

# HTAP Queries & Data Fabrics



Atif Rahman  
@mantaq10

7<sup>th</sup> December, 2018  
PostgreSQL Down Under  
Melbourne, Australia

## Background

OLTP vs OLAP vs HTAP

The Problem Statement

## Patterns

Data Fabrics

Key PostgreSQL Features

## Components

Foreign Data Wrappers

Distributed Cache



# The agenda

# OLTP

Smaller transactions

Lots of them

Lots of Updates

**ACID (Transactional)  
CRUD (Commands)**

# OLAP

complex queries

Large working sets

Bulk loads & offloads

**INDEXING  
AGGREGATIONS**

# HTAP

Mixed workloads on the same system

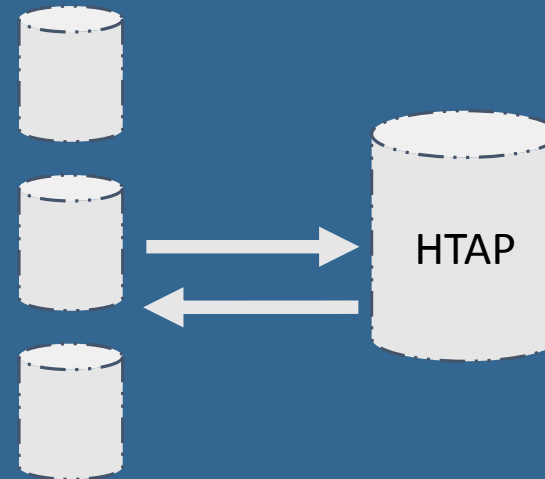
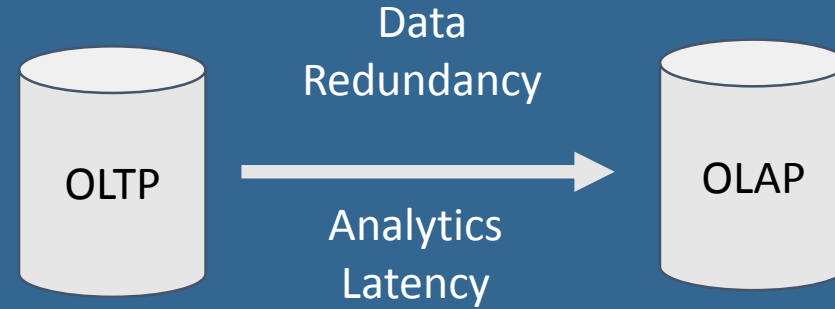
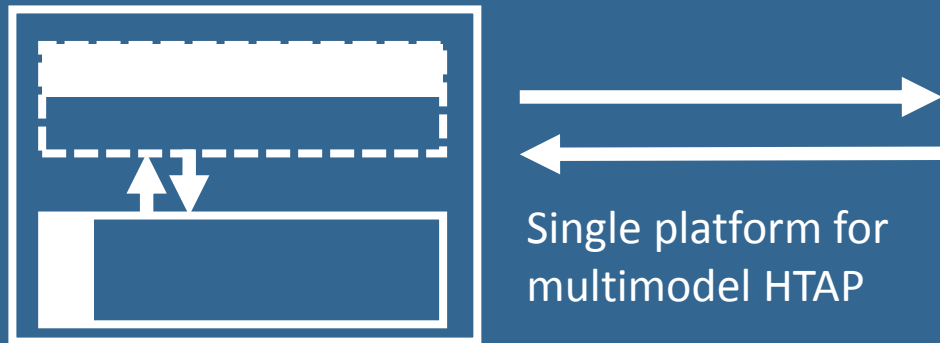
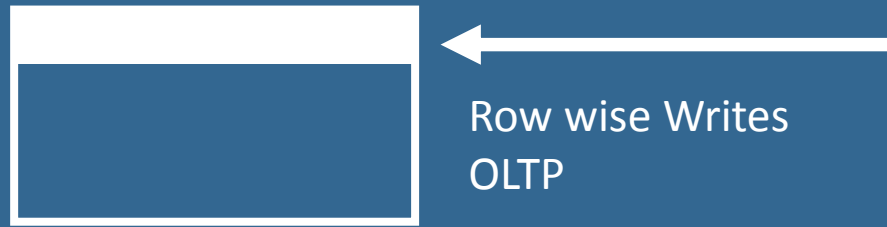
Analytics on 'inflight' transactional data

Addresses resource contention

**FEDERATION  
SYNCHRONISATION**

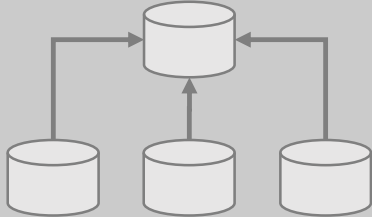
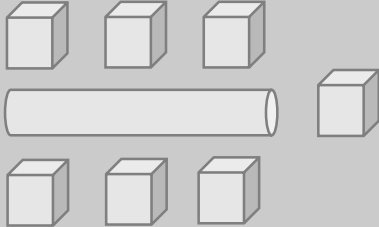
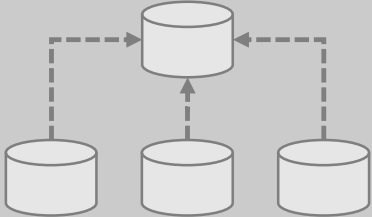
**OLTP vs OLAP vs HTAP**

# Data Organisation



# Hybrid Transactional / Analytical Processing



	<b>Data Integration (ETL/ELT)</b>	<b>Application Integration (ESB / API)</b>	<b>Data Virtualisation (DV)</b>
Low Fidelity View			
Integration Type	Physical Movement and Consolidation	Synchronization and Propagation	Abstraction, Virtual Consolidation, Federation
Purpose	Database to Database	Application to Application	Database to Application
Agility*	Weeks, Months	Minutes, Hours	Hours, Days
Repository	Warehouse / Lake	Transactional System	Semantic Layer
Run Time*	Typically Scheduled	Event Driven	Typically OnDemand

# Data & Application Integration

# Data Warehouse

- Schema on write
- (early binding)
- BI and analysts
- Arguably better governed.
- MPP / SMP databases

# Data Lake

- Schema on read
- (Late binding)
- Data scientists
- Arguably more flexible
- MapReduce et al.

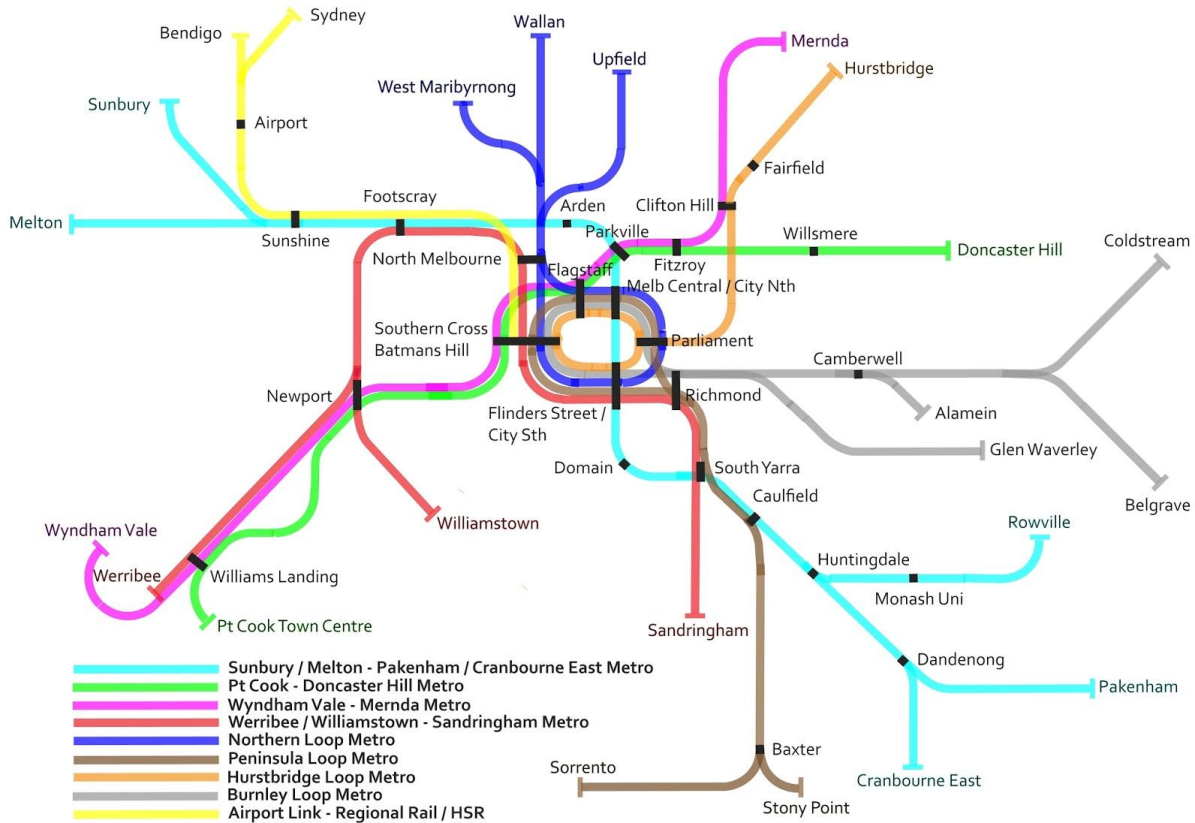
## Data Warehouse vs Data Lakes



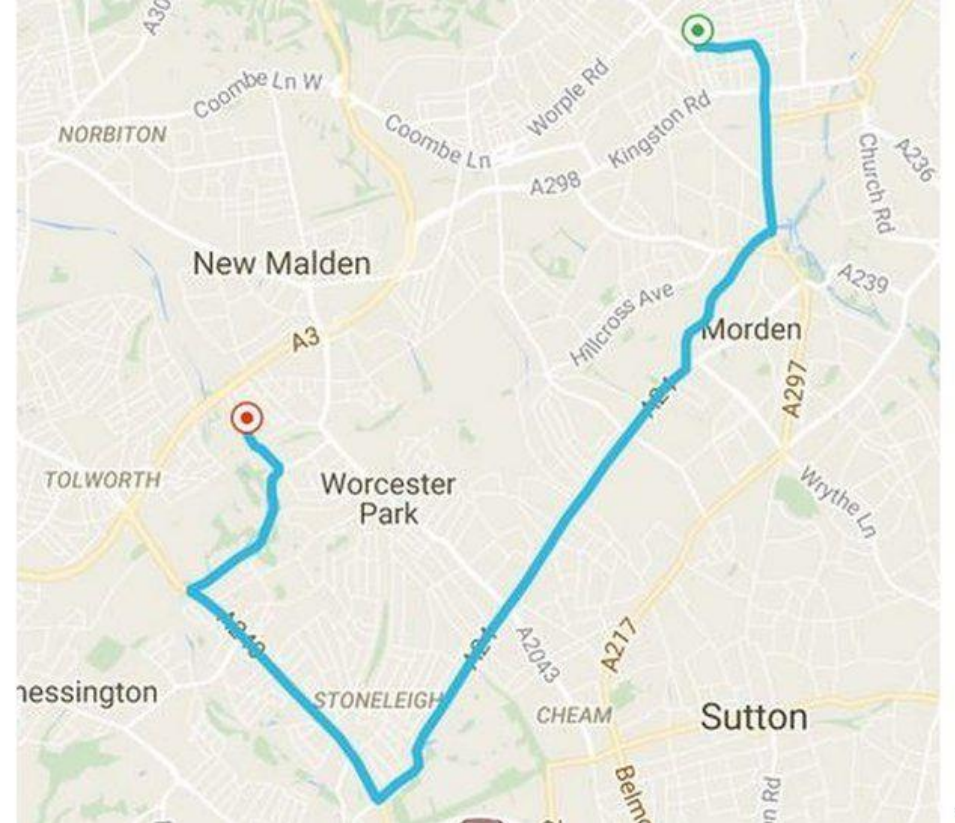


# Data Warehouse

# Data Lake



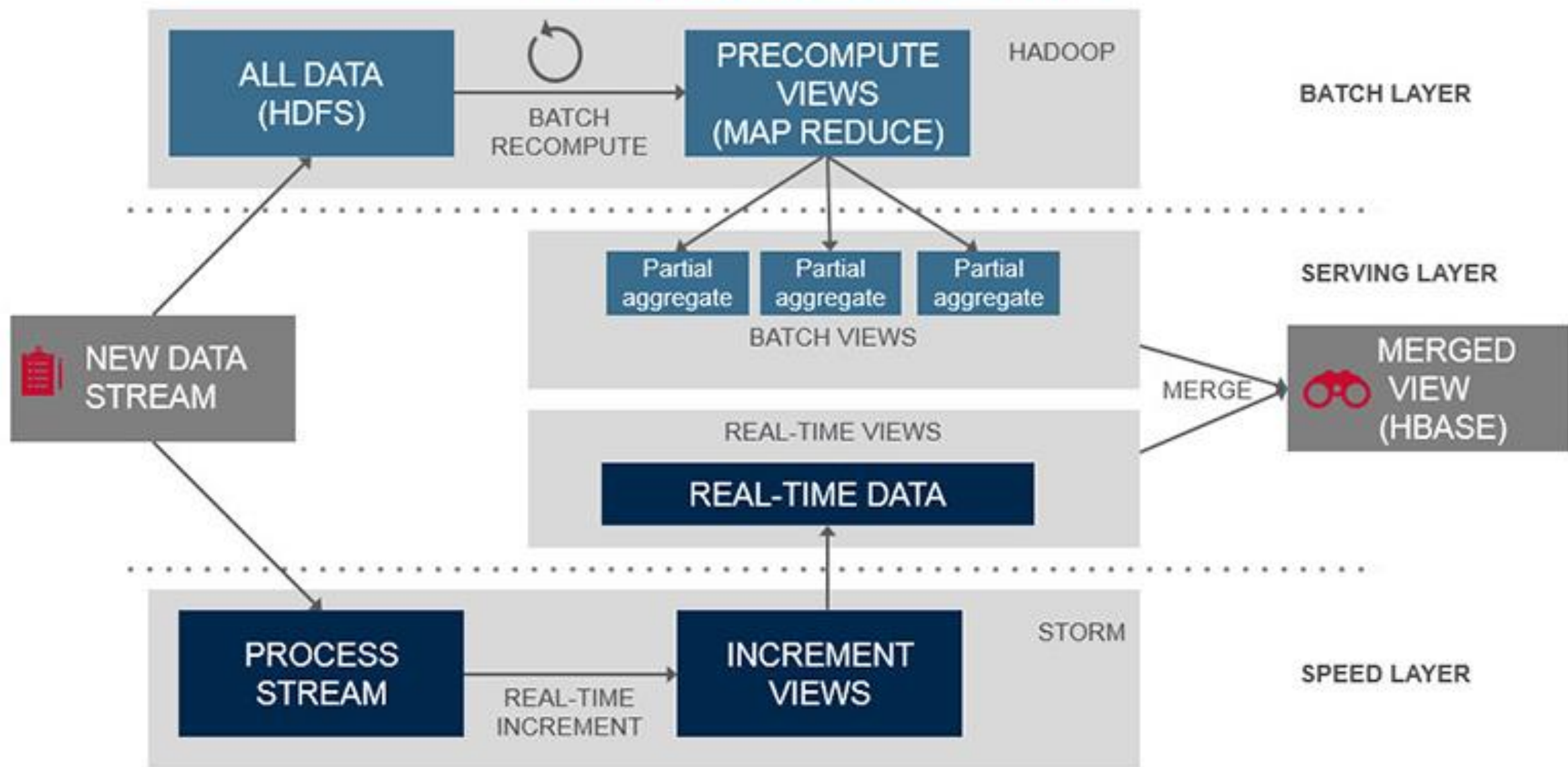
Metro



Uber

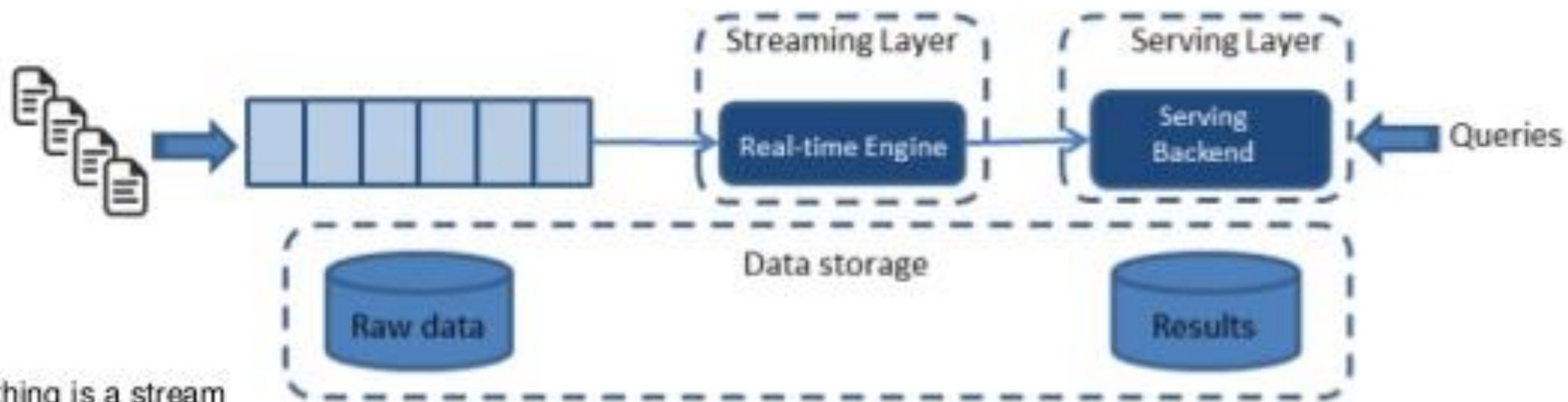
# Data Warehouse vs Data Lakes

# Lambda Architecture



# Kappa architecture

## Stream Processing with Scalable Storages



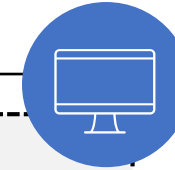
- Everything is a stream
- Immutable unstructured data sources
- Single analytics framework
- Windows on Streaming Layer
- Linearly scalable Serving Layer
- Interactive querying

- **ACID**
  - **Atomicity**
  - **Concurrency**
  - **Isolation**
  - **Durability**
- **BASE**
  - **Basic Availability**
  - **Soft States**
  - **Eventual Consistency**
- **CAP Theorem (Consistency vs Availability vs Partitions)**
- **CQRS (Command Query Response Segregation)**

**The Problems**



# Data Fabric Architecture

Unified queries



Metrics & Analytics  Complex Alerts  Scheduled Feeds  Adhoc Queries  Detail Data 

[Distributed Cache]

Logical Unified Data Model  Subject Area Logical Data Models 

Business & Exceptions Rules

Normalisation and Standardisation Rules

Top Down Data Quality (and profiling)

Connectors (JDBC, APIs, ODBC, etc)

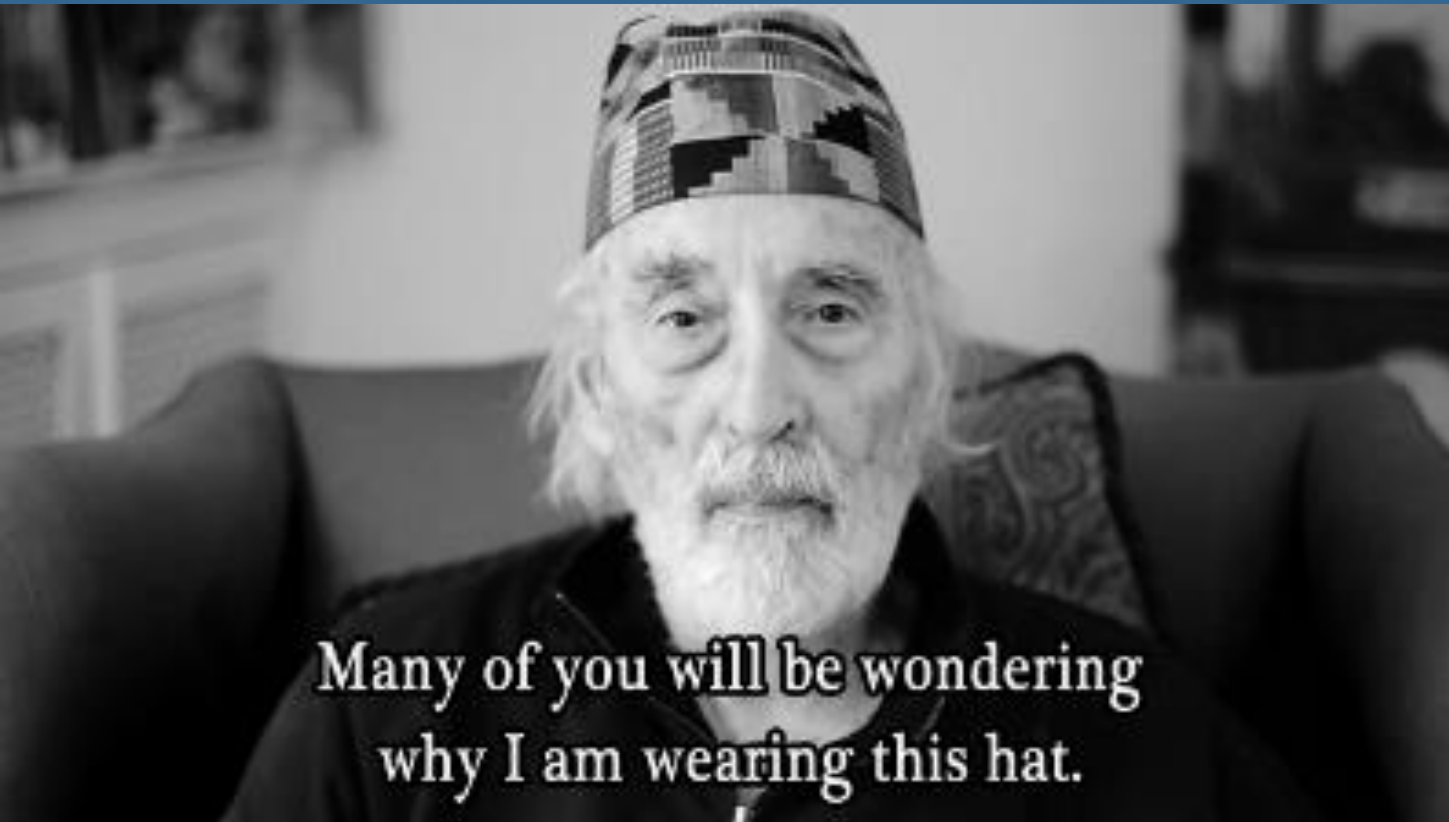
Schedulers and Refresh Jobs Rules

Query Planners & Optimiser(s)

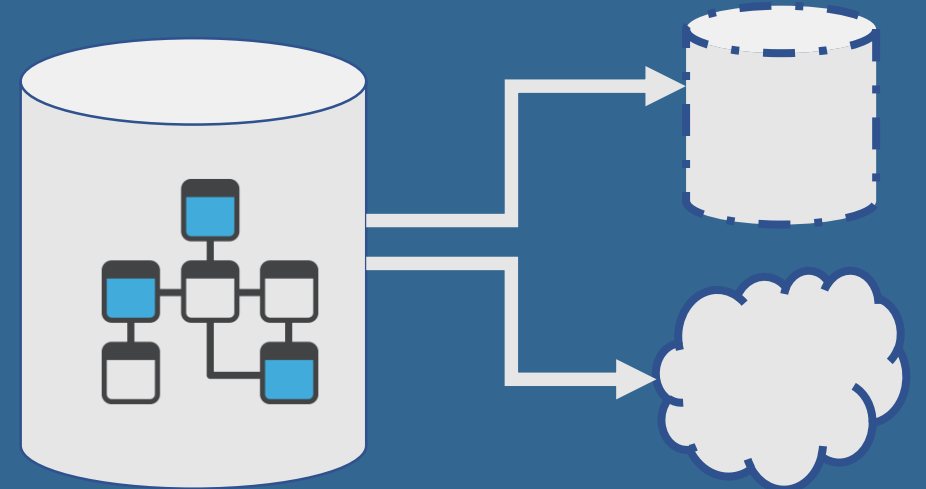
Enterprise Data Fabric Cluster



- Schema Store
- Distributed Cache
- Query Federation
- Query Optimisation
- Semantic Normalisation



One database to rule them all,  
One database to find them,  
One database to bring them all,  
And in a wrapper bind them.



# Foreign Data Wrappers

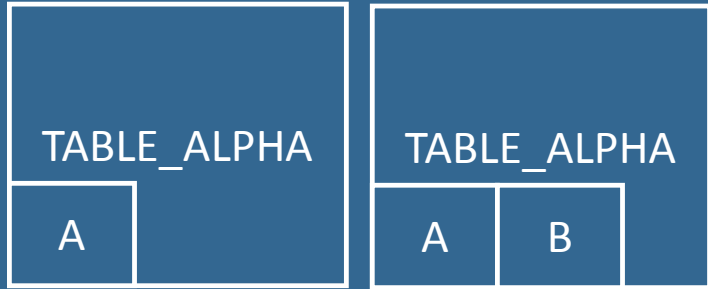
- Uses the standard compliant SQL/MED
- \*Data Type Translations (SQL, NoSQL etc)
- \*Push Down Predicates
  - WHERE and ORDER BY are propagated
  - Required COLUMNS
- \*Supports CRUD
- \*Two Way Joins
- Import Foreign Schemas

\* May vary based on specific wrapper

## Foreign Data Wrappers

```
BEGIN;  
INSERT INTO  
TABLE_ALPHA  
VALUES ('B');  
COMMIT;
```

Dirty  
Page(s)



```
BEGIN;  
INSERT INTO  
TABLE_ALPHA  
VALUES ('B');  
COMMIT;
```

Dirty  
Page(s)



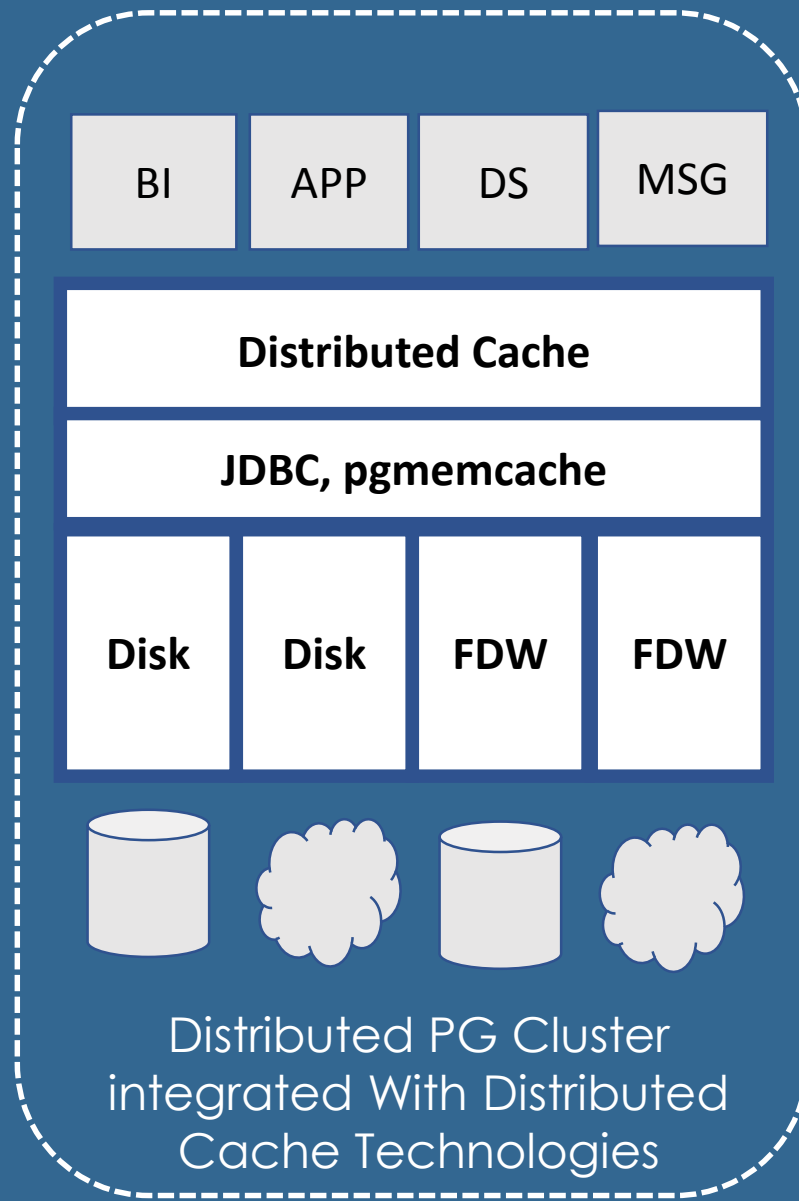
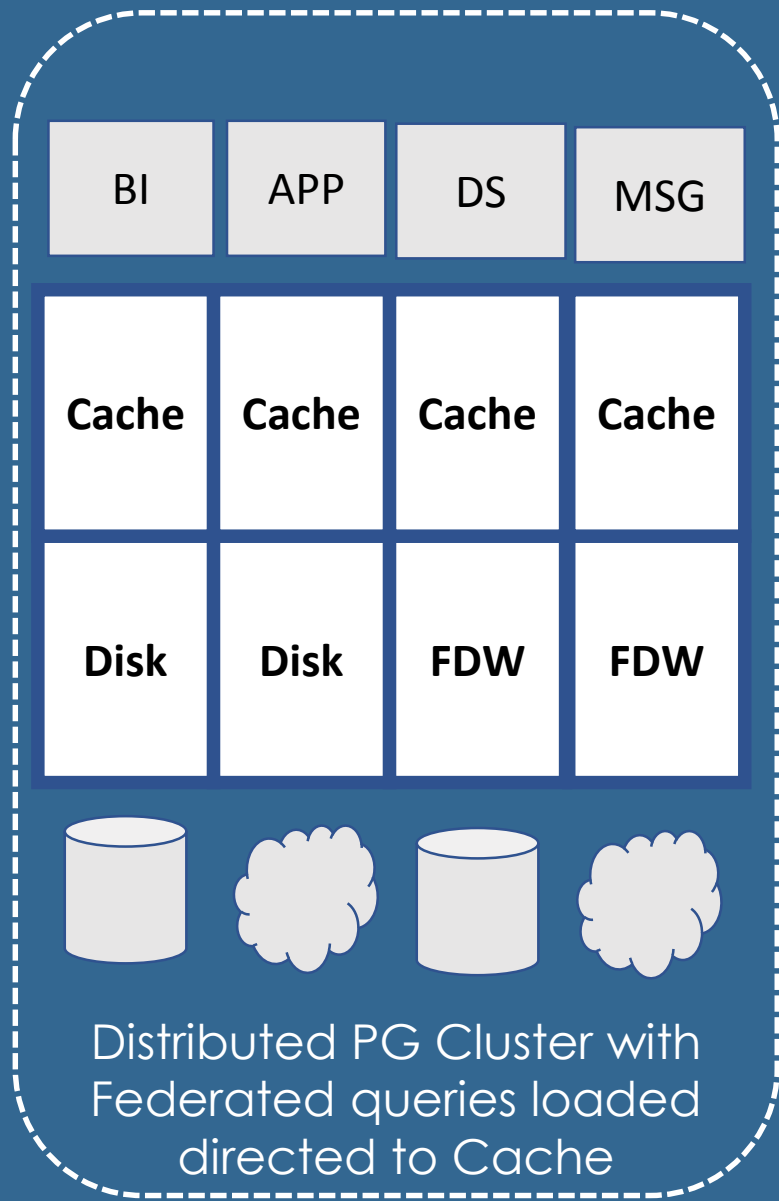
WAL  
Checkpoints  
write back to  
disk



pg\_prewarm()  
Loads OS and  
buffer cache(s)

# (Reflections) the PostgreSQL way

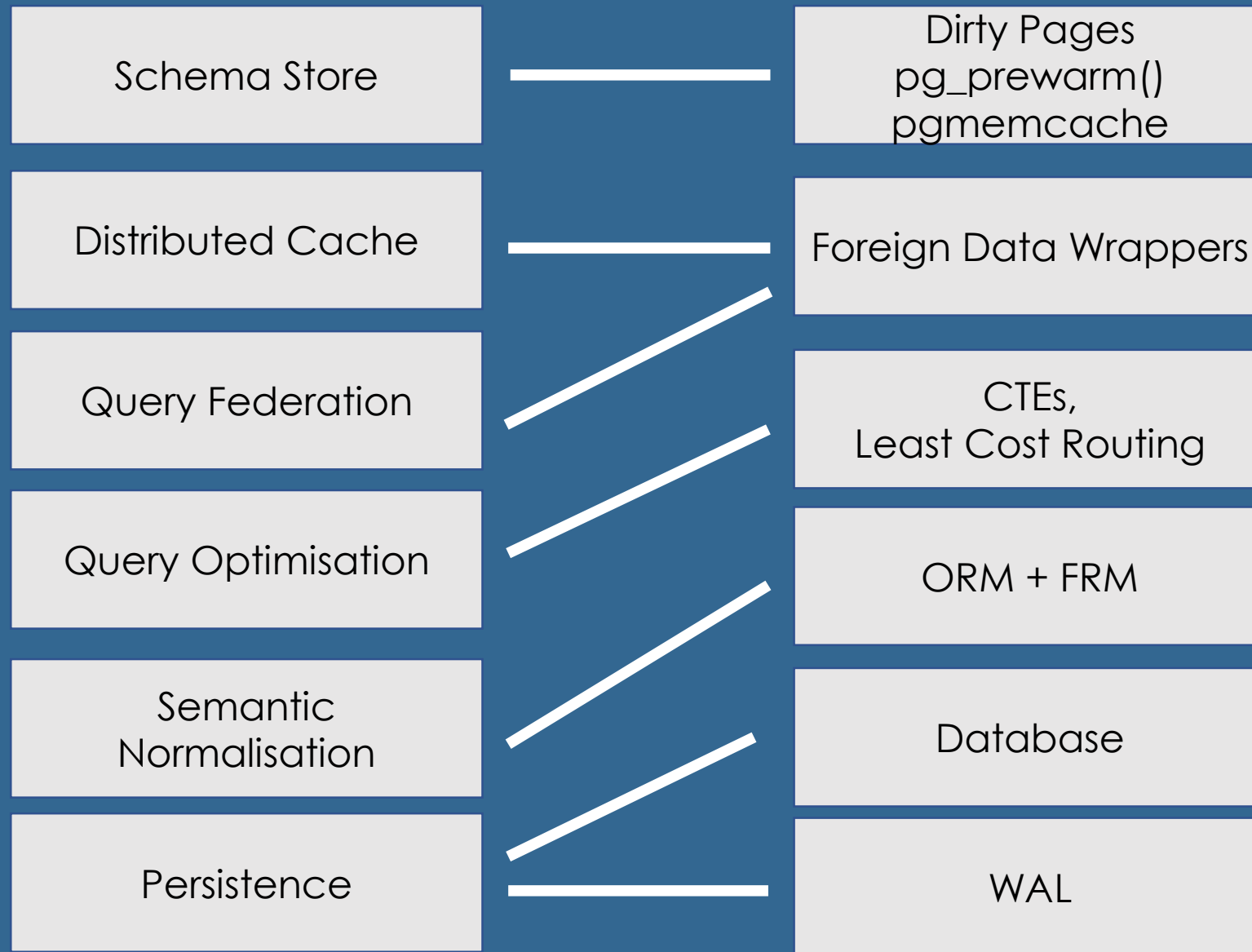




### Some Technologies

- Apache Ignite
- Dremio
- Terracotta
- memcached

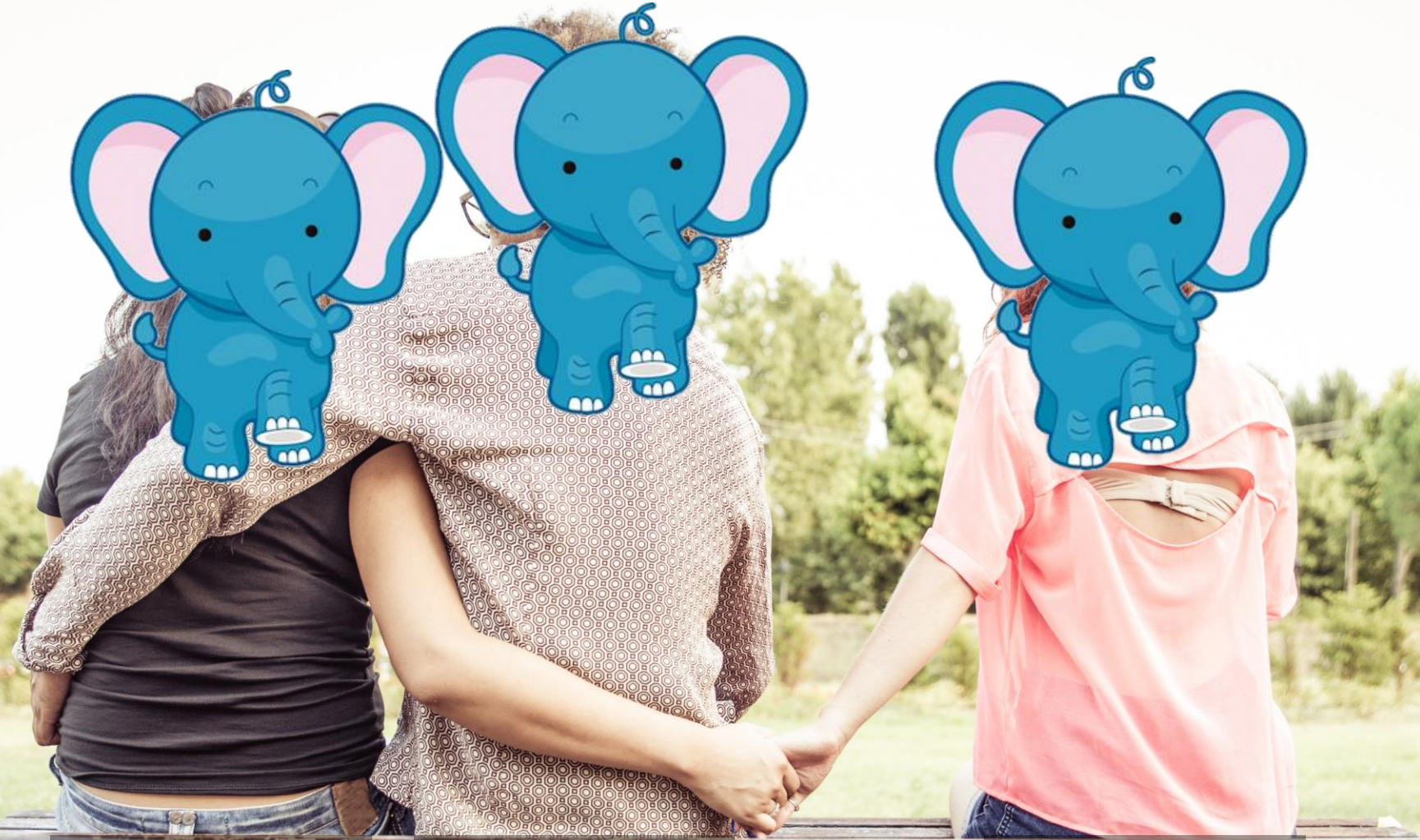
# Distributed Cache with PostgreSQL



# Key PostgreSQL Features for Fabric

- **Cloud migrations**
- **Jurisdictions (privacy and availability zones)**
- **Computing at the edge of the network.**
- **Data Virtualisation**
- **Rights to be Forgotten (GDPR)**
- **Query Lineage & Audit across ALL data**
- **Areas for future development.**

**Key Takeaways**



**HTAP**

**OLAP**

**OLTP**