Redpill Linpro

# New features in PostgreSQL 9.0

## Open Source Days, March 2010
## Copenhagen, Denmark

Magnus Hagander
Redpill Linpro AB

Redpill Linpro

# It's a big release!

- New feature patches: 204
  - Not including those directly added by committers
- Submitters: 84
  - Not including 14 committers
- 1860 files changed, 150951 insertions(+), 82558 deletions(-)

Redpill Linpro

# Breaking it down

- Performance
- Developer
- DBA
- Version-bumping

Redpill Linpro

# Join Removal

- JOINs to unnecessary tables are automatically removed
- Particularly useful with VIEWs
- Another good reason not to use «SELECT *»

Redpill Linpro

# Rewrite of LISTEN/NOTIFY

- Asynchronous notifications
- Now in-memory
  - Much faster!
- Support for payloads
  - NOTIFY mychannel, 'fooh'
  - Ex. cache-expire specific items by pk

Redpill Linpro

# EXPLAIN BUFFERS

```
postgres=# EXPLAIN (ANALYZE, BUFFERS) SELECT * FROM
pg_attribute;

                    QUERY PLAN
-----------------------------------------
 Seq Scan on pg_attribute
                (cost=0.00..52.46 rows=1946 width=167)
                (actual time=0.011..8.834 rows=1946 loops=1)
   Buffers: shared hit=16 read=17
 Total runtime: 23.080 ms
(3 rows)
```

Redpill Linpro

# Machine readable EXPLAIN

```
postgres=# EXPLAIN (ANALYZE, BUFFERS, FORMAT XML)
SELECT * FROM pg_attribute;
                          QUERY PLAN
---------------------------------------------------------------
 <explain xmlns="http://www.postgresql.org/2009/explain">
   <Query>
     <Plan>
       <Node-Type>Seq Scan</Node-Type>
       <Relation-Name>pg_attribute</Relation-Name>
       <Alias>pg_attribute</Alias>
       <Startup-Cost>0.00</Startup-Cost>
       <Total-Cost>54.05</Total-Cost>
       <Plan-Rows>2005</Plan-Rows>
       <Plan-Width>167</Plan-Width>
       <Actual-Startup-Time>0.009</Actual-Startup-Time>
       <Actual-Total-Time>17.376</Actual-Total-Time>
```

Redpill Linpro

# Machine readable EXPLAIN

```
postgres=# EXPLAIN (ANALYZE, BUFFERS, FORMAT YAML)
SELECT * FROM pg_attribute;
            QUERY PLAN
--------------------------------
 - Plan:                       +
     Node Type: Seq Scan       +
     Relation Name: pg_attribute+
     Alias: pg_attribute       +
     Startup Cost: 0.00        +
     Total Cost: 52.46         +
     Plan Rows: 1946           +
     Plan Width: 167           +
     Actual Startup Time: 0.018 +
     Actual Total Time: 9.448  +
     Actual Rows: 1946         +
```

Redpill Linpro

# Extended Windowing Functions

- Moving average!

```
postgres=# SELECT i,v,avg(v) OVER
(ORDER BY i ROWS BETWEEN 1 PRECEDING AND 1 FOLLOWING) FROM t;
 i | v |         avg
---+---+--------------------
 1 | 1 | 2.000000000000000
 2 | 3 | 2.333333333333333
 3 | 3 | 3.333333333333333
 4 | 4 | 3.666666666666667
 5 | 4 | 4.333333333333333
 6 | 5 | 4.000000000000000
 7 | 3 | 3.666666666666667
 8 | 3 | 2.333333333333333
 9 | 1 | 2.000000000000000
(9 rows)
```

Redpill Linpro

# Breaking it down

- Performance
- Developer
- DBA
- Version-bumping

Redpill Linpro

# Anonymous code blocks!

```
DO $$
BEGIN
  FOR x IN 1..10 LOOP
     RAISE NOTICE 'Now at %', x;
  END LOOP;
END;
$$;
```

Redpill Linpro

# Anonymous code blocks!

```
DO $$
for (my $x = 1; $x < 11; $x++) {
   elog(NOTICE, "Now at $x");
}
$$ language 'plperl';
```

Redpill Linpro

# Anonymous code blocks!

```
DO $$
HAI
    I HAS A VAR ITZ 0
    IM IN YR LOOP
        VAR R SUM OF VAR AN 1
        VISIBLE VAR
        BOTH SAEM VAR AN 10, O RLY?
            YA RLY, GTFO
        OIC
    IM OUTTA YR LOOP
KTHXBYE
$$ language 'pllolcode';
```

Redpill Linpro

# pl/perl and pl/python

- Error context tracking
- Support for python 3.1
- Datatype mapping
- Major overhauls

Redpill Linpro

# Trigger enhancements

- Column level triggers (UPDATE)

```
CREATE TRIGGER t_1
  BEFORE UPDATE OF v ON t
  EXECUTE PROCEDURE test();
```

- Conditional triggers

```
CREATE TRIGGER t_2
  BEFORE UPDATE ON t
  FOR EACH ROW WHEN (OLD.v != NEW.v)
  EXECUTE PROCEDURE test();
```

Redpill Linpro

# Deferrable UNIQUE constraints

- Defer uniqueness checks until end of command/transaction

```
postgres=# create table t2(i int UNIQUE, v int);
postgres=# insert into t2 values (1,0), (2,0);
postgres=# update t2 set i=i+1;
ERROR:  duplicate key value violates unique constraint "t2_i_key"
DETAIL:  Key (i)=(2) already exists.
```

```
postgres=# create table t2(i int UNIQUE DEFERRABLE, v int);
postgres=# insert into t2 values (1,0), (2,0);
postgres=# update t2 set i=i+1;
UPDATE 2
```

Redpill Linpro

# Bytea hex format

- New output format for bytea
  - Default!

```
postgres=# select 'test test'::bytea;
----------------------
 \x7465737420746573 74
```

Redpill Linpro

# hstore

- «Less inefficient key/value store»
- No size limit
- New opclasses -> GROUP BY, DISTINCT etc
- New functions and operators

Redpill Linpro

# hstore

```
postgres=# CREATE TABLE x(h hstore);
postgres=# CREATE INDEX ix ON x USING gist(h);
postgres=# INSERT INTO x VALUES ('k' => 'v1');
postgres=# INSERT INTO x VALUES ('k=>v2, k2=>v3');

postgres=# SELECT * FROM x;
 "k"=>"v1"
 "k"=>"v2", "k2"=>"v3"

postgres=# SELECT * FROM x WHERE h ? 'k2';
 "k"=>"v2", "k2"=>"v3"

postgres=# SELECT h -> 'k' FROM x;
 v1
 v2
```

Redpill Linpro

# Breaking it down

- Performance
- Developer
- DBA
- Version-bumping

Redpill Linpro

# No more VACUUM FULL

- It's gone!

Redpill Linpro

# The new VACUUM FULL

- Welcome the new VACUUM FULL
- No bloating!
- Faster!
- Rewrite-based, just like CLUSTER
- Keep avoiding it, but it's not disastrous anymore

Redpill Linpro

# GRANT ON ALL

- Grant permissions on all existing objects, whithout naming
  - Tables (incl views), sequences, functions
- Most requested feature on IRC?

```
postgres=# GRANT SELECT,INSERT,UPDATE ON
           ALL TABLES IN SCHEMA PUBLIC TO mha;
```

Redpill Linpro

# DEFAULT PRIVILEGES

- What about new objects?
- They're covered by DEFAULT PRIVILEGES

```
postgres=# ALTER DEFAULT PRIVILEGES
           IN SCHEMA public
           GRANT ALL ON TABLES TO mha;
```

Redpill Linpro

# RADIUS authentication

- Authenticate users with RADIUS
- Alternative to LDAP etc
- OTP
- Policy Based access, etc
- Still need to create db user

Redpill Linpro

# Per user/per database GUC

- Configuration changes based on combination of user and database
  - Previously user *or* database

```
postgres=# ALTER ROLE mha IN DATABASE webdb
           SET work_mem=10MB;
```

Redpill Linpro

# Breaking it down

- Performance
- Developer
- DBA
- Version-bumping

Redpill Linpro

# Win64 support

- Better late than never
- Not all that necessary before
- Larger shared_buffers and work_mem in OLAP scenarios
- Bad 3rd party support still!

Redpill Linpro

# Exclusion Constraints

Redpill Linpro

# Exclusion Constraints

!= Constraint Exclusion

(yes, we suck at names)

Redpill Linpro

# Exclusion Constraints

- Generalize the idea of UNIQUE constraints
- Not just equals - any indexable commutative operator
- Based on high performance GiST indexes

Redpill Linpro

# Exclusion Constraints

- Between-rows constraint
  - Like UNIQUE, unlike anything else
- Each row can potentially conflict with *any other row in the table*
- UNIQUE is just a subset

Redpill Linpro

# Exclusion Constraint

- Most common usecase: non-overlapping «something»
- Ex: non overlapping geometric or geographical regions
- Ex: non overlapping *time intervals*
  - Scheduling application, anyone?

Redpill Linpro

# Exclusion Constraint example

- Introducing the PERIOD datatype
  - Currently on pgFoundry, will be in 9.1
- Start and end time in single column
  - Inclusive of exclusive
- Set of useful operators
  - Especially &&, meaning «overlaps»

Redpill Linpro

# Exclusion Constraint example

```
CREATE TABLE bookings (
  title text,
  room text,
  during period
)
```

- How do we encforce that there are no conflicting bookings?
- Ideas?

Redpill Linpro

# Exclusion Constraint example

```
CREATE TABLE bookings (
  title text,
  room text,
  during period,
  EXCLUDE USING gist
     (room WITH =,
      during WITH &&)
)
```

Redpill Linpro

# Some replication changes

- Many options exist
- Most exist outside of core
- Only one is in core:

Redpill Linpro

# Log based replication

- Ship transaction log from master to slave
- Physical database changes
- Highly reliable, uses PITR code
- Entire database cluster
- pg_standby in <= 8.4

Redpill Linpro

# Streaming Replication

- 9.0 adds streaming mode
- No longer limited to 16Mb files
- Near-realtime
- Regular libpq connection!
- Still uses regular archive logging
  - to get started or when fallen behind

Redpill Linpro

# Hot Standby

- WAL replicas used to be replay only
- Still are with streaming replication
- Hot Standby changes this
  - Read queries allowed on slaves!
  - Some caveats of course

Redpill Linpro

# More about HS & SR

Saturday,

15:00

Redpill Linpro

# We **need** your help

- We've started shipping Alpha releases
- 9.0 alpha4 released Feb 24th
- Download and test!
  - Both your existing applications and new features!
  - Beware – dump/reload required!

Redpill Linpro

# Thank You!

*Questions?*

*magnus@hagander.net*
*http://blog.hagander.net/*
*Twitter: magnushagander*
*FreeNode: #postgresql:magnush*