LOXODATA

THE FABULOUS DESTINY OF 0000000200000008000000BB

FOSDEM

2018-02-03

Patrick Francelle
Loxodata

LOXODATA

WHO

Patrick Francelle

- PostgreSQL consultant and trainer
- First contact with PostgreSQL in 1999
- never stopped using it
- `@pharrek`

LOXODATA

Company built on 3 essential pillars

PostgreSQL

DevOps

Cloud

LOXODATA

WHAT

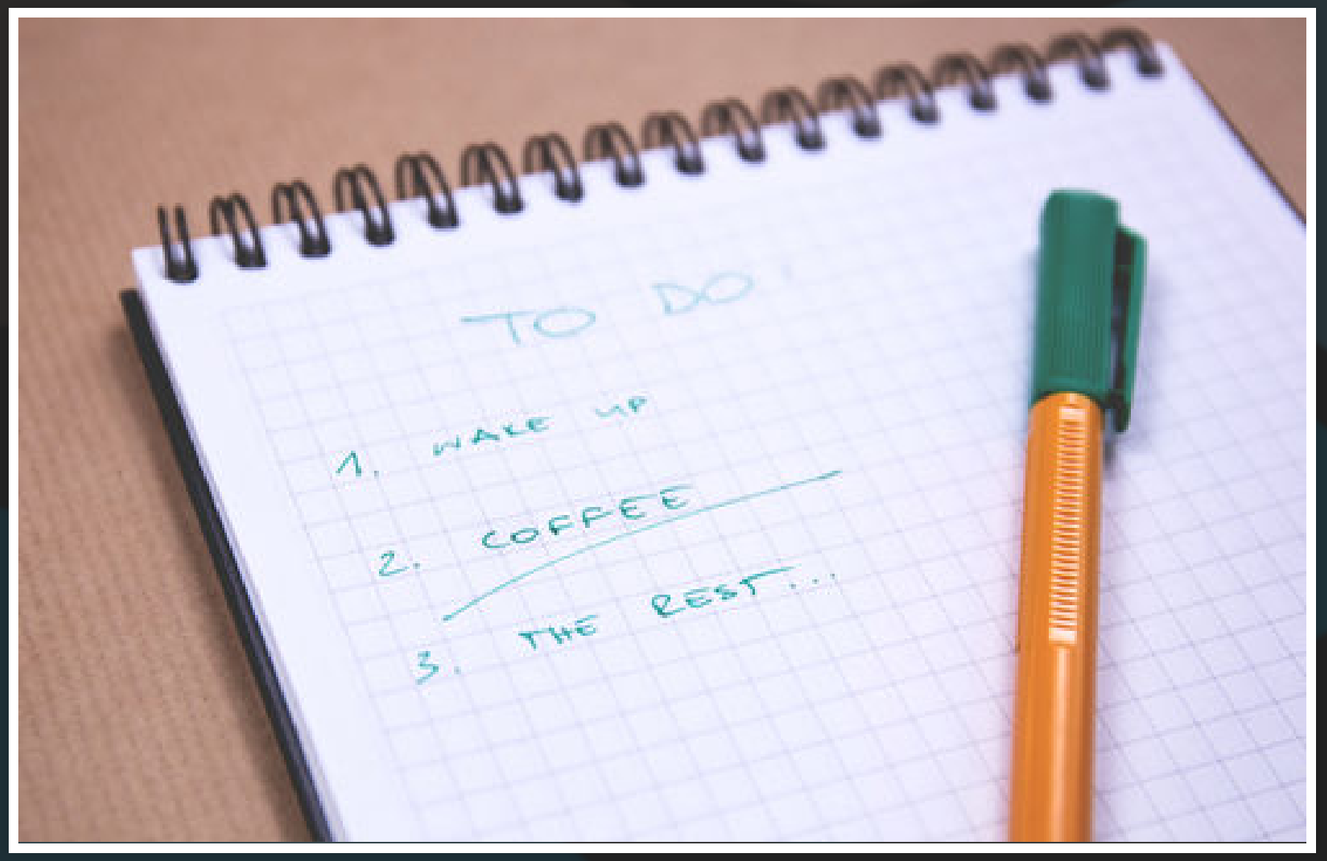
The many possible lives of a WAL

HI !

SOON, I WILL BE A WAL FILE

LOXODATA

WHAT AM I ?



- transaction log
- REDO log
- **W**rite **A**head **L**og

LOXODATA

WAL ?

- record data changes ASAP
- bring data consistency
- help restore data
- be the pillar of replication

LOXODATA

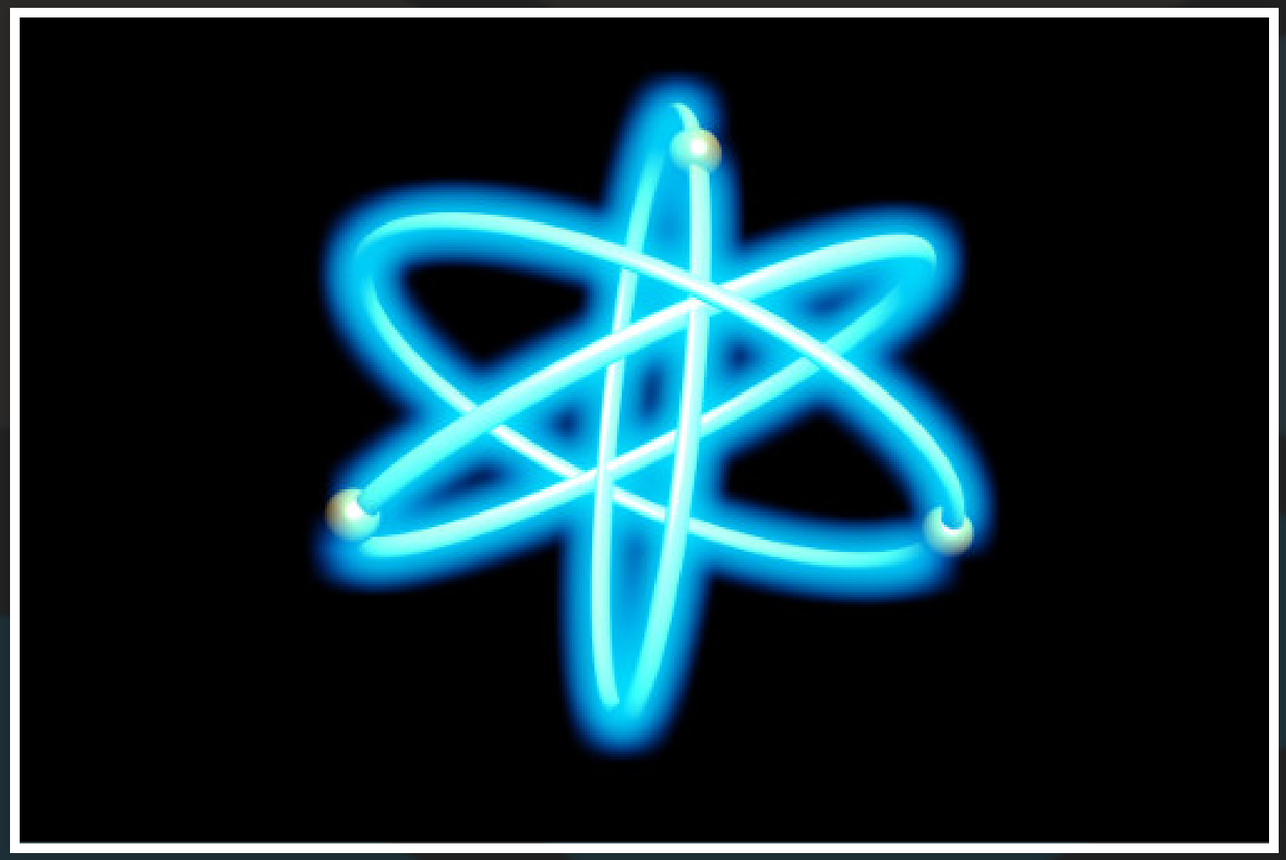
SOME THEORY

ACID

- Atomicity
- Consistency
- Isolation
- Durability

LOXODATA

SOME THEORY

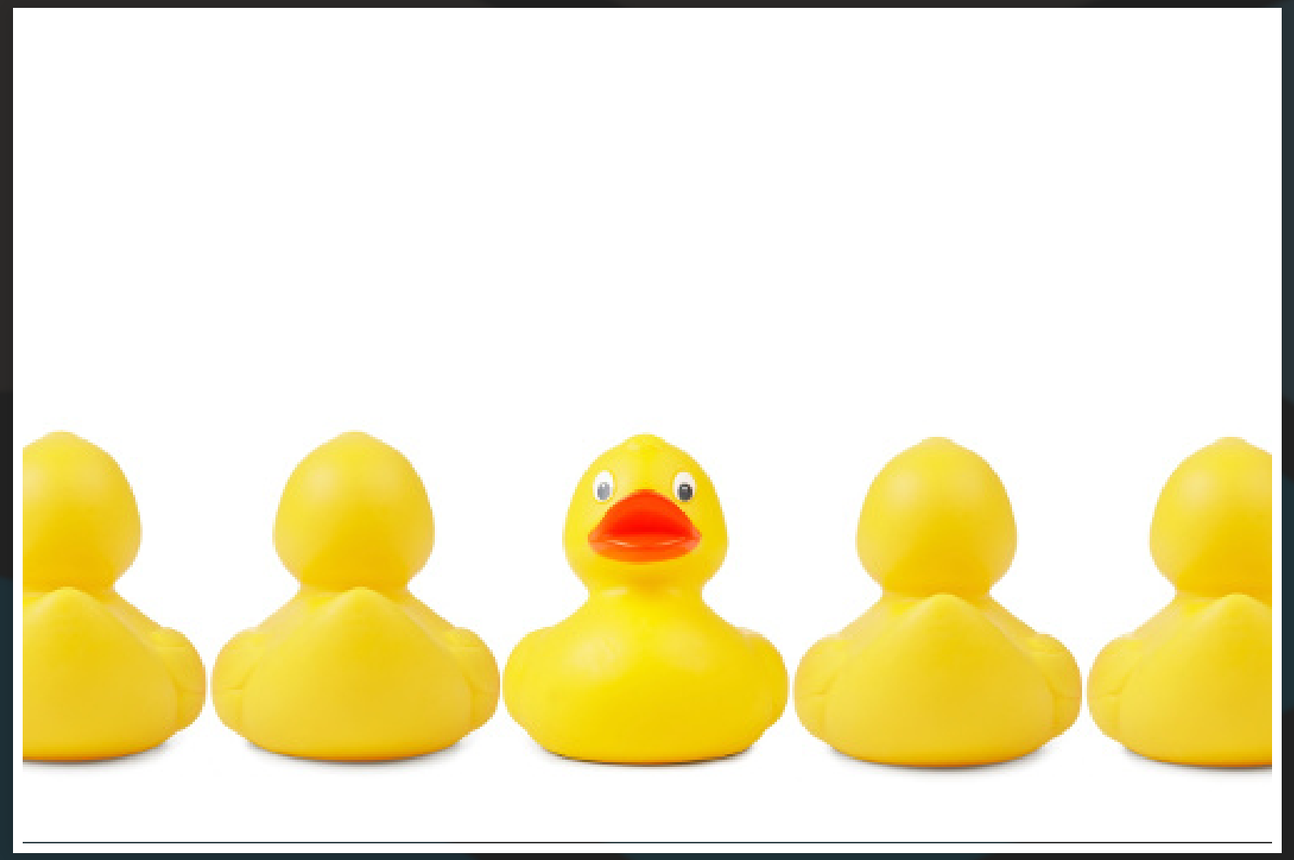


Atomicity

Atomicity requires that each transaction be "all or nothing": if one part of the transaction fails, then the entire transaction fails, and the database state is left unchanged.

LOXODATA

SOME THEORY



Consistency

The consistency property ensures that any transaction will bring the database from one valid state to another.

LOXODATA

SOME THEORY

Isolation

The isolation property ensures that the concurrent execution of transactions results in a system state that would be obtained if transactions were executed sequentially, i.e., one after the other.

LOXODATA

SOME THEORY

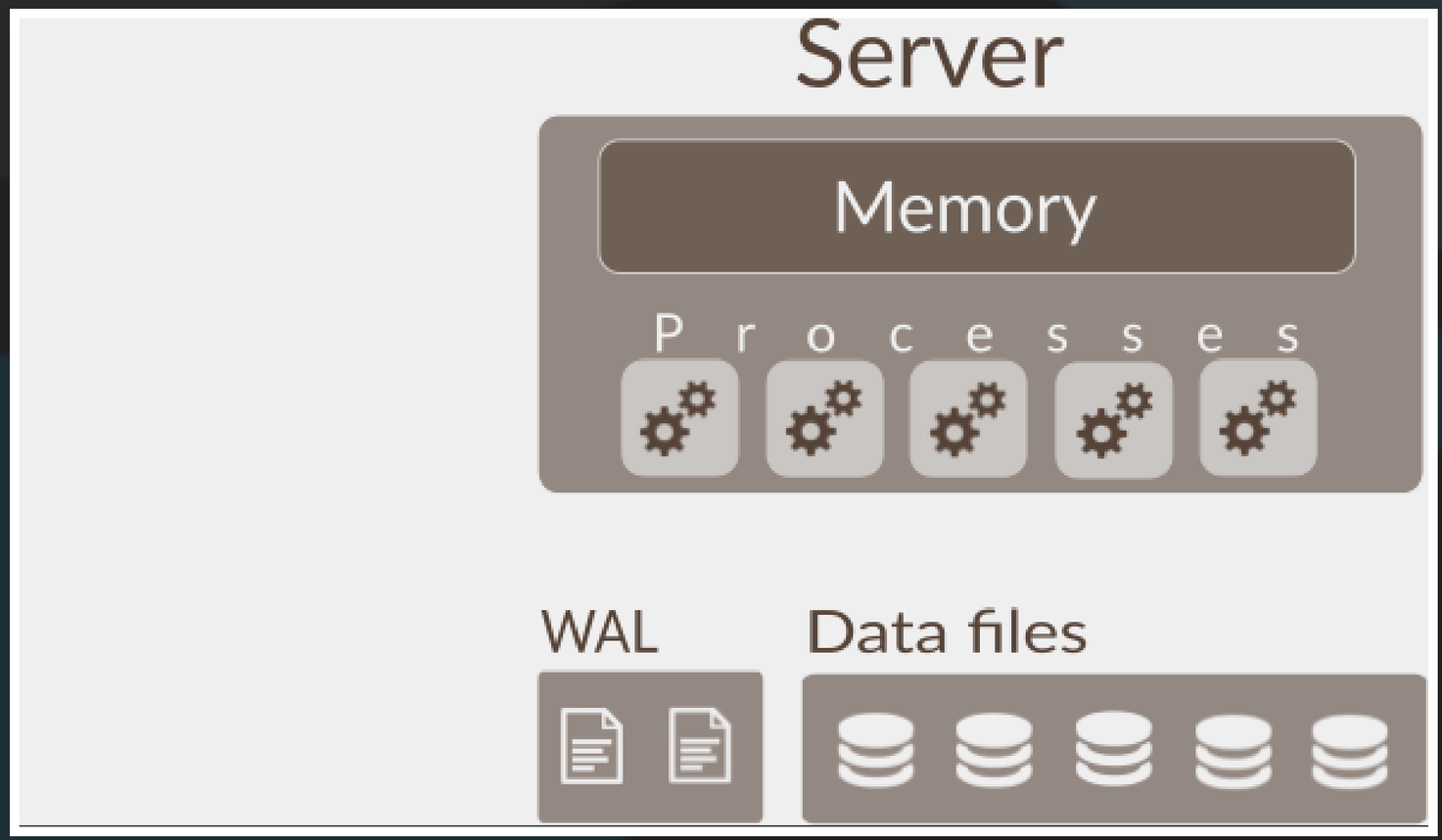
Durability

The durability property ensures that once a transaction has been committed, it will remain so, even in the event of power loss, crashes, or errors.

(All quotes from Wikipedia)

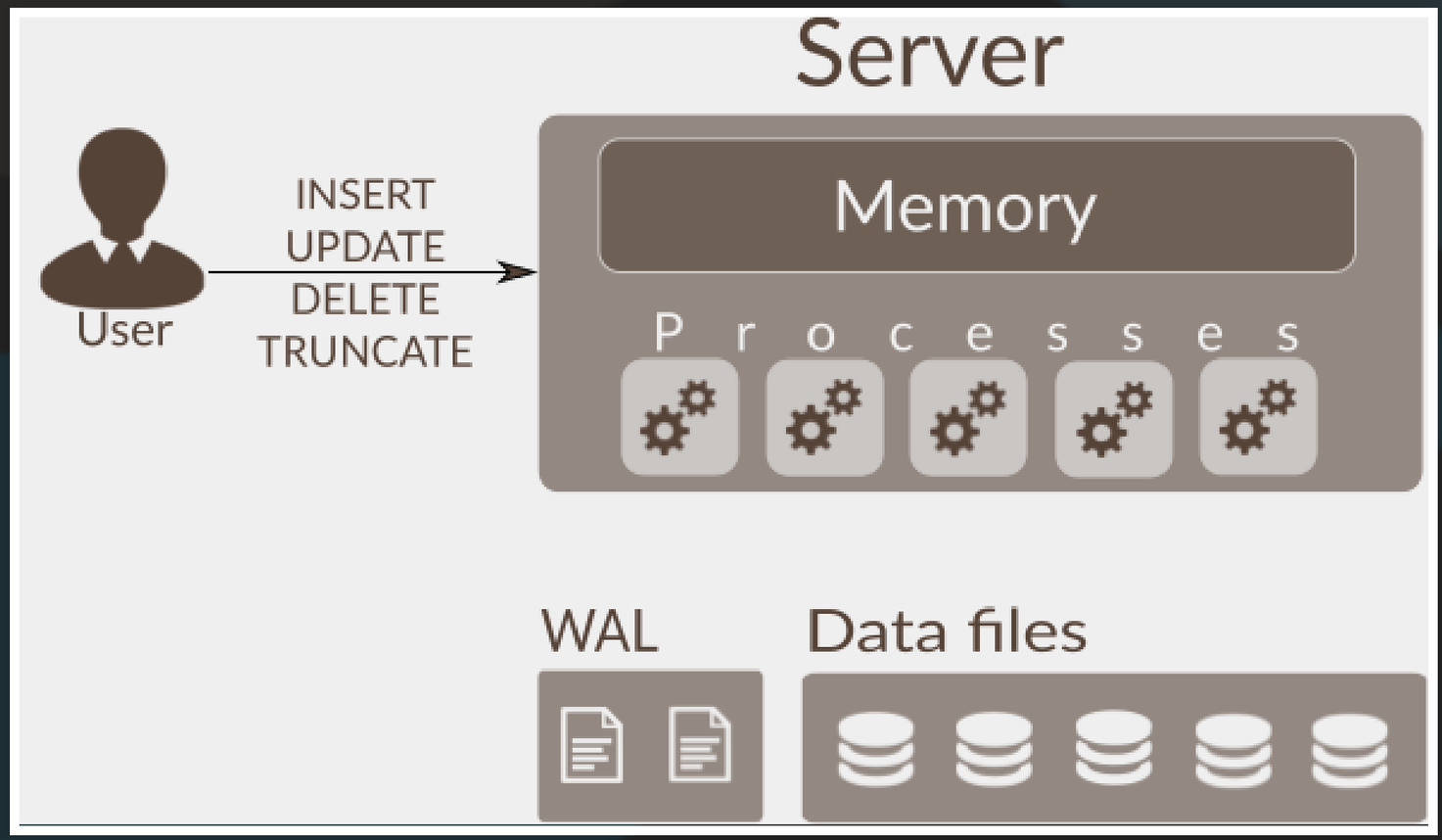
LOXODATA

SOME THEORY



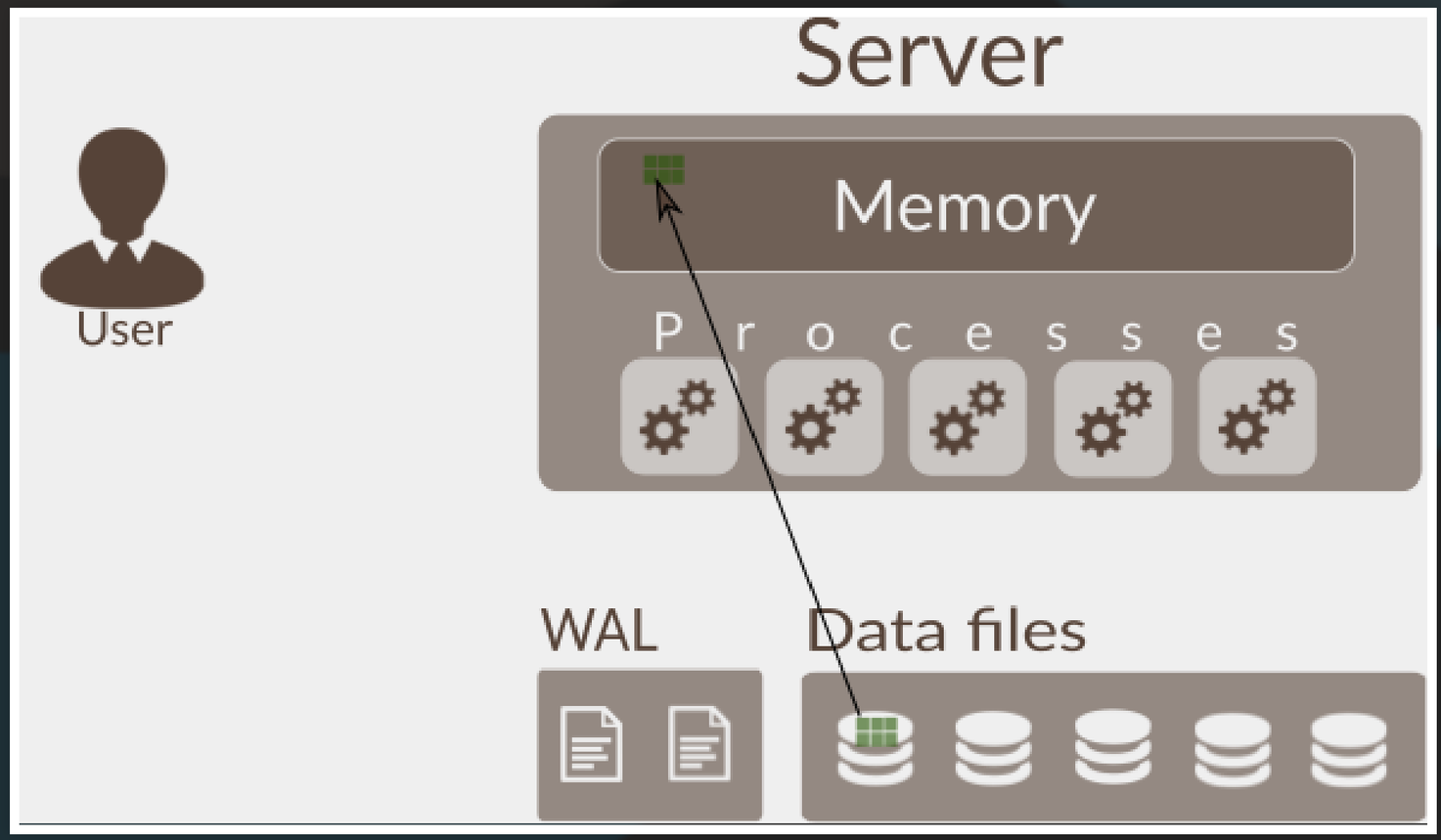
LOXODATA

SOME THEORY



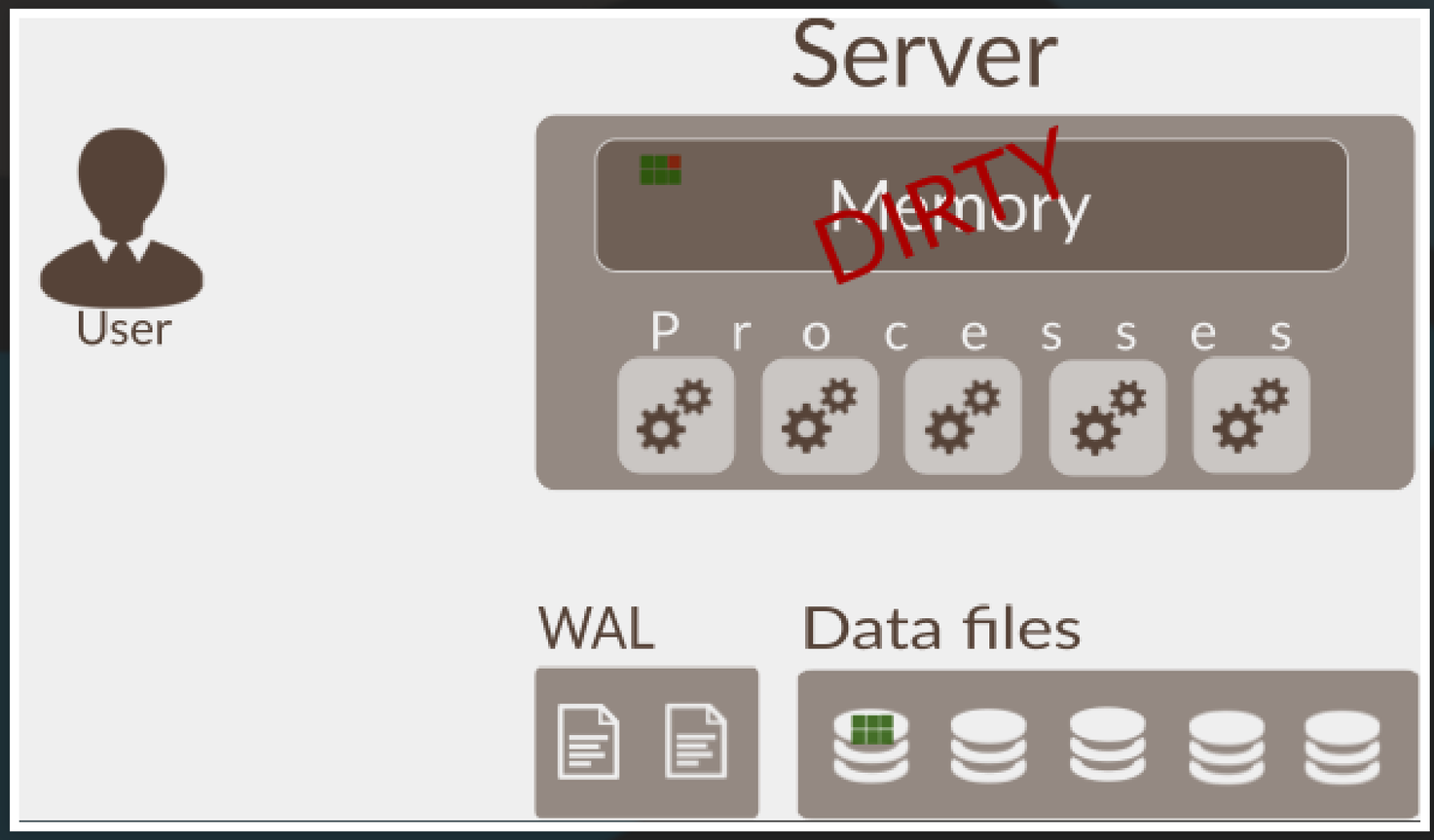
LOXODATA

SOME THEORY



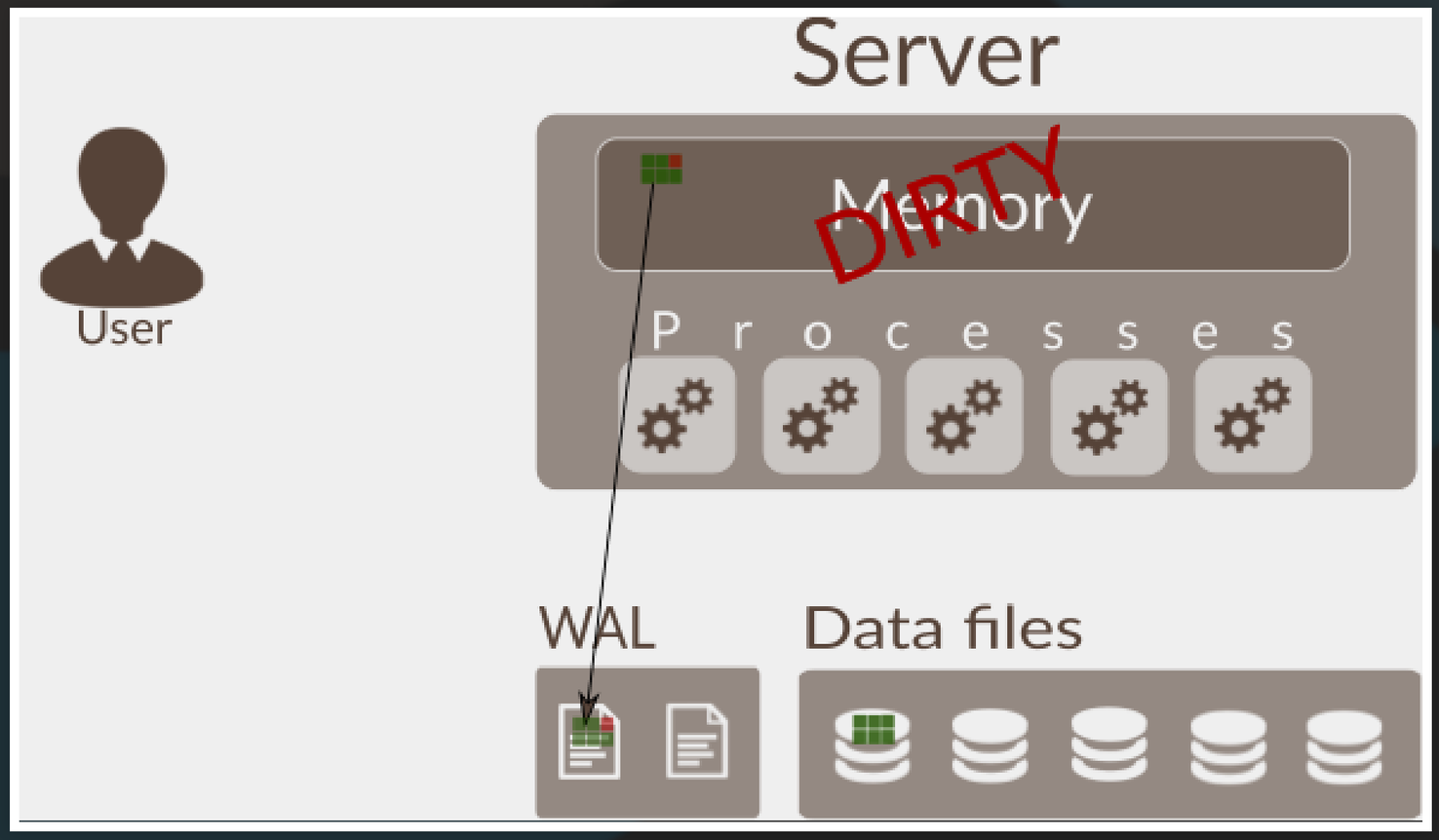
LOXODATA

SOME THEORY



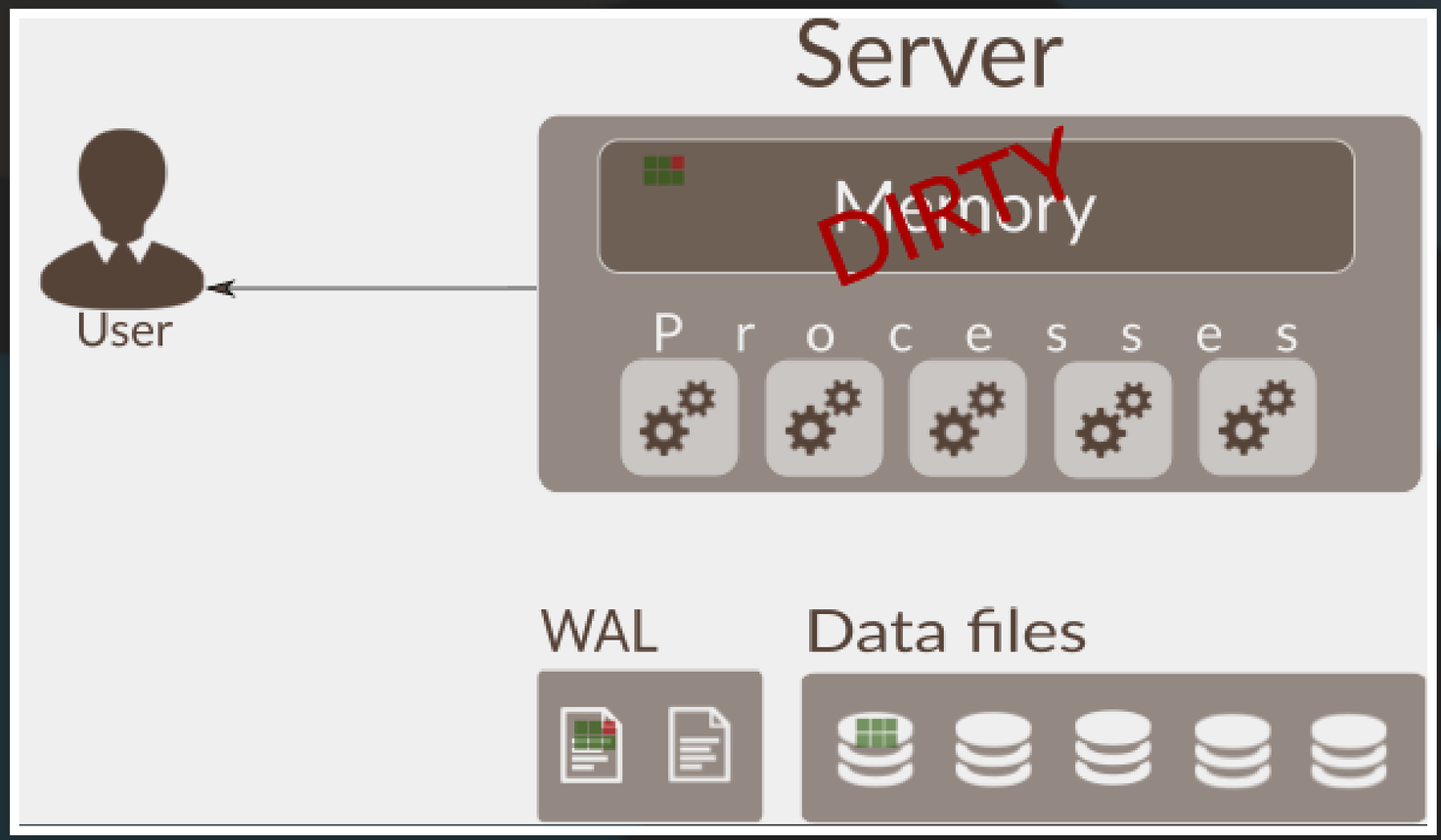
LOXODATA

SOME THEORY



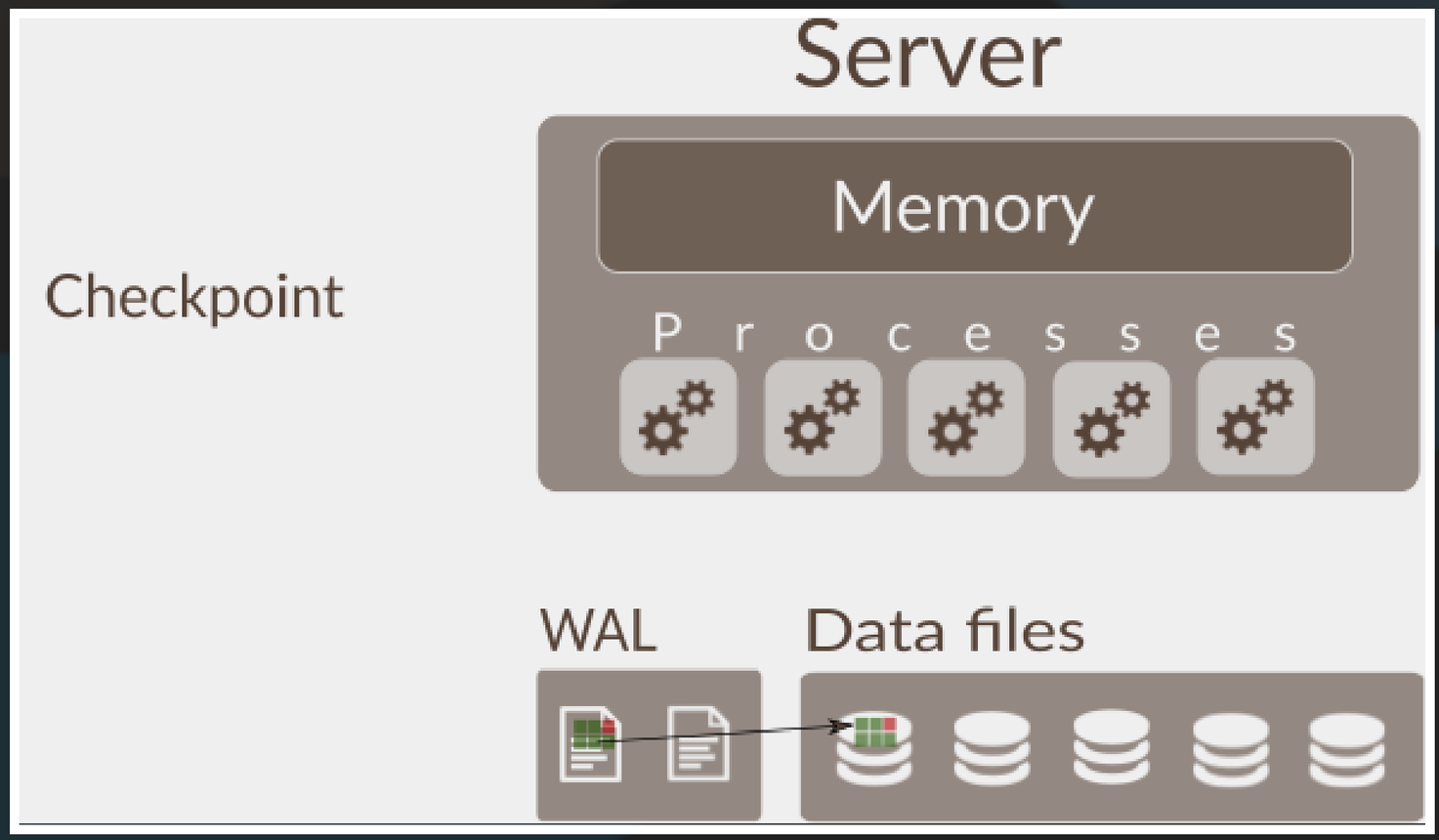
LOXODATA

SOME THEORY



LOXODATA

SOME THEORY



LOXODATA

MY VISION OF "TIME"

- not human time
- depends on activity
- "soon" : microseconds to years

LOXODATA

MY LIFE WISHES

- reach the end of file
- travel!
- not be involved in a "disaster"
- not end up in "cryo chamber"

LOXODATA

MY DESTINY

- no choice
- all my goals may be achievable

LOXODATA

LIFE

LOXODATA

"BIRTH"

- preallocated
- recycled
- allocated on demand

LOXODATA

JOB

- record "events"
  - data changes
  - replication events
  - checkpoints
- writings in append-only mode
  - except for metadata

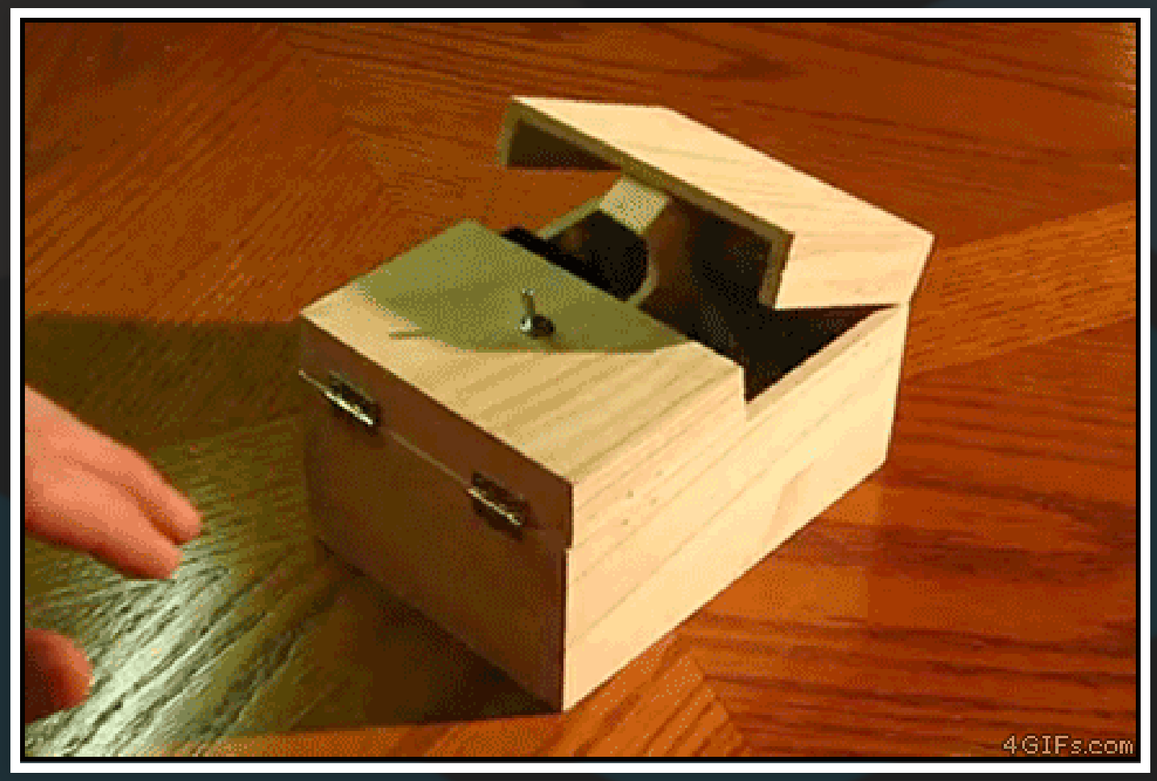
LOXODATA

WORK TIME

- after switch from previous WAL file
- until switch to next WAL file
- continuation of other's work

LOXODATA

THE SWITCH



- normal switch at EOF
- manual switch with *pg_wal_switch()*
- special PITR / promote

LOXODATA

CRYONICS

- at the end of working period
- may be copied to another location
- this is called archiving

LOXODATA

ARCHIVING

- external command in *archive_command*
- enable with *archive_mode = on*
- may retain WAL longer than expected

LOXODATA

ARCHIVES

- many possible destinations
  - local or remote filesystem
  - tape band or permanent storage
- would likely never been used again

LOXODATA

"DEATH"

- checkpointer process : deleted or recycled
- "death" may be delayed in some cases
- manual (human) action: ERROR
- (not) Schrödinger paradox

LOXODATA

DEFROST

- only when recovering
- copied from archives
- fully read to REDO transactions

LOXODATA

IDENTITY

LOXODATA

MY NAME

00000002 00000008 000000BB

It's made of 3 parts, 8 digits each.

- TimeLine ID
  - starts at 1
- Logical file ID
  - starts at 0
- Physical file ID
  - from 00 to FF
- First of all WAL:
  000000010000000000000001

LOXODATA

MY NAME

00000002 00000008 000000BB

- (unofficial) nickname: 8/BB
- the 0xBBth segment in the 0x8th logical file
- all my bytes have an address
- LSN: 8/BB3CB0D2 is my byte 3 977 426

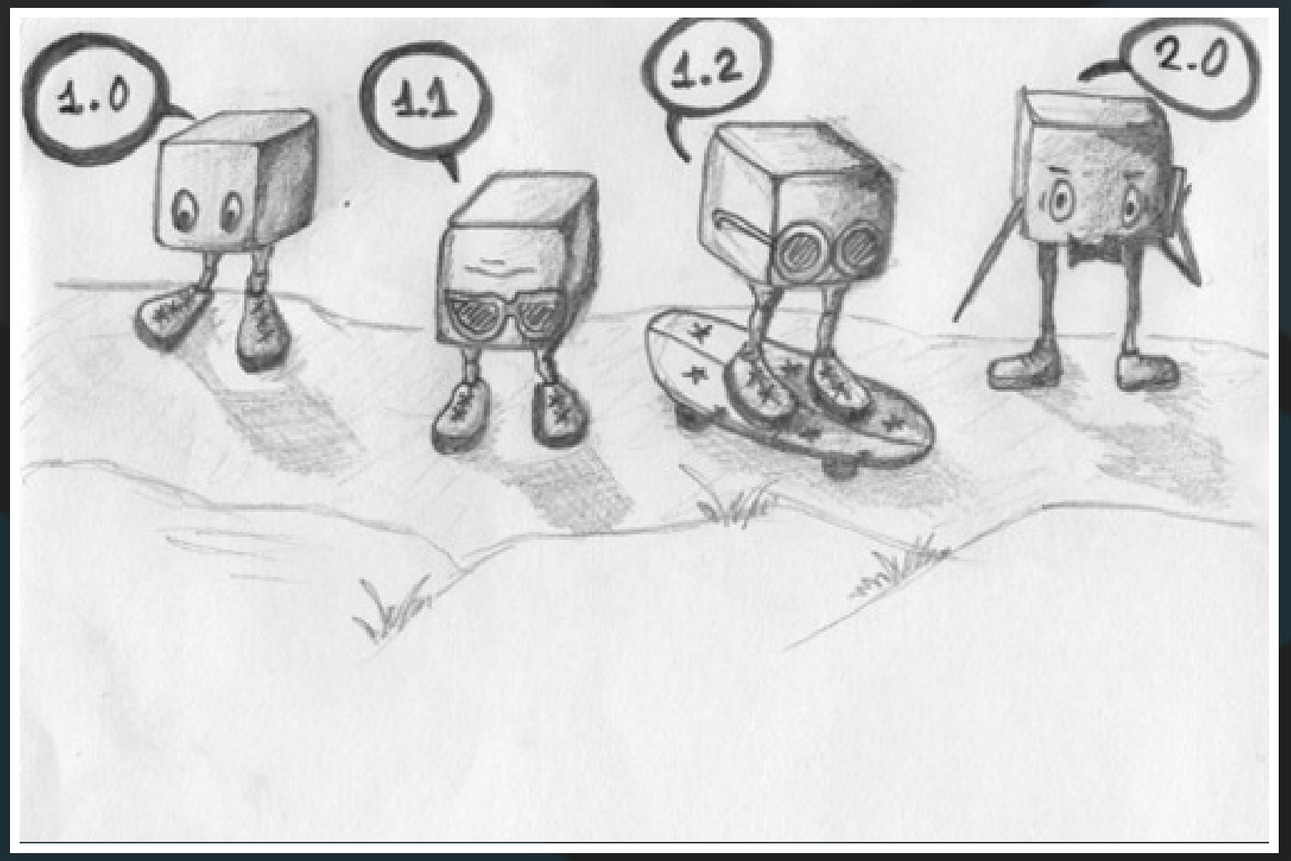
LOXODATA

TIMELINE ?

Something horrible happened.
That's why the TLID is 2 and not 1
A part of the family was abandonned.
Some informations in file 00000002.history

LOXODATA

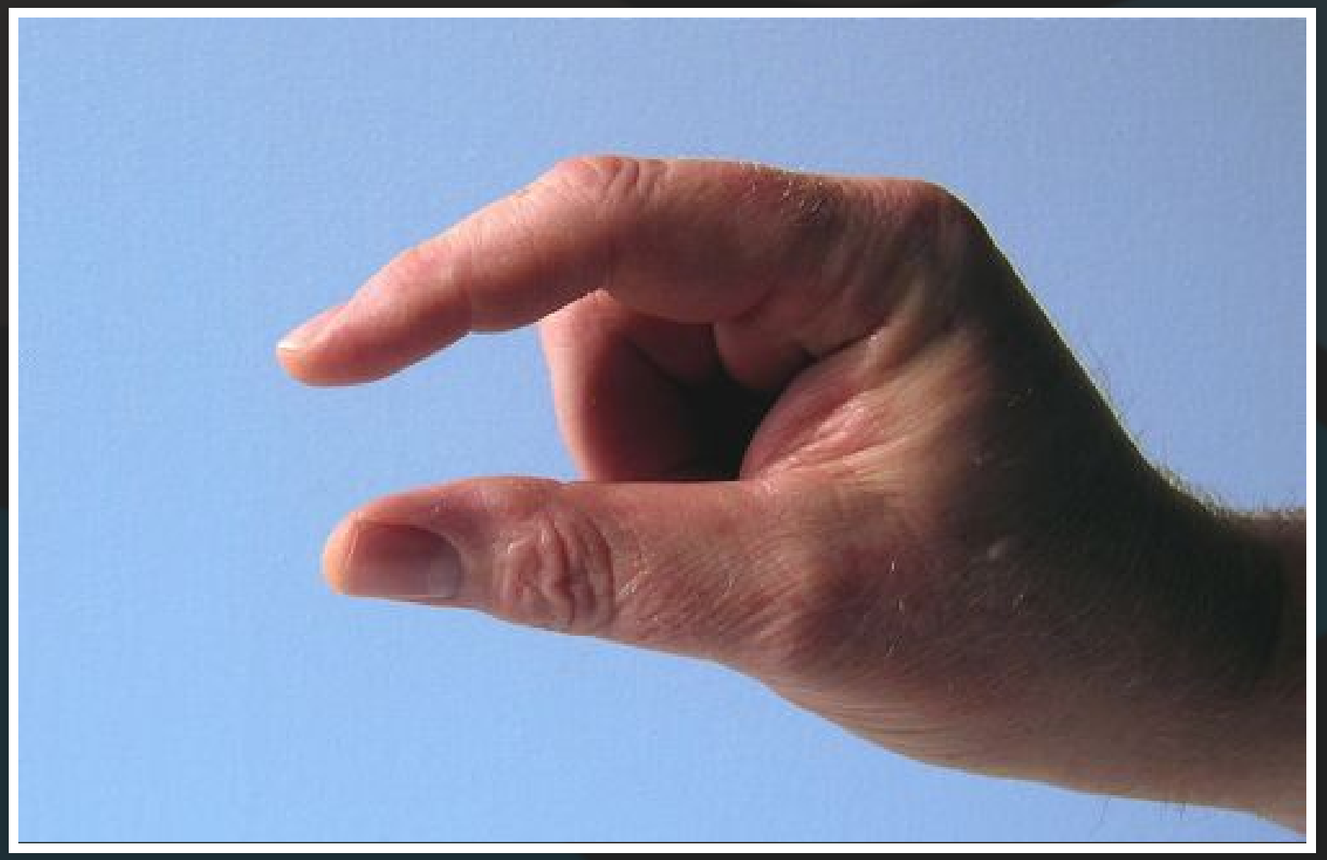
VERSION



- I am tied to the PostgreSQL version
- my internals may differ
  from one major version to another

LOXODATA

SIZE



- default size: 16 MB
- divided into blocks, by default 8 kB each
- full size when allocated

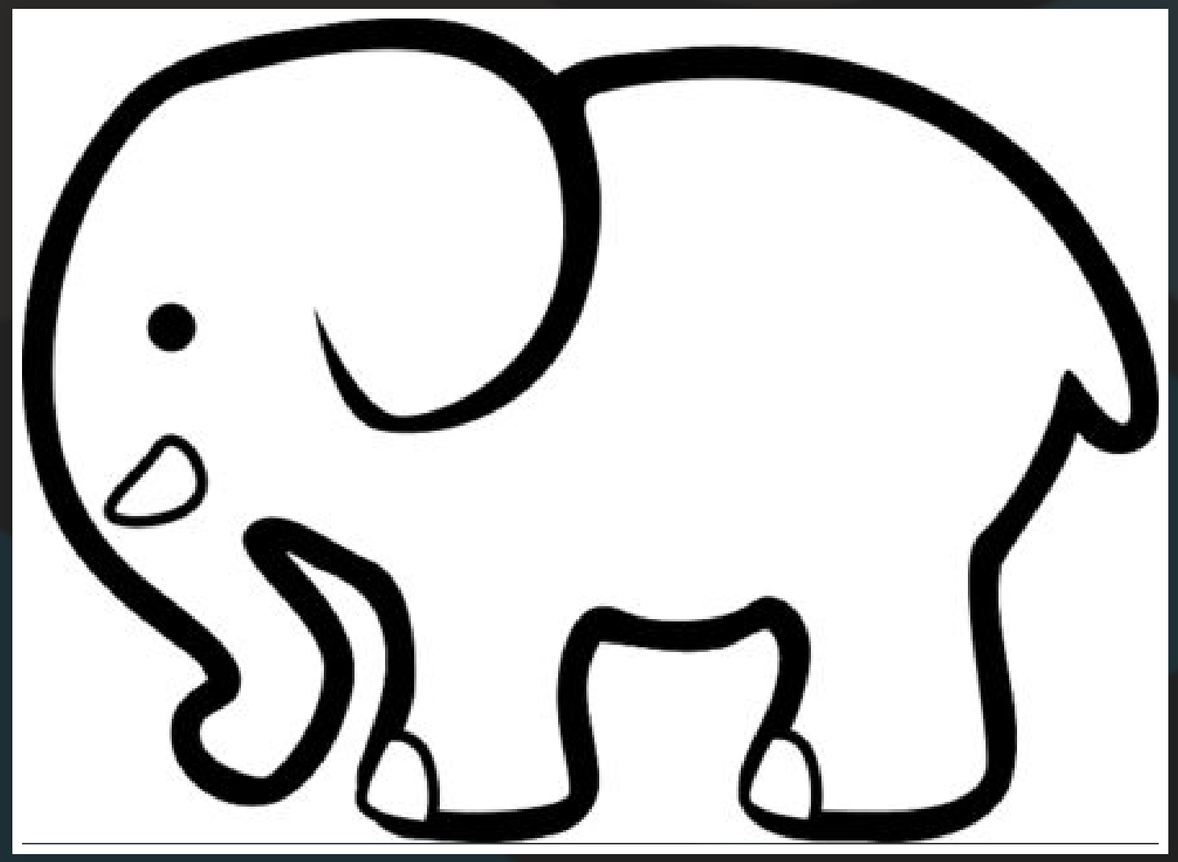
LOXODATA

WAL LEVEL

- 3 different levels available
- *wal_level* in configuration
- allows different life opportunities
- might change over time

LOXODATA

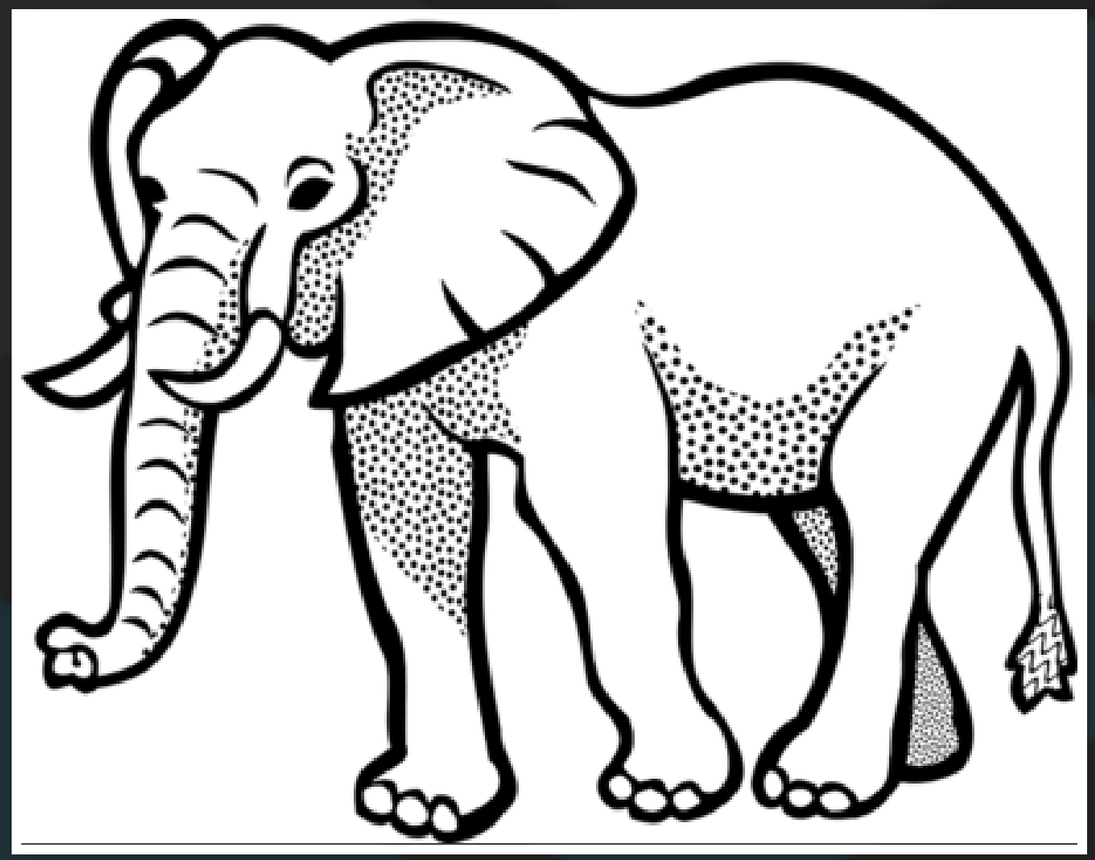
LEVEL "MINIMAL"



- recovery, only from "crash"
- data consistency
- short life :'(

LOXODATA

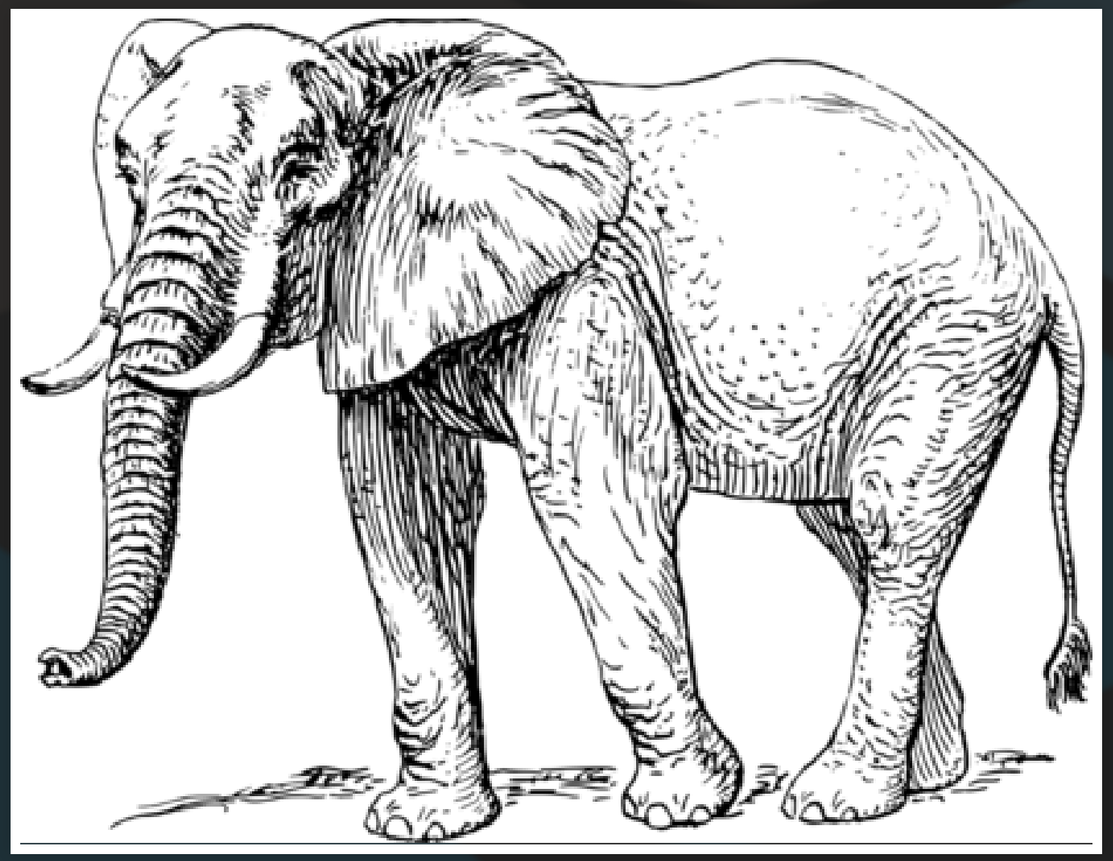
LEVEL "REPLICA"



- archiving
- physical replication (travel!)
- read-only queries on standby
- more informations stored

LOXODATA

LEVEL "LOGICAL"



- logical decoding
- logical replication
- even more informations stored

LOXODATA

MY PLACE

- pg_wal directory in $PGDATA
- or any directory symlinked as pg_wal

LOXODATA

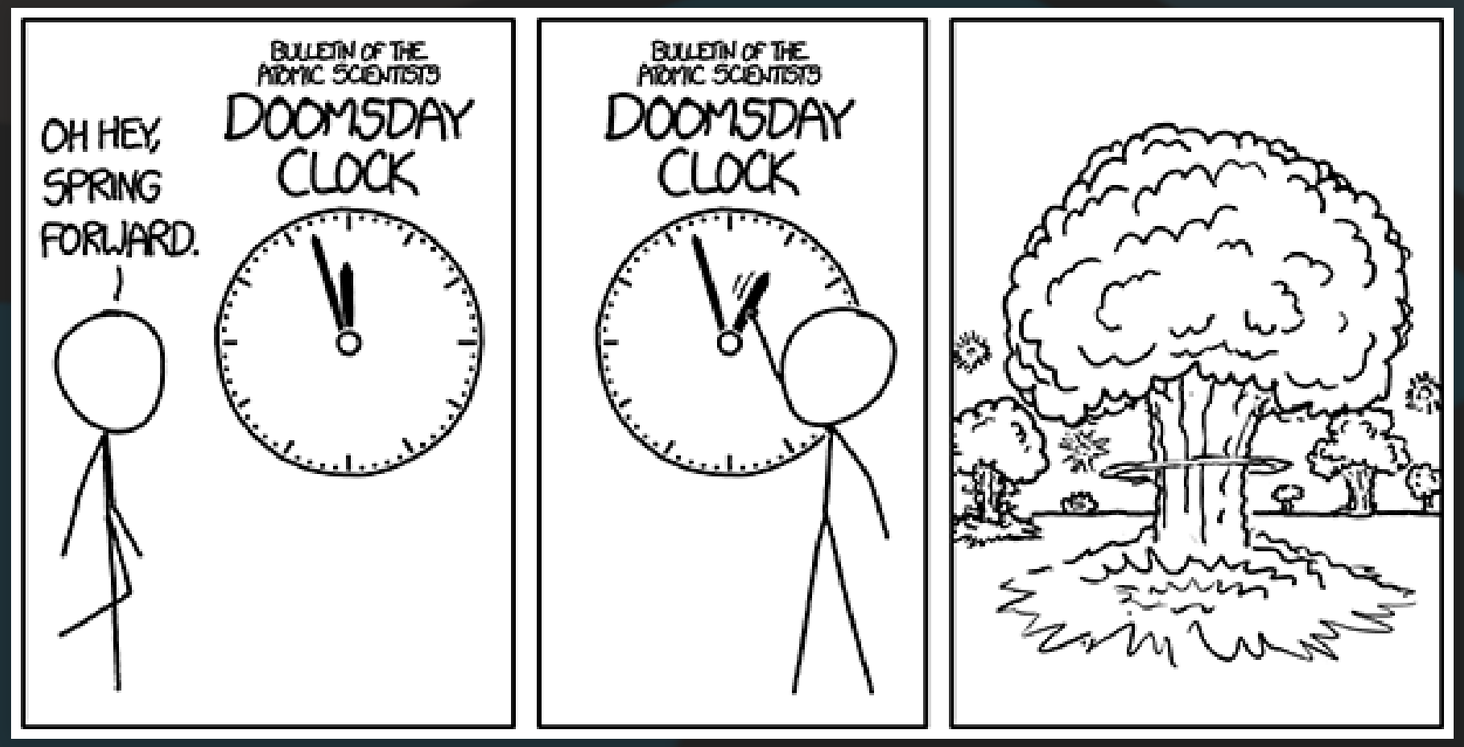
PASSPORT

- date of birth ?
- date of issue/expiration ?
- photo ?

Not a human passport

LOXODATA

DISASTER MANAGEMENT



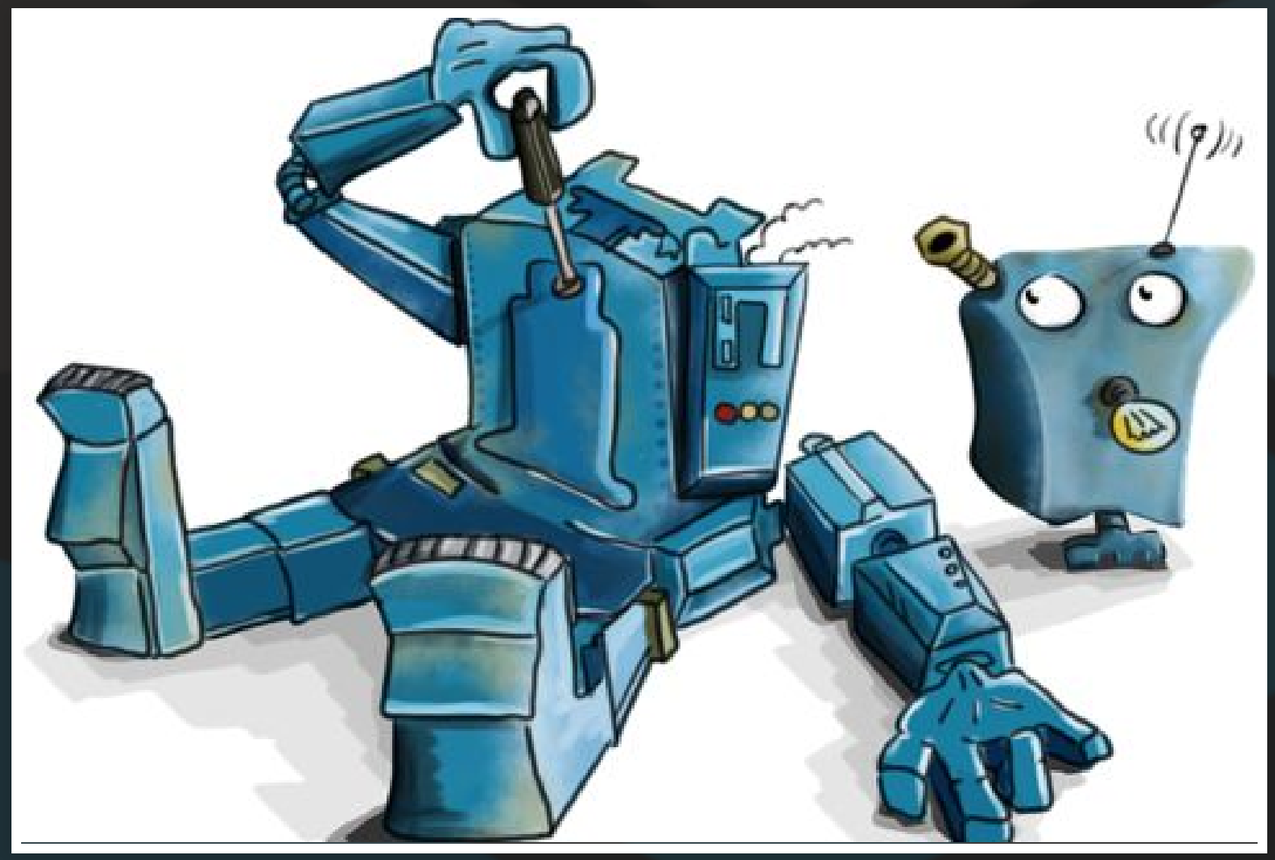
LOXODATA

DISASTER

- restore a backup
- replay work after that backup
- maybe stop at some point
- go back in production

LOXODATA

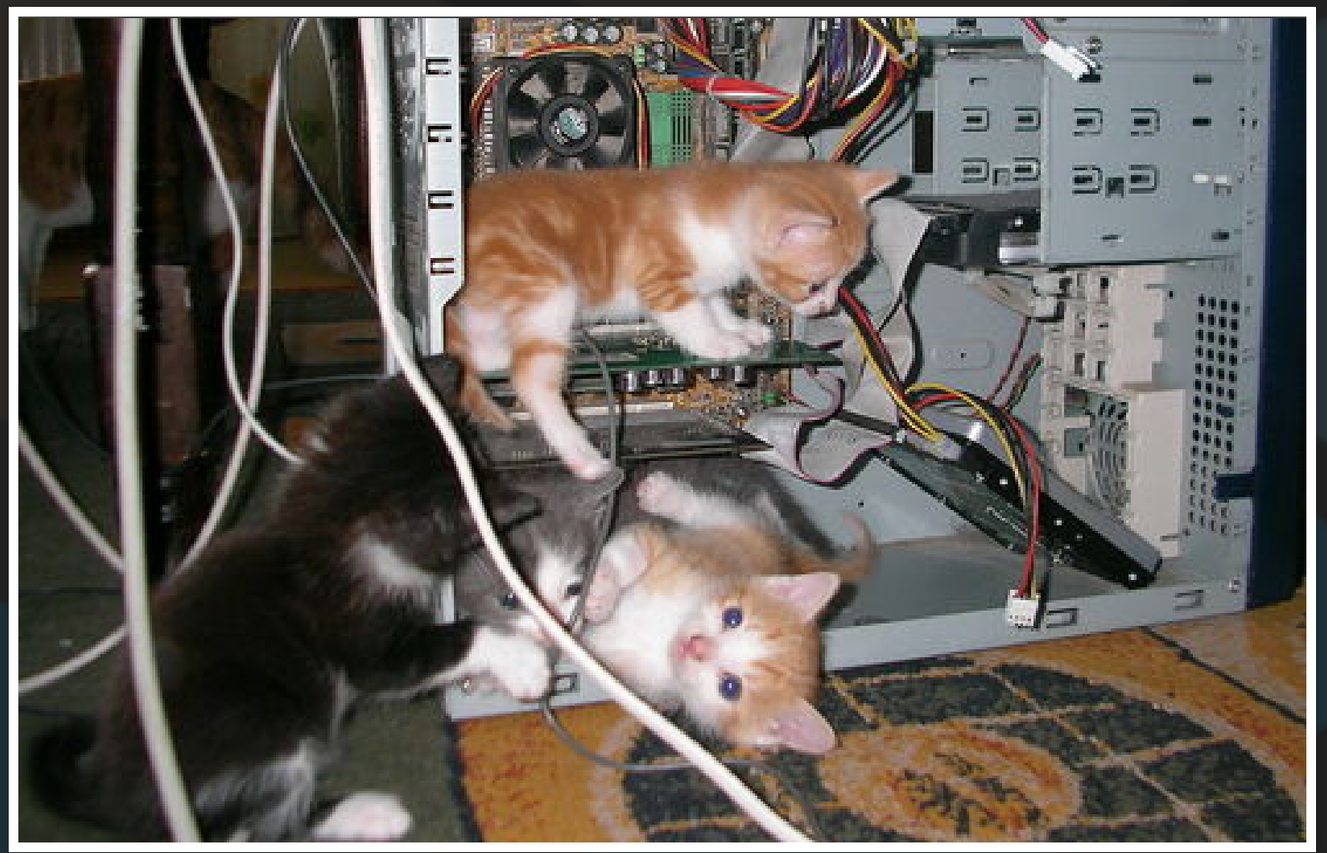
AUTOMATIC RECOVERY



- after a brutal stop
- no need to restore a backup
- last checkpoint lookup
- transactions replay

LOXODATA

DELIBERATE RECOVERY



- start from a physical backup
- write file *recovery.conf*
- *restore_command* to fetch WAL
- same as automatic recovery
- timeline change

LOXODATA

PITR

- point-in-time recovery
- deliberate recovery
- specify end of recovery
- end of recovery action

LOXODATA

TIMELINE CHANGE

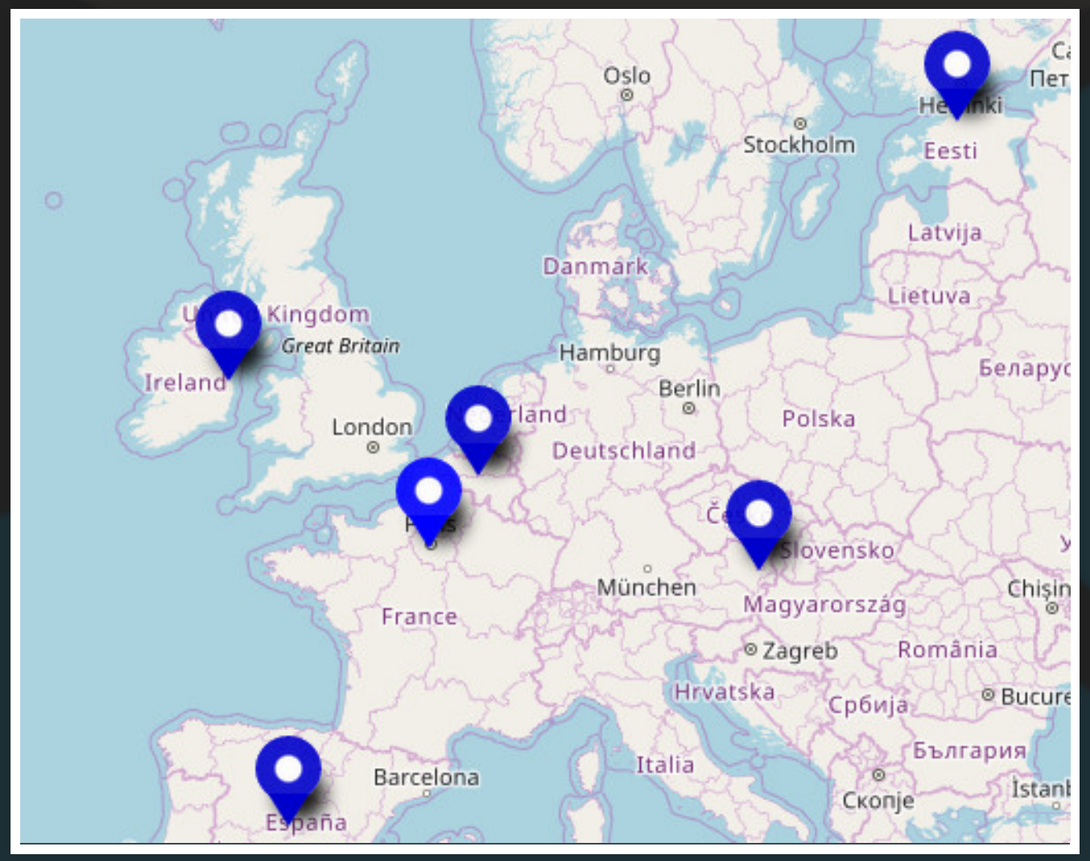
- last WAL of recovery copied
- name differs by TLID
- content differs from recovery point
- subsequent WAL in old timeline abandonned

LOXODATA

TRAVELLING

LOXODATA

TRIP



- origin, source
  - primary, provider, publisher
- destinations
  - standby, subscriber
- transport method
  - via archives
  - streaming replication

LOXODATA

CONCEPT

- continuously up-to-date clone of data
- copy data, then replay transactions
- who's the best at recording transactions ?

LOXODATA

PHYSICAL REPLICATION

- duplication of WAL file
- from one cluster to another
- streaming replication

LOXODATA

LOGICAL REPLICATION

- decoded on publisher side
- information transformed
- sent to feed another WAL out there
- no travel

LOXODATA

WAL SENDER

- gets replication connections
- runs replication protocol commands
- sends WAL content

LOXODATA

WAL RECEIVER

- fetchs data
- permits REDO events
- sends feedback

LOXODATA

PG_RECEIVEWAL

- special receiver process
- collects and stores (no REDO)
- streamed archive

LOXODATA

REPLICATION SLOTS

- client dedicated resource
- stores replication status
- forbids deletion until replicated

LOXODATA

TIME TO GET TO WORK

LOXODATA

THANK YOU FOR YOUR ATTENTION