新手机 新应用 新娱乐

9.1 特性体验

Digoal.Zhou

10/17/2011

SKY mobi
新手机 新应用 新娱乐

目录

- 新增功能
- 性能
- 管理
- 复制与恢复
- SQL
- 示例
- 讨论

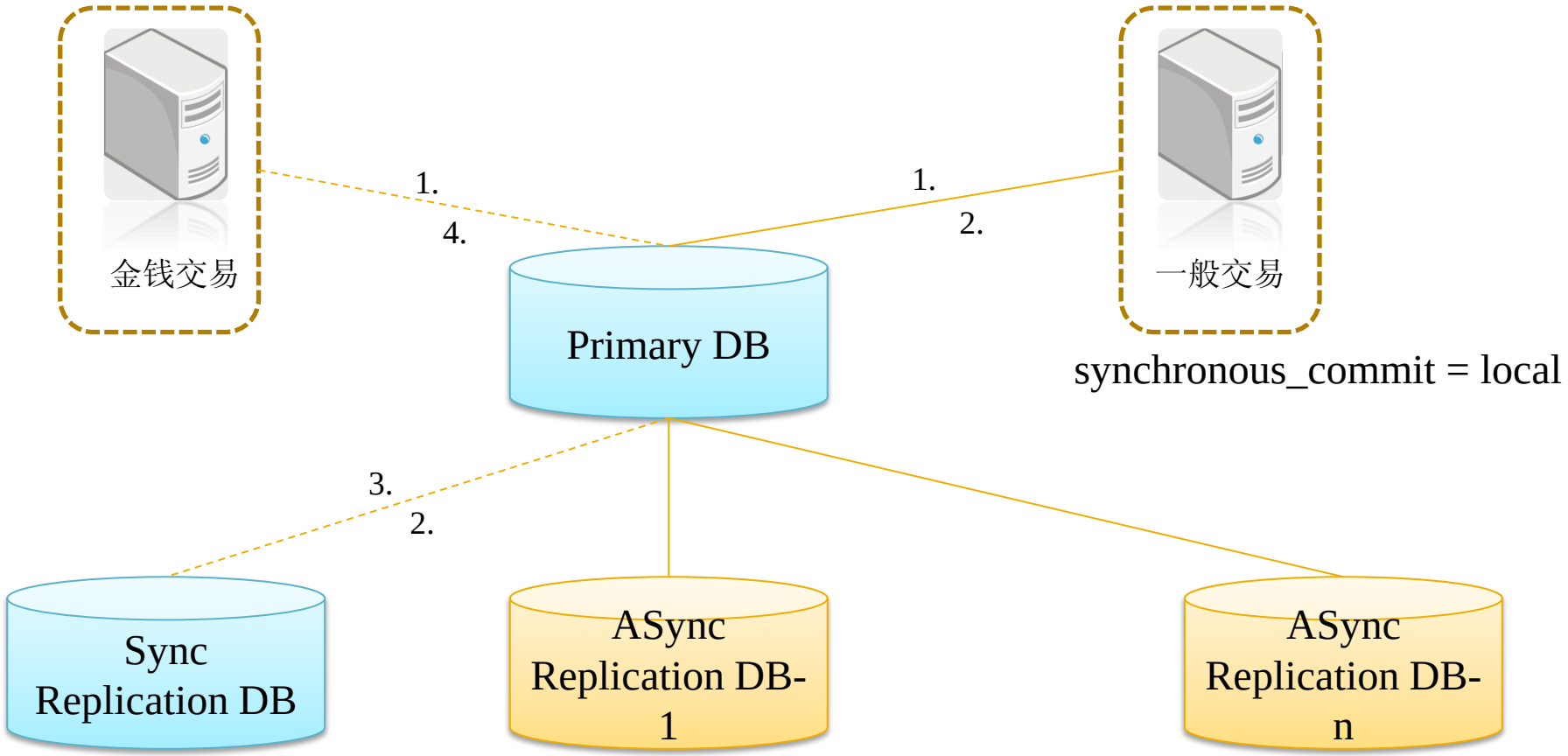
SKY mobi
新手机 新应用 新娱乐

新增功能

- 同步流复制
  - 快速数据迁移
  - 异地容灾
  - 多机房业务部署
  - 读请求负载均衡
  - HA
- FOREIGN DATA WRAPPER
  - 文件表
  - 外部表(远程数据库,PG,Oracle,MySQL,CouchDB,Redis,LDAP,ODBCAPI,...)
  - PL/Proxy 集群配置
- 列级Collation
- Add directory format to pg_dump
- INSTEAD OF Trigger ON Views

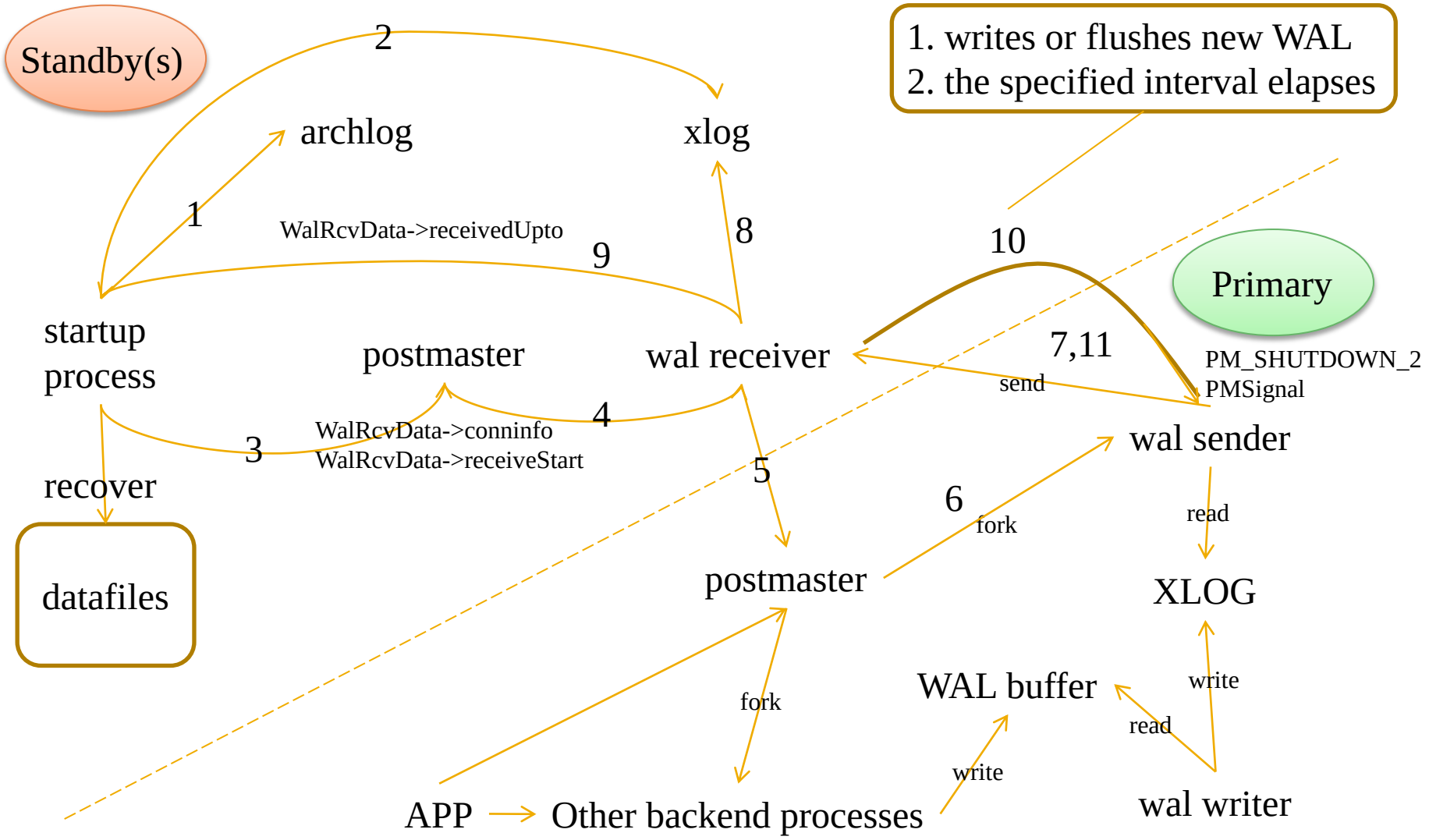
SKY mobi
新手机 新应用 新娱乐

流复制应用场景 - 1



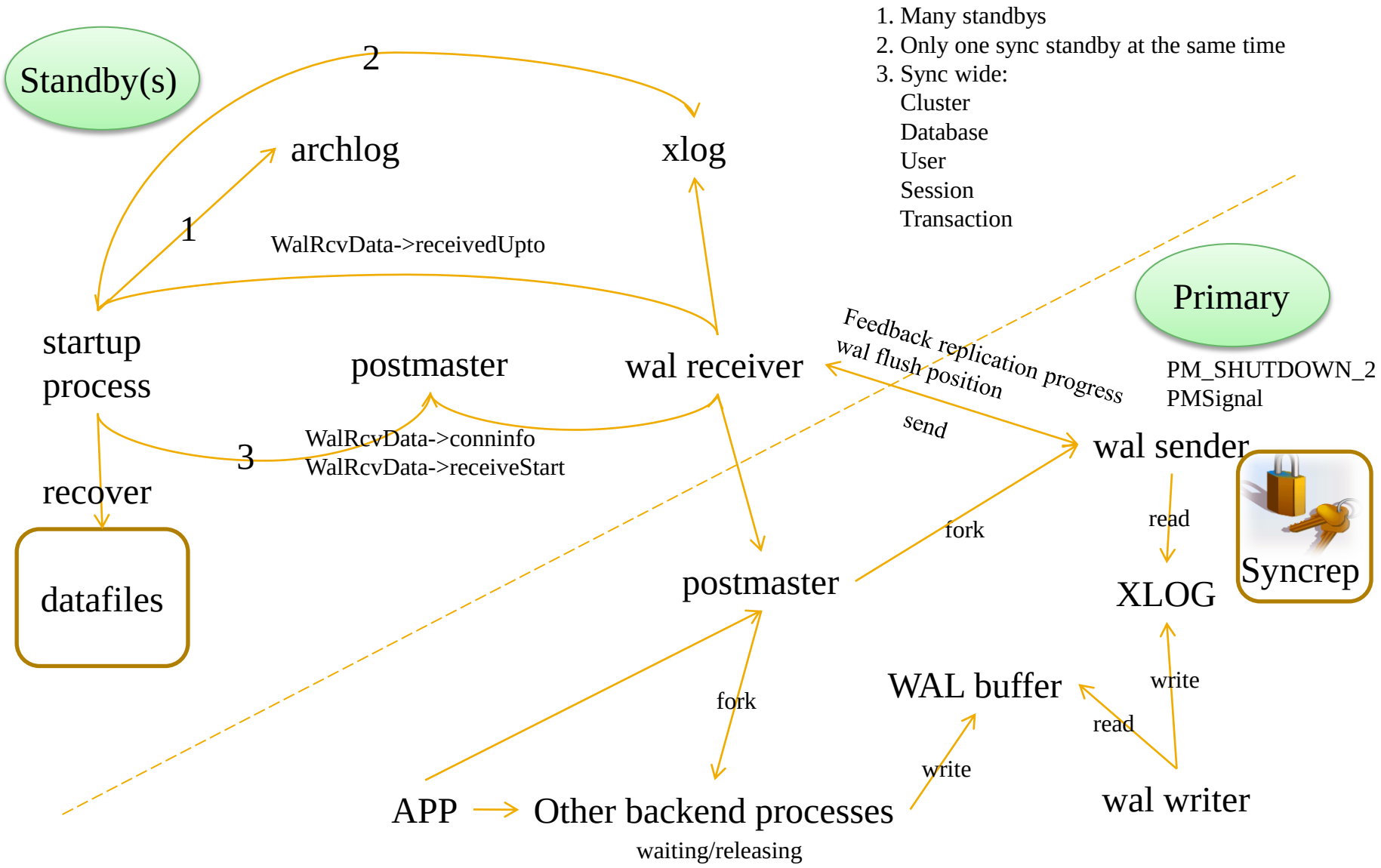
SKY mobi
新手机 新应用 新娱乐

异步流复制



SKY mobi
新手机 新应用 新娱乐

同步流复制



SKY mobi
新手机 新应用 新娱乐

流复制核心模块

- WALsender, 主节点
  - 与WALreceiver一对一连接,从磁盘读取XLOG文件记录发送给WALreceiver.
  - 主节点关库时WALreceiver 在所有用户进程关闭, 数据库做完shutdown CHECKPOINT后将未发送的WAL以及CHKPOINT信息发送给WALreceiver. 从而STANDBY节点与主节点一致.
- WALreceiver, Standby节点
  - 与WALsender一对一连接, 告知WALreceiver进程需要的XLOG起始点, 接收XLOG信息, FlushXLOG信息到磁盘, 更新共享内存WalRcv->receivedUpto变量.
- Startup Process, Standby节点
  - 从WalRcv->receivedUpto中获取已知的XLOG信息, 从归档,XLOG中读取XLOG信息完成实时数据恢复.
- Syncrep, 主节点
  - 当执行同步事务时被调用, 持有同步事务用户进程Ordered Queue, 等待Sync连接的WalSndCtlData共享内存变量状态(每个WALreceiver一个区域).
  - 同步事务用户进程状态, SYNC_REP_NOT_WAITING , SYNC_REP_WAITING , SYNC_REP_WAIT_COMPLETE

SKY mobi
新手机 新应用 新娱乐

同步复制压力测试

- 测试服务器配置 :
- 主库: 8核, 24G, 1块本地SAS10K转硬盘, 1块1GB网卡, x86服务器
- standby库1: 8核, 8G, 1块本地SAS10K转硬盘, 1块1GB网卡, x86服务器
- standby库2: 8核, 14G, 1块本地SAS10K转硬盘, 1块1GB网卡, x86服务器
- 测试表结构
- create table user_info
- (userid int,
- engname text,
- cnname text,
- occupation text,
- birthday date,
- signname text,
- email text,
- qq numeric,
- crt_time timestamp without time zone,
- mod_time timestamp without time zone
-);

SKY mobi
新手机 新应用 新娱乐

同步复制压力测试

- create table user_login_rec
- (userid int,
- login_time timestamp without time zone,
- ip inet
-);

- create table user_logout_rec
- (userid int,
- logout_time timestamp without time zone,
- ip inet
-);
- 测试数据
- insert into user_info (userid,engname,cnname,occupation,birthday,signname,email,qq,crt_time,mod_time)
- select generate_series(1,50000000),
- 'digoal.zhou',
- '德哥',

SKY mobi
新手机 新应用 新娱乐

同步复制压力测试

- 'DBA',
- '1970-01-01'
- ,E'公益是一辈子的事, I\'m Digoal.Zhou, Just do it!',
- 'digoal@126.com',
- 276732431,
- clock_timestamp(),
- NULL;
- alter table user_info add constraint pk_user_info primary key (userid) using index tablespace digoal_idx;

- 测试逻辑:
- 用户登录
  - 查询用户表，记录登录日志
- 用户退出
  - 记录登出日志

SKY mobi
新手机 新应用 新娱乐

同步复制压力测试

- 登录函数

登录函数：

```
create or replace function f_user_login
(i_userid int,
OUT o_userid int,
OUT o_engname text,
OUT o_cnname text,
OUT o_occupation text,
OUT o_birthday date,
OUT o_signname text,
OUT o_email text,
OUT o_qq numeric
)
as $BODY$
declare
begin
select userid,engname,cnname,occupation,birthday,signname,email,qq
into o_userid,o_engname,o_cnname,o_occupation,o_birthday,o_signname,o_email,o_qq
from user_info where userid=i_userid;
insert into user_login_rec (userid,login_time,ip) values (i_userid,now(),inet_client_addr());
return;
end;
$BODY$
language plpgsql;
```

SKY mobi
新手机 新应用 新娱乐

同步复制压力测试

- 退出函数

退出函数 :

```
create or replace function f_user_logout
(i_userid int,
OUT o_result int
)
as $BODY$
declare
begin
insert into user_logout_rec (userid,logout_time,ip) values (i_userid,now(),inet_client_addr());
o_result := 0;
return;
exception
when others then
o_result := 1;
return;
end;
$BODY$
language plpgsql;
```

SKY mobi
新手机 新应用 新娱乐

同步复制压力测试

- pgbench脚本
- pgbench测试脚本1 f_user_login.sql :
- \setrandom userid 1 50000000
- select * from f_user_login(:userid);
- pgbench测试脚本2 f_user_logout.sql :
- \setrandom userid 1 50000000
- select * from f_user_logout(:userid);
- 测试结果

TPS	单节点	1主,2异步复制库	1主,1同步,1异步复制库
登录	10431	10446	9882
退出	12486	12685	12235
性能损失	-	-0.9%	3.5%

SKY mobi
新手机 新应用 新娱乐

异步复制角色切换

- WALsender wake up Interval
  - wal_sender_delay (>=10ms)
  - every transaction commit
- 计划内
  - shutdown 主库(smart/fast)
  - 无数据丢失, 角色切换可逆
- 计划外
  - 相当于shutdown 主库(immediate)
  - 丢失未发送的事务信息, 角色切换不可逆

SKY mobi
新手机 新应用 新娱乐

同步复制角色切换

- WALsender wake up Interval
  - wal_sender_delay (>=10ms)
  - every transaction commit
- 计划内
  - shutdown 主库(smart/fast)
  - 无数据丢失, 角色切换可逆
- 计划外
  - 相当于shutdown 主库(immediate)
  - 无数据丢失, 角色切换可逆(完全sync)

SKY mobi
新手机 新应用 新娱乐

其他复制/恢复增强

- 添加支持发送文件系统备份的协议，可以非常方便的通过流复制连接创建备库。
- 添加流复制权限专属角色。
- 添加复制连接超时参数（以前只有等TCP超时，修改操作系统内核参数）。
- 新增监控视图。
  - pg_stat_replication
  - 主 : the current WAL sender state and transaction log location, display sync_priority from synchronous_standby_names
  - 备 : the last transaction log position it received and wrote, the last position it flushed to disk, and the last position it replayed,
- 监控函数。
  - pg_is_in_recovery()
  - pg_last_xlog_receive_location()
  - pg_last_xlog_replay_location()
  - pg_last_xact_replay_timestamp()
  - pg_is_xlog_replay_paused()
  - pg_xlog_replay_pause() / pg_xlog_replay_resume()

SKY mobi
新手机 新应用 新娱乐

其他复制/恢复增强

- hot_tandby反馈相关改进
  - hot_standby_feedback , 避免standby长SQL冲突退出.
- pg_stat_database_conflicts
  - show queries that have been canceled and the reason. Cancellations can occur because of dropped tablespaces, lock timeouts, old snapshots, pinned buffers, and deadlocks.
- 加入pg_stat_database 的 conflicts 计数
- 加大参数最大值, max_standby_archive_delay, max_standby_streaming_delay
- 允许创建还原时间点和还原到指定还原时间点. pg_create_restore_point(), pause_at_recovery_target, recovery_target_name.
- 参考
  - http://blog.163.com/digoal@126/blog/static/163877040201011303423 2645/
  - http://blog.163.com/digoal@126/blog/static/163877040201011305382 5671/
  - http://blog.163.com/digoal@126/blog/static/163877040201141154024306/
  - http://blog.163.com/digoal@126/blog/static/163877040201104420508 08/
  - http://blog.163.com/digoal@126/blog/static/163877040201151711114803/
  - http://blog.163.com/digoal@126/blog/static/163877040201182395310376/

SKY mobi
新手机 新应用 新娱乐

FOREIGN DATA WRAPPER

- Connect To Remote DB
- Connect To File

couchdb_fdw 0.1.0	PostgreSQL SQL/MED FDW for CouchDB
ldap_fdw 0.0.2	LDAP Foreign Data Wrapper for PostgreSQL 9.1+
mysql_fdw 1.0.0	MySQL FDW for PostgreSQL 9.1+
odbc_fdw 0.1.0	PostgreSQL SQL/MED FDW for ODBC API
oracle_fdw 0.9.1 (testing)	A Foreign Data Wrapper for Oracle
redis_fdw 1.0.0	Redis FDW for PostgreSQL 9.1+
s3_fdw 0.2.0 (unstable)	foreign-data wrapper for Amazon S3
twitter_fdw 1.0.0	a foreign data wrapper to Twitter Search

SKY mobi
新手机 新应用 新娱乐

FOREIGN DATA WRAPPER

- 只读,其他操作与普通表类似,如支持JOIN等操作
- 模块下载
  - pgxn.org
- 文件外部表
  - http://blog.163.com/digoal@126/blog/static/16387704020114164114831 1/
- 远程PostgreSQL表
- 远程Oracle表
  - http://blog.163.com/digoal@126/blog/static/16387704020118151162340/
  - http://blog.163.com/digoal@126/blog/static/16387704020118150533 1588/
- 远程Redis表
  - http://blog.163.com/digoal@126/blog/static/16387704020119181188247/

SKY mobi
新手机 新应用 新娱乐

性能改进

- UNLOGGED TABLE
- inheritance table scans return meaningfully-sorted results
- Add gist Nearest-neighbor order by operator

SKY mobi
新手机 新应用 新娱乐

UNLOGGED TABLE

- 特点
  - 不写wal日志, 不被流复制或者log shipping类的复制。
  - 基于unlogged table的索引也不写WAL日志。
  - 不能在UNLOGGED TABLE上建GiST索引。
  - 无法满足crash-safe。
- 应用场景
  - 临时数据快速数据入库。
  - 无需永久保留的数据。如会话保持类数据。
- 参考
  - http://blog.163.com/digoal@126/blog/static/16387704020115140232128/

SKY mobi
新手机 新应用 新娱乐

优化器改进

- 继承表查询排序优化
  - 当查询父表，并对输出结果执行了排序，如果子表在排序字段有索引存在，可以利用这个索引，不需要额外的排序过程。而在9.1以前的版本必须是先MERGE RESULT再进行一次排序。
- 参考
  - http://blog.163.com/digoal@126/blog/static/163877040201191922618396/

SKY mobi
新手机 新应用 新娱乐

近邻查询排序优化

- 添加GiST索引point_ops操作类的排序支持。
- 应用场景。
  - 找出二维平面上与某点最近的点。
- 参考
  - http://blog.163.com/digoal@126/blog/static/163877040201191810141210 5/

SKY mobi
新手机 新应用 新娱乐

管理改进

- ADD PG_STAT_XACT_*
  - pg_stat_xact_xxx_tables , pg_stat_xact_user_functions(set track_functions='pl|all|none')

```
View "pg_catalog.pg_stat_xact_all_tables"
    Column      |  Type   | Modifiers
----------------+---------+-----------
 relid          | oid     |
 schemaname     | name    |
 relname        | name    |
 seq_scan       | bigint  |
 seq_tup_read   | bigint  |
 idx_scan       | bigint  |
 idx_tup_fetch  | bigint  |
 n_tup_ins      | bigint  |
 n_tup_upd      | bigint  |
 n_tup_del      | bigint  |
 n_tup_hot_upd  | bigint  |
```

SKY mobi
新手机 新应用 新娱乐

管理改进

- stats_reset in database-level and background writer statistics views
- Number of vacuum and analyze ops in pg_stat_xxx_tables
- pg_stat_bgwriter : number of times a backend fsyncs a buffer
- Auto-tuning wal_buffers
- Porting Contrib to Extension
- 参考
  - http://www.postgresql.org/docs/9.1/static/monitoring-stats.html

SKY mobi
新手机 新应用 新娱乐

管理改进

- pg_stat_xxx_xxxs

```
          View "pg_catalog.pg_stat_all_tables"
      Column       |           Type           | Modifiers
-------------------+--------------------------+-----------
 relid             | oid                      |
 schemaname        | name                     |
 relname           | name                     |
 seq_scan          | bigint                   |
 seq_tup_read      | bigint                   |
 idx_scan          | bigint                   |
 idx_tup_fetch     | bigint                   |
 n_tup_ins         | bigint                   |
 n_tup_upd         | bigint                   |
 n_tup_del         | bigint                   |
 n_tup_hot_upd     | bigint                   |
 n_live_tup        | bigint                   |
 n_dead_tup        | bigint                   |
 last_vacuum       | timestamp with time zone |
 last_autovacuum   | timestamp with time zone |
 last_analyze      | timestamp with time zone |
 last_autoanalyze  | timestamp with time zone |
 vacuum_count      | bigint                   |
 autovacuum_count  | bigint                   |
 analyze_count     | bigint                   |
 autoanalyze_count | bigint                   |
```

SKY mobi
新手机 新应用 新娱乐

管理改进

- pg_stat_bgwriter :
  - checkpoints_timed : Number of times the background writer has started timed checkpoints (because the checkpoint_timeout time has expired)
  - checkpoints_req : Number of times the background writer has started checkpoints based on requests from backends because the checkpoint_segments has been exceeded or because the CHECKPOINT command has been issued
  - buffers_checkpoint : Number of buffers written by the background writer during checkpoints
  - buffers_clean : Number of buffers written by the background writer for routine cleaning of dirty pages
  - maxwritten_clean : Number of times the background writer has stopped its cleaning scan because it has written more buffers than specified in the bgwriter_lru_maxpages parameter
  - buffers_backend : Number of buffers written by backends because they needed to allocate a new buffer
  - buffers_backend_fsync : how many times those backends had to execute their own fsync calls (normally the background writer handles those even when the backend does its own write)
  - buffers_alloc : Total number of buffer allocations
  - stats_reset : Time of the last statistics reset for the background writer, updated when executing pg_stat_reset_shared('bgwriter') on the database cluster.

SKY mobi
新手机 新应用 新娱乐

恢复增强

- Add functions to control streaming replication replay
- allows a recovery server to be queried to check whether the recovery point is the one desired
- Add the ability to create named restore points
- Allow standby recovery to switch to a new timeline automatically
- allows external cluster management software to control whether the database server restarts or not

SKY mobi
新手机 新应用 新娱乐

SQL增强

- ISOLATION LEVEL SERIALIZABLE
- data-modification commands (INSERT/UPDATE/DELETE) in WITH
- Allow non-group by columns in the query target list when the pk is in the group by clause
- Enum type can use alter type add new values
- Transaction level advisory locks
- Copy to/from add encoding option
- Explain verbose add functionScan

SKY mobi
新手机 新应用 新娱乐

示例 Per-column collation

- digoal=> create table t_posix (id int,info text collate "POSIX");
- digoal=> create table t_c (id int,info text collate "C");
- digoal=> create table t_utf8 (id int,info text collate "zh_CN");
- digoal=> create table t_en_US (id int,info text collate "en_US");

```
      Table "digoal.t_zh_cn"
 Column |  Type   |   Modifiers
--------+---------+---------------
 id     | integer |
 info   | text    | collate zh_CN
```

```
      Table "digoal.t_en_us"
 Column |  Type   |   Modifiers
--------+---------+---------------
 id     | integer |
 info   | text    | collate en_US
```

```
      Table "digoal.t_posix"
 Column |  Type   |   Modifiers
--------+---------+---------------
 id     | integer |
 info   | text    | collate POSIX
```

```
      Table "digoal.t_c"
 Column |  Type   | Modifiers
--------+---------+-----------
 id     | integer |
 info   | text    | collate C
```

- digoal=> select * from t_zh_cn order by info collate "en_US";

- 参考
- http://blog.163.com/digoal@126/blog/static/16387704020117300354723 6/

SKY mobi
新手机 新应用 新娱乐

示例 extension

- Before 9.1
  - cd postgresql-9.0.4/contrib/dblink
  - make
  - su make install PG_CONFIG=/opt/pgsql/bin/pg_config
  - psql -h 127.0.0.1 -U postgres -d digoal -f /opt/pgsql/share/contrib/dblink.sql
- 9.1
  - psql -h 127.0.0.1 -U postgres -d digoal
  - digoal=# create extension dblink;

```
cd /opt/pgsql/share/extension/
ll|grep dblink
 Oct 21 16:49 dblink--1.0.sql
 Oct 21 16:49 dblink.control
 Oct 21 16:49 dblink--unpackaged--1.0.sql
```

SKY mobi
新手机 新应用 新娱乐

示例 FDW

- PostgreSQL TO PostgreSQL
- FILE TO PostgreSQL
- ORACLE TO PostgreSQL
- REDIS TO PostgreSQL

- 参考
- http://blog.163.com/digoal@126/blog/static/16387704020114164114831 1/
- http://blog.163.com/digoal@126/blog/static/16387704020118151162340/
- http://blog.163.com/digoal@126/blog/static/16387704020118150533 1588/
- http://www.postgresql.org/docs/9.1/static/contrib-dblink-connect.html
- http://blog.163.com/digoal@126/blog/static/16387704020119181188247/

SKY mobi
新手机 新应用 新娱乐

PostgreSQL 2 PostgreSQL

```
HOST B : 172.16.3.33
\c digoal digoal
CREATE TABLE fdw_test (id INT);
INSERT INTO fdw_test SELECT generate_series(1,10);

HOST A : 172.16.3.40
\c digoal postgres
CREATE EXTENSION dblink;
CREATE FOREIGN DATA WRAPPER pgsql VALIDATOR postgresql_fdw_validator;
CREATE SERVER hostb FOREIGN DATA WRAPPER pgsql OPTIONS (host '172.16.3.33', dbname 'digoal', port '1921');
CREATE USER MAPPING FOR digoal SERVER hostb OPTIONS (user 'digoal', password 'digoal');
GRANT USAGE ON FOREIGN SERVER hostb TO digoal;

\c digoal digoal
SELECT dblink_connect('myconn', 'hostb');
SELECT * FROM dblink('myconn','SELECT * FROM fdw_test') AS t(id int);
 id
----
  1
  2
  3
  4
  5
  6
  7
  8
  9
 10
(10 rows)
```

SKY mobi
新手机 新应用 新娱乐

File 2 PostgreSQL

```
文件
postgres@db-digoal-> cd /home/postgres/zzz
sar -q|grep -v "^$"|grep -v "Average"|grep -v "^Linux" |sed -e 's/[[:space:]][[:space:]]*/ /g' >./test.csv
postgres@db-digoal-> ll
total 4.0K
-rw-rw-r-- 1 postgres postgres 2.9K Oct 28 14:28 test.csv
postgres@db-digoal-> less test.csv
12:00:01 AM runq-sz plist-sz ldavg-1 ldavg-5 ldavg-15
12:10:01 AM 0 199 0.00 0.00 0.00
12:20:01 AM 0 187 0.23 0.06 0.02

\c digoal postgres
CREATE EXTENSION file_fdw;
CREATE FOREIGN DATA WRAPPER fdw_file handler file_fdw_handler VALIDATOR file_fdw_validator;
CREATE SERVER file_server FOREIGN DATA WRAPPER file_fdw;
CREATE FOREIGN TABLE digoal.ext_sar (
  crt_time text,am_pm text,runq_sz int, plist_sz  int, ldavg_1 numeric,  ldavg_5 numeric, ldavg_15 numeric
  ) SERVER file_server
  OPTIONS (format 'csv', filename '/home/postgres/zzz/test.csv', header 'true', delimiter ' ', null '');
GRANT SELECT ON digoal.ext_sar TO digoal;
\c digoal digoal
SELECT * FROM ext_sar LIMIT 2;
 crt_time | am_pm | runq_sz | plist_sz | ldavg_1 | ldavg_5 | ldavg_15
----------+-------+---------+----------+---------+---------+----------
 12:10:01 | AM    |       0 |      199 |    0.00 |    0.00 |     0.00
 12:20:01 | AM    |       0 |      187 |    0.23 |    0.06 |     0.02
(2 rows)
```

SKY mobi
新手机 新应用 新娱乐

Oracle 2 PostgreSQL

```
sqlplus /nolog
conn digoal/digoal_123
create table tbl_oracle_fdw (id int,firstname varchar2(32),lastname varchar2(32),crt_time date);
insert into tbl_oracle_fdw values(1,'zhou','digoal',sysdate);
insert into tbl_oracle_fdw values(2,'周','德哥',sysdate);
commit;

\c digoal postgres
CREATE EXTENSION oracle_fdw;
CREATE SERVER ora FOREIGN DATA WRAPPER oracle_fdw OPTIONS (dbserver '//172.16.x.xxx:1521/digoal');
CREATE USER MAPPING FOR digoal SERVER ora OPTIONS (user 'digoal',password 'digoal_123');
# plan_costs如果为true, 需要ORACLE用户有查询v$SQL的权限
CREATE FOREIGN TABLE tbl_oracle_fdw
   (id int, firstname varchar(32), lastname varchar(32), crt_time timestamp without time zone)
   SERVER ora OPTIONS (table 'tbl_oracle_fdw',schema 'digoal',plan_costs 'true');
GRANT SELECT ON tbl_oracle_fdw TO digoal;
\c digoal digoal
digoal=> select * FROM tbl_oracle_fdw;
 id | firstname | lastname |      crt_time
----+-----------+----------+---------------------
  1 | zhou      | digoal   | 2011-09-15 10:55:32
  2 | 周        | 德哥     | 2011-09-15 11:18:41
(2 rows)
如果配置出现问题, 需要先中断连接.
digoal=# select oracle_close_connections();
```

SKY mobi
新手机 新应用 新娱乐

Redis 2 PostgreSQL

```
CREATE EXTENSION redis_fdw;
CREATE SERVER redis_server1
  FOREIGN DATA WRAPPER redis_fdw
  OPTIONS (address '172.xxx.xxx.xxx', port '6379');
CREATE USER MAPPING FOR digoal
  SERVER redis_server1 OPTIONS (password 'DIGOAL');
GRANT USAGE ON FOREIGN SERVER redis_server1 TO digoal;
CREATE FOREIGN TABLE redis_db0 (key text, value text)
  SERVER redis_server1 OPTIONS (database '0');
CREATE FOREIGN TABLE redis_db1 (key text, value text)
  SERVER redis_server1 OPTIONS (database '1');
digoal=> select * from redis_db0 limit 5;
          key            | value
-------------------------+-------
 visits:digoal13:totals  | 13
 visits:digoal23:totals  | 23
 visits:digoal33:totals  | 33
```

SKY mobi
新手机 新应用 新娱乐

示例 Serializable isolation Level

- Oracle 实现的Serializable级别描述(摘自Oracle10G手册)：
  - The SERIALIZABLE setting specifies serializable transaction isolation mode as defined in the SQL92 standard. If a serializable transaction contains data manipulation language (DML) that attempts to update any resource that may have been updated in a transaction uncommitted at the start of the serializable transaction, then the DML statement fails.
- PostgreSQL 8的版本已经支持Oracle提到的Serializable级别.
- PostgreSQL测试：
  - Session A :
  - create table isolate_test(id int,info text);
  - insert into isolate_test values (1,'digoal');
  - begin;
  - update isolate_test set id=2 where id=1;
  - Session B :
  - begin isolation level serializable; select 1;
  - Session A : commit;
  - Session B :
  - update isolate_test set id=3 where id=1;
  - ERROR: could not serialize access due to concurrent update

SKY mobi
新手机 新应用 新娱乐

示例 Serializable isolation Level

- PostgreSQL 9.1实现了真正的serializable的隔离级别.
  - All statements of the current transaction can only see rows committed before the first query or data-modification statement was executed in this transaction. If a pattern of reads and writes among concurrent serializable transactions would create a situation which could not have occurred for any serial (one-at-a-time) execution of those transactions, one of them will be rolled back with a serialization_failure SQLSTATE.
- 测试 :
  - truncate table isolate_test ;
  - insert into isolate_test values (1,'digoal');
  - insert into isolate_test values (2,'digoal');
  - SESSION A :
  - begin transaction isolation level serializable read write; select 1;
  - SESSION B :
  - begin transaction isolation level serializable read write; select 1;
  - SESSION A :
  - update isolate_test set id=11 where id=1;
  - UPDATE 1
  - select * from isolate_test where id=2;

SKY mobi
新手机 新应用 新娱乐

示例 Serializable isolation Level

- 测试数据(续) :
  - id | info
  - ----+--------
  - 2 | digoal
  - SESSION B :
  - update isolate_test set id=22 where id=2;
  - UPDATE 1
  - select * from isolate_test where id=1;
  - id | info
  - ----+--------
  - 1 | digoal
  - SESSION A/B : commit;
  - COMMIT
  - SESSION B/A :
  - commit;
  - ERROR: could not serialize access due to read/write dependencies among transactions
  - DETAIL: Reason code: Canceled on identification as a pivot, during commit attempt.
  - HINT: The transaction might succeed if retried.
- 参考
  - http://blog.163.com/digoal@126/blog/static/163877040201181629506 91/
  - http://blog.163.com/digoal@126/blog/static/163877040201192715948181/

SKY mobi
新手机 新应用 新娱乐

示例 UNLOGGED TABLE

- create UNLOGGED table unlogged_test (id int);
- create table logged_test (id int);
- select pg_xlogfile_name(pg_current_xlog_location());
  - 000000010000001C00000017
- insert into unlogged_test select generate_series(1,50000000);
  - INSERT 0 50000000
  - Time: 33527.923 ms
- select pg_xlogfile_name(pg_current_xlog_location());
  - 000000010000001C00000017
- insert into logged_test select generate_series(1,50000000);
  - INSERT 0 50000000
  - Time: 62378.872 ms
- select pg_xlogfile_name(pg_current_xlog_location());
  - 000000010000001D00000008

SKY mobi
新手机 新应用 新娱乐

示例 WITH

- 摘自PostgreSQL 9.1 手册
- http://www.postgresql.org/docs/9.1/static/queries-with.html

```
WITH moved_rows AS (
    DELETE FROM products
    WHERE
        "date" >= '2010-10-01' AND
        "date" < '2010-11-01'
    RETURNING *
)
INSERT INTO products_log
SELECT * FROM moved_rows;
```

SKY mobi
新手机 新应用 新娱乐

示例 Transaction level advisory locks

- PostgreSQL9.1新增事务级的advisory lock.
- 背景：
  - 长时间持有锁带来的数据库问题(Oracle里面可能导致snapshot too old,PG里面可能导致某些dead tuple空间无法释放),从而诞生了advisory lock.
- advisory lock 应用场景：
  - 比如数据库里面存储了文件和ID的对应关系，应用程序需要长时间得获得一个锁，然后对文件进行修改，再释放锁。
  - SESSION A:
  - digoal=> select pg_advisory_lock(id),file_path from tbl_file_info where id=1;
  - pg_advisory_lock | file_path
  - ----------+----------
  - | /home/postgres/advisory_lock_1.txt
  - (1 row)
  - 应用程序对/home/postgres/advisory_lock_1.txt文件进行编辑之后，再释放这个advisory锁。
  - SESSION B:
  - 当SESSIONA在编辑/home/postgres/advisory_lock_1.txt这个文件的时候，无法获得这个锁，所以可以确保不会同时编辑这个文件。

SKY mobi
新手机 新应用 新娱乐

示例 Transaction level advisory locks

- 如果使用行锁 :
- SESSION A:
- digoal=> begin;
- BEGIN
- digoal=> select file_path from tbl_file_info where id=1 for update;
  - /home/postgres/advisory_lock_1.txt
- 应用程序对/home/postgres/advisory_lock_1.txt文件进行编辑之后，再释放这个锁。
- digoal=> end;

- SESSION A 在编辑文件的时候，需要保持这个事务，一方面降低了数据库使用的并发性，另一方面带来了写操作的长事务，导致这段时间产生的DEAD TUPLE存储空间无法回收利用。

- 参考
- http://blog.163.com/digoal@126/blog/static/163877040201172492217830/
- http://blog.163.com/digoal@126/blog/static/163877040201011162912604/

SKY mobi
新手机 新应用 新娱乐

示例 Nearest-neighbor search

- PostgreSQL 9.1 新增了gist 索引按距离排序的操作符(order by operator)
- create table gist_test (id serial primary key,location point);
- insert into gist_test (location) select cast ('(1,'||generate_series(1,1000000)||')' as point);
- create index idx_gist_test_1 on gist_test using gist (location point_ops);

- explain verbose select * from gist_test order by location <-> cast ('(1,2)' as point) limit 10;
- QUERY PLAN
- Limit (cost=0.00..0.50 rows=10 width=20)
- Output: id, location, ((location <-> '(1,2)'::point))
- -> Index Scan using idx_gist_test_1 on digoal.gist_test (cost=0.00..50374.43 rows=1000000 width=20)
- Output: id, location, (location <-> '(1,2)'::point)
- Order By: (gist_test.location <-> '(1,2)'::point)

- 参考
- http://blog.163.com/digoal@126/blog/static/163877040201191810141210 5/

SKYMOBI 新手机 新应用 新娱乐

示例 pg_trgm 模糊查询,相似匹配

- 应用场景(忽略大小写, 修正拼写错误) :
- 忽略大小写 : SKYMOBI 和 skyMOBI
- 修正拼写错误 : digoal 和 dagoal
- digoal=> select * from trgm_test where info <-> 'skyMOBI' < 0.7;
- id | info
- ----+---------
- 1 | SKYMOBI

- digoal=> select * from trgm_test where info <-> 'dagoal' < 0.7;
- id | info
- ----+--------
- 2 | digoal

- create index idx_trgm_1 on trgm_test using gist (info gist_trgm_ops);
- 参考
- http://blog.163.com/digoal@126/blog/static/16387704020119188255380 3/

SKY mobi
新手机 新应用 新娱乐

示例 pg_trgm 模糊查询,相似匹配

- 字符串分组拆分.
- digoal=> select show_trgm('digoal');
- {" d"," di","al ",dig,goa,igo,oal}

- digoal=> select show_trgm('dagoal');
- {" d"," da",ago,"al ",dag,goa,oal}

- digoal=> select similarity('digoal','dagoal');
- 0.4

- digoal=> select 4/10.0;
- 0.40000000000000000000

- 相似度 = (overlaps的分组个数 / 总分组数)

SKY mobi
新手机 新应用 新娱乐

示例 pg_trgm 模糊查询,相似匹配

- 模糊查询
- digoal=> select show_limit();
- 0.3
- digoal=> select 'digoal' % '%Di%';
- f
- digoal=> select 'digoal' % '%Dig%';
- t
- digoal=> select set_limit(0.4);
- 0.4
- digoal=> select 'digoal' % '%Dig%';
- f
- digoal=> select set_limit(0.1);
- 0.1
- digoal=> select 'digoal' % '%Di%';
- t
- digoal=> select 'digoal' % '%D%';
- t

SKY mobi
新手机 新应用 新娱乐

示例 pg_bench

- 添加report per-statement latencies

- transaction type: Custom query
- scaling factor: 1
- query mode: extended
- number of clients: 8
- number of threads: 8
- duration: 180 s
- number of transactions actually processed: 855429
- tps = 4752.346080 (including connections establishing)
- tps = 4752.383363 (excluding connections establishing)
- statement latencies in milliseconds:
- 0.002288 \setrandom userid 0 50000000
- 1.677890 SELECT f_user_logout(:userid);

SKY mobi
新手机 新应用 新娱乐

示例 pg_test_fsync

- 添加了O_DIRECT测试支持

```
postgres@db5-> pg_test_fsync -o 160
160 operations per test
O_DIRECT supported on this platform for open_datasync and open_sync.

Compare file sync methods using one 8kB write:
(in wal_sync_method preference order, except fdatasync
is Linux's default)
        open_datasync                                  n/a
        fdatasync                         157.925 ops/sec
        fsync                             156.809 ops/sec
        fsync_writethrough                             n/a
        open_sync                         165.424 ops/sec

Compare file sync methods using two 8kB writes:
(in wal_sync_method preference order, except fdatasync
is Linux's default)
        open_datasync                                  n/a
        fdatasync                         151.582 ops/sec
        fsync                             116.479 ops/sec
        fsync_writethrough                             n/a
        open_sync                          82.551 ops/sec

Compare open_sync with different write sizes:
(This is designed to compare the cost of writing 16kB
in different write open_sync sizes.)
        16kB open_sync write              159.632 ops/sec
         8kB open_sync writes              82.296 ops/sec
         4kB open_sync writes              44.675 ops/sec
         2kB open_sync writes              19.838 ops/sec
         1kB open_sync writes              10.126 ops/sec

Test if fsync on non-write file descriptor is honored:
(If the times are similar, fsync() can sync data written
on a different descriptor.)
        write, fsync, close               159.365 ops/sec
        write, close, fsync               162.906 ops/sec

Non-Sync'ed 8kB writes:
        write                          169491.525 ops/sec
```

SKY mobi
新手机 新应用 新娱乐

讨论

- Thanks !

- 参考
  - http://www.postgresql.org/docs/9.1/static/release-9-1.html

- Author : Digoal.Zhou
- MSN : zzzqware@hotmail.com
- QQ : 276732431
- Email : digoal@126.com
- BLOG : http://blog.163.com/digoal@126/