



АДМИНИСТРИРОВАНИЕ POSTGRESQL

глазами Oracle DBA



Задачи администратора

- **доступность базы**
- **производительность**
- **безопасность**

О чем будем говорить

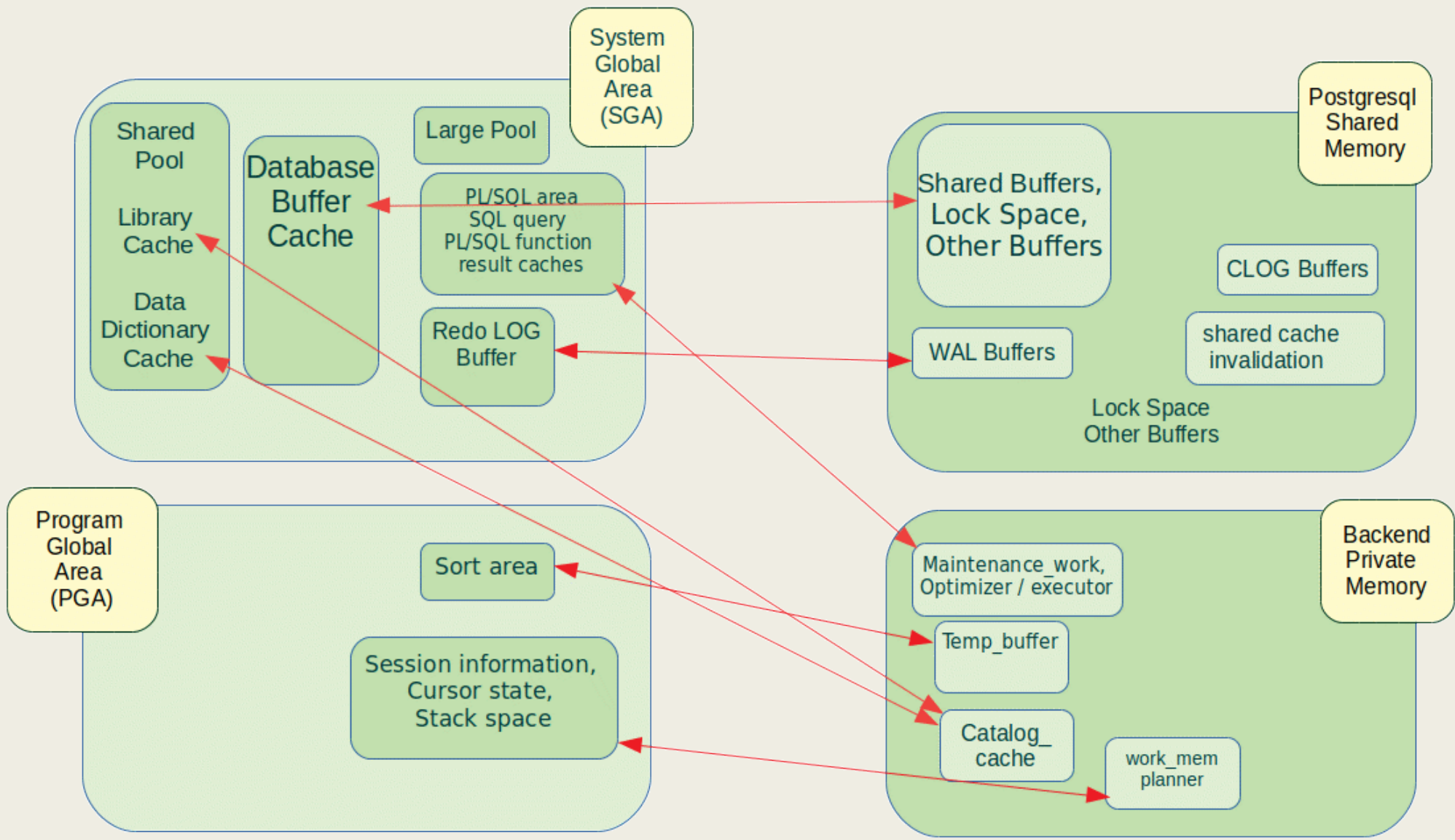
- **особенности архитектуры**
- **инструмент для работы**

Память

Фрагмент памяти	Oracle	PostgreSQL
Разделяемая память	SGA system global area	PSM postgres shared memory
	SGA buffer cache	PSM shared_buffers , CLOG Buffers, Lock Space, Other Buffers
	SGA log buffer	PSM WAL Buffers
	SGA shared pool - library cache, data dictionary cache	PBM Catalog cache
	SGA shared pool - PL/SQL area	PBM Maintenance_work, Optimizer / executor
	SGA shared pool - SQL query and PL/SQL function result caches	PBM Maintenance_work, Optimizer / executor
Неразделяемая память	PGA Process Global Area	PBM Privat Backend Memory
	Sort area	PBM Temp_buffer
	Session information, Cursor state, Stack space	PBM Maintenance_work, Work_mem

ORACLE

POSTGRESQL



Процессы

Процесс	Oracle	PostgreSQL
Пишет модифицированные данные из буферного кэша на диск	DBWn	Bgwriter
Писатель журнальных файлов	LGWR	Wal writer
Обрабатывает контрольную точку	CKPT	Checkpointner
Процесс монитора процессов	PMON	Postmaster
Процесс монитора системы	SMON	Autovacuum, Startup
Архивирует заполненные файлы журналов повторного выполнения	ARCn	Wal archiver
Диспетчер процессов	D000	Backend
Слушает порт	Listener 1521	Postmaster 5432

Файлы на диске

Процесс	Oracle	PostgreSQL
Файлы данных	+	+
Файл undo	+	-
Управляющие файлы	+	+
Временные файлы	+	+
Файлы паролей	+	+
Файлы параметров	+	+
Файлы логов	+	+

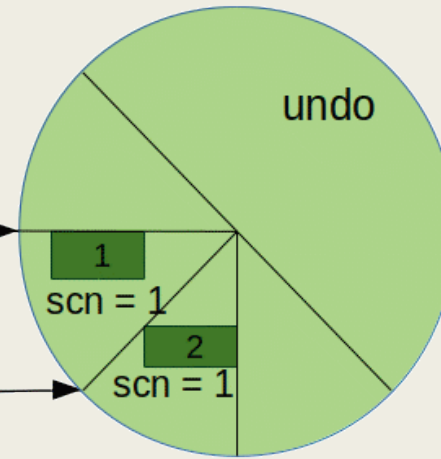
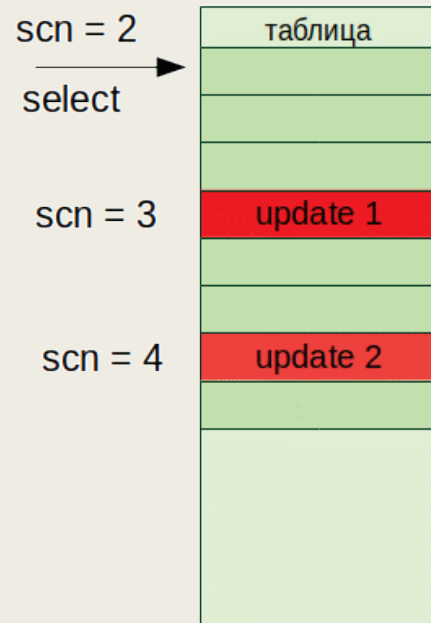
Версионность

Postgresql

table

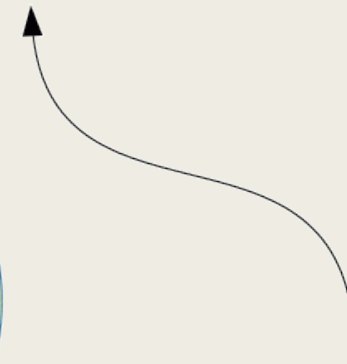
	xid	status	xmin	xmax	infomask	hhu	hot	ctid	data
(0,1)	1	normal	100	102	1	t		(0,2)	1,AAA
(0,2)	2	normal	102	0	1		t	(0,2)	1,XXX

Oracle



xlog

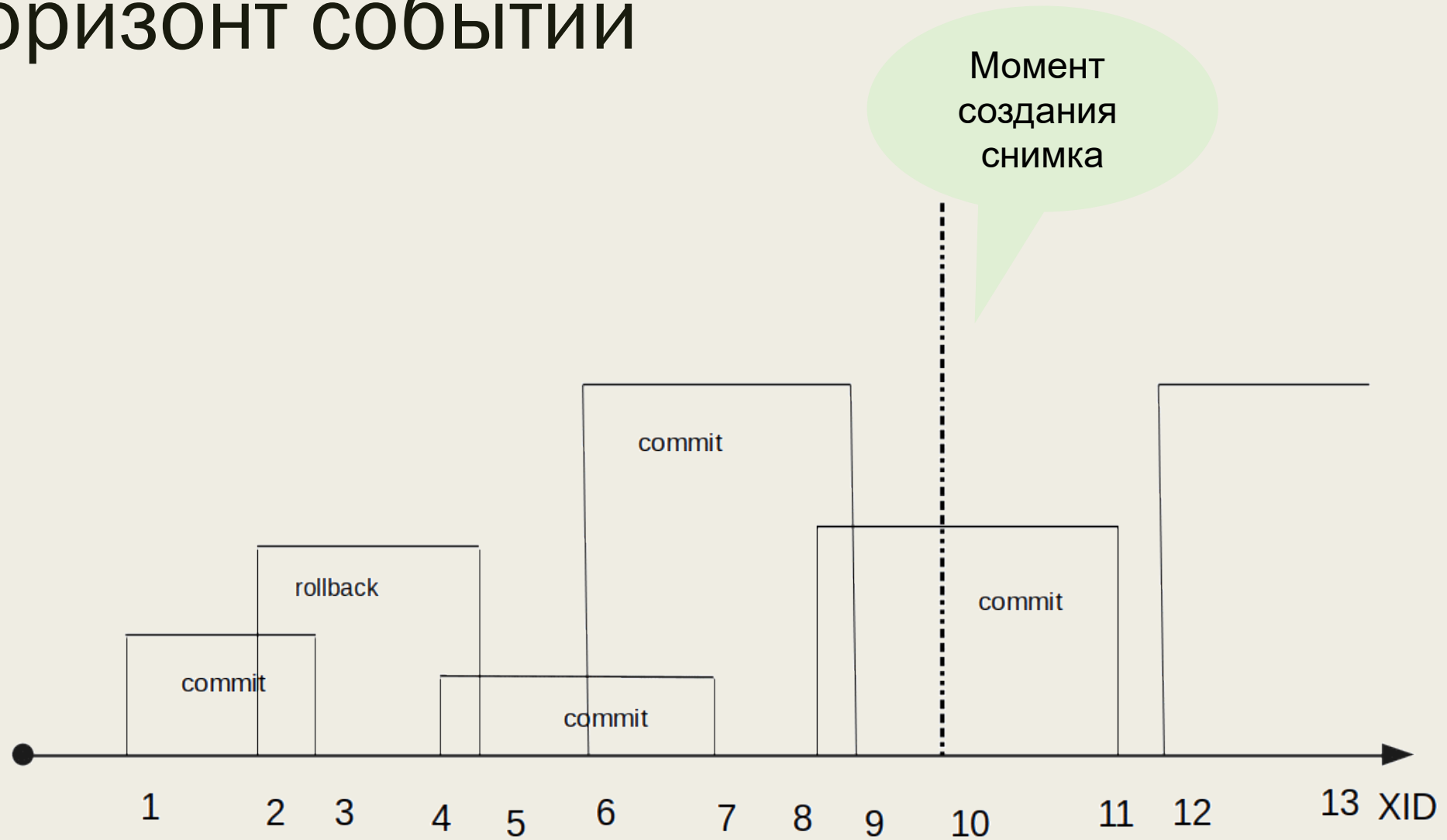
	committed	aborted
100	t	
101		t
102	t	



Вакуум

- Внутри страничная очистка
- Vacuum Autovacuum
- Vacuum Full
- Pg_repack (extention)

Горизонт событий



Мониторинг

- pg_stat_*
- pg_locks
- pg_qualstats
- pg_statio_*
- Как часто снимать статистику
- Куда писать, где хранить
- Как агрегировать
- Куда выводить результат

Инструменты

pg_profile

pgbender

Zabbix

EMS

pgAdmin

telegram

Postgres profile report (StartID: 1, EndID: 7)

Report interval: 2019-02-04 20:41:10+03 - 2019-02-04 22:30:41+03

Report sections

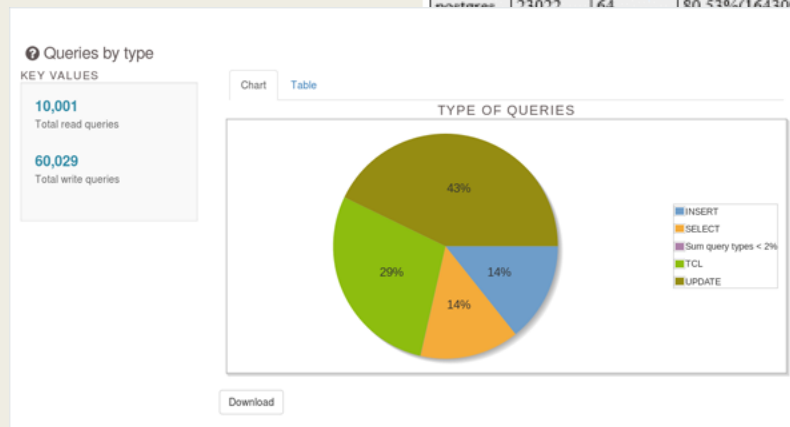
- [Cluster statistics](#)
 - [Databases stats](#)
 - [Statements stats by database](#)
 - [Cluster stats](#)
- [SQL Query stats](#)
 - [Top SQL by elapsed time](#)
 - [Top SQL by executions](#)
 - [Top SQL by I/O wait time](#)
 - [Top SQL by gets](#)
 - [Top SQL by temp usage](#)
 - [Complete List of SQL Text](#)
- [Schema objects stats](#)
 - [Most scanned tables](#)
 - [Top DML tables](#)
 - [Top Delete/Update tables with vacuum run count](#)
 - [Top growing tables](#)
 - [Top growing indexes](#)
 - [Unused indexes](#)
- [I/O Schema objects stats](#)
 - [Top tables by I/O](#)
 - [Top indexes by I/O](#)
- [User function stats](#)
 - [Top functions by total time](#)
 - [Top functions by executions](#)
- [Vacuum related stats](#)
 - [Tables ordered by dead tuples ratio](#)
 - [Tables ordered by modified tuples ratio](#)

Cluster statistics

Databases stats

Database	Commits	Rollbacks	BlkHit%(read/hit)	Tup Ret/Fet	Tup Ins	Tup Del	Temp Size(Files)	Growth	Deadlocks
postgres	73022	64	80.5384(164300/679754)	5535525/188697	772430	20387	5376 kB(8)	103 MB	0

l gets	Shared dirtied	Local dirtied	Work_r (blk)	Work_w (blk)	Local_r (blk)	Local_w (blk)	Statements
--------	----------------	---------------	--------------	--------------	---------------	---------------	------------



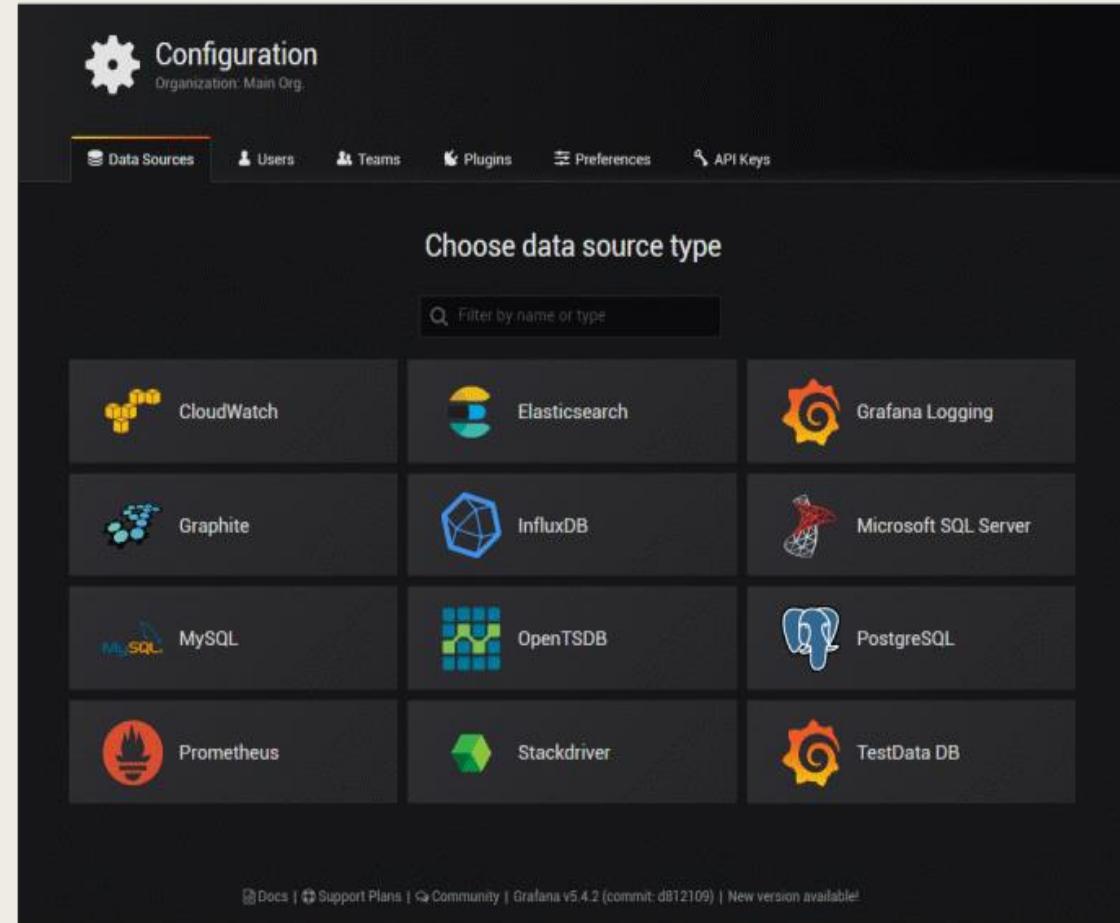
Инструменты postgresql

PoWA PostgreSQL Workload Analyzer

grafana

graphite

brubeck



Восстановление

POSTGRESQL

pg_dump

pg_dumpall

pg_basebackup

barman

pg_probackup

ORACLE

exp imp

data pump

rman

flashback

Отказоустойчивость

POSTGRESQL

patroni
replication
...

ORACLE

RAC
replication
ASM

Что радует

- гибкий SQL
- КОМПИЛЯТОРЫ ДЛЯ КУЧИ ЯЗЫКОВ
- MVCC
- ТИПЫ
- гибкие уровни изоляции
- ИНДЕКСЫ
- ...

```
with s as (  
    select object_id  
    from acme_state_tmp  
)  
,  
upd as (  
    update acme_obj_value  
    set date_value = least(l_dt, d.date_value)  
    from s  
    where acme_obj_value.object_id = s.object_id  
    returning acme_obj_value.object_id  
)  
insert into acme_obj_value(id, object_id, date_value)  
select nextval('acme_param_sequence'), s.object_id, l_dt  
from s  
where s.object_id not in (select object_id from upd)
```


Итого

Oracle	PostgreSql
организация многоверсионности через UNDO	организация многоверсионности через хранение версий в блоке
позволяет настроить shared servers, для обслуживания многих соединений одним процессом	каждое клиентское соединение обслуживается отдельным серверным процессом, если не использовать pool
может работать с raw-устройствами и использовать IO минуя кэш ядра	двойное кеширование работает с FS
коробка	открытый код, расширяемость
\$	-