POSTGRESQL is an open-source, full-featured relational database. This presentation covers advanced administration topics.
1. Installation
2. Configuration
3. Maintenance
4. Monitoring
5. Recovery
1. Installation

• Click-through Installers
  • MS Windows
  • Linux
  • OS X

• Ports
  • RPM
  • DEB
  • PKG
  • other packages

• Source
  • obtaining
  • build options
  • installing
$ initdb /u/pgsql/data
The files belonging to this database system will be owned by user "postgres".
This user must also own the server process.

The database cluster will be initialized with locale "en_US.UTF-8".
The default database encoding has accordingly been set to "UTF8".
The default text search configuration will be set to "english".

Data page checksums are disabled.

fixing permissions on existing directory /u/pgsql/data ... ok
creating subdirectories ... ok
selecting default max_connections ... 100
selecting default shared_buffers ... 128MB
selecting dynamic shared memory implementation ... posix
creating configuration files ... ok
running bootstrap script ... ok
performing post-bootstrap initialization ... ok
syncing data to disk ... ok

WARNING: enabling "trust" authentication for local connections
You can change this by editing pg_hba.conf or using the option -A, or
--auth-local and --auth-host, the next time you run initdb.

Success. You can now start the database server using:

    pg_ctl -D /u/pgsql/data -l logfile start
$ pg_control

pg_control version number: 1002
Catalog version number: 201707211
Database system identifier: 654463619067825437
Database cluster state: shut down
pg_control last modified: Sun 15 Apr 2018 07:20:58 AM EDT
Latest checkpoint location: 0/15C09E0
Prior checkpoint location: 0/15C0708
Latest checkpoint's REDO location: 0/15C09E0
Latest checkpoint's REDO WAL file: 000000010000000000000000
Latest checkpoint's TimeLineID: 1
Latest checkpoint's PrevTimeLineID: 1
Latest checkpoint's full_page_writes: on
Latest checkpoint's NextXID: 0:555
Latest checkpoint's NextOID: 12296
Latest checkpoint's NextMultiXactId: 1
Latest checkpoint's NextMultiOffset: 0
Latest checkpoint's oldestXID: 548
Latest checkpoint's oldestXID's DB: 1
Latest checkpoint's oldestActiveXID: 0
Latest checkpoint's oldestMultiXid: 1
Latest checkpoint's oldestMulti's DB: 1
Latest checkpoint's oldestCommitTsXid: 0
Latest checkpoint's newestCommitTsXid: 0
Time of latest checkpoint: Sun 15 Apr 2018 07:20:58 AM EDT
Fake LSN counter for unlogged rels: 0/1
Minimum recovery ending location: 0/0
Min recovery ending loc's timeline: 0
Backup start location: 0/0
Backup end location: 0/0
End-of-backup record required: no
wal_level setting: replica
wal_log_hints setting: off
max_connections setting: 100
max_worker_processes setting: 8
System Architecture
Session Creation

The diagram illustrates the creation of a session in a database system. It shows the process starting with a `postmaster` process, which forks into a `postgres` process. The `postgres` process then runs multiple instances of itself, each with its own program (Text), data, shared memory, and stack.

- The `postmaster` process forks into a `postgres` process.
- The `postgres` process creates multiple instances of itself, each with its own environment.
Starting Postmaster

- manually
- `pg_ctl start`
- on boot
Stopping Postmaster

2018-04-15 07:23:47.317 EDT [12055] LOG: received fast shutdown request
2018-04-15 07:23:47.318 EDT [12055] LOG: aborting any active transactions
2018-04-15 07:23:47.327 EDT [12055] LOG: database system is shut down

- manually
- pg_ctl stop
- on shutdown
Connections

- local — unix domain socket
- host — TCP/IP, both SSL or non-SSL
- hostssl — only SSL
- hostnossl — never SSL
Authentication

- trust
- reject
- passwords
  - scram-sha-256
  - md5
  - password (cleartext)
- local authentication
  - socket permissions
  - ’peer’ socket user name passing
  - host ident using local identd
• remote authentication
  • host ident using pg_ident.conf
  • kerberos
    • gss
    • sspi
  • pam
  • ldap
  • radius
  • cert
Access

- hostname and network mask
- database name
- role name (user or group)
- filename or list of databases, role
- IPv6
# TYPE    DATABASE    USER    ADDRESS    METHOD

# "local" is for Unix domain socket connections only
local all all all trust

# IPv4 local connections:
host all all 127.0.0.1/32 trust

# IPv6 local connections:
host all all ::1/128 trust

# Allow replication connections from localhost, by a user with the
# replication privilege.
#local replication postgres trust
#host replication postgres 127.0.0.1/32 trust
#host replication postgres ::1/128 trust
# TYPE DATABASE USER ADDRESS METHOD

# "local" is for Unix domain socket connections only
local all all all trust

# IPv4 local connections:
host all all 127.0.0.1/32 trust

# IPv6 local connections:
host all all ::1/128 trust

# disable connections from the gateway machine
host all all 192.168.1.254/32 reject

# enable local network
host all all 192.168.1.0/24 scram-sha-256

# require SSL for external connections, but do not allow the superuser
hostssl all postgres 0.0.0.0/0 reject
hostssl all all 0.0.0.0/0 scram-sha-256
Permissions

- Host connection permissions
- Role permissions
  - create roles
  - create databases
  - table permissions
- Database management
  - template1 customization
  - system tables
  - disk space computations
Data Directory

```
$ ls -CF
base/
  pg_ident.conf  pg_serial/
  pg_tblspc/
  postmaster.opts
global/
  pg_logical/
  pg_snapshots/
  pg_twophase/
  postgresql.conf
pg_commit_ts/
  pg_multixact/
  pg_stat/
  PG_VERSION
  postgresql.auto.conf
pg_dynshmem/
  pg_notify/
  pg_stat_tmp/
  pg_wal/
  postmaster.opts
pg_hba.conf
  pg_replslot/
  pg_subtrans/
  pg_xact/
```
Database Directories

$ ls -CF global/
1136 1214_fsm 1261_vm 2671 2846 2967 6000_vm
1136_fsm 1214_vm 1262 2672 2846_vm 3592 6001
1136_vm 1232 1262_fsm 2676 2847 3592_vm 6002
1137 1233 1262_vm 2677 2964 3593 pg_control
1213 1260 2396 2694 2964_vm 4060 pg_filenode.map
1213_fsm 1260_fsm 2396_fsm 2695 2965 4060_vm pg_internal.init
1213_vm 1260_vm 2396_vm 2697 2966 4061
1214 1261 2397 2698 2966_vm 6000

$ ls -CF base/
1/ 12406/ 12407/ 16384/

$ ls -CF base/16384
112 1249_fsm 2606_vm 2652 2699 3081 3598_vm
113 1249_vm 2607 2653 2701 3085 3599
12242 1255 2607_fsm 2654 2702 3118 3600
12242_fsm 1255_fsm 2607_vm 2655 2703 3118_vm 3600_fsm
12242_vm 1255_vm 2608 2656 2704 3119 3600_vm
12244 1259 2608_fsm 2657 2753 3164 3601
12246 1259_fsm 2608_vm 2658 2753_fsm 3256 3601_fsm

...
$ ls -CF pg_wal/
00000001000000000000000001 archive_status/

$ ls -CF pg_xact/
0000
Configuration Directories

$ ls -CF share/
conversion_create.sql  postgres.bki  snowball_create.sql
extension/  postgres.description  sql_features.txt
information_schema.sql  postgresql.conf.sample  system_views.sql
pg_hba.conf.sample  postgres.shdescription
pg_ident.conf.sample  psqlrc.sample
pg_service.conf.sample  recovery.conf.sample
timezone/
timezonesets/
tsearch_data/
2. Configuration

https://www.flickr.com/photos/mwichary/
# PostgreSQL configuration file

# This file consists of lines of the form:
#
#   name = value
#
# (The "=" is optional.) Whitespace may be used. Comments are introduced with
# "#" anywhere on a line. The complete list of parameter names and allowed
# values can be found in the PostgreSQL documentation.
#
# The commented-out settings shown in this file represent the default values.
# Re-commenting a setting is NOT sufficient to revert it to the default value;
# you need to reload the server.
This file is read on server startup and when the server receives a SIGHUP signal. If you edit the file on a running system, you have to SIGHUP the server for the changes to take effect, run "pg_ctl reload", or execute "SELECT pg_reload_conf()". Some parameters, which are marked below, require a server shutdown and restart to take effect.

Any parameter can also be given as a command-line option to the server, e.g., "postgres -c log_connections=on". Some parameters can be changed at run time with the "SET" SQL command.

Memory units: kB = kilobytes Time units: ms = milliseconds
MB = megabytes s = seconds
GB = gigabytes min = minutes
TB = terabytes h = hours
d = days
# The default values of these variables are driven from the -D command-line option or PGDATA environment variable, represented here as ConfigDir.

#data_directory = 'ConfigDir'
    # use data in another directory
    # (change requires restart)

#hba_file = 'ConfigDir/pg_hba.conf'
    # host-based authentication file
    # (change requires restart)

#ident_file = 'ConfigDir/pg_ident.conf'
    # ident configuration file
    # (change requires restart)

# If external_pid_file is not explicitly set, no extra PID file is written.
#external_pid_file = ''
    # write an extra PID file
    # (change requires restart)
#listen_addresses = 'localhost'

# port = 5432
max_connections = 100
#superuser_reserved_connections = 3
#unix_socket_directories = '/tmp'

#unix_socket_group = ''
#unix_socket_permissions = 0777

#bonjour = off

#bonjour_name = ''
# authentication_timeout = 1min  # 1s-600s
# ssl = off
# ssl_ciphers = 'HIGH:MEDIUM:+3DES:!aNULL'  # allowed SSL ciphers
# ssl_prefer_server_ciphers = on
# ssl_ecdh_curve = 'prime256v1'
# ssl_dh_params_file = ''
# ssl_cert_file = 'server.crt'
# ssl_key_file = 'server.key'
# ssl_ca_file = ''
# ssl_crl_file = ''
# password_encryption = md5  # md5 or scram-sha-256
# db_user_namespace = off
# row_security = on

# GSSAPI using Kerberos
# krb_server_keyfile = ''
# krb_caseins_users = off
TCP/IP Control

#tcp_keepalives_idle = 0
# TCP_KEEPIDLE, in seconds;
# 0 selects the system default

#tcp_keepalives_interval = 0
# TCP_KEEPINTVL, in seconds;
# 0 selects the system default

#tcp_keepalives_count = 0
# TCP_KEEPCNT;
shared_buffers = 128MB  # min 128kB  # (change requires restart)
巨大页面 = try  # on, off, or try  # (change requires restart)
temp_buffers = 8MB  # min 800kB
max_prepared_transactions = 0  # zero disables the feature  # (change requires restart)

# Caution: it is not advisable to set max_prepared_transactions nonzero unless
# you actively intend to use prepared transactions.
work_mem = 4MB  # min 64kB
maintenance_work_mem = 64MB  # min 1MB
replacement_sort_tuples = 150000  # limits use of replacement selection sort
autovacuum_work_mem = -1  # min 1MB, or -1 to use maintenance_work_mem
max_stack_depth = 2MB  # min 100kB
dynamic_shared_memory_type = posix  # the default is the first option
# supported by the operating system:
# posix
# sysv
# windows
# mmap
# use none to disable dynamic shared memory  # (change requires restart)
Memory Usage (Continued)

Query and Checkpoint Operations

PostgreSQL Shared Buffer Cache

Kernel Disk Buffer Cache

Transaction Durability

Write-Ahead Log

fsync

fsync

Disk Blocks
# Sizing Shared Memory

The diagram illustrates the memory footprint of PostgreSQL sessions and their corresponding buffer caches. The shared memory is divided into several sections:

- **Kernel**: Directly managed by the operating system and not shared.
- **Kernel Disk Buffer Cache**: A portion of shared memory used for caching disk blocks.
- **Shared Buffer Cache (shared_buffers)**: Part of the shared memory used for caching database pages.
- **Postgres Session (work_mem)**: Memory allocated per database session for working sets.

The diagram also shows the flow of memory pages between RAM and swap space, indicating

- **Page Out**: Pages are moved from RAM to swap space when they exceed the available RAM capacity.
- **Page In (bad)**: Pages are loaded back into RAM, but the process is inefficient, often leading to increased page faults.

The visual representation helps in understanding the memory management strategies and potential bottlenecks in PostgreSQL operations.
# - Disk -

# temp_file_limit = -1

# limits per-process temp file space
# in kB, or -1 for no limit

# - Kernel Resource Usage -

# max_files_per_process = 1000

# min 25
# (change requires restart)

# shared_preload_libraries = ''

# (change requires restart)
# - Cost-Based Vacuum Delay -

```sql
#vacuum_cost_delay = 0  # 0-100 milliseconds
#vacuum_cost_page_hit = 1  # 0-10000 credits
#vacuum_cost_page_miss = 10  # 0-10000 credits
#vacuum_cost_page_dirty = 20  # 0-10000 credits
#vacuum_cost_limit = 200  # 1-10000 credits
```

# - Background Writer -

```sql
#bgwriter_delay = 200ms  # 10-10000ms between rounds
#bgwriter_lru_maxpages = 100  # 0-1000 max buffers written/round
#bgwriter_lru_multiplier = 2.0  # 0-10.0 multiplier on buffers scanned/round
#bgwriter_flush_after = 512kB  # measured in pages, 0 disables
```

# - Asynchronous Behavior -

```sql
#effective_io_concurrency = 1  # 1-1000; 0 disables prefetching
#max_worker_processes = 8  # (change requires restart)
#max_parallel_workers_per_gather = 2  # taken from max_parallel_workers
#max_parallel_workers = 8  # maximum number of max_worker_processes that can be used in parallel queries
#old_snapshot_threshold = -1  # 1min-60d; -1 disables; 0 is immediate
```
Write-Ahead Log (WAL)

#wal_level = replica
# minimal, replica, or logical
# (change requires restart)
# flush data to disk for crash safety
# (turning this off can cause
# unrecoverable data corruption)
# synchronous_commit = on
# synchronization level;
# off, local, remote_write, remote_apply, or
# the default is the first option
# supported by the operating system:
# open_datsasync
# fdatasync (default on Linux)
# fsync
# fsync_writethrough
# open_sync
# recover from partial page writes
# enable compression of full-page writes
# also do full page writes of non-critical
# (change requires restart)
# min 32kB, -1 sets based on shared_buffers
# (change requires restart)
# 1-10000 milliseconds
# measured in pages, 0 disables
# range 0-100000, in microseconds

#wal_level = replica

#fsync = on

#synchronous_commit = on

#wal_sync_method = fsync
# the default is the first option supported by the operating system:
# open_datsasync
# fdatasync (default on Linux)
# fsync
# fsync_writethrough
# open_sync
# recover from partial page writes
# enable compression of full-page writes
# also do full page writes of non-critical
# (change requires restart)
# min 32kB, -1 sets based on shared_buffers
# (change requires restart)
# 1-10000 milliseconds
# measured in pages, 0 disables
# range 0-100000, in microseconds

#full_page_writes = on
# wal_compression = off
# wal_log_hints = off

#full_page_writes = on
# wal_compression = off
# wal_log_hints = off

#wal_buffers = -1
# min 32kB, -1 sets based on shared_buffers

#wal_buffers = -1
# min 32kB, -1 sets based on shared_buffers

#wal_writer_delay = 200ms
# 1-10000 milliseconds

#wal_writer_delay = 200ms
# 1-10000 milliseconds

#wal_writer_flush_after = 1MB
# measured in pages, 0 disables

#wal_writer_flush_after = 1MB
# measured in pages, 0 disables

#commit_delay = 0
# range 0-100000, in microseconds

#commit_delay = 0
# range 0-100000, in microseconds

#commit_siblings = 5
# range 1-1000

#commit_siblings = 5
# range 1-1000
Write-Ahead Logging (Continued)

Query and Checkpoint Operations

PostgreSQL Shared Buffer Cache

Transaction Durability

Write-Ahead Log

Kernel Disk Buffer Cache

fsync

fsync

Disk Blocks

Postgres Backend

Postgres Backend

Postgres Backend

Recovery
# Checkpoints and Archiving

#checkpoint_timeout = 5min # range 30s-1d
#max_wal_size = 1GB
#min_wal_size = 80MB
#checkpoint_completion_target = 0.5 # checkpoint target duration, 0.0 - 1.0
#checkpoint_flush_after = 256kB # measured in pages, 0 disables
#checkpoint_warning = 30s # 0 disables

# - Archiving -

#archive_mode = off # enables archiving; off, on, or always
# (change requires restart)
#archive_command = '' # command to use to archive a logfile segment
# placeholders: %p = path of file to archive
# %f = file name only
# e.g., 'test ! -f /mnt/server/archivedir/%f && cp
#archive_timeout = 0 # force a logfile segment switch after this
# number of seconds; 0 disables
Write-Ahead Logging (Continued)

PostgreSQL Shared Buffer Cache

Begin 1

End 1

Rotate

Write-Ahead Log
Sending Server

# Set these on the master and on any standby that will send replication data.

#max_wal_senders = 10  # max number of walsender processes  # (change requires restart)
#wal_keep_segments = 0  # in logfile segments, 16MB each; 0 disables
#wal_sender_timeout = 60s  # in milliseconds; 0 disables

#max_replication_slots = 10  # max number of replication slots  # (change requires restart)
#track_commit_timestamp = off  # collect timestamp of transaction commit  # (change requires restart)
# These settings are ignored on a standby server.

#synchronous_standby_names = '' # standby servers that provide sync rep
# method to choose sync standbys, number of sync standbys,
# and comma-separated list of application_name
# from standby(s); '*' = all

#vacuum_defer_cleanup_age = 0 # number of xacts by which cleanup is delayed
# These settings are ignored on a master server.

```
#hot_standby = on  # "off" disallows queries during recovery
# (change requires restart)
#max_standby_archive_delay = 30s  # max delay before canceling queries
# when reading WAL from archive;
# -1 allows indefinite delay
#max_standby_streaming_delay = 30s  # max delay before canceling queries
# when reading streaming WAL;
# -1 allows indefinite delay
#wal_receiver_status_interval = 10s  # send replies at least this often
# 0 disables
#hot_standby_feedback = off  # send info from standby to prevent
# query conflicts
#wal_receiver_timeout = 60s  # time that receiver waits for
# communication from master
# in milliseconds; 0 disables
#wal_retrieve_retry_interval = 5s  # time to wait before retrying to
# retrieve WAL after a failed attempt
```
# These settings are ignored on a publisher.

#max_logical_replication_workers = 4  # taken from max_worker_processes
#  # (change requires restart)
#max_sync_workers_per_subscription = 2  # taken from max_logical_replication_workers
Planner Method Tuning

#enable_bitmapscan = on
#enable_hashagg = on
#enable_hashjoin = on
#enable_indexscan = on
#enable_indexonlyscan = on
#enable_material = on
#enable_mergejoin = on
#enable_nestloop = on
#enable_seqscan = on
#enable_sort = on
#enable_tidscan = on
Planner Constants

#seq_page_cost = 1.0  # measured on an arbitrary scale
#random_page_cost = 4.0  # same scale as above
#cpu_tuple_cost = 0.01  # same scale as above
#cpu_index_tuple_cost = 0.005  # same scale as above
#cpu_operator_cost = 0.0025  # same scale as above
#parallel_tuple_cost = 0.1  # same scale as above
#parallel_setup_cost = 1000.0  # same scale as above
#min_parallel_table_scan_size = 8MB
#min_parallel_index_scan_size = 512kB
#effective_cache_size = 4GB
Planner GEQO

#geqo = on
#geqo_threshold = 12
#geqo_effort = 5
#geqo_pool_size = 0
#geqo_generations = 0
#geqo_selection_bias = 2.0
#geqo_seed = 0.0
Miscellaneous Planner Options

#default_statistics_target = 100 # range 1-10000
#constraint_exclusion = partition # on, off, or partition
#cursor_tuple_fraction = 0.1 # range 0.0-1.0
#from_collapse_limit = 8 # 1 disables collapsing of explicit JOIN clauses
#join_collapse_limit = 8
#force_parallel_mode = off
Where To Log

#log_destination = 'stderr'
# Valid values are combinations of
# stderr, csvlog, syslog, and eventlog,
# depending on platform. csvlog
# requires logging_collector to be on.

# This is used when logging to stderr:
#logging_collector = off
# Enable capturing of stderr and csvlog
# into log files. Required to be on for
# csvlogs.
# (change requires restart)

# These are only used if logging_collector is on:
#log_directory = 'log'
# directory where log files are written,
# can be absolute or relative to PGDATA

#log_filename = 'postgresql-%Y-%m-%d_%H%M%S.log'
# log file name pattern,
# can include strftime() escapes

#log_file_mode = 0600
# creation mode for log files,
# begin with 0 to use octal notation
#log_truncate_on_rotation = off

# If on, an existing log file with the same name as the new log file will be truncated rather than appended to. But such truncation only occurs on time-driven rotation, not on restarts or size-driven rotation. Default is off, meaning append to existing files in all cases.

#log_rotation_age = 1d

# Automatic rotation of logfiles will happen after that time. 0 disables.

#log_rotation_size = 10MB

# Automatic rotation of logfiles will happen after that much log output. 0 disables.
Where to Log (syslog)

#syslog_facility = 'LOCAL0'
#syslog_ident = 'postgres'
#syslog_sequence_numbers = on
#syslog_split_messages = on

# This is only relevant when logging to eventlog (win32):
# (change requires restart)
#event_source = 'PostgreSQL'
When to Log

client_min_messages = notice

log_min_messages = warning
When to Log (Continued)

```
#log_min_error_statement = error       # values in order of decreasing detail:
#   debug5
#   debug4
#   debug3
#   debug2
#   debug1
#   info
#   notice
#   warning
#   error
#   log
#   fatal
#   panic (effectively off)

#log_min_duration_statement = -1       # -1 is disabled, 0 logs all statements
# and their durations, > 0 logs only
# statements running at least this number
# of milliseconds
```
What to Log

#debug_print_parse = off
#debug_print_rewritten = off
#debug_print_plan = off
#debug_pretty_print = on
#log_checkpoints = off
#log_connections = off
#log_disconnections = off
#log_duration = off
#log_error_verbosity = default  # terse, default, or verbose messages
#log_hostname = off
What To Log: Log_line_prefix

```
#log_line_prefix = '%m [%p] '

# special values:
#  %a = application name
#  %u = user name
#  %d = database name
#  %r = remote host and port
#  %h = remote host
#  %p = process ID
#  %t = timestamp without milliseconds
#  %m = timestamp with milliseconds
#  %n = timestamp with milliseconds (as a
#      #  %i = command tag
#  %e = SQL state
#  %c = session ID
#  %l = session line number
#  %s = session start timestamp
#  %v = virtual transaction ID
#  %x = transaction ID (0 if none)
#  %q = stop here in non-session
#      processes
#  %% = '
#  # e.g., '<%u%%d> '
```

#log_lock_waits = off # log lock waits >= deadlock_timeout
#log_statement = 'none' # none, ddl, mod, all
#log_replication_commands = off
#log_temp_files = -1 # log temporary files equal or larger
# than the specified size in kilobytes; 
# -1 disables, 0 logs all temp files

log_timezone = 'US/Eastern'

# - Process Title -

#cluster_name = ''

#update_process_title = on # added to process titles if nonempty # (change requires restart)
# - Query/Index Statistics Collector -

#track_activities = on
#track_counts = on
#track_io_timing = off
#track_functions = none # none, pl, all
#track_activity_query_size = 1024 # (change requires restart)
#stats_temp_directory = 'pg_stat_tmp'

# - Statistics Monitoring -

#log_parser_stats = off
#log_planner_stats = off
#log_executor_stats = off
#log_statement_stats = off
#log_statement_stats = off
# autovacuum = on
# Enable autovacuum subprocess? 'on'
# requires track_counts to also be on.
# -1 disables, 0 logs all actions and
# their durations, > 0 logs only
# actions running at least this number
# of milliseconds.
# max number of autovacuum subprocesses
# (change requires restart)
# time between autovacuum runs
# min number of row updates before
# vacuum
# min number of row updates before
# analyze
# fraction of table size before vacuum
# fraction of table size before analyze
# maximum XID age before forced vacuum
# (change requires restart)
# maximum multixact age
# before forced vacuum
# (change requires restart)
# default vacuum cost delay for
# autovacuum, in milliseconds;
# -1 means use vacuum_cost_delay
# default vacuum cost limit for
Statement Behavior

```
#search_path = '"$user", public'  # schema names
#default_tablespace = ''  # a tablespace name, '' uses the default
#temp_tablespaces = ''  # a list of tablespace names, '' uses
# only default tablespace

#check_function_bodies = on
#default_transaction_isolation = 'read committed'
#default_transaction_read_only = off
#default_transaction_deferrable = off
#session_replication_role = 'origin'
#statement_timeout = 0  # in milliseconds, 0 is disabled
#lock_timeout = 0  # in milliseconds, 0 is disabled
#idle_in_transaction_session_timeout = 0  # in milliseconds, 0 is disabled
#vacuum_freeze_min_age = 50000000
#vacuum_freeze_table_age = 150000000
#vacuum_multixact_freeze_min_age = 5000000
#vacuum_multixact_freeze_table_age = 150000000
#bytea_output = 'hex'  # hex, escape
#xmlbinary = 'base64'
#xmloption = 'content'
#gin_fuzzy_search_limit = 0
#gin_pending_list_limit = 4MB
```
datestyle = 'iso, mdy'
#intervalstyle = 'postgres'
timezone = 'US/Eastern'
#timezone_abbreviations = 'Default'

#extra_float_digits = 0
#client_encoding = sql_ascii

# These settings are initialized by initdb, but they can be changed.
lc_messages = 'en_US.UTF-8'  # locale for system error message strings

lc_monetary = 'en_US.UTF-8'  # locale for monetary formatting
lc_numeric = 'en_US.UTF-8'   # locale for number formatting
lc_time = 'en_US.UTF-8'      # locale for time formatting

# default configuration for text search
default_text_search_config = 'pg_catalog.english'
Other Defaults

#dynamic_library_path = '$libdir'
#local_preload_libraries = ''
#session_preload_libraries = ''
Lock Management

#deadlock_timeout = 1s
#max_locks_per_transaction = 64

#max_pred_locks_per_transaction = 64
#max_pred_locks_per_relation = -2
#max_pred_locks_per_page = 2

# min 10
# (change requires restart)
# min 10
# (change requires restart)
# negative values mean
# (max_pred_locks_per_transaction
# / -max_pred_locks_per_relation) - 1
# min 0
# - Previous PostgreSQL Versions -

#array_nulls = on
#backslash_quote = safe_encoding  # on, off, or safe_encoding
#default_with_oids = off
#escape_string_warning = on
#lo_compat_privileges = off
#operator_precedence_warning = off
#quote_all_identifiers = off
#standard_conforming_strings = on
#synchronize_seqscans = on

# - Other Platforms and Clients -

#transform_null_equals = off
Error Handling

#exit_on_error = off # terminate session on any error?
#restart_after_crash = on # reinitialize after backend crash?
#include_dir = 'conf.d'
#include_if_exists = 'exists.conf'
#include = 'special.conf'

# include files ending in '.conf' from directory 'conf.d'
# include file only if it exists
# include file
Interfaces

- Installing
  - Compiled Languages (C, ecpg)
  - Scripting Language (Perl, Python, PHP)
  - SPI

- Connection Pooling
$ ls -CF include/
ecpg_config.h      libpq/        pgtypes_date.h    sql3types.h
ecpgerrno.h         libpq-events.h pgtypes_error.h   sqlca.h
ecpg_informix.h     libpq-fe.h    pgtypes_interval.h sqlda-compat.h
ecpglib.h           pg_config_ext.h pgtypes_numeric.h sqlda.h
ecpgotype.h         pg_config.h    pgtypes_timestamp.h sqlda-native.h
informix/           pg_config_manual.h postgres_ext.h
internal/           pg_config_os.h  server/
Library Files

$ ls -CF lib/
ascii_and_mic.so*  libpgcommon.a  utf8_and_ascii.so*
cyrillic_and_mic.so*  libpgfeutils.a  utf8_and_big5.so*
dict_snowball.so*  libpgport.a  utf8_and_cyrillic.so*
euc2004_sjis2004.so*  libpgtypes.a  utf8_and_euc2004.so*
euc_cn_and_mic.so*  libpgtypes.so@  utf8_and_euc_cn.so*
euc_jp_and_sjis.so*  libpgtypes.so.3@  utf8_and_euc_jp.so*
euc_kr_and_mic.so*  libpgtypes.so.3.10*  utf8_and_euc_kr.so*
euc_tw_and_big5.so*  libpq.a  utf8_and_euc_tw.so*
latin2_and_win1250.so*  libpq.so@  utf8_and_gb18030.so*
latin_and_mic.so*  libpq.so.5@  utf8_and_gbk.so*
libecpg.a  libpq.so.5.10*  utf8_and_iso8859_1.so*
libecpg_compat.a  libpqwalreceiver.so*  utf8_and_iso8859.so*
libecpg_compat.so@  pgoutput.so*  utf8_and_johab.so*
libecpg_compat.so.3@  pgxs/  utf8_and_sjis2004.so*
libecpg_compat.so.3.10*  pkgconfig/  utf8_and_sjis.so*
libecpg.so@  plperl.so*  utf8_and_uhc.so*
libecpg.so.6@  plpgsql.so*  utf8_and_win.so*
libecpg.so.6.10*  plpython2.so*
3. Maintenance
Backup

- File system-level (physical)
  - tar, cpio while shutdown
  - file system snapshot
  - rsync, shutdown, rsync, restart

- `pg_dump/pg_dumpall` (logical)

- Restore/pg_restore with custom format
Continuous Archiving / Point-In-Time Recovery (PITR)

File System–Level Backup

Continuous Archive (WAL)

WAL

02:00

09:00

11:00

13:00
1. `archive_mode = on`
2. `wal_level = archive`
3. `archive_command = 'cp -i %p /mnt/server/pgsql/%f < /dev/null'`
4. `SELECT pg_start_backup('label');`
5. Perform file system-level backup (can be inconsistent)
6. `SELECT pg_stop_backup();`

`pg_basebackup` does this automatically.
PITR Recovery

File System–Level Backup  Continuous Archive (WAL)
PITR Recovery Procedures

1. Stop postmaster
2. Restore file system-level backup
3. Make adjustments as outlined in the documentation
4. Create recovery.conf
5. `restore_command = 'cp /mnt/server/pgsql/%f %p'`
6. Start the postmaster
Continuous Archive Management

Simplify backups and WAL archive file management with:

- *pgBackRest*
- *barman*
Data Maintenance

- **VACUUM** (nonblocking) records free space into .fsm (free space map) files
- **ANALYZE** collects optimizer statistics
- **VACUUM FULL** (blocking) shrinks the size of database disk files
Autovacuum handles vacuum and analyze tasks automatically.
Checkpoints

- Write all dirty shared buffers
- Sync all dirty kernel buffers
- Recycle WAL files
- Controlled by `checkpoint_timeout` and `max_wal_size`
$ ps -f -Upostgres
postgres 825 1 0 Tue12AM ?? 0:06.57 /u/pgsql/bin/postmaster -i
postgres 829 825 0 Tue12AM ?? 0:35.03 writer process (postmaster)
postgres 830 825 0 Tue12AM ?? 0:16.07 wal writer process (postmaster)
postgres 831 825 0 Tue12AM ?? 0:11.34 autovacuum launcher process (postmaster)
postgres 832 825 0 Tue12AM ?? 0:07.63 stats collector process (postmaster)
postgres 13003 825 0 3:44PM ?? 0:00.01 postgres test [local] idle (postmaster)
postgres 13002 12997 0 3:44PM ttyq1 0:00.03 /u/pgsql/bin/psql test
$ top -c
  top - 10:29:47 up 23 days, 18:53, 6 users, load average: 1.73, 1.49, 0.81
Tasks: 387 total, 2 running, 385 sleeping, 0 stopped, 0 zombie
%Cpu(s): 5.9 us, 0.5 sy, 0.0 ni, 93.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 24734444 total, 19187724 used, 5546720 free, 532280 buffers
KiB Swap: 6369276 total, 168292 used, 6200984 free. 16936936 cached Mem

  PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
  32037 postgres 20 0 190980 27940 21420 R 100.0 0.1 0:09.74 postgres: postgres test [local] INSERT
  32061 root 20 0 26056 3240 2444 R 0.7 0.0 0:00.09 top -c
Query Monitoring

test=> SELECT * FROM pg_stat_activity;
...
| datid | 16384 |
| datname | test |
| pid | 16382 |
| usesysid | 10 |
| usename | postgres |
| application_name | psql |
| client_addr | |
| client_hostname | |
| client_port | -1 |
| backend_start | 2018-04-15 09:00:26.467813-04 |
| xact_start | 2018-04-15 09:00:48.028667-04 |
| query_start | 2018-04-15 09:00:48.028667-04 |
| state_change | 2018-04-15 09:00:48.028671-04 |
| wait_event_type | |
| wait_event | |
| state | active |
| backend_xid | |
| backend_xmin | 556 |
| query | SELECT * FROM pg_stat_activity; |
| backend_type | client backend |
pg_stat_all_indexes | view | postgres
pg_stat_all_tables  | view | postgres
pg_stat_database    | view | postgres
pg_stat_sys_indexes | view | postgres
pg_stat_sys_tables  | view | postgres
pg_stat_user_indexes| view | postgres
pg_stat_user_tables | view | postgres
pg_statio_all_indexes| view | postgres
pg_statio_all_sequences| view | postgres
pg_statio_all_tables | view | postgres
pg_statio_sys_indexes | view | postgres
pg_statio_sys_sequences| view | postgres
pg_statio_sys_tables  | view | postgres
pg_statio_user_indexes | view | postgres
pg_statio_user_sequences| view | postgres
pg_statio_user_tables | view | postgres
```
SELECT * FROM pg_stat_database;

- [ RECORD 4 ]-+----------
datid     | 16384
datname   | test
numbackends | 1
xact_commit | 188
xact_rollback | 0
blks_read  | 95
blks_hit   | 11832
tup_returned | 64389
tup_fetched | 2938
tup_inserted | 0
tup_updated | 0
tup_deleted | 0
```
### Table Activity

```sql
SELECT * FROM pg_stat_all_tables;
```

<table>
<thead>
<tr>
<th>relid</th>
<th>2616</th>
</tr>
</thead>
<tbody>
<tr>
<td>schemaname</td>
<td>pg_catalog</td>
</tr>
<tr>
<td>relname</td>
<td>pg_opclass</td>
</tr>
<tr>
<td>seq_scan</td>
<td>2</td>
</tr>
<tr>
<td>seq_tup_read</td>
<td>2</td>
</tr>
<tr>
<td>idx_scan</td>
<td>99</td>
</tr>
<tr>
<td>idx_tup_fetch</td>
<td>99</td>
</tr>
<tr>
<td>n_tup_ins</td>
<td>0</td>
</tr>
<tr>
<td>n_tup_upd</td>
<td>0</td>
</tr>
<tr>
<td>n_tup_del</td>
<td>0</td>
</tr>
<tr>
<td>n_tup_hot_upd</td>
<td>0</td>
</tr>
<tr>
<td>n_live_tup</td>
<td>0</td>
</tr>
<tr>
<td>n_dead_tup</td>
<td>0</td>
</tr>
<tr>
<td>last_vacuum</td>
<td></td>
</tr>
<tr>
<td>last_autovacuum</td>
<td></td>
</tr>
<tr>
<td>last_analyze</td>
<td></td>
</tr>
<tr>
<td>last_autoanalyze</td>
<td></td>
</tr>
</tbody>
</table>
```

81 / 113
Table Block Activity

test=> SELECT * FROM pg_statio_all_tables;

<table>
<thead>
<tr>
<th>relid</th>
<th>2602</th>
</tr>
</thead>
<tbody>
<tr>
<td>schemaname</td>
<td>pg_catalog</td>
</tr>
<tr>
<td>relname</td>
<td>pg_amop</td>
</tr>
<tr>
<td>heap_blks_read</td>
<td>3</td>
</tr>
<tr>
<td>heap_blks_hit</td>
<td>114</td>
</tr>
<tr>
<td>idx_blks_read</td>
<td>5</td>
</tr>
<tr>
<td>idx_blks_hit</td>
<td>303</td>
</tr>
<tr>
<td>toast_blks_read</td>
<td></td>
</tr>
<tr>
<td>toast_blks_hit</td>
<td></td>
</tr>
<tr>
<td>tidx_blks_read</td>
<td></td>
</tr>
<tr>
<td>tidx_blks_hit</td>
<td></td>
</tr>
</tbody>
</table>
Analyzing Activity

- Heavily used tables
- Unnecessary indexes
- Additional indexes
- Index usage
- TOAST usage
The image shows the output of the `vmstat 5` command, which provides a snapshot of system performance metrics in 5-second intervals. The command output includes five columns for process and memory statistics, as well as disk usage and fault details. The `vmstat` output is a tool for monitoring system performance and is particularly useful for identifying performance bottlenecks.

The `vmstat 5` command provides the following metrics:

- **progs**: Process states (running, blocked, waiting, etc.)
- **memory**: Memory usage statistics
- **page**: Details about page faults and page misses
- **disks**: Disk I/O statistics
- **faults**: Details about kernel faults and memory faults
- **cpu**: CPU usage statistics

The output is structured in a tabular format, with columns for each metric and data points for different time intervals. The specific output shown includes a timestamp at the top left corner, indicating the last update time, and the number of rows indicating the number of data points collected during the observation period.
```
$ iostat 5

<table>
<thead>
<tr>
<th></th>
<th>tty</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>tin</td>
<td>tout</td>
<td>sps</td>
<td>tps</td>
<td>mps</td>
<td>sps</td>
<td>tps</td>
<td>mps</td>
<td>sps</td>
</tr>
<tr>
<td>7</td>
<td>119</td>
<td>244</td>
<td>11</td>
<td>6.1</td>
<td>0</td>
<td>0</td>
<td>27.3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>86</td>
<td>20</td>
<td>1</td>
<td>1.4</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>82</td>
<td>61</td>
<td>4</td>
<td>3.6</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>65</td>
<td>6</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>90</td>
<td>31</td>
<td>2</td>
<td>5.4</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>24</td>
<td>173</td>
<td>6</td>
<td>0</td>
<td>4.9</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>91</td>
<td>3594</td>
<td>63</td>
<td>4.6</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
```
test=> \df *size*

<table>
<thead>
<tr>
<th>Schema</th>
<th>Name</th>
<th>Result data type</th>
<th>Argument data types</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>pg_catalog</td>
<td>pg_column_size</td>
<td>integer</td>
<td>&quot;any&quot;</td>
<td>normal</td>
</tr>
<tr>
<td>pg_catalog</td>
<td>pg_database_size</td>
<td>bigint</td>
<td>name</td>
<td>normal</td>
</tr>
<tr>
<td>pg_catalog</td>
<td>pg_database_size</td>
<td>bigint</td>
<td>oid</td>
<td>normal</td>
</tr>
<tr>
<td>pg_catalog</td>
<td>pg_indexes_size</td>
<td>bigint</td>
<td>regclass</td>
<td>normal</td>
</tr>
<tr>
<td>pg_catalog</td>
<td>pg_relation_size</td>
<td>bigint</td>
<td>regclass</td>
<td>normal</td>
</tr>
<tr>
<td>pg_catalog</td>
<td>pg_relation_size</td>
<td>bigint</td>
<td>regclass, text</td>
<td>normal</td>
</tr>
<tr>
<td>pg_catalog</td>
<td>pg_size_pretty</td>
<td>text</td>
<td>bigint</td>
<td>normal</td>
</tr>
<tr>
<td>pg_catalog</td>
<td>pg_table_size</td>
<td>bigint</td>
<td>regclass</td>
<td>normal</td>
</tr>
<tr>
<td>pg_catalog</td>
<td>pg_tablespace_size</td>
<td>bigint</td>
<td>name</td>
<td>normal</td>
</tr>
<tr>
<td>pg_catalog</td>
<td>pg_tablespace_size</td>
<td>bigint</td>
<td>oid</td>
<td>normal</td>
</tr>
<tr>
<td>pg_catalog</td>
<td>pg_total_relation_size</td>
<td>bigint</td>
<td>regclass</td>
<td>normal</td>
</tr>
</tbody>
</table>
$ oid2name
All databases:
---------------------------------
18720 = test1
 1  = template1
18719 = template0
18721 = test
18735 = postgres
18736 = cssi
Table File Mapping

```
$ cd /usr/local/pgsql/data/base
$ oid2name
All databases:
---------------------------------
16817 = test2
16578 = x
16756 = test
1  = template1
16569 = template0
16818 = test3
16811 = floattest

$ cd 16756
$ ls 1873*
18730 18731 18732 18735 18736 18737 18738 18739
```
$ oid2name -d test -o 18737
Tablename of oid 18737 from database "test":
---------------------------------
18737  = ips

$ oid2name -d test -t ips
Oid of table ips from database "test":
---------------------------------
18737  = ips

$ # show disk usage per database
$ cd /usr/local/pgsql/data/base
$ du -s * | 
> while read SIZE OID
>   do
>     echo "$SIZE `oid2name -q | grep ^$OID' '`
>   done | 
> sort -rn
2256   18721  = test
2135   18735  = postgres
Disk Balancing

- Move pg_wal to another drive using symlinks
- Tablespaces
Per-Database Tablespaces

DB1  DB2  DB3  DB4

Disk 1  Disk 2  Disk 3
Per-Object Tablespaces

- tab1
- tab2
- index
- constraint

Disk 1
Disk 2
Disk 3
Analyzing Locking

$ ps -f -U postgres

<table>
<thead>
<tr>
<th>PID</th>
<th>TT</th>
<th>STAT</th>
<th>TIME</th>
<th>COMMAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>9874</td>
<td>??</td>
<td>I</td>
<td>0:00.07</td>
<td>postgres test [local] idle in transaction (postmaster)</td>
</tr>
<tr>
<td>9835</td>
<td>??</td>
<td>S</td>
<td>0:00.05</td>
<td>postgres test [local] UPDATE waiting (postmaster)</td>
</tr>
<tr>
<td>10295</td>
<td>??</td>
<td>S</td>
<td>0:00.05</td>
<td>postgres test [local] DELETE waiting (postmaster)</td>
</tr>
</tbody>
</table>

test=> SELECT * FROM pg_locks;

<table>
<thead>
<tr>
<th>relation</th>
<th>database</th>
<th>transaction</th>
<th>pid</th>
<th>mode</th>
<th>granted</th>
</tr>
</thead>
<tbody>
<tr>
<td>17143</td>
<td>17142</td>
<td></td>
<td>9173</td>
<td>AccessShareLock</td>
<td>t</td>
</tr>
<tr>
<td>17143</td>
<td>17142</td>
<td></td>
<td>9173</td>
<td>RowExclusiveLock</td>
<td>t</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9380</td>
<td>ExclusiveLock</td>
<td>t</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9338</td>
<td>ShareLock</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9338</td>
<td>ExclusiveLock</td>
<td>t</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9380</td>
<td>AccessShareLock</td>
<td>t</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9338</td>
<td>AccessShareLock</td>
<td>t</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9338</td>
<td>RowExclusiveLock</td>
<td>t</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9173</td>
<td>ExclusiveLock</td>
<td>t</td>
</tr>
</tbody>
</table>
Miscellaneous Tasks

- Log file rotation, syslog
- Upgrading
  - pg_dump, restore
  - pg_upgrade
  - Slony
- Migration
Administration Tools

- pgadmin
- phppgadmin
External Monitoring Tools

- Alerting: check_postgres, tail_n_mail, Nagios
- Server analysis: Munin, Cacti, Zabbix, Nagios, MRTG, Prometheus, Grafana
- Queries: pg_stat_statements, auto_explain, pgbadger
- Commercial: Postgres Enterprise Manager (PEM), Circonus, VividCortex
Monitoring Summary

Operating System: vmstat, iostat

Process/Session: ps, top -c, pg_stat_activity

Query: pg_stat_activity.query, log_statement, log_statement_stats

Parser: log_parser_stats

Planner: log_planner_stats

Executor: log_executor_stats, pg_locks, log_lock_waits, pg_stat_activity.wait_event

Reporting

Alterting / Aggregation

sar, check_postgres, log_temp_files
tail_n_mail
pg_stat_statements, log_min_duration_statement, pgbadger
auto_explain
e.g. pg_stat_all_tables

time
5. Recovery

https://www.flickr.com/photos/coastguardnews/
Nothing Required. Transactions in progress are rolled back.
Nothing Required. Transactions in progress are rolled back.
Abrupt Postgres Server Crash

Nothing Required. Transactions in progress are rolled back.
Operating System Crash

Nothing Required. Transactions in progress are rolled back. Partial page writes are repaired.
Disk Failure

Restore from previous backup or use PITR.
Accidental DELETE

Recover table from previous backup, perhaps using pg_restore. It is possible to modify the backend code to make deleted tuples visible, dump out the deleted table and restore the original code. All tuples in the table since the previous vacuum will be visible. It is possible to restrict that so only tuples deleted by a specific transaction are visible.
Write-Ahead Log (WAL) Corruption

See pg_resetwal. Review recent transactions and identify any damage, including partially committed transactions.
It may be necessary to create an empty file with the deleted file name so the object can be deleted, and then the object restored from backup.
Accidental DROP TABLE

Restore from previous backup.
Accidental DROP INDEX

Recreate index.
Accidental DROP DATABASE

Restore from previous backup.
Restart problems are usually caused by write-ahead log problems. See pg_resetwal. Review recent transactions and identify any damage, including partially committed transactions.
Use REINDEX.
Table Corruption

Try reindexing the table. Try identifying the corrupt OID of the row and transfer the valid rows into another table using `SELECT...INTO...WHERE oid != ###`. Use `pageinspect` to analyze the internal structure of the table.
Conclusion

https://momjian.us/presentations