9.1 Mystery Tour

Josh Berkus
pgCon 2011
May 2011, Ottawa

PGX
POSTGRESQL
EXPERTS, INC.

Home

Replication
Tools

Synchronous
Replication

Per-Column
Collations

Other
Features

SE-Postgres

wCTEs
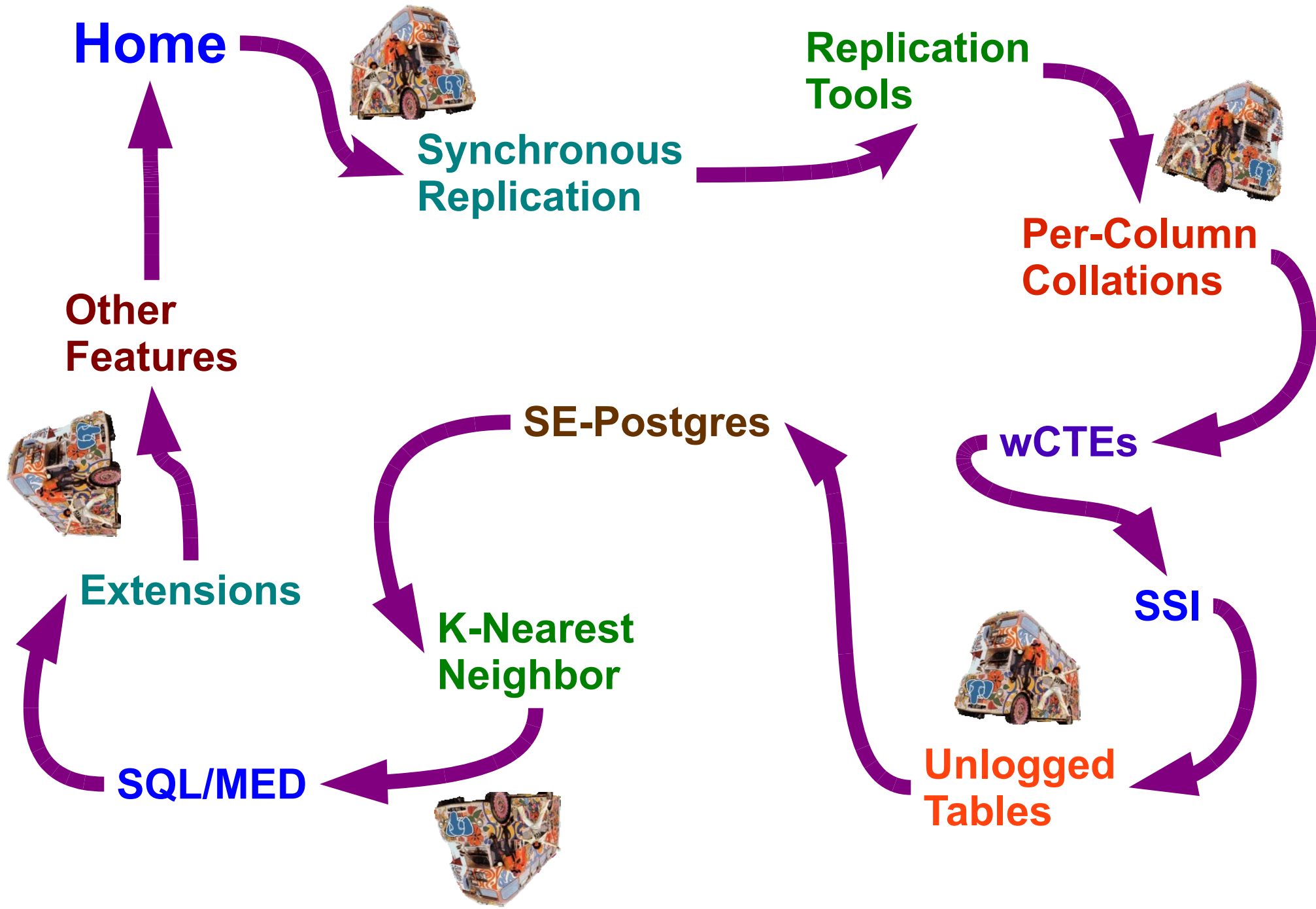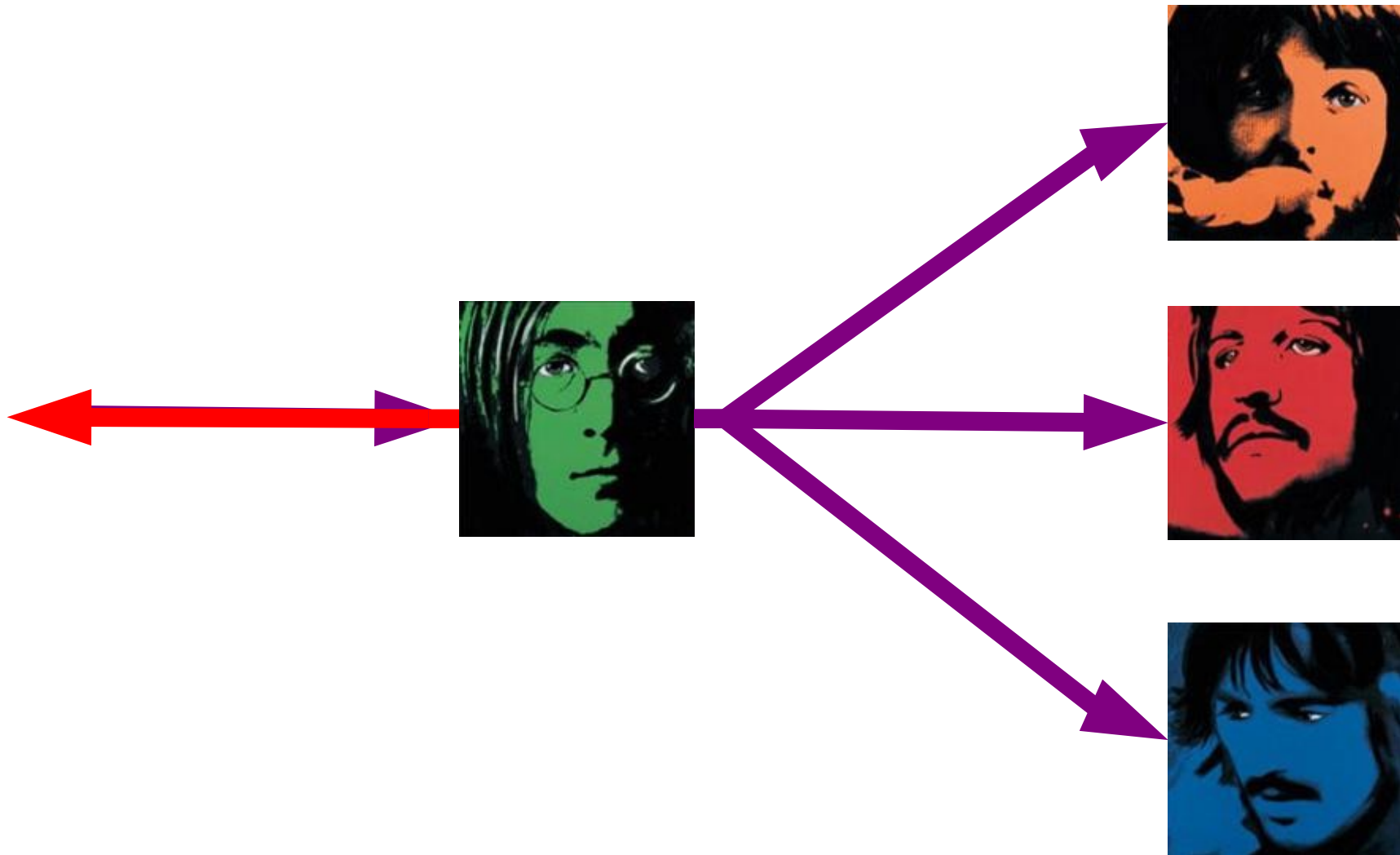
Extensions
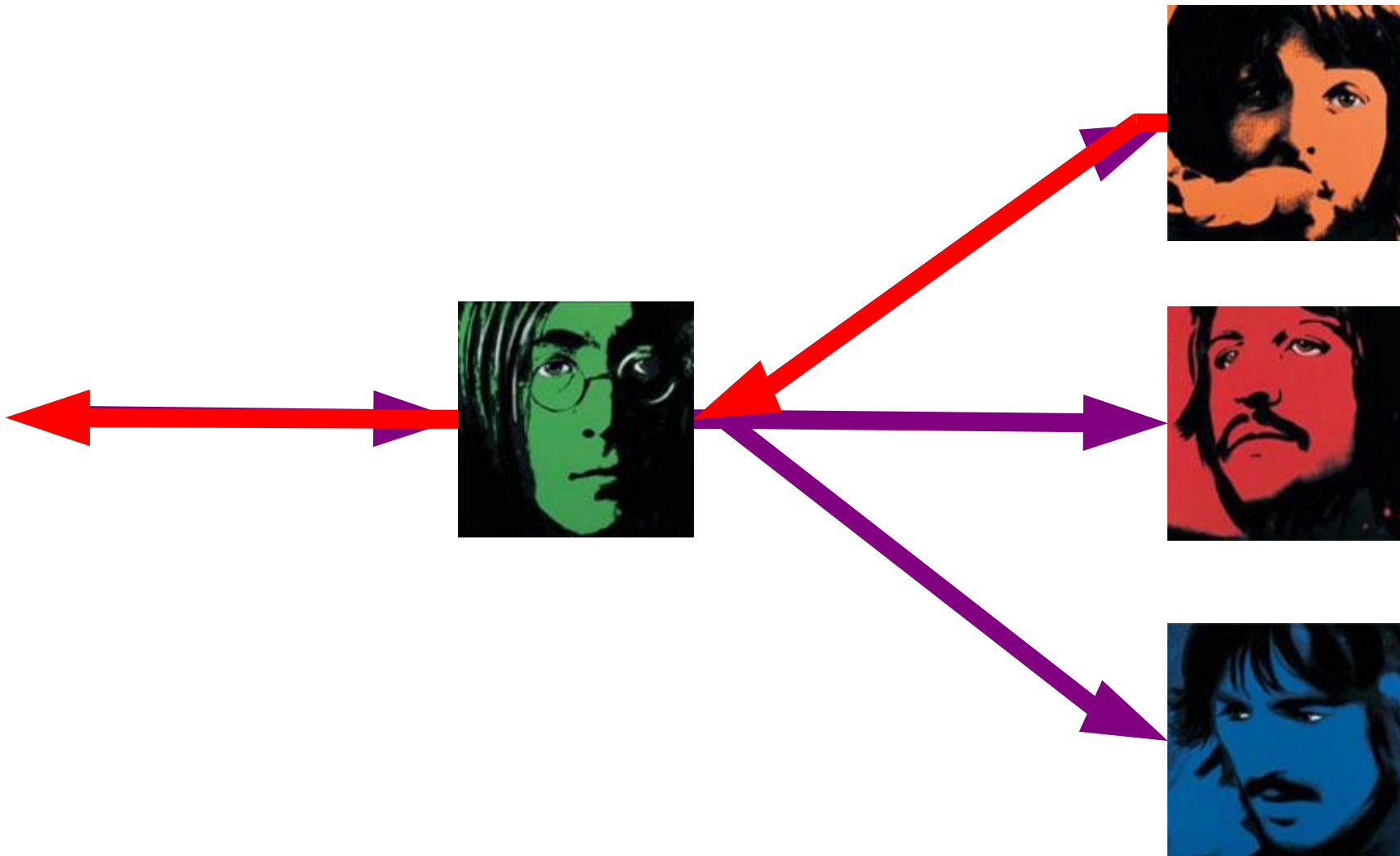
K-Nearest
Neighbor

SSI

SQL/MED

Unlogged
Tables

Synchronous Replication

Synchronous Replication

Synchronous Replication

standby Paul:
```
primary_conninfo = 'host=john
  port=5432 user=replication
  application_name=paul'
```

master John:
```
synchronous_commit = on
synchronous_standby_names =
  'paul,george,ringo'
```

Synchronous Replication

```
postgres=# select application_name as appname, state,
sync_state, sync_priority as prty, replay_location
from pg_stat_replication;
 appname |    state    |  sync_state  | prty | replay_loc
---------+-------------+--------------+------+------------
 paul    | STREAMING   | SYNC         |    1 | 0/29154EB8
 george  | STREAMING   | POTENTIAL    |    2 | 0/291452F0
 ringo   | STREAMING   | POTENTIAL    |    3 | 0/29154EB8
```

Transaction Control

```
# set synchronous_commit = on;
# begin;

# update user_balance
# set balance = balance - 300
# where user = 14301;

# update user_balance
# set balance = balance + 300
# where user = 13221;

# commit;
```

Transaction Control

```
# set synchronous_commit = 'local';
# insert into user_messages
# values ( 14301, 13221,
# 'hello a/s/l?' );
```

New Replication Tools

- pg_stat_replication view
    - shows all replicas and their statuses
- pg_basebackup
    - single-command cloning over port 5432
- re-mastering
    - promote a replica to be the new master
- new functions
    - pg_last_xact_replay_timestamp()
    - pg_xlog_replay_pause()

Per-Column Collations

ID	english	日本語
787	error	エラー
12	unique	ユニークな
456	index	インデックス
333	documentation	ドキュメンテーション
217	open source	オープンソース
2134	penguin	ペンギン

Per-Column Collations

```
$ create table collated_polish (some_text
text collate "pl_PL.utf8");
$ copy collated_polish from '/tmp/polish';

$ select * from collated_polish order by
some_text;
some_text
-----------
alfa
car
coś
ćma
lama
łódka
```

Psycadelic Queries

Writable CTEs

```sql
WITH deleted_posts AS (
   DELETE FROM posts
   WHERE created < now()
      - '6 months'::INTERVAL
   RETURNING *
)
SELECT user_id, count(*) FROM
deleted_posts group BY 1;
```

Writable CTEs

```
WITH deleted_posts AS (
   DELETE FROM posts
   WHERE created < now()
      - '6 months'::INTERVAL
   RETURNING *
),
deleted_per_user as (
   SELECT user_id, count(*)
   FROM deleted_posts
   GROUP BY 1
)
UPDATE counts
SET posts_count = posts_count - d.count
FROM deleted_per_user d
WHERE d.user_id = counts.user_id;
```

Serialized Snapshot Isolation

- Allows true serialization of concurrent transactions

- No more explicit locking!

- No more deadlocks!

- As long as each transaction works, all will work.

- Including *limited predicate locking*.

SSI

```
update othello set
pieces = 'white'
where pieces =
'black';


commit;

ERROR:  could not
serialize access
due to read/write
dependencies among
transactions
```

```
update othello set
pieces = 'black'
where pieces =
'white';

commit;
```

Unlogged Tables

```
CREATE UNLOGGED TABLE
cleaned_log_import
AS SELECT hit_time, page
FROM raw_hits, hit_watermark
WHERE hit_time > last_watermark
  AND is_valid(page);
```

SE-Postgres

- Now a contrib module!

  - uses "hooks" enabled at build time

- Integrates with SE-Linux

  - SE-Linux "security labels" become object permissions

  - covers all database object permissions

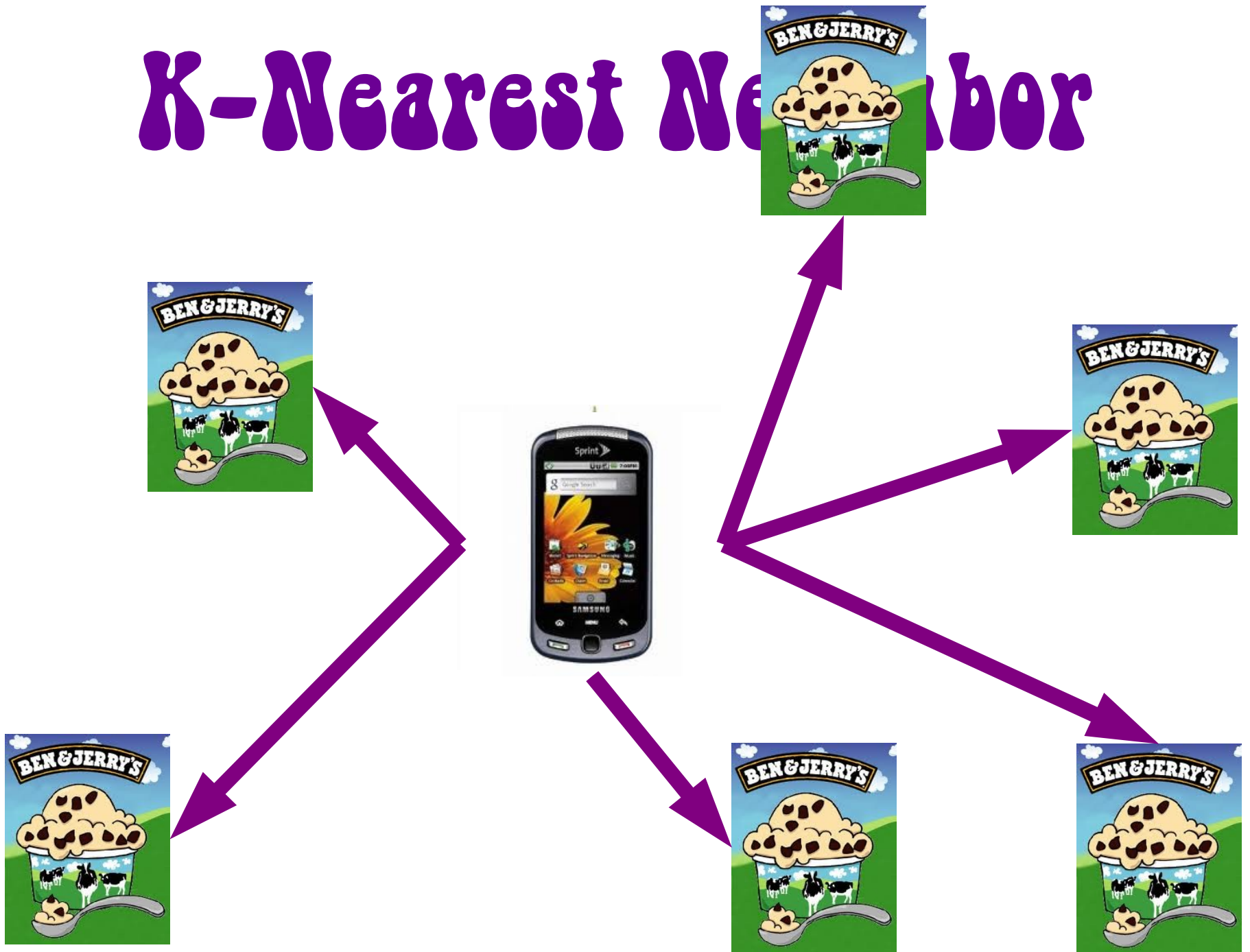- row-level access control not implemented yet

SE-Postgres

SE-Postgres

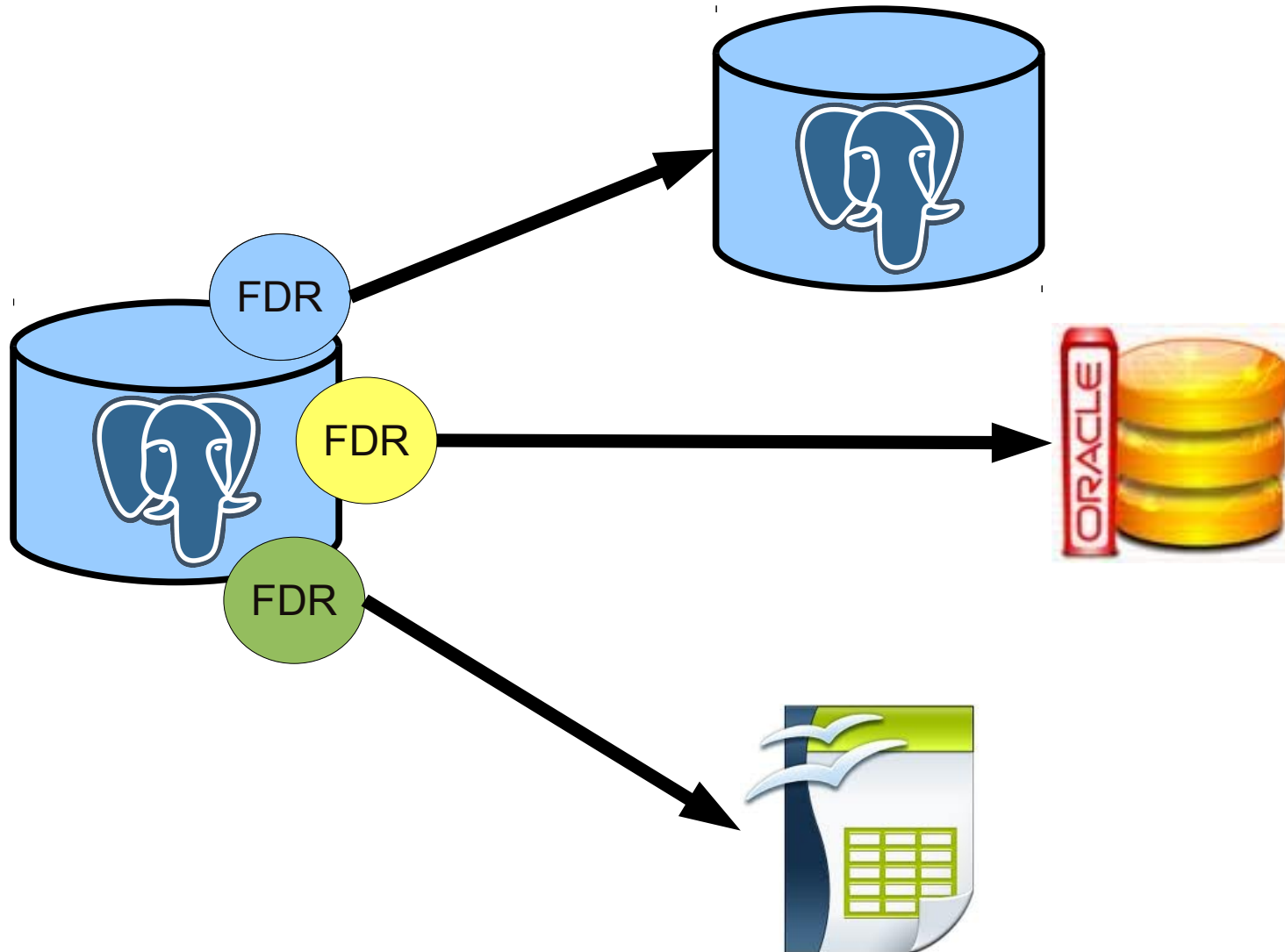THE PENTAGON

WASHINGTON
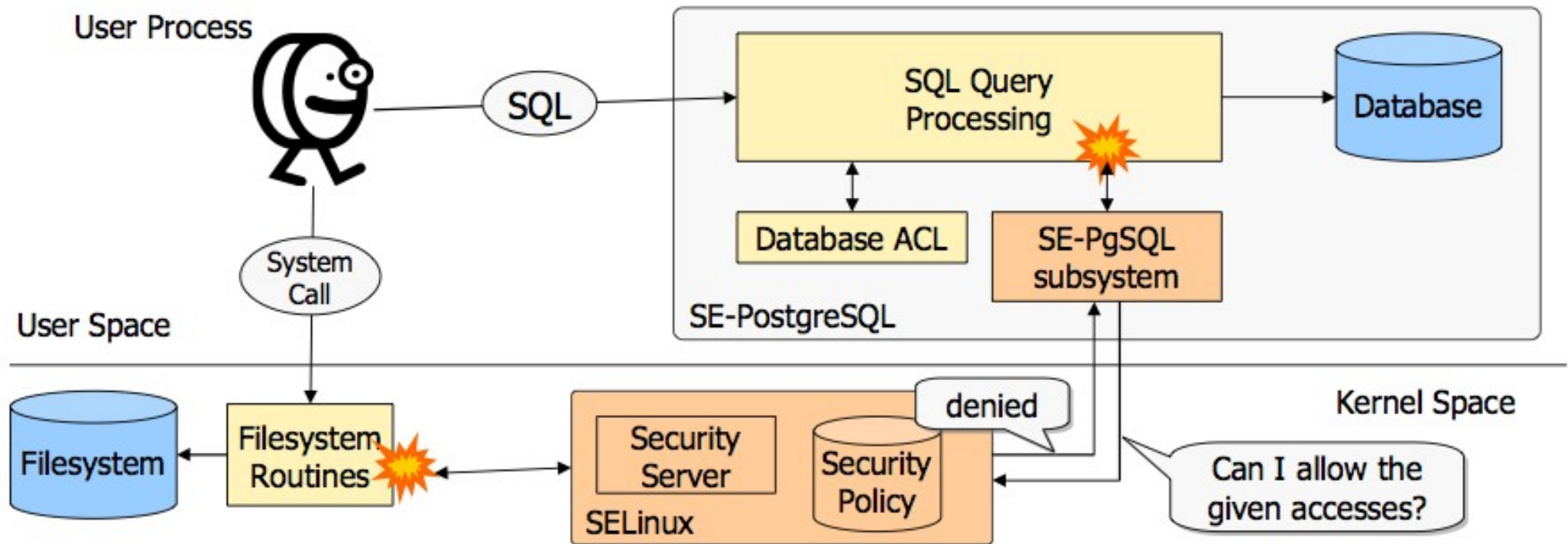
K-Nearest Neighbor

K-Nearest Neighbor

```
SELECT *,
   position <-> $my_current_location
   as distance
FROM ben_n_jerry
ORDER BY position <-> $my_current_location
LIMIT 3;
```

SQL-MED

SQL-MED

```
CREATE FOREIGN TABLE raw_hits
( hit_time TIMESTAMP,
  page TEXT )
SERVER file_fdw
OPTIONS (format 'csv', delimiter
';', filename '/var/log/hits.log');
```

SQL-MED

```
SELECT page, count(*)
FROM raw_hits
WHERE hit_time >
  '2011-04-16 16:00:00' AND
  hit_time <= '2011-04-16 17:00:00'
GROUP BY page;
```

Extensions

Extensions

```
CREATE EXTENSION IF NOT EXISTS citext
WITH SCHEMA ext;

\dx
Schema | Name   | Version   | Description

-------+--------+-----------+------------
ext    | citext | 9.1devel  | case-
                              insensitive character string type
```

Extensions

```
ALTER EXTENSION citext UPDATE TO 9.2;


DROP EXTENSION citext;
```

Extensions

PGXN
PostgreSQL Extension Network

recent   users   about   faq

analyze   api   arithmetic   data types   datatype   dictionary
explain   explain analyze   fdw   france   full-text search   gmp
hash   integer   internet   ispell   italian   key value   key value
pair   md5   node   ordered pair   pair   plan   rational
semantic version   semver   sha   sha1   statistics   table   tap
tddd   test driven database development   testing   tinyint
twitter   unit testing   variadic function   version   version

Other Features

- PL/Python overhaul

- Valid-on-creation Foreign Keys

- Extensible ENUMs

- Triggers on Views

- New Trigram implementation

- Reduced NUMERIC size

- ALTER TYPE without rewrite

- pg_dump directory format

9.1 Sessions at PgCon

- Real Federation Database System leveraging PostgreSQL FDW - Thursday 13:30

- Label-based Mandatory Access Control: Friday 10:00

- Writeable CTEs: Friday 11:00

- Distributing Extensions on PGXN: Friday 15:00

- Extensions Development: Friday 16:00

- Serializable Snapshot Isolation: Friday 16:00

Contact

- Josh Berkus: josh@pgexperts.com
  - blog: blogs.ittoolbox.com/database/soup
- PostgreSQL: www.postgresql.org
  - pgexperts: www.pgexperts.com
- Upcoming Events
  - pgOpen: September 14-16, Chicago
  - OpenSourceBridge: Portland, June

PGX
POSTGRESQL
EXPERTS, INC.