SELECT * FROM changes; – Part 2

Logical Decoding

Also known as:

Changeset Extraction
Logical Replication
Data Change Streaming
Change Data Capture
...

Streams

Andres Freund

What's that?

- The ability to get all modifications of database content (=>DML); with the following properties:
  - Consistent (COMMIT) order
  - Configurable format
  - Safe against concurrent DDL
  - Low overhead

The SQL Interface:

```
postgres=# SELECT * FROM
pg_create_logical_replication_slot(
        'my_stream', 'test_decoding');
 slotname  | xlog_position
-----------+---------------
 my_stream | 0/1BE2A18
(1 row)
```

The SQL Interface:

```
postgres=# SELECT slot_name, plugin, database FROM
pg_replication_slots;
 slot_name |    plugin     | database
-----------+---------------+----------
 my_stream | test_decoding | postgres
(1 row)
```

The SQL Interface:

```
postgres=# CREATE TABLE talks(
    talkname text primary key, description text);

postgres=# INSERT INTO talks(talkname, description)
VALUES ('SELECT * FROM changes', 'Explains logical
decoding.');

postgres=# ALTER TABLE talks ADD COLUMN review text;

postgres=# UPDATE talks SET review = 'who is going to
use this?';

postgres=# DROP TABLE talks;
```

The SQL Interface:

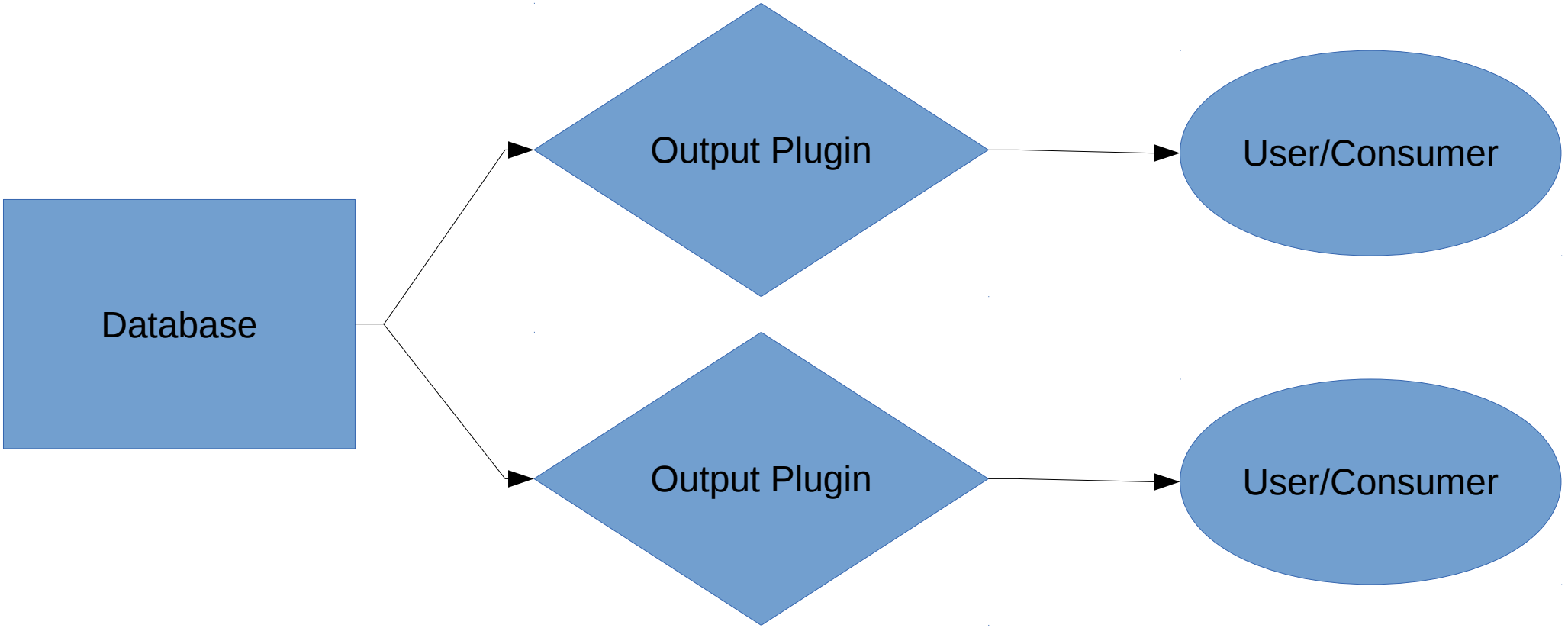
```
Postgres=#
SELECT location, data
FROM pg_logical_slot_get_changes(
     'my_stream', NULL, NULL);
 location  |  data

 0/1C41DA0 | BEGIN 1013
 0/1C41DA0 | table public.talks: INSERT:
     talkname[text]:'SELECT * FROM changes'
     description[text]:'Explains logical decoding.'
 0/1C41F1F | COMMIT 1013

 0/1C42278 | BEGIN 1015
 0/1C42278 | table public.talks: UPDATE:
     talkname[text]:'SELECT * FROM changes'
     description[text]:'Explains logical decoding.'
     review[text]:'who is going to use this?'
 0/1C4234F | COMMIT 1015

(12 rows)
```

Data Flow



Andres Freund

Output Plugins

- Transform Data
- Can restrict which data gets decoded
- User supplied
- Callback based
  - startup
  - begin
  - change
  - commit
  - shutdown
- Access to the transaction's xid, commit time, ...

Output Plugins

- test_decoding (contrib)
- Json (Euler Taveira de Oliveira)
  - https://github.com/eulerto/wal2json
- SQL (Michael Paquier)
  - http://michael.otacoo.com/postgresql-2/postgres-9-4-feature-highlight-output-plugin-logical-replication/
  - https://github.com/michaelpq/pg_plugins/tree/master/decoder_raw
- ...

Replication Identifiers

- What if my table doesn't have a primary key?
- What if I want the entire old row?
- => ALTER TABLE … REPLICA IDENTITY
  - DEFAULT
  - USING INDEX index_name
  - FULL
  - NOTHING

Use Cases

- Replication
- Cache Invalidation
- Auditing
- Aggregation/Federation/Rollup/...
- ...

How To Receive Changes

- SQL Interface
  - Simple
  - High startup costs
  - No streaming
- Walsender Interface
  - More Complex
  - Streaming and synchronous replication are supported
  - pg_recvlogical commandline client
- User written Interfaces
  - C Code

Problems

- Only committed changes are streamed
  - Otherwise much more complex to use
  - Some additional complexity needed while decoding
- Replicate DDL as well
  - Event Triggers/DDL normalization
- Additional Features
  - 2PC (optionally synchronous replication!)
  - More detailed configuration about what to replicate

What we need it for: BDR

- Asynchronous Multimaster
- Last Update Wins
- Pluggable conflict handlers
- Open Source
- Hopefully will get integrated into postgres, bit by bit.

Thanks

- Intel Security/McAfee for funding
- Community for Review (particularly Robert Haas)
- People providing consumers/output plugins (Steve Singer, Euler Taveira de Olivera, Michael Paquier)
- 2ndq for letting me work on it...