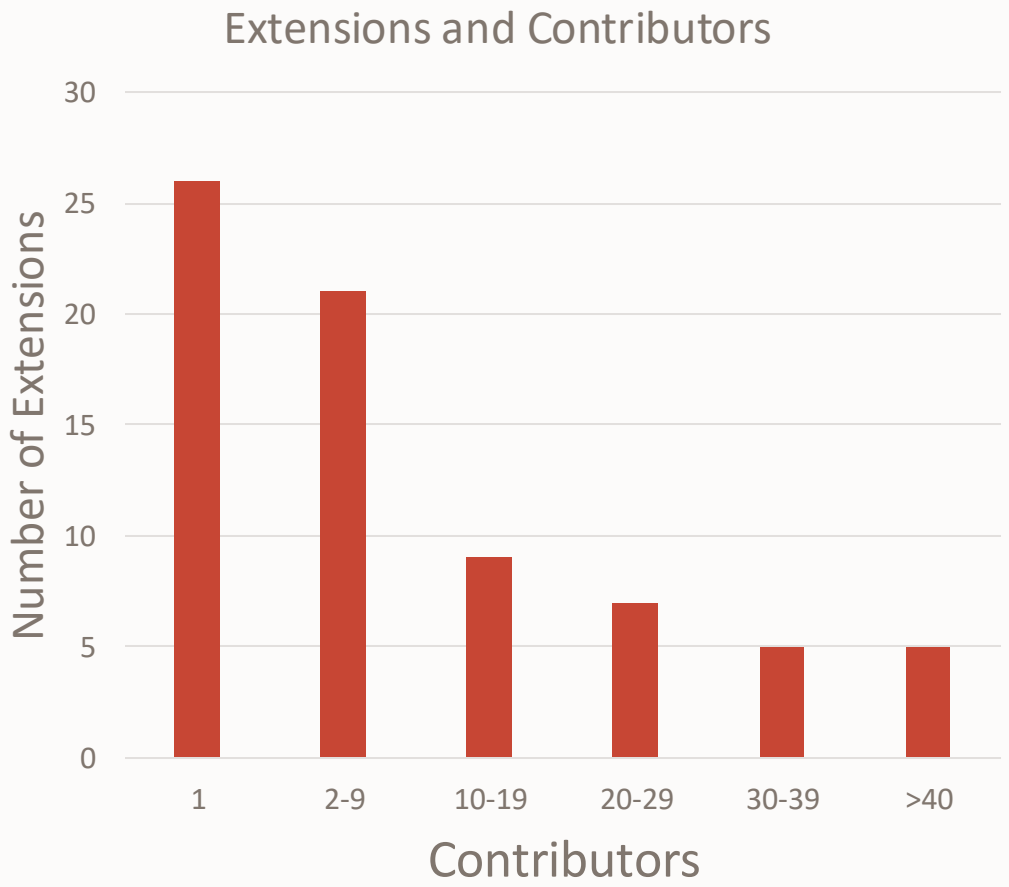
Extension Management can be Problematic

https://www.postgresql.eu/events/pgconfde2025/sessions/session/6474/slides/695/Extend%20your%20PostgreSQL_%20The%20world%20of%20PostgreSQL%20extensions%20-%202025.pdf

PostgreSQL Extensions

"The real strength of PostgreSQL is its extensibility" (quot.)

- Too many extensions (1000+) (1)
- It's common to install 20+ of them
- Some distributions offer 200+ extensions
- Different source repos and OS requirements
- Different licenses
- Complex governance
- Security concerns (2)
- Who creates and maintains them?



* Checked against the 70 most popular extensions, excluding *contrib*

(1) https://gist.github.com/joelonsql/e5aa27f8cc9bd22b8999b7de8aee9d47
(2) https://www.postgresql.org/docs/current/extend-extensions.html#EXTEND-EXTENSIONS-SECURITY

Security issues can arise from "not secure extensions

Risks of insecure extensions

- **Malicious code injection**: Insecure extensions can act as an entry point for attackers. A malicious extension could have a backdoor, allowing a user to execute arbitrary code on the server or a user with insufficient privileges to escalate their access to superuser status.
- **Arbitrary code execution**: A compromised or poorly written extension can allow an attacker to execute code on the server, potentially leading to a full system compromise.
- **Privilege escalation**: An attacker might exploit a vulnerability in an extension to gain higher privileges than they are supposed to have, potentially allowing them to access sensitive data or control the database system

Security issues can arise from "not secure extensions

aqua Platform Solutions Resources Company Get Started

Aqua Blog

PG_MEM: A Malware Hidden in the Postgres Processes

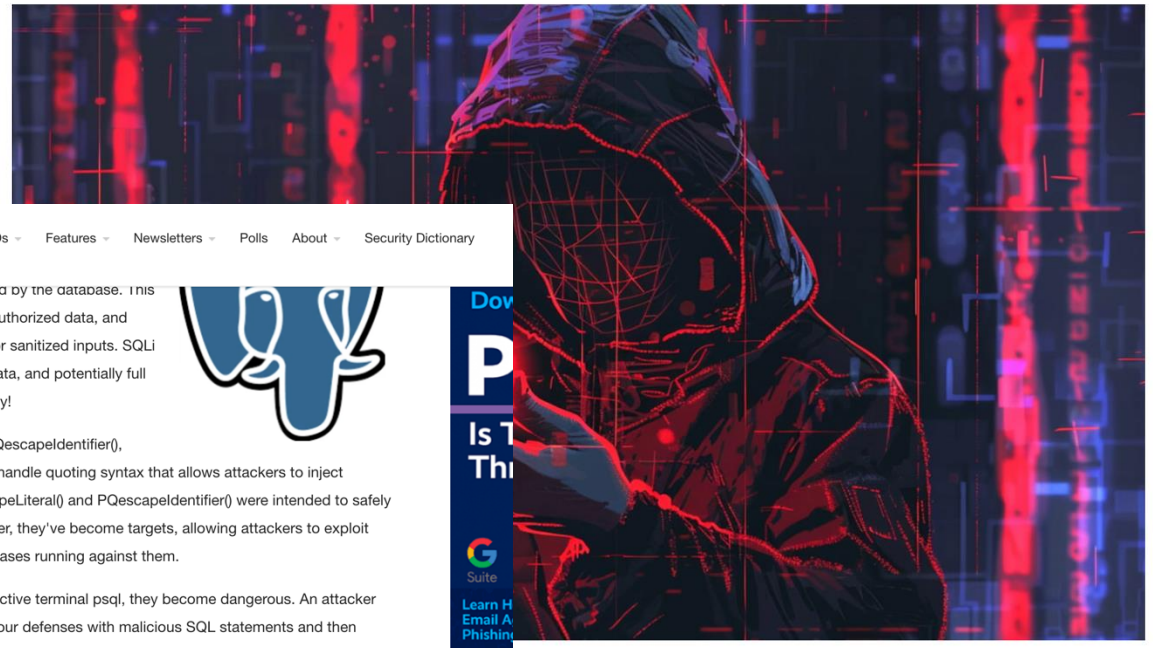
Assaf Morag
August 19, 2024

Aqua Nautilus researchers have uncovered PG_MEM, a new Pos
brute forces its way into PostgreSQL databases, delivers payloa
and mines cryptocurrency. In this blog, we explain this attack, t
threat actor, and how to detect and protect your environments

PostgreSQL flaw exploited as zero-day in BeyondTrust breach

By Sergiu Gatlan

February 14, 2025 09:15 AM 0



kers exploited a PostgreSQL security flaw as a zero-day to
ement company BeyondTrust in December.

its systems and 17 Remote Support SaaS instances in early
-12356 and CVE-2024-12686) and a stolen API key.

e U.S. Treasury Department disclosed that its network
n Remote Support SaaS API key to compromise its

LinuxSecurity ESTABLISHED 1996

News Advisories HOWTOs Features Newsletters Polls About Security Dictionary

malicious SQL code into an input field, which is then executed by the database. This enables the attacker to manipulate the database, access unauthorized data, and perform privileged operations by exploiting poorly validated or sanitized inputs. SQLi vulnerabilities can lead to unauthorized access to sensitive data, and potentially full system compromise - so they are not something to take lightly!

In the specific case of CVE-2025-1094, PQescapeLiteral(), PQescapeIdentifier(), PQescapeString(), and PQescapeStringConn() all improperly handle quoting syntax that allows attackers to inject malicious SQL commands using key libpq functions. PQescapeLiteral() and PQescapeIdentifier() were intended to safely escape user input and prevent injection attacks. Now, however, they've become targets, allowing attackers to exploit them and insert malicious SQL commands directly into databases running against them.

When you use these functions to feed into PostgreSQL interactive terminal psql, they become dangerous. An attacker could exploit how these functions' process input to bypass your defenses with malicious SQL statements and then execute remote commands against vulnerable versions of BeyondTrust Remote Support to gain complete control of impacted systems.

Not only are libpq functions vulnerable, but command line utilities of PostgreSQL may also be susceptible. This becomes especially apparent when client_encoding is set to BIG5 while server_encoding differs - opening up an opportunity to inject potentially dangerous SQL commands through command-line operations.

Affected Versions and Patches

It is crucial to understand which PostgreSQL versions are at risk and the patches released address to address the issue. Specifically, those using versions prior to 17.3, 16.7, 15.11, 14.16 or 13.19 could be vulnerable to this exploit

Patches for this issue have already been released to address it. PostgreSQL 16.7, 15.11, 14.16, and 13.19 versions were

Known Issues and Cases in Extensions

- **Privilege Escalation via Extensions:**
  - **CVE-2023-32305 (aiven-extras extension):** A flaw allowed an unprivileged database user to elevate their privileges to superuser level through a specific bug in this third-party extension.
  - **CVE-2023-39417 (Extension installation scripts):** A vulnerability in how extension installation scripts handle substitutions (`@extowner@`) could allow SQL injection and arbitrary code execution with higher privileges during installation.
  - **CVE-2022-2625 (Various extensions):** core server impacted several extensions certain objects to execute arbitrary SQL.
  - **CVE-2018-1058:** An attacker could use t functions are resolved) to execute code could be leveraged via extensions or cus
- **Arbitrary Code Execution via Language Extensions:**
  - **CVE-2024-10979 (PL/Perl):** A high-severity flaw in the PL/Perl extension allowed an unprivileged user to alter environment variables, which could lead to arbitrary code execution even without an operating system user account on the server.
  - **Buffer Overflows:** Older vulnerabilities in extensions like `pgcrypto` have included memory or buffer overflows that could potentially lead to code execution or denial of service, requiring an authenticated user to exploit.

Survey and Evaluation of Database Management System Extensibility

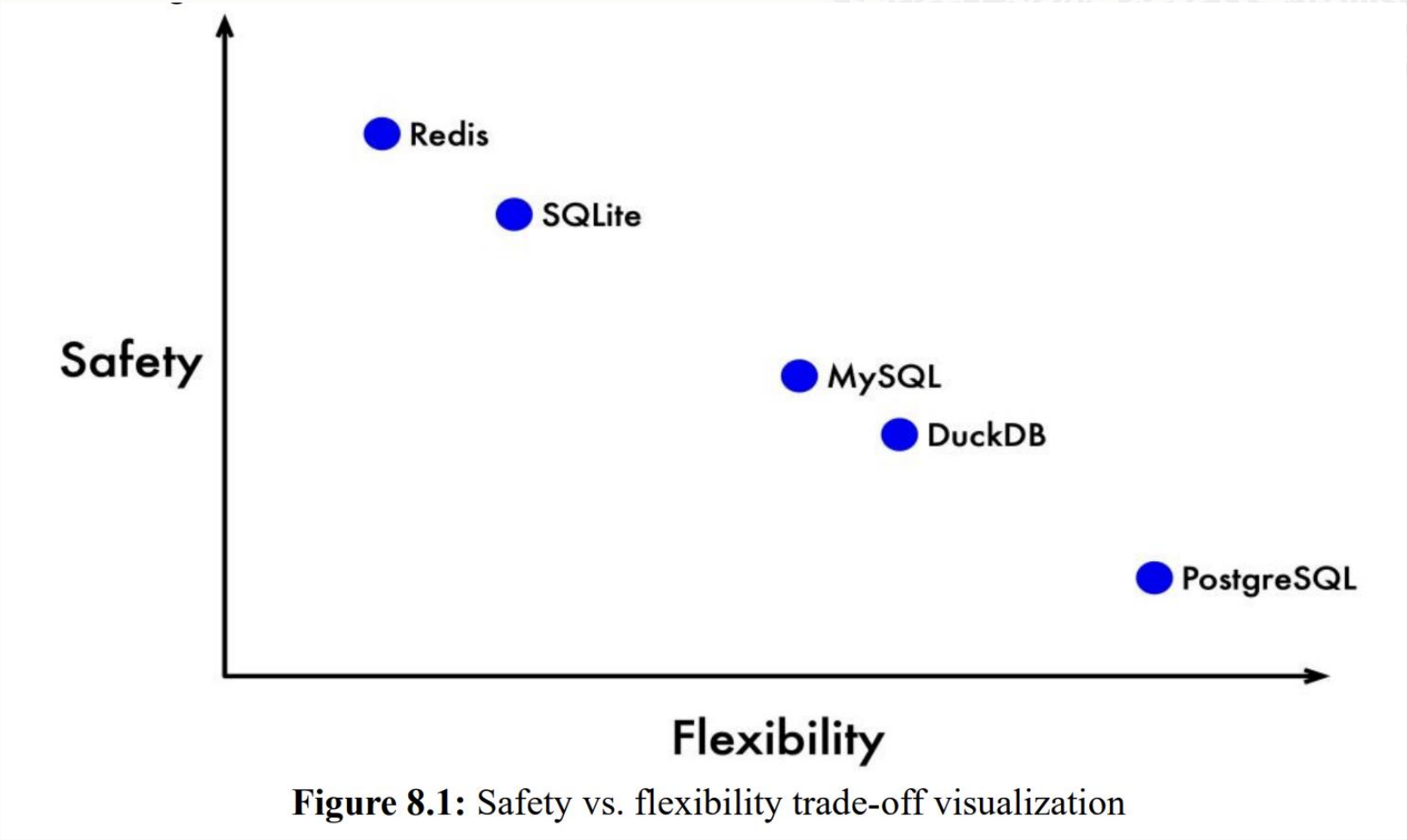
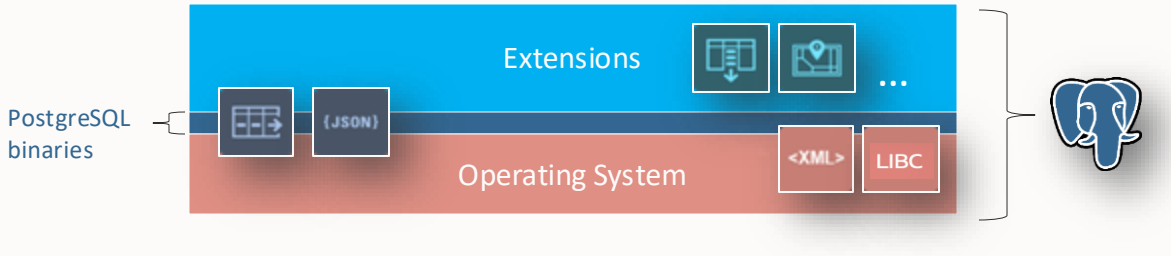


Figure 8.1: Safety vs. flexibility trade-off visualization

http://reports-archive.adm.cs.cmu.edu/anon/2023/CMU-CS-23-144.pdf…
https://www.vldb.org/pvldb/vol18/p1962-kim.pdf

Architecture

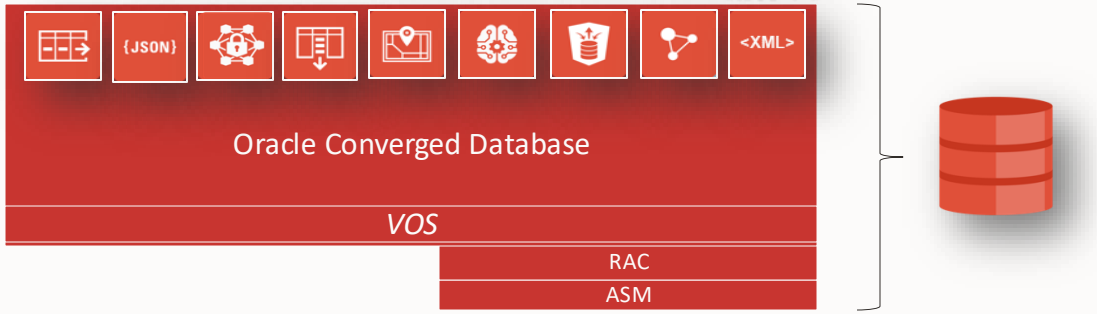
PostgreSQL vs. Oracle Database Architecture



Relies on OS andexternal libraries as much as possible

- Sub-optimal memory utilization (1)
- Unpredictable behavior (2)
- Various issues can arise during OS updates (3)
- No central governance in the supply chain

1.4 Million lines of code



Zero external dependencies

- End-to-end optimization and compatibility
- Consistent behavior everywhere
- Comprehensive and OS independent
- Strict control and governance of the whole stack

14 Million lines of code

(1) (2) https://www.dbi-services.com/blog/postgresql-shared-buffers-vs-free-ram/
(3) https://gitlab.com/gitlab-org/gitlab/-/issues/369573 (probably solved in PostgreSQL 17)

PostgreSQL vs. Oracle Database Architecture

Source: https://www.youtube.com/watch?v=df-l2PxUidI
Any Pavlo, Carnegie Mellow University (CMU)

“Good enough”

but possibly

unfit for mission-critical environments

PostgreSQL Has a Bad MVCC Implementation

"Our research at Carnegie Mellon University […] have shown that its MVCC implementation is the worst among the other widely used relational DBMSs, including MySQL, Oracle, and Microsoft SQL Server."

Source: https://ottertune.com/blog/the-part-of-postgresql-we-hate-the-most

Most annoying issues:

- Data duplication
- Table bloat
- Index write amplification
- Transaction Wraparound
- Vacuum Management
- Transactional DDLs (2)

(2) https://postgres.ai/blog/20210923-zero-downtime-postgres-schema-migrations-lock-timeout-and-retries

Oracle Database’s maturity
is the result of 45+ years of running the world's most significant and critical workloads.

Kyle Hailey • 1st

Co-Founder Innovage AI

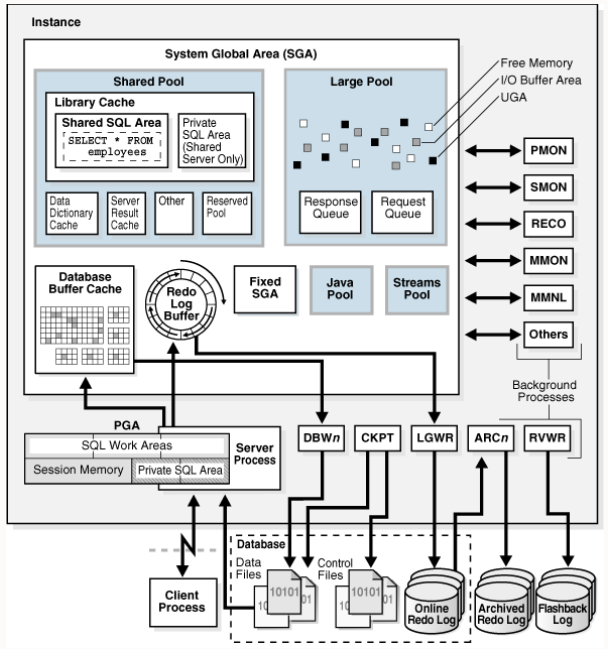
3w • Edited •

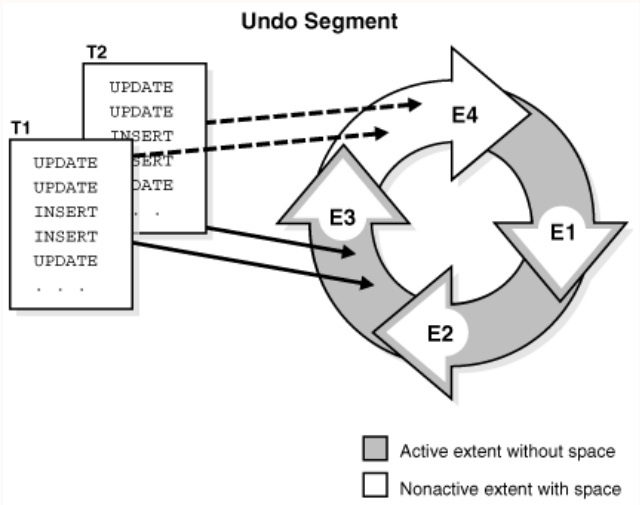
The best database out there, by far, is Oracle. It works in the highest percentage of situations. Now, if you think PostgreSQL works for you, you're lucky, because it works for some cases and it doesn't for others. If you're using MySQL and it works, you're even luckier. It's not because MySQL or PostgreSQL are as good as Oracle, it's just you're lucky.

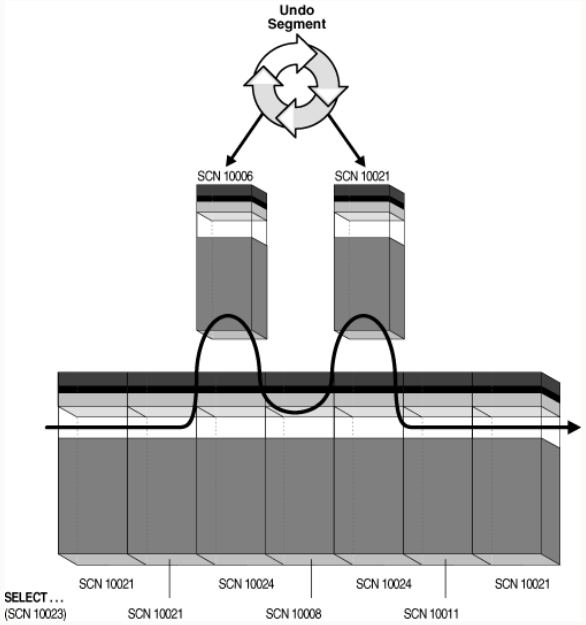
PostgreSQL is the first database I would turn to for my RDBMS needs, but if I had a critical database with super high throughput and high reliability ideas, I'd use Oracle (Update, yes money matters, if I'm not making money I'm going to use Postgres and not Oracle. If I'm doing 50,000 transactions a second I'm most likely making money and can afford Oracle)

Oracle Database MVCC is Hard to Beat

20+ years later, Automatic Undo Management is probably still the best implementation





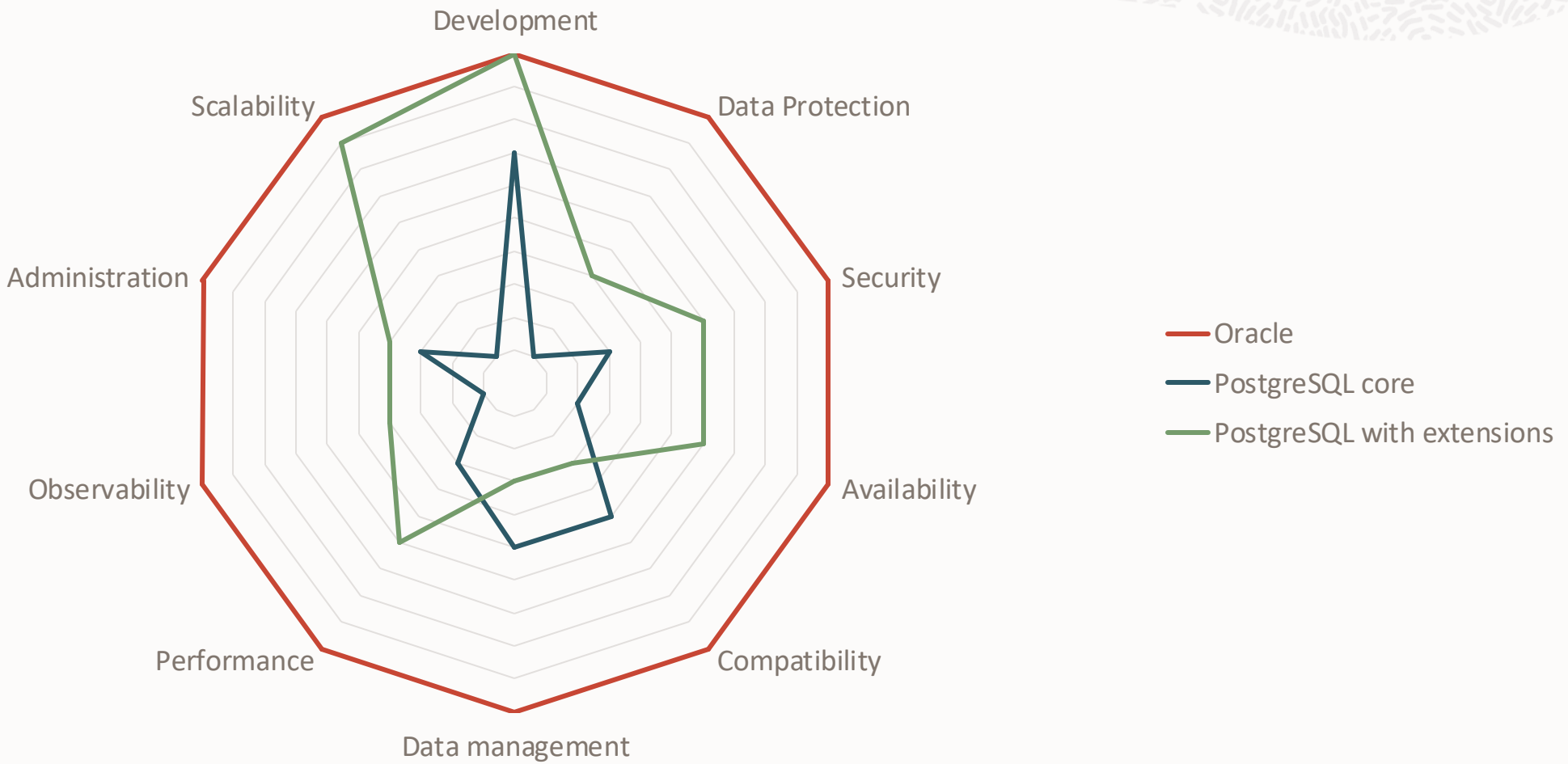


EnterpriseDB started the Zheap project in 2013 to implement UNDO-based MVCC.

The last commit was in 2019, and the project is at a dead point.

https://wiki.postgresql.org/wiki/Zheap

Product Maturity and Completeness (1)



(1) The figures shown here **are subjective** and estimated considering the advanced Oracle features highlighted for each category during the rest of the presentation.

Data Protection

Don’t Compromise on Data Protection

Some important Oracle data protection features that PostgreSQL is missing:

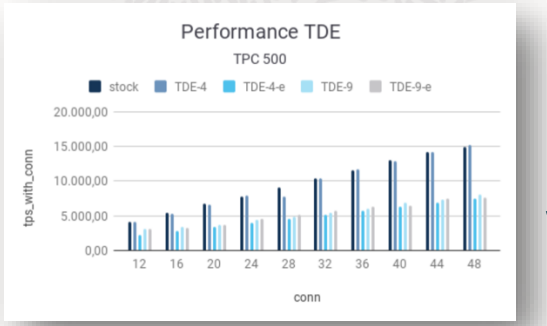
- On-disk physical block corruption detection *and automatic repair (ASM, Active Data Guard)*
- Lost write detection
- Flashback (time travel capabilities) to recover “human errors.”
  - At the query, table, or database level
  - At the transaction level (with supplemental logging)
- Point-in-time recovery at the Pluggable Database level
- Automatic, fast reinstate of the standby after failover (Data Guard)
- Checksum validation from OS to HBA to Disk using DIX/DIF (Exadata)
- Hardware-Assisted Resilient Data disk scrub and repair (Exadata)
- Corruption detection and periodical validation during backup and restore (ZDLRA)

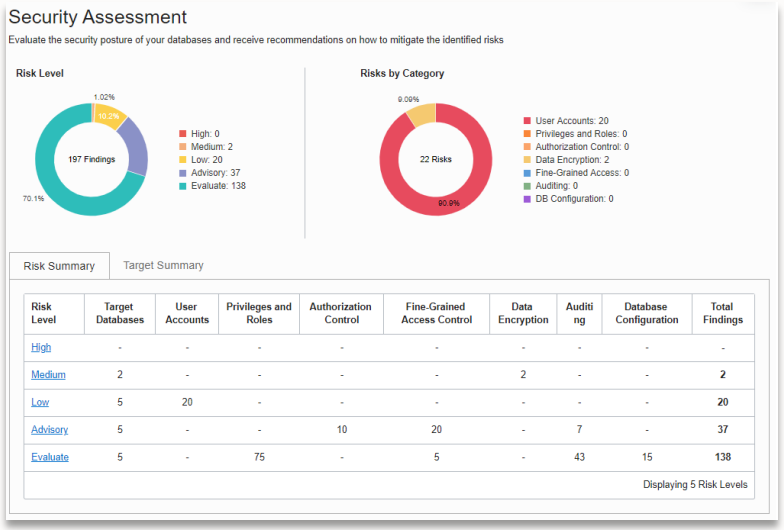
Security

Don’t Compromise on Security

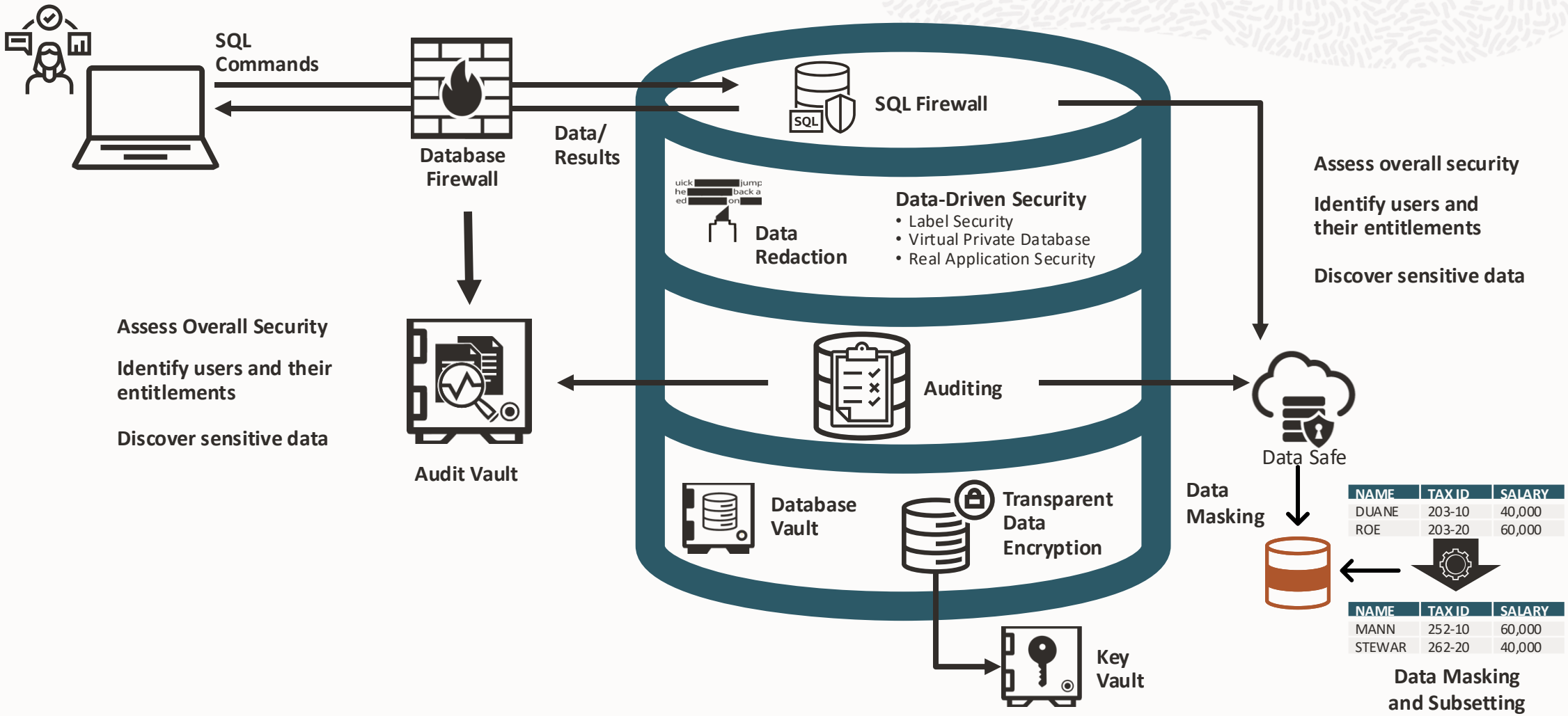
Oracle offers enterprise-level security features missing in PostgreSQL

- In-transit Data Integrity with chained checksums
- Data Redaction hides sensitive data
- Data Masking and sub-setting mask cloned production data
- Database Vault prevents insider threats, data leakage, privileged access
- Label Security protects data based on sensitivity
- Lockdown profiles prevent specific operations at the database (PDB) level
- Transparent Data Encryption (Cybertec sells a variant of TDE, but there’s a performance impact)
- Database Security Assessment Tool (DBSAT)
- Audit Vault & Database Firewall
- Native SQL Firewall (23ai)
- Privilege Analysis
- Key Vault
- Data Safe





Oracle's Maximum Security Architecture



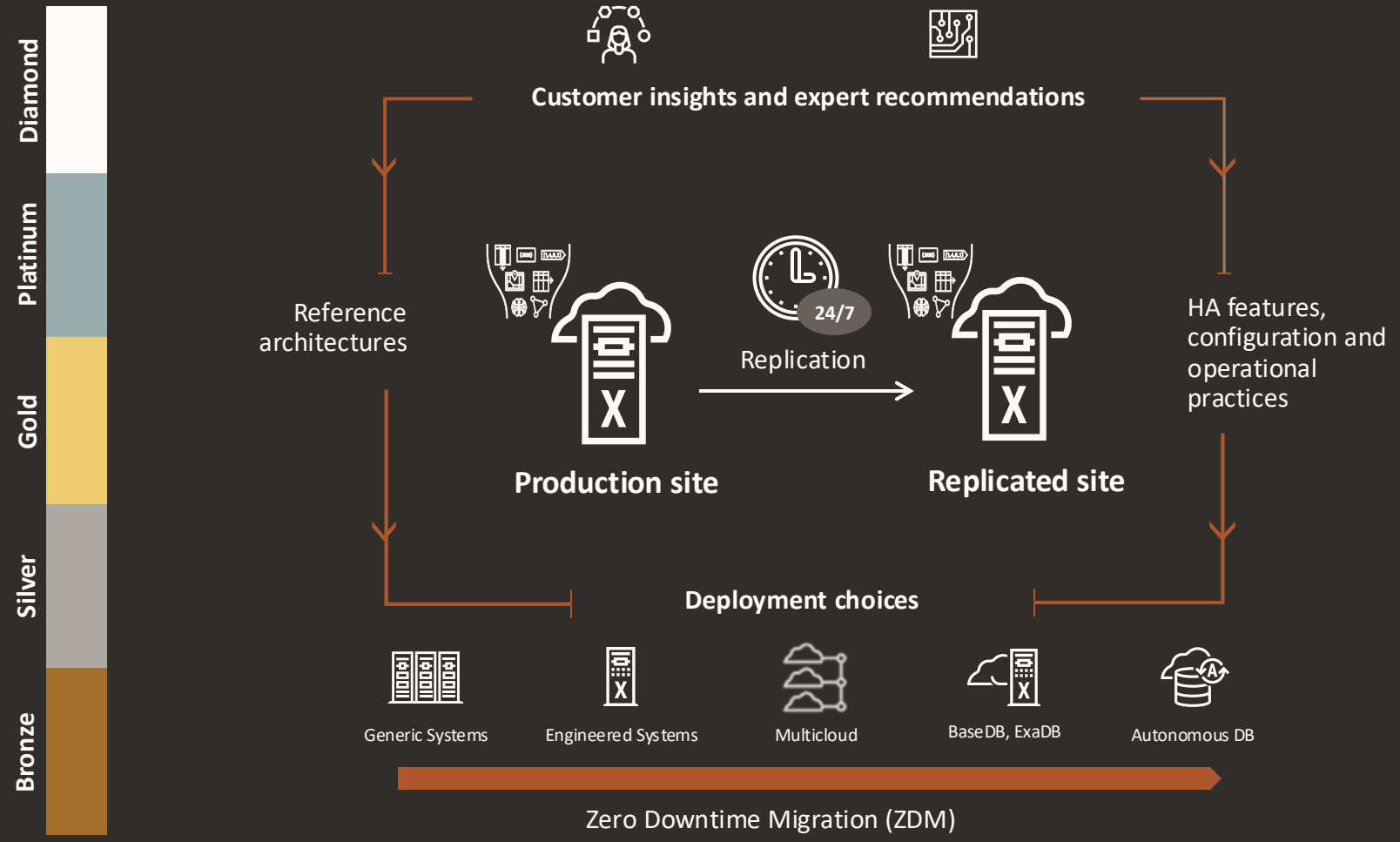
Availability

Don’t Compromise on Availability

Oracle has some unique high-availability features

- Active-active high availability with Real Application Clusters and Sharding
- Flashback technologies for faster data recovery
- Fast Application Notification and session draining
- Online maintenance:
  - At the infrastructure level with Real Application Clusters and Exadata
  - At the database level with DBMS_ROLLING, including major upgrades
  - At the schema level with Online Operations, DBMS_REDEFINITION, and Edition-Based Redefinition
- Fast incremental backup at the block level
- Application Continuity
- Exadata’s unique protection features

Next Gen Maximum Availability Architecture (MAA)



High performance

Resource Management · Database In-Memory · True Cache

Continuous availability

Application Continuity · Online Redefinition · Edition-based Redefinition

Data protection

Flashback · RMAN · ZDLRA+ ZRCV

Active replication

Active Data Guard · Full Stack DR · GoldenGate

Scale out & Lifecycle

RAC · Globally Distributed Database · FPP · Real Application Testing

Next-Gen MAA Reference Architectures

Availability service levels for the next generation of Oracle Database

Bronze	Silver	Gold	Platinum	Diamond (NEW)
Dev, test, prod	**Prod/departmental**	**Business critical**	**Mission critical**	**Extreme availability**
Single instance DB Restartable Backup/restore	**Bronze +** Database HA with RAC or Local Data Guard Client failover HA best practices Application Continuity (optional)	**Silver with RAC +** DB replication with (Active) Data Guard with automatic failover Client failover DR best practices	**Gold with** Exadata *and either option:* Option 1: GoldenGate with Oracle Database 19c ***OR*** Option 2: (Active) Data Guard with Oracle AI Database 26ai	**Configuration** GoldenGate 26ai replicas, *each running*: Oracle AI Database 26ai + RAC on Exadata + (Active) Data Guard
Recoverable local failure: Minutes to hour Disasters: Hours to days RPO < 15 min	Recoverable local failure: seconds to minutes Disasters: Hours to days RPO < 15 min	Recoverable local failure: Less than 60 seconds Disasters: < 5 min RPO = zero or near zero	Recoverable local failure: Less than 20 seconds Disasters: < 30 secs RPO = zero or near zero	Recoverable local failures: Less than 10 seconds Disasters zero to 10 secs RPO = zero or near zero

Compatibility

PostgreSQL Compatibility has Challenges

Major versions make complex changes, so the contents of the data directory cannot be maintained in a backward compatible way. A dump/reload of the database or use of the pg_upgrade application is required for major upgrades.

Source: https://www.postgresql.org/support/versioning/

- Extension upgrades can be problematic (2)
- Replicas require identical OS package versions (3)
- Different vendors offer solutions for the same problems (e.g., backups) (4)

(1) https://www.endpointdev.com/blog/2015/07/how-fast-is-pgupgrade-anyway/
(2) https://repost.aws/knowledge-center/rds-postgresql-upgrade-postgis
(3) https://stackoverflow.com/questions/68204965/is-it-a-problem-to-mix-different-os-versions-in-a-postgresql-cluster
(4) https://wiki.postgresql.org/wiki/Ecosystem:Backup

Oracle has an Amazing Backward Compatibility

```
SQL> set transaction use rollback segment my_old_app_was_using_in_8i;

Transaction set.

SQL>
```

- Every deprecation and desupport goes through deep scrutiny
- Most commands and tools work across releases
- The `compatibility` parameter allows downgrading the database in case of problems
- Exadata streamlines upgrading the whole stack (firmware, OS, binaries)

Observability

Oracle Database has Superior Observability and Instrumentation

10x more wait events (~2300 vs ~230)

50x more performance views

Automatic Workload Repository
vs. pg_profile

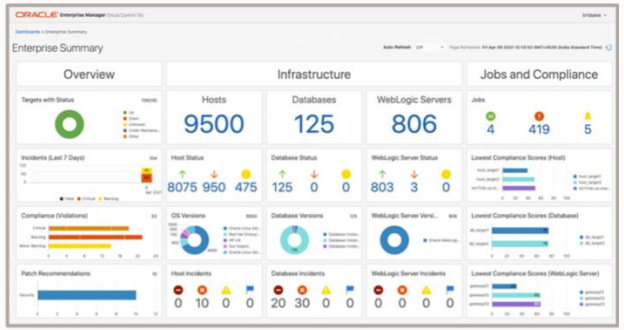
Automatic Database Diagnostic Monitor

Active Session History

Traces and Events
(e.g. 10053 and 10046)

Real-Time SQL and Operations Monitoring

Enterprise Manager



Performance

Oracle Database has Outstanding Performance Capabilities

Postgres architectural limits slow down the workload

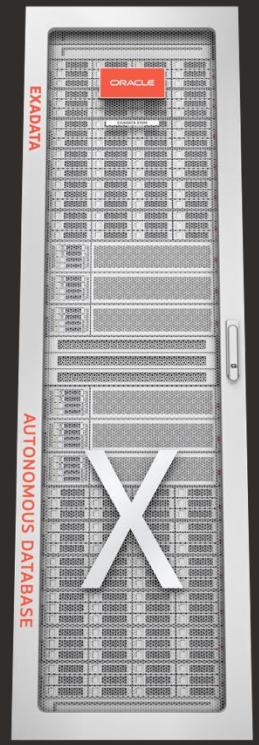
- Vacuum Management
- XID Wraparound
- Write amplification
- Prepared statements can be problematic (1)
- No asynchronous I/O

The XID wraparound issue requires careful monitoring and, in some cases, throttling the workload to avoid encountering it.

Oracle advanced performance capabilities

- In-memory
- Adaptive Cursor Sharing
- Adaptive Statistics
- Many performance advisors
- SQL Plan Management
- Automatic Indexing
- SQL Performance Analyzer
- Superior partitioning and materialized views
- Real-time statistics
- Autonomous Database's unique features (Automatic Partitioning and MVs)
- …

Exadata performance features



(1) https://pganalyze.com/blog/5mins-postgres-high-memory-usage-prepared-statements

Scalability

Choose Performance, Consistency, and Scalability

PostgreSQL cannot scale out writes without extensions or forks.

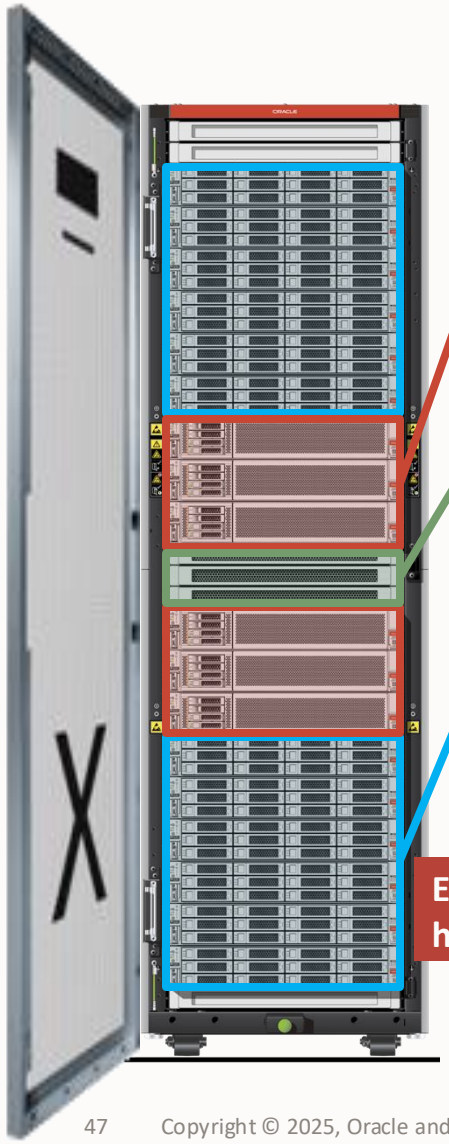
- Consistent scale-out requires synchronous replication that increases throughput but also latency.
- Some scalable forks are either good at OLTP or OLAP but not both. (1)
- Some scalable, multi-purpose forks are not consistent across shards (2)

Oracle Real Application Clusters and Exadata are the foundation for extreme performance.

- Only Real Application Clusters delivers **transaction performance and scalability together.**
- Oracle Exadata pushes performance and scalability to the extreme (<17μs OLTP I/O latency).
- Oracle Globally Distributed Database unlocks inter-cluster horizontal scalability.

(1) https://docs.yugabyte.com/preview/faq/general/#when-is-yugabytedb-not-a-good-fit
(2) https://dev.to/yugabyte/citus-is-not-acid-but-eventually-consistent-3711

Exadata X10M: Extreme Performance Scale-out Database Platform



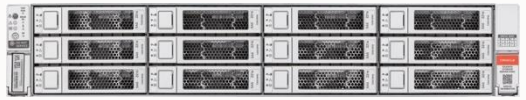
- Scale-Out 2 Socket Database Servers
  - Latest 96-core AMD EPYC™ CPUs
  - Up to 3 TB Memory
- Ultra-fast 100Gb/s RDMA over Converged Ethernet (RoCE) Internal Fabric
- Scale-Out ***intelligent*** 2-Socket Storage Servers
  - Latest 32-core AMD EPYC™ CPUs
  - 1.25 TB Exadata RDMA Memory (XRMEM) per storage server
  - Three tiers of storage: XRMEM, NVMe Flash, HDD

Each rack has up to...

Compute Capacity	Storage Capacity
2,880 DB Cores	1088 Storage Cores
33 TB Memory	21.25 TB XRMEM
	462.4 TB NVMe Flash
	2 PB Raw Flash Capacity
	4.2 PB Raw Disk Capacity

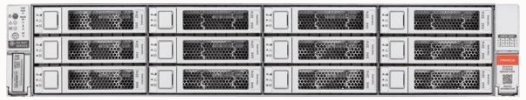
Database Server

High-Capacity (HC) Storage



Extreme Flash (EF) Storage

Extended (XT) Storage



Innovation

PostgreSQL Core New Features (v11 to v17[(1)])

PG	Feature	Oracle	1997 1999 2001 2003 2005 2007 2009 2011 2013 2015 2017 2019 2021 2023 1998 2000 2002 2004 2006 2008 2010 2012 2014 2016 2018 2020 2022 2024
	Incremental backups	8	27 years
	Logon triggers	8i	26 years
17	Other JSON functions like JSON_SERIALIZE	19c	5
	Other JSON functions like JSON_SCALAR	21c	3
	MERGE and SPLIT PARTITIONS	8i	26 years
	JSON_ARRAY(), JSON_ARRAYAGG(), and IS JSON	18c	5
16	Client load balancing	8i	25 years
	I/O statistics view	8	26 years
	Backup compression	10g	19 years
15	MERGE statement	9i	21 years
	Advanced REGEXP functions	10g	19 years
	LOB compression	11g	15 years
14	OUT parameters in stored procedures		?
	Extended (multi-column) statistics	11g	15 years
12	Online Index Rebuild	8i	21 years
11	Transactions inside stored procedures	8i	20 years

(1) https://pgpedia.info/postgresql-versions/postgresql-17.html https://pgpedia.info/postgresql-versions/postgresql-16.html
https://pgpedia.info/postgresql-versions/postgresql-15.html https://pgpedia.info/postgresql-versions/postgresql-14.html
https://pgpedia.info/postgresql-versions/postgresql-12.html https://pgpedia.info/postgresql-versions/postgresql-11.html

Planned PostgreSQL 18 features

- Transparent Column Encryption
- Asynch I/O (developer preview in 17)
- Statistic export/import
- Logical Replication DDL support
- Logical Replication Conflict D&R
- Row Pattern Recognition
- Wire Protocol Compression
- Better Partition Pruning
- Parallel Vacuum and Index Creation

In yellow: Long-time available features in Oracle

Jonathan Katz Wishlist

- System versioned tables
- Schema variables
- Autonomous Transactions
- Bulk functions in PL/pgSQL
- Add online operations/DDLs
- Bitemporal queries
- Parallel recovery
- Native global temporary tables

Source: https://jkatz05.com/post/postgres/postgresql-2024/

In yellow: Long-time available features in Oracle

Some Unique Oracle pre-23ai Features

- APEX
- Advanced Queueing
- Blockchain and immutable tables
- In-Database Machine Learning
- REST Data services
- Polymorphic functions
- In-Memory
- Edition-Based Redefinition
- Application Containers
- SQL macros
- Result Cache
- Analytic Views

Oracle Database 23ai – 300+ New Features

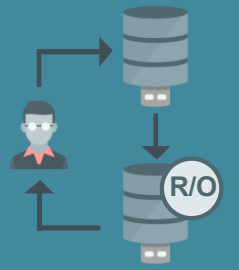
Data Use Case Domains

Boolean Datatype

Oracle Database

23ai

Bring AI to your data

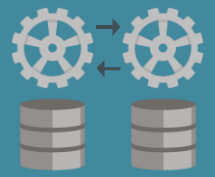


Readable PDB Standby

Property Graphs

Real-time SQL Plan Management

JSON Schema

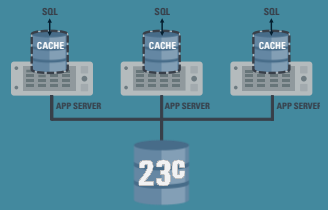


Microservice Sagas

JSON / Relational Duality

AI Vector Search

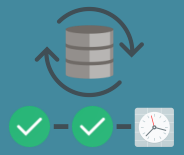
True Cache



SQL Firewall

Priority Transactions

Globally Distributed Database



Rolling Patching

JS Stored Procedures

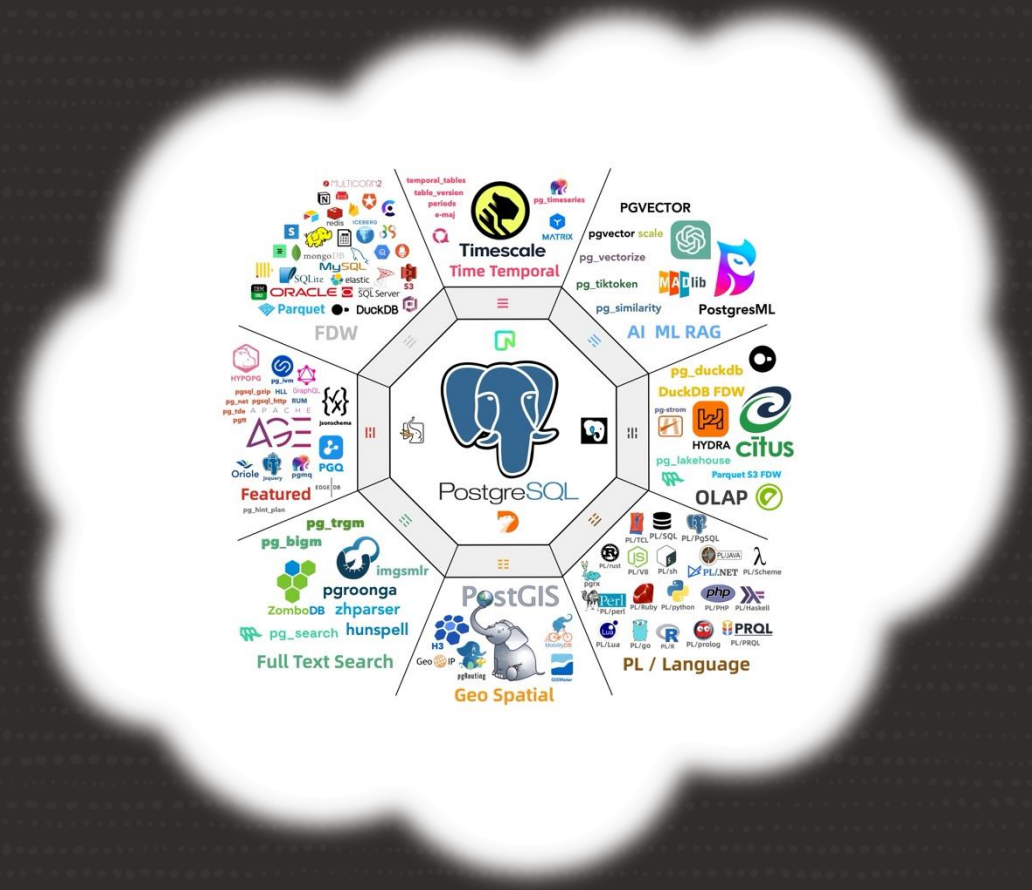
Developer Role

Shrink Tablespace

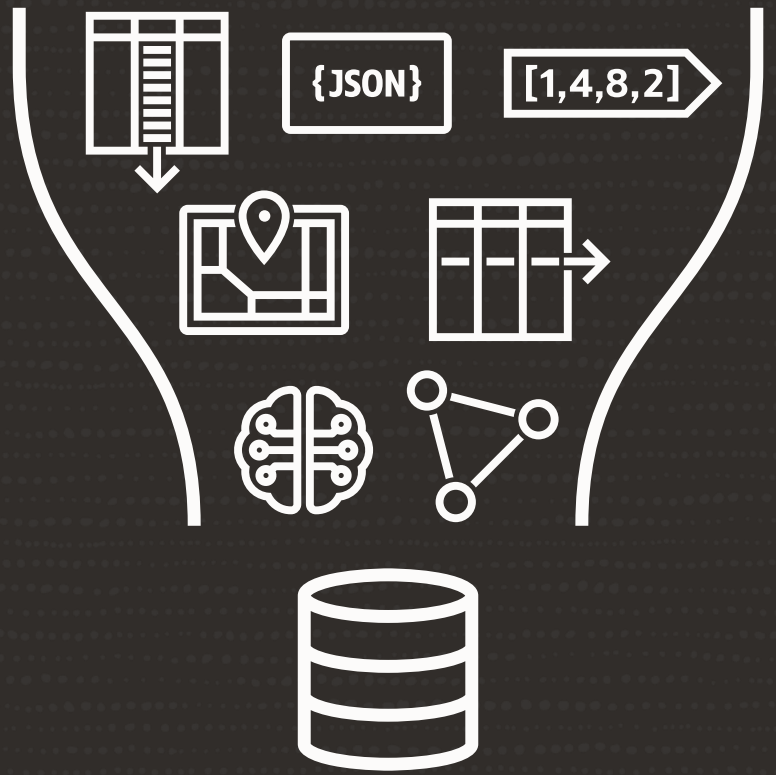
Schema Privileges

Some words about Total Cost of Ownership

Spending time on Integration?



Focus on Business!



Thank you


ORACLE

Our mission is to help people see data in new ways, discover insights, unlock endless possibilities.