

Change Data Capture Solutions with Debezium and Memphis.dev

RJ Nowling

Memphis.dev

Chicago Data Professionals Meetup

July 2023

Introducing Myself



- Developer Advocate at Memphis.dev
- Associate Professor of Computer Science at the Milwaukee School of Engineering (5 years)
- ML engineer in industry (4 years)

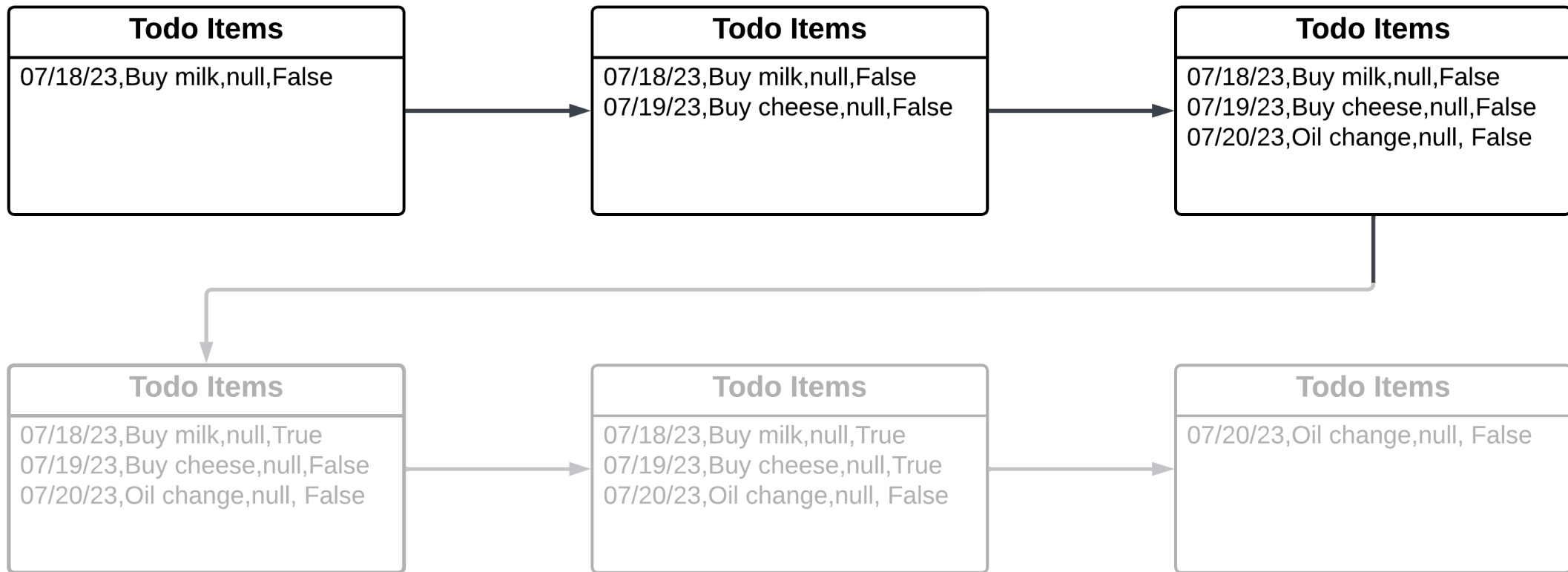
Life of a Database Table



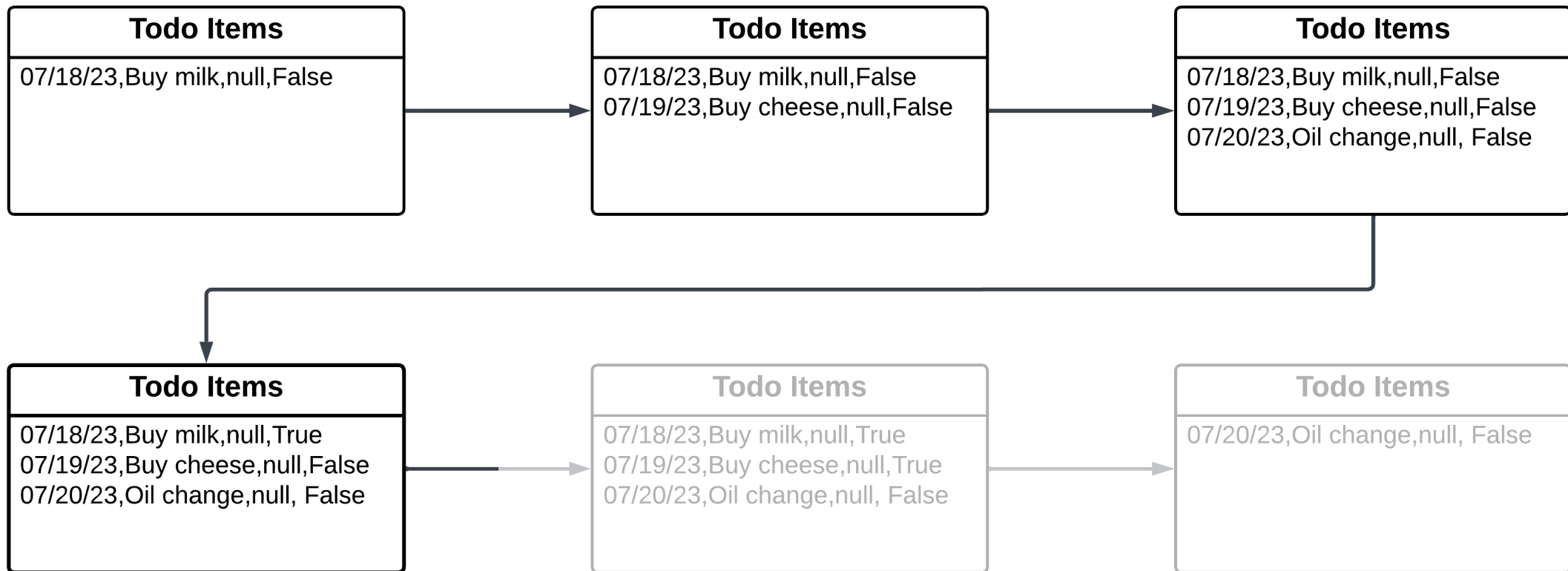
Life of a Database Table



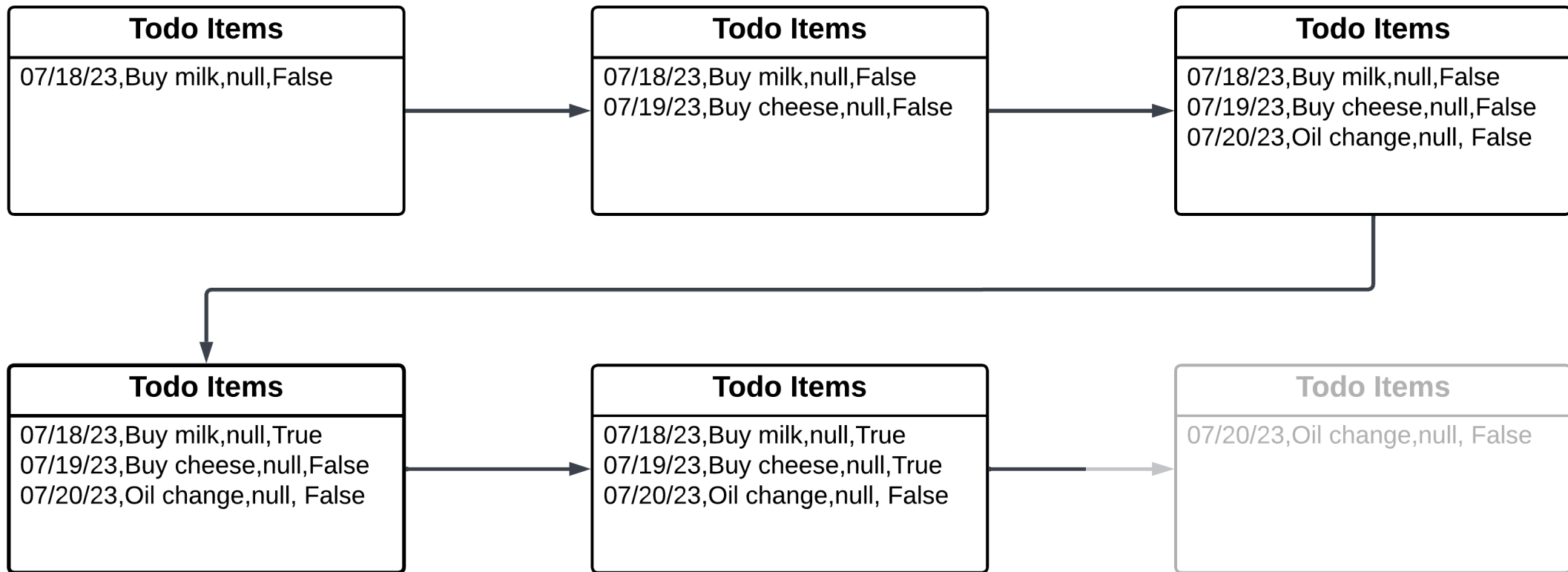
Life of a Database Table



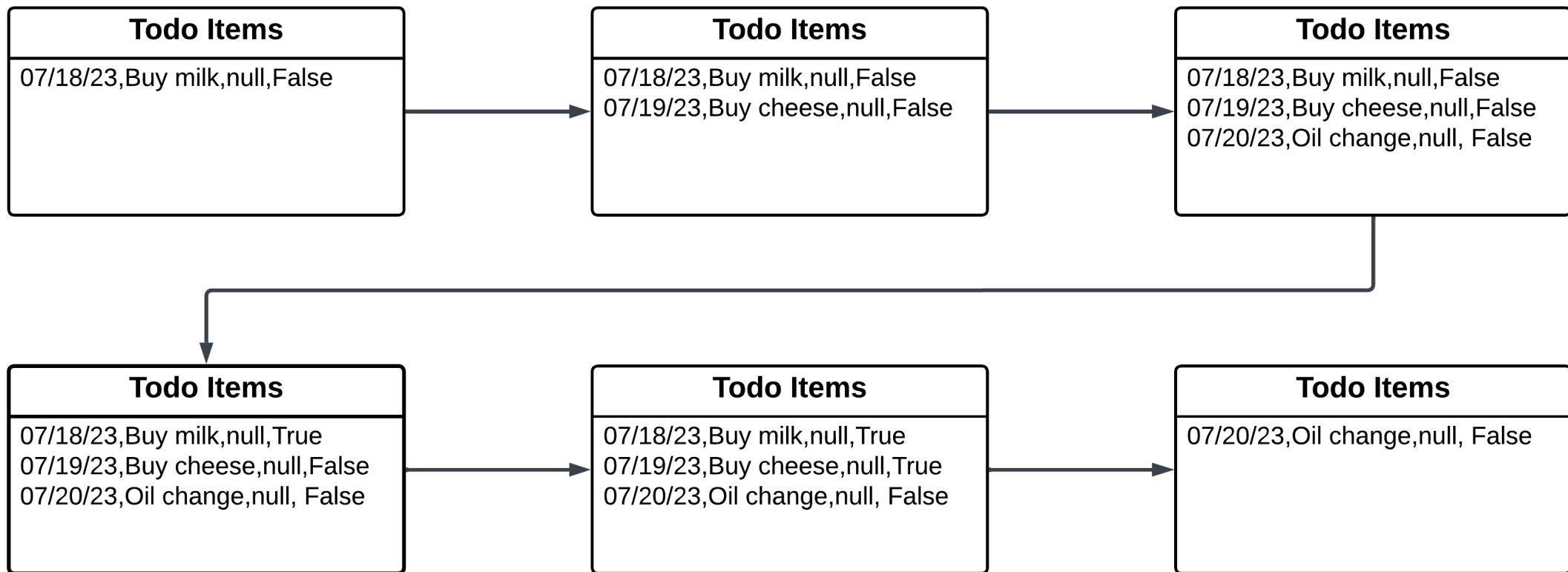
Life of a Database Table



Life of a Database Table



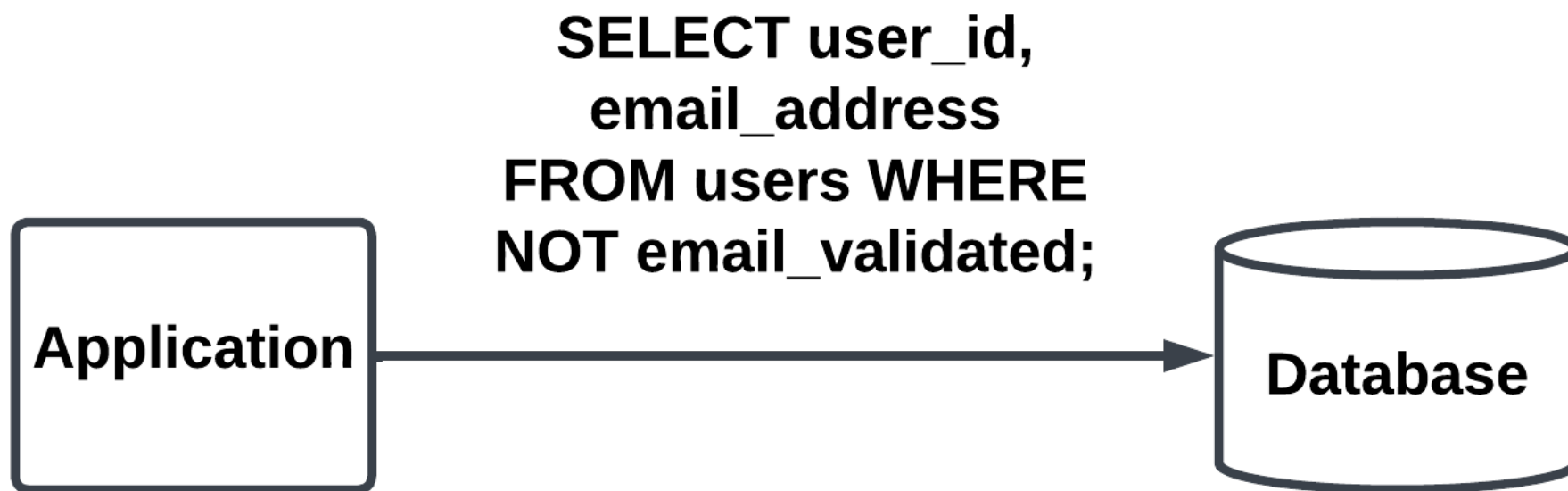
Life of a Database Table



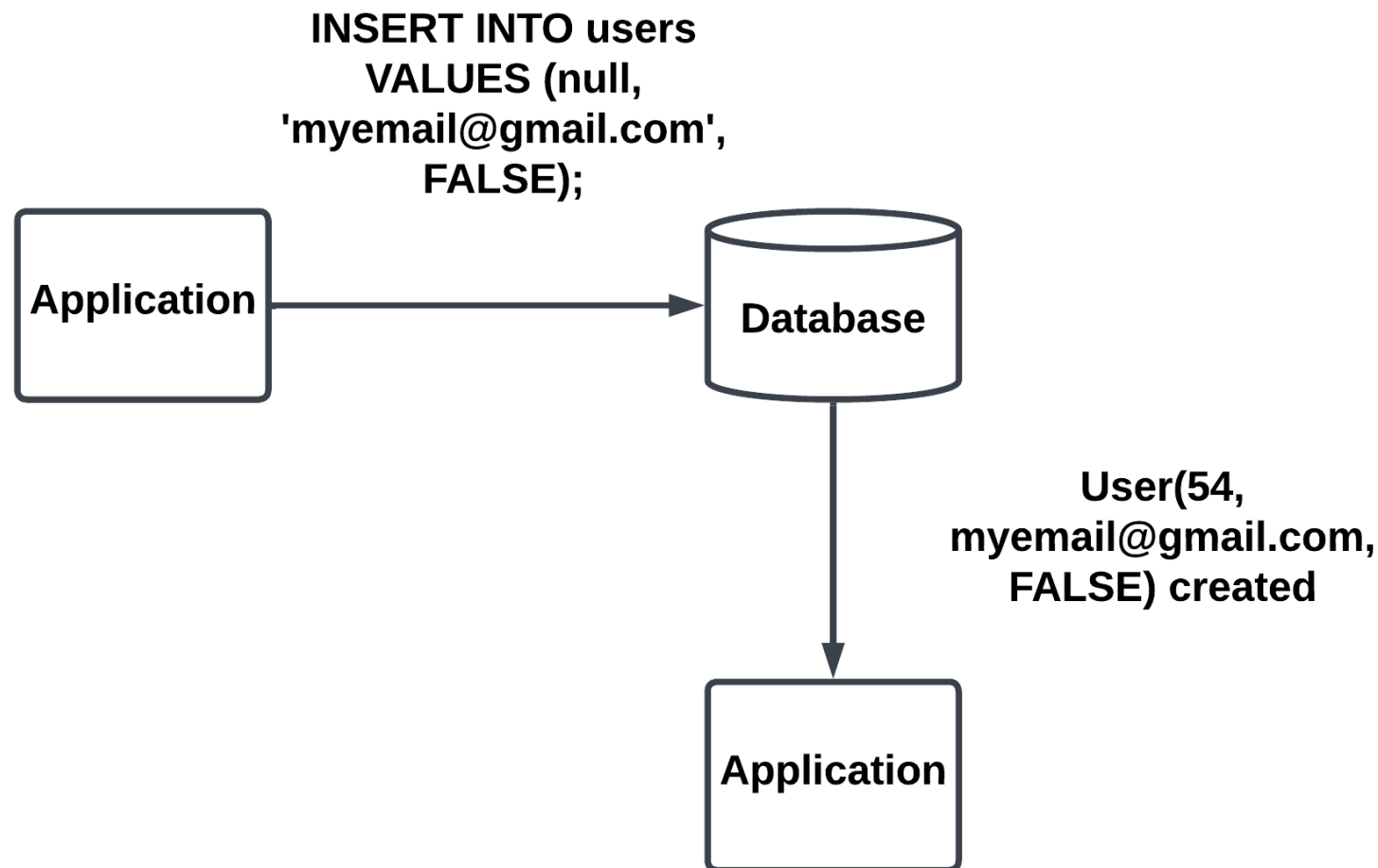
Change Log

1. Create "07/19/23,Buy cheese,null,False"
2. Create "07/20/23,Oil change,null, False"
3. Update "07/18/23,Buy milk,null,True"
4. Update "07/19/23,Buy cheese,null,True"
5. Delete "07/18/23,Buy milk,null,True"
"07/19/23,Buy cheese,null,True"

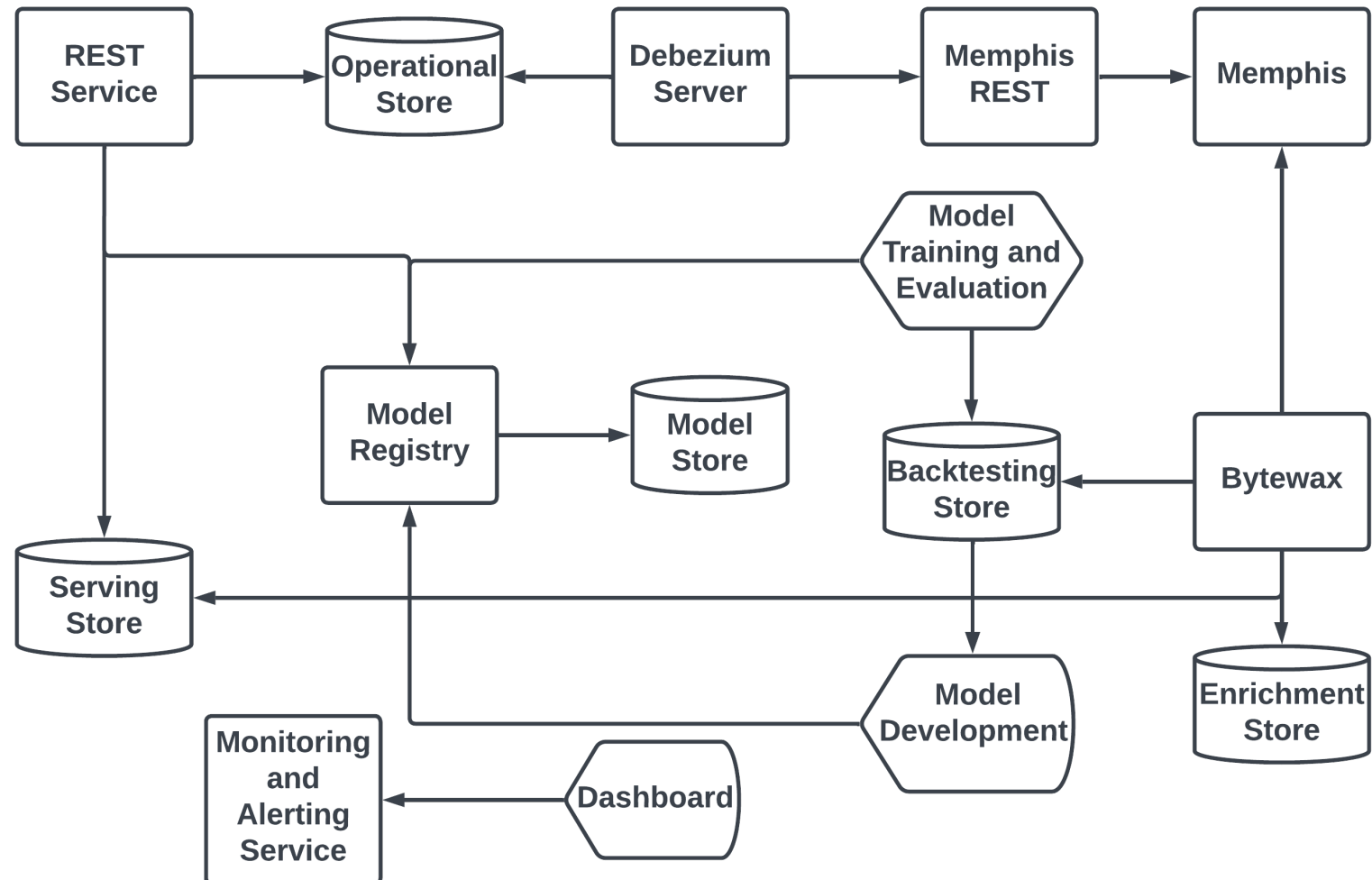
Application-Initiated I/O



Streaming Change Data Capture

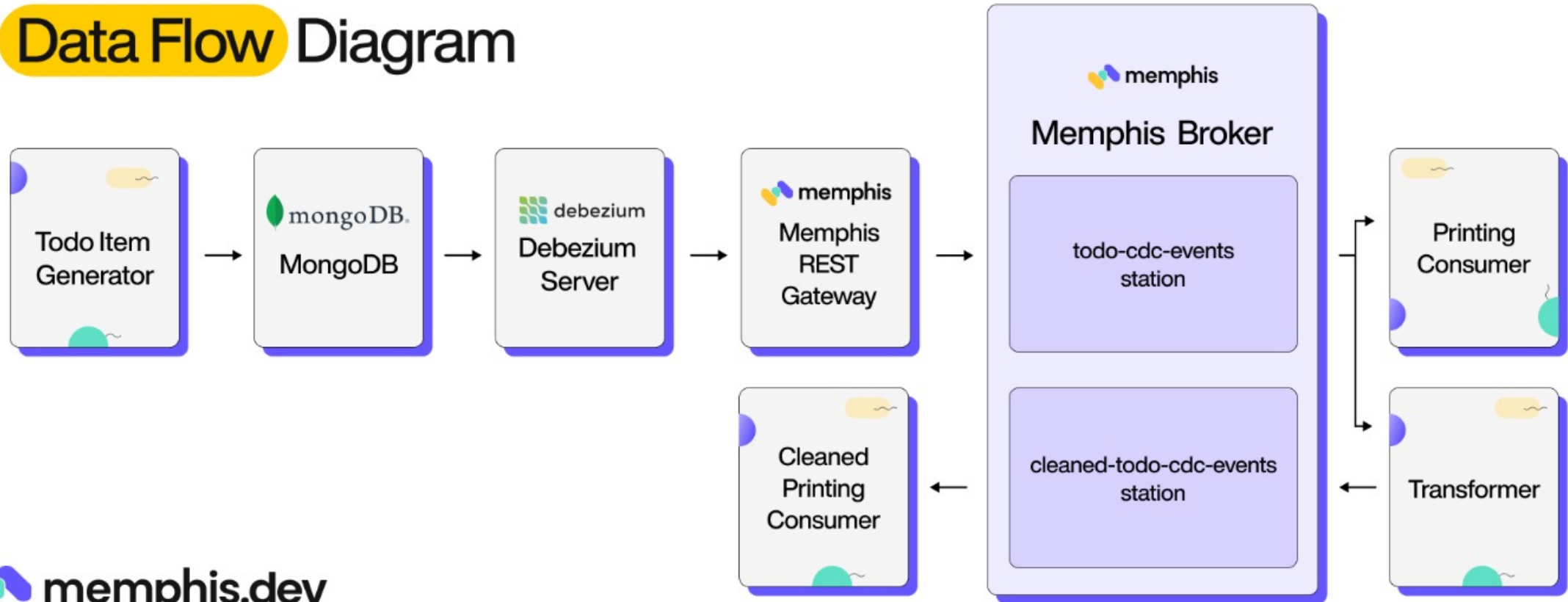


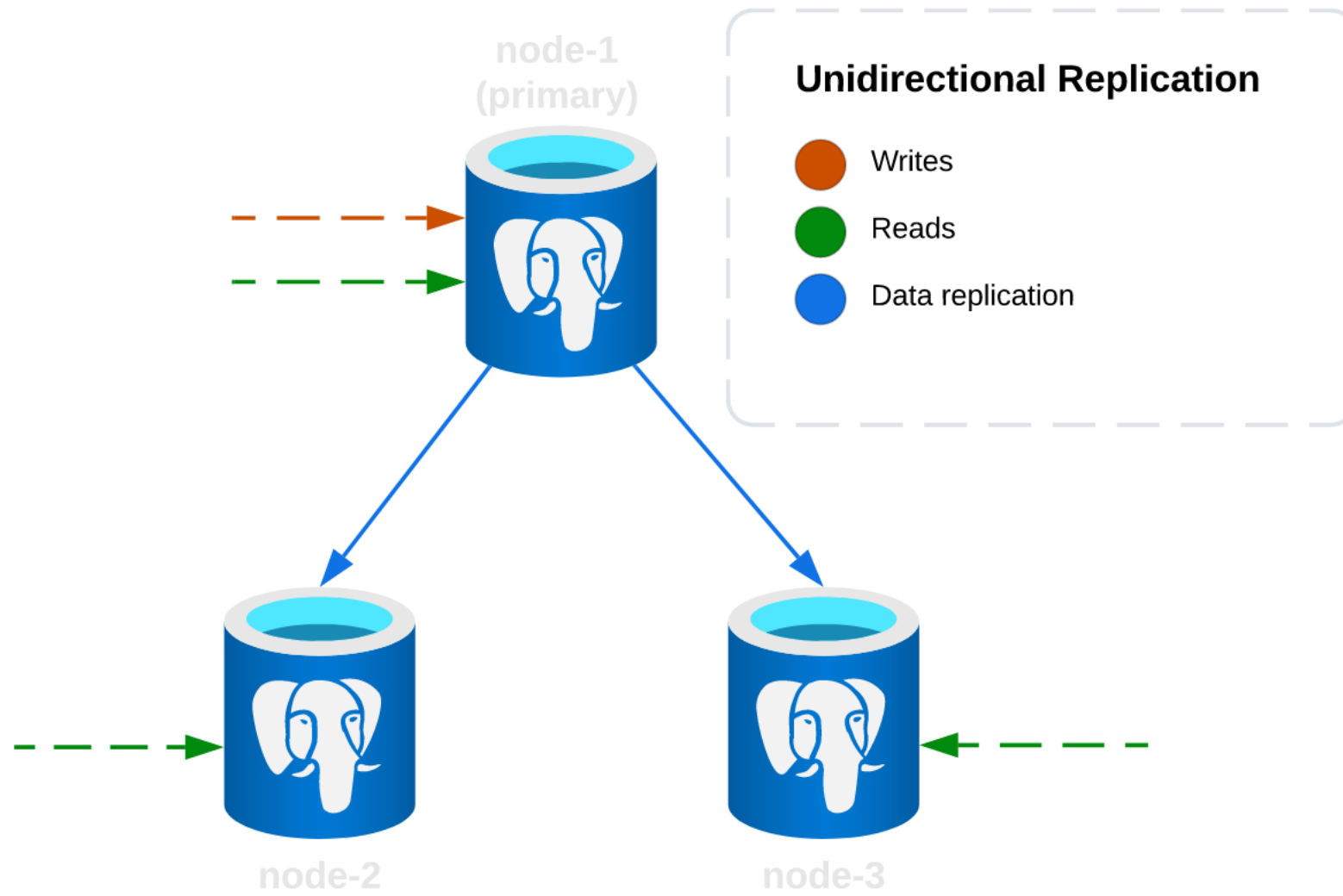
Example Architecture

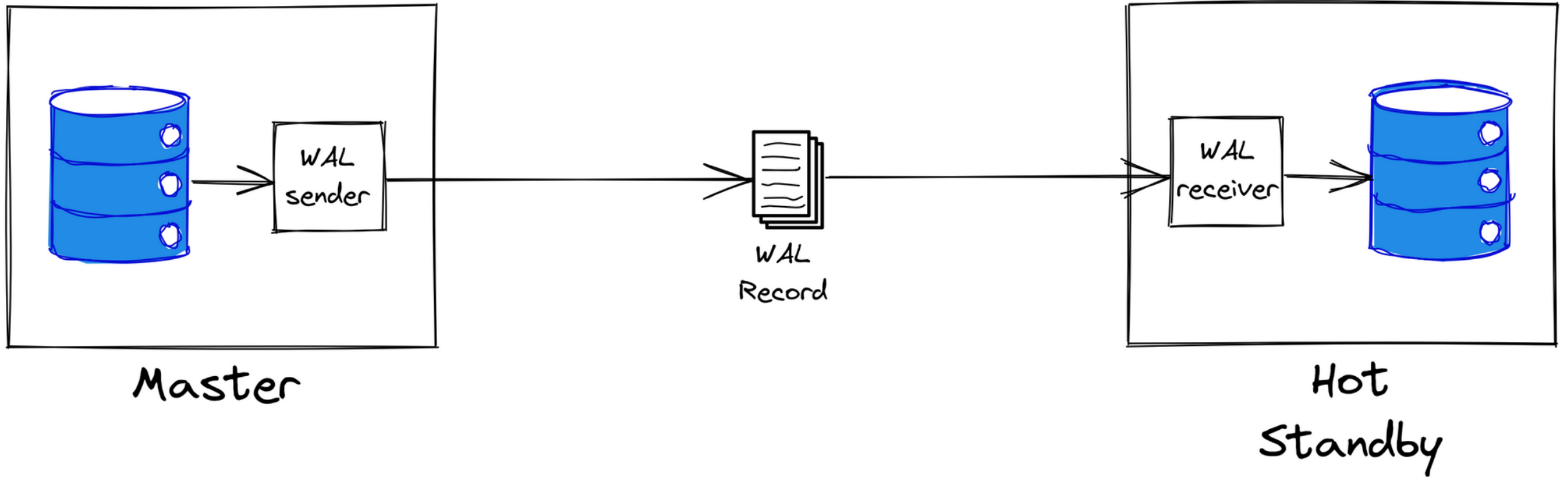


CDC Pipeline Architecture

Data Flow Diagram

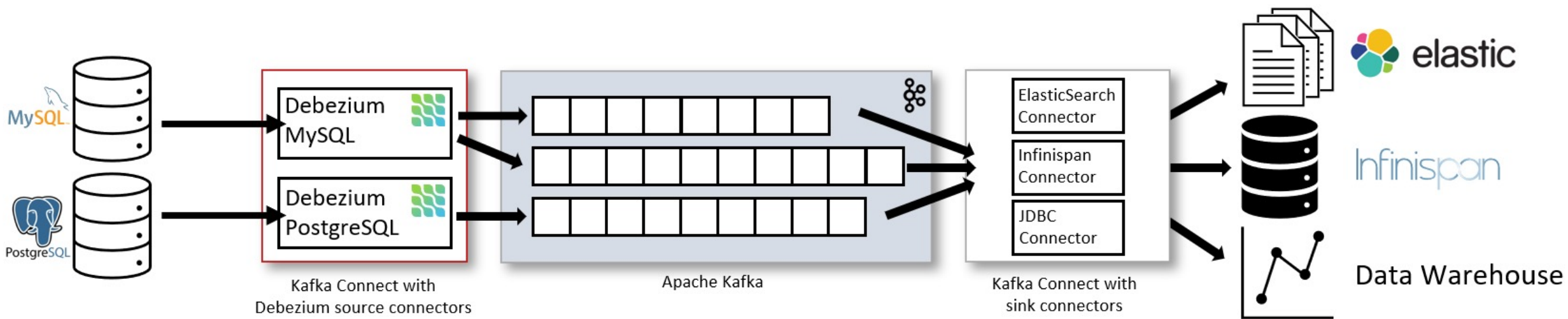








debezium



Schema on Read / Schema on Write

Schema on Write

- Tables are defined ahead of time in SQL databases
 - Values for required fields must be present
 - The values must be of the appropriate types
 - Cannot have additional fields
- Only conforming records can be added to the tables

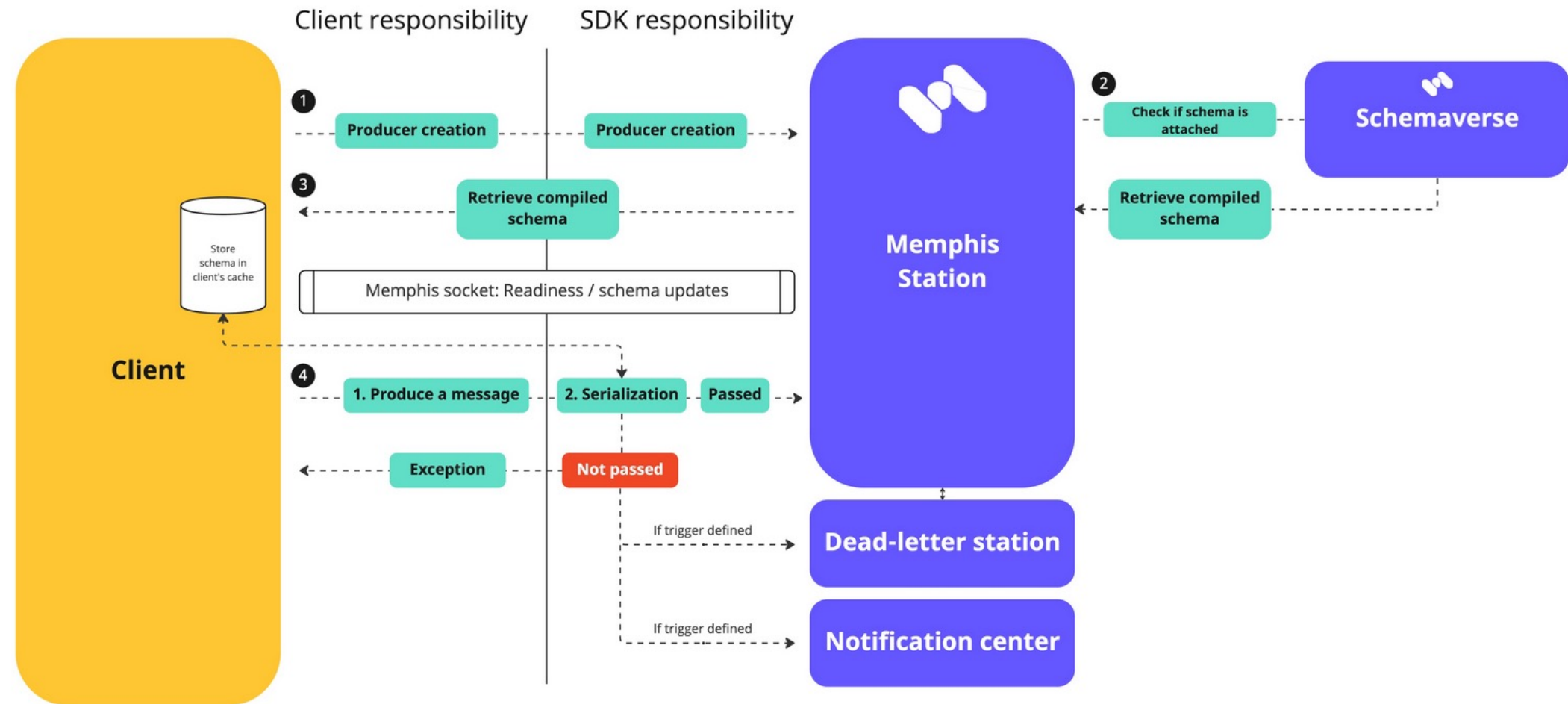
Schema on Read

- Collections in NoSQL databases can store a diverse range of objects: fields and value types can vary
- Objects are validated at read time -- the application must check that the desired fields exist and have the right types

Pipeline Reliability

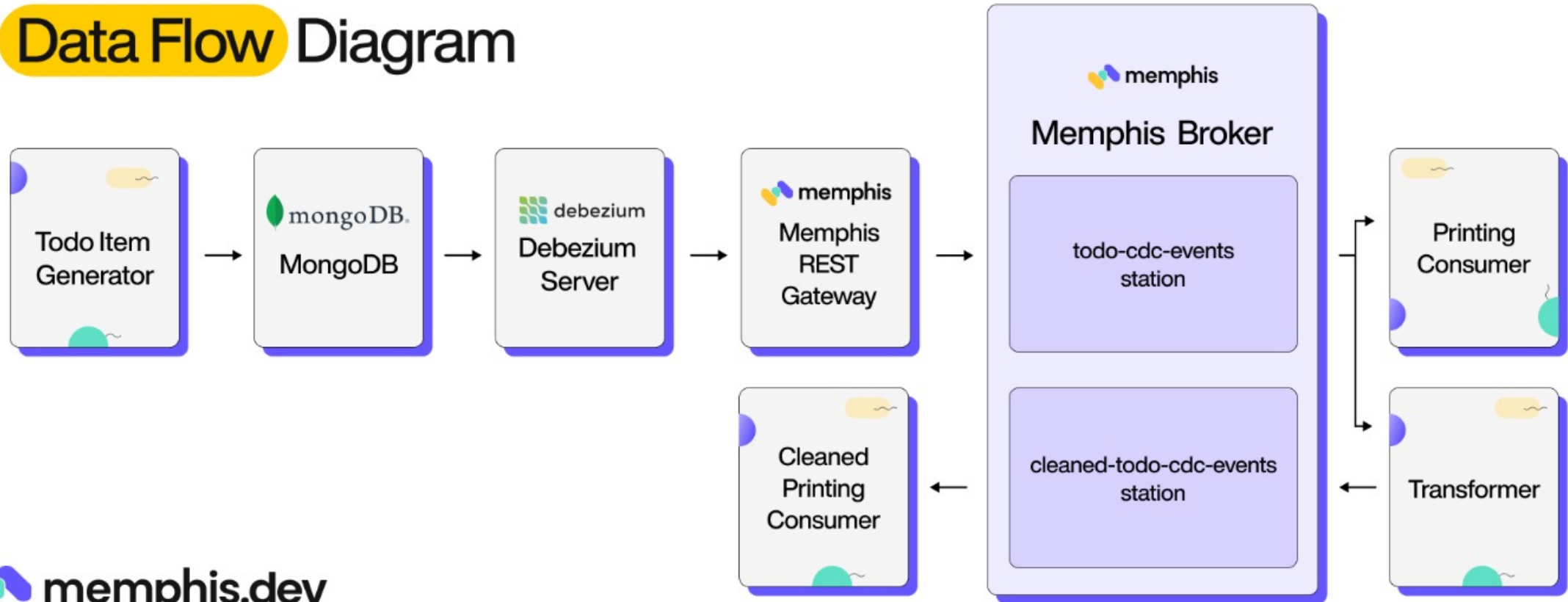


Schemaverse



CDC Pipeline Architecture

Data Flow Diagram



CDC Messages

```
{
  "schema" : ...,
  "payload" : {
    "before" : null,
    "after" : "{\\"_id\\": {\\"$oid\\":
\\\"645fe9eaf4790c34c8fcc2ed\\\"}, \\"creation_timestamp\\":
{\\"$date\\": 1684007402978}, \\"due_date\\":
{\\"$date\\": 1684266602978}, \\"description\\": \\"buy
milk\\", \\"completed\\": false}",
    ...
  }
}
```


Transformed Messages

```
{
  "schema" : ...,
  "payload" : {
    "before" : null,
    "after" : {
      "_id": { "$oid": "645fe9eaf4790c34c8fcc2ed" },
      "creation_timestamp": { "$date": 1684007402978 },
      "due_date": { "$date" : 1684266602978 },
      "description": "buy milk",
      "completed": false
    },
    ...
  }
}
```

JSON Schema

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "type" : "object",
  "properties" : {
    "payload" : {
      "type" : "object",
      "properties" : {
        "before" : { "oneOf" : [{ "type" : "null" }, { "$ref" : "#/$defs/todoItem" }] },
        "after" : { "oneOf" : [{ "type" : "null" }, { "$ref" : "#/$defs/todoItem" }] },
      },
      "required" : ["before", "after"]
    }
  },
  "required" : ["payload"],
}
```


JSON Schema

```
"$defs" : {  
  "todoItem" : {  
    "type" : "object",  
    "properties" : {  
      "_id" : {  
        "type" : "object",  
        "properties" : {  
          "$oid" : { "type" : "string" }  
        }  
      },  
    },  
  },  
}
```

JSON Schema

```
"description" : { "type" : "string" },  
"creation_timestamp" : {  
  "type" : "object",  
  "properties" : {  
    "$date" : { "type" : "integer" }  
  }  
},
```

JSON Schema

```
"due_date" : {  
  "anyOf" : [{  
    "type" : "object",  
    "properties" : {  
      "$date" : { "type" : "integer" }  
    }  
  },  
  { "type" : "null" } ]  
},  
"completed" : { "type" : "boolean" }  
},  
"required" : [ "_id", "description", "creation_timestamp", "completed" ]  
}  
}
```

Demo

- Let's switch to our live demo...

memphis-example-solutionsPublic

Edit PinsWatch1Fork0Star1

master3 branches0 tags

Go to fileAdd fileCode

rnowling-memphis Remove curl (#18)5d5a39d2 weeks ago28 commits

mongodb-debezium-cdc-example Remove curl (#18)2 weeks ago


postgres-debezium-cdc-example Use Python Debian images instead of vanilla Debian images (#15)last month

CODE_OF_CONDUCT.md Add license, code of conduct, and README (#1)3 months ago


LICENSE Add license, code of conduct, and README (#1)3 months ago

README.md Add license, code of conduct, and README (#1)3 months ago

README.md



Memphis is a next-generation alternative to traditional message brokers.



[Docs](#) - [Twitter](#) - [YouTube](#)

Introduction to this Repository

This repository contains examples of integrating [Memphis](#), the next-generation message broker, with other open-source technologies to solve technical use cases.

About

Example solutions using Memphis

Readme

View license

Code of conduct

Activity

1 star

1 watching

0 forks

Report repository

Releases

No releases published

Packages

No packages published

Contributors3

rnowling-memphis RJ Nowling

rnowling RJ Nowling

idanasulinmemphis Idan Asulin

Languages

Python94.6%

Dockerfile5.4%



<https://github.com/memphisdev/memphis-example-solutions>

Thanks!