Scaling Python to thousands of nodes with Ray

Rob de Wit-Liezenga

PyData Eindhoven // 2025-12-09

14 books + 1 prequel

11 526 pages

2787 named characters

~7 times Lord of the Rings

~4 times the bible

“Who is Tylin Quintara Mitsobar?”

Who she is

- Tylin is the nominal Queen of Altara (a nation in The Wheel of Time world). wot.fandom.com +1
- She is head of House Mitsobar. laruedadeltiempo... +1

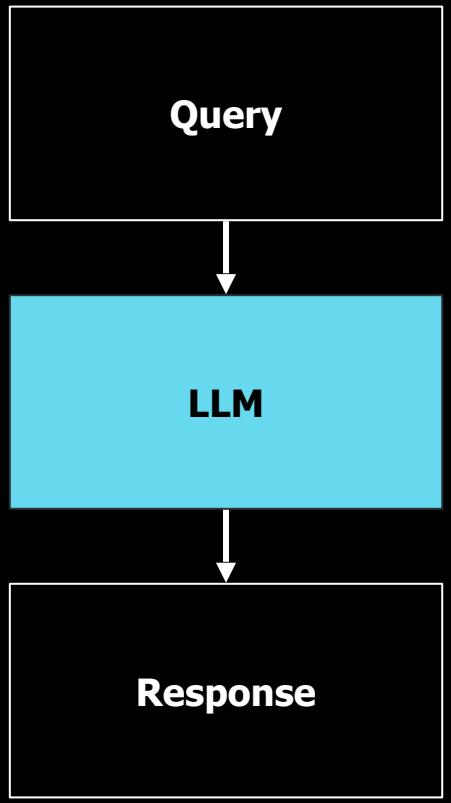
Background & Role

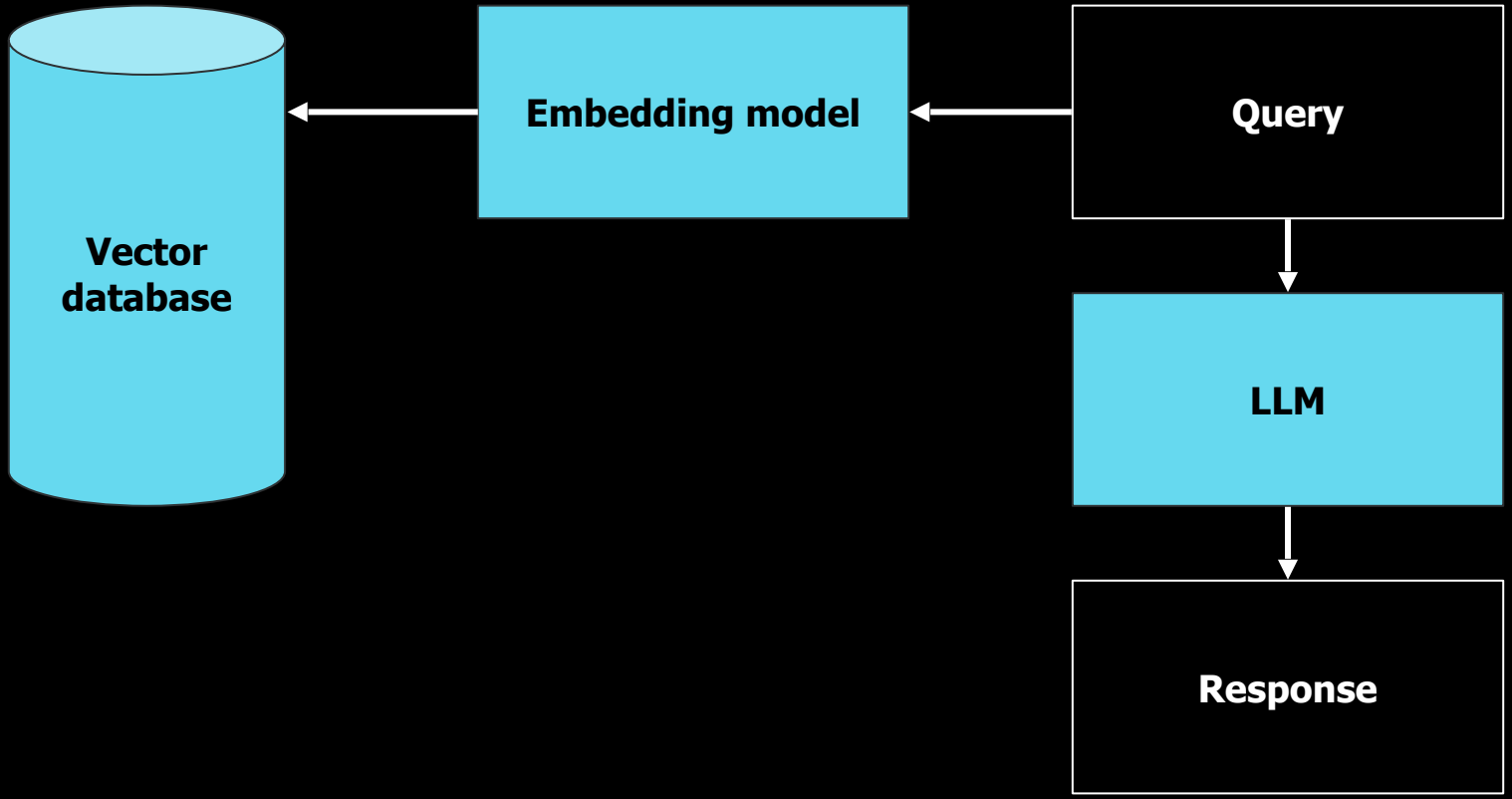
- Physically she is described as short-ish, with long black hair (graying at the temples), dark eyes, and two faint scars on her cheeks. wot.fandom.com +1
- She is a widow, had five children (four sons and one daughter), but by the time of the main storyline only one son, Beslan Mitsobar, remains alive — the others died in duels. wot.fandom.com +1

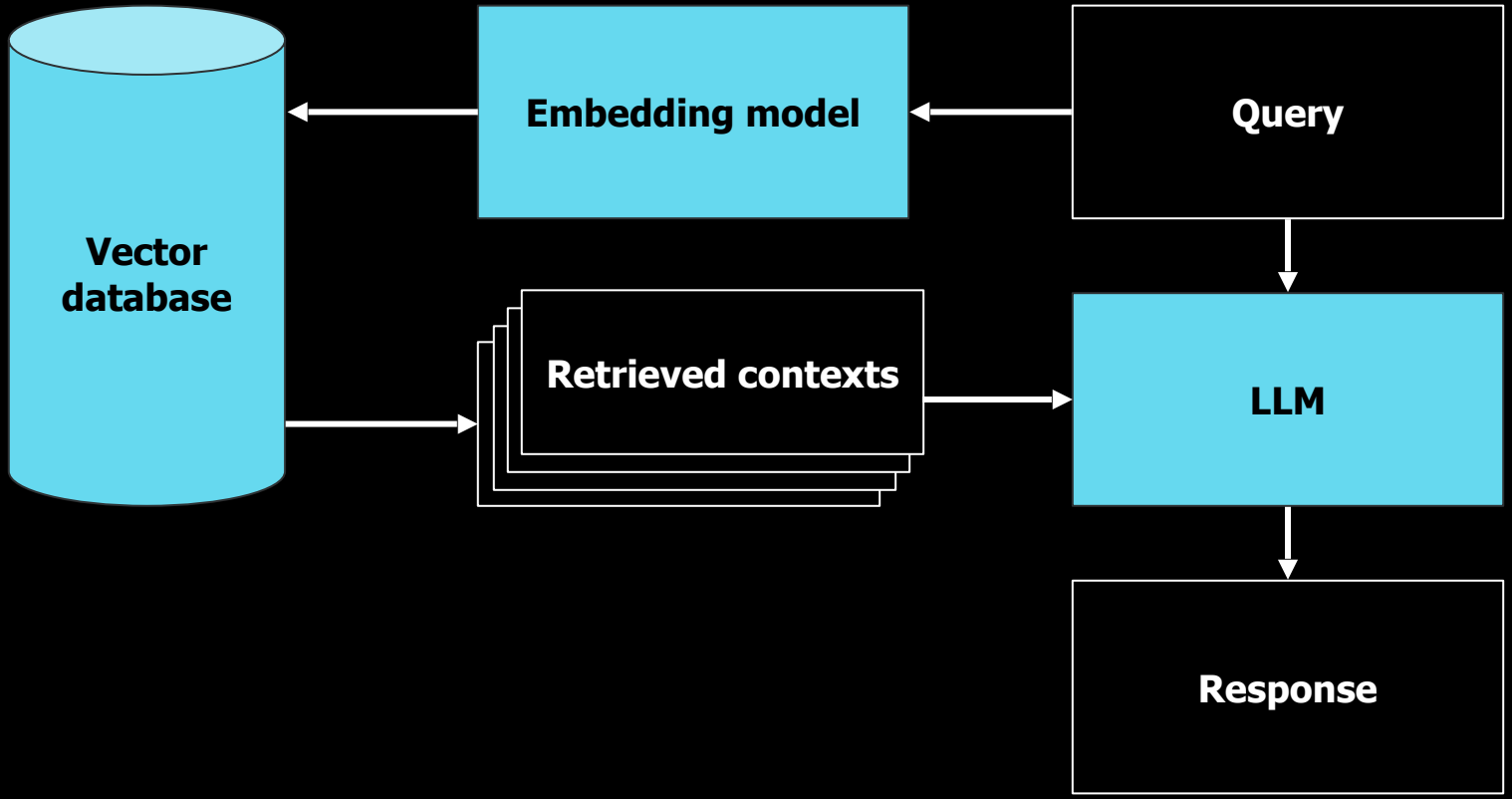
In-Story Events

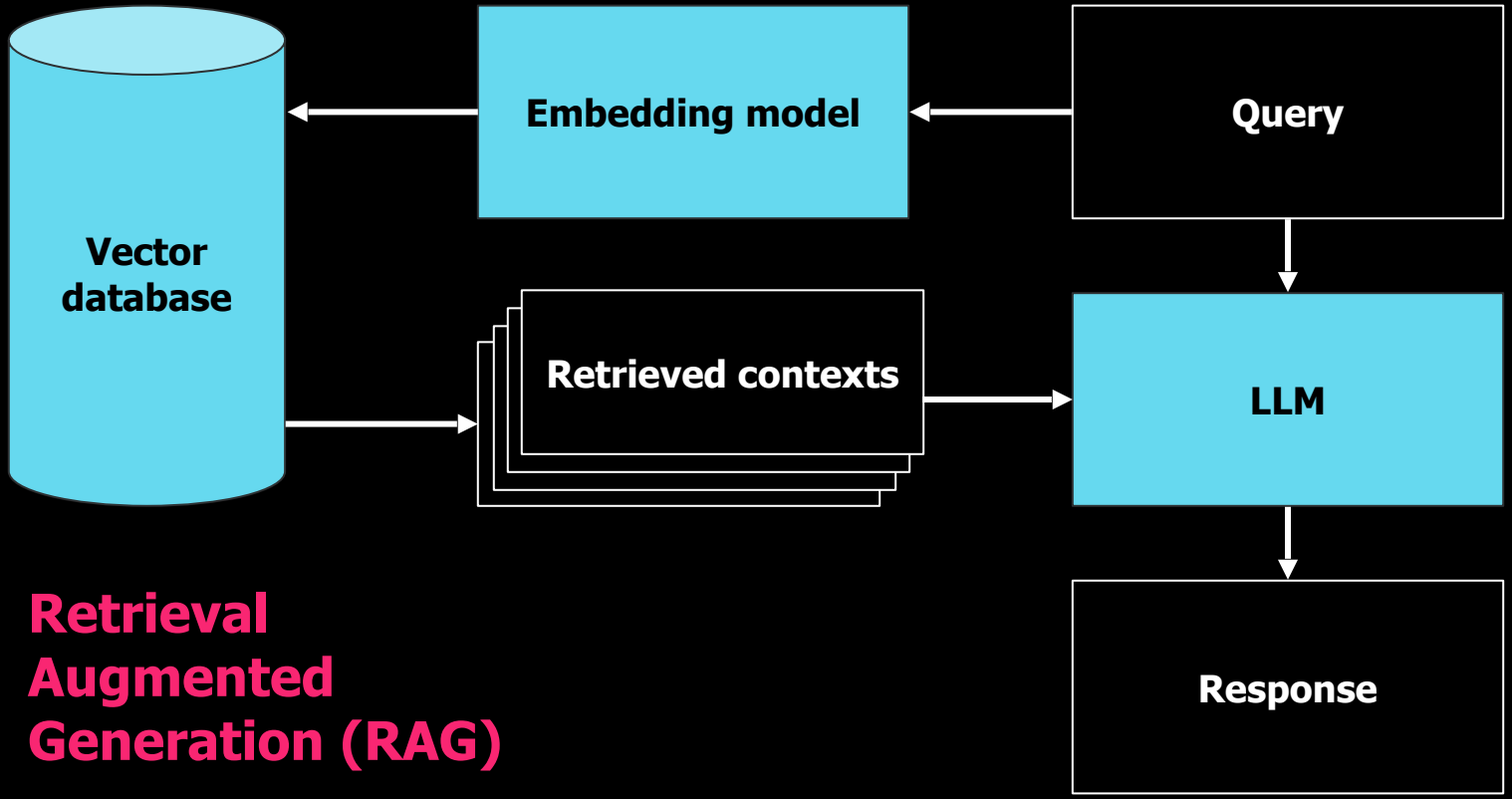
- She rules from the city Ebou Dar and interacts with other major characters when they visit seeking magical artifacts. karolginter.pl +1
- At one point [redacted], which many interpret as [redacted] though he later [redacted]. wot.fandom.com +1
- Later, during turmoil and political upheaval, she [redacted] over Altara [redacted] laruedadeltiempo... +1
- Ultimately she meets a tragic fate: [redacted] wot.fandom.com +1

What if…?


Query
LLM
Response


Vector database
Embedding model
Query
LLM
Response


Vector database
Embedding model
Query
Retrieved contexts
LLM
Response



Retrieval Augmented Generation (RAG)

About me

Rob de Wit-Liezenga

Freelance engineer

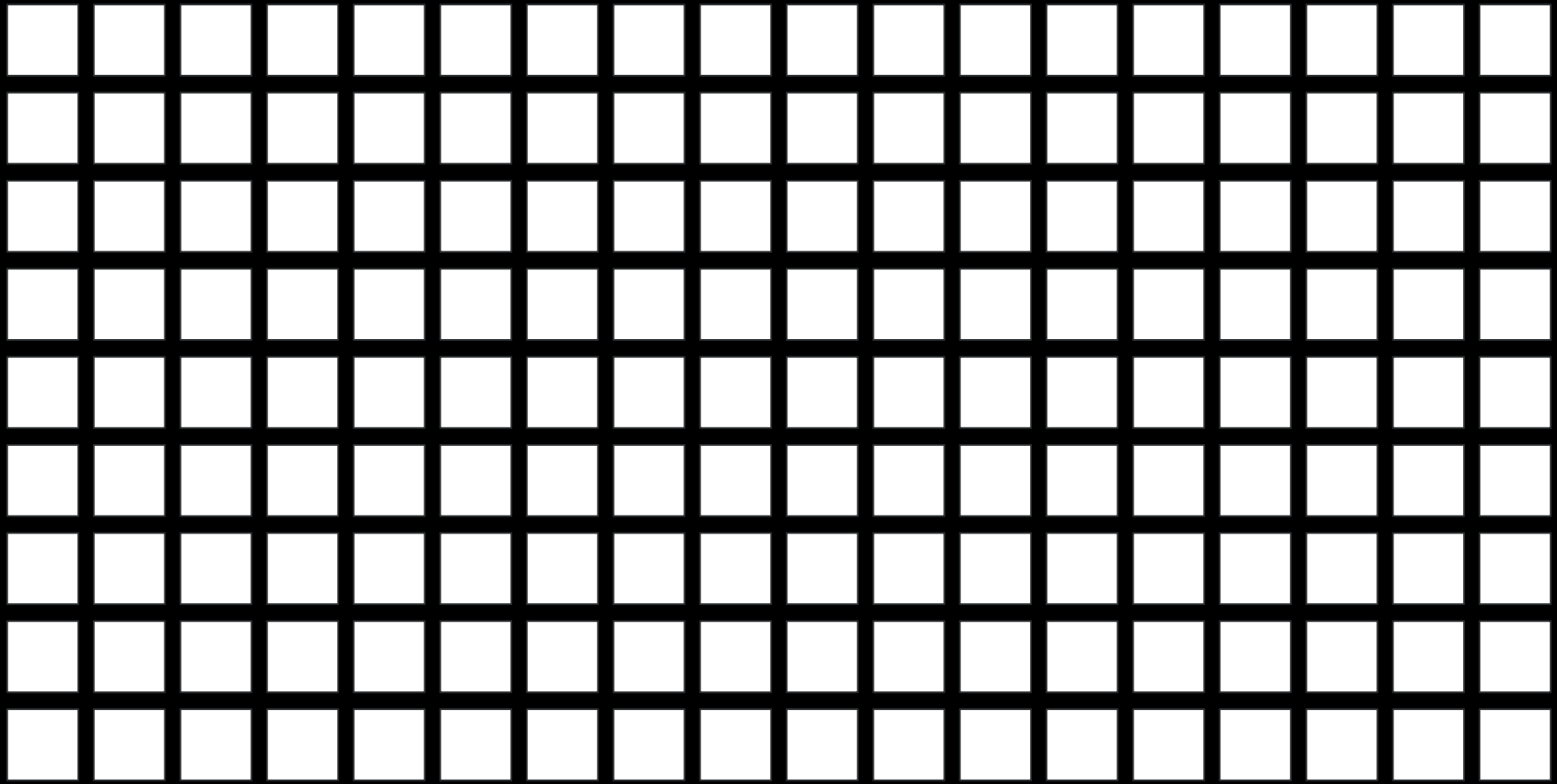
Driebergen, the Netherlands

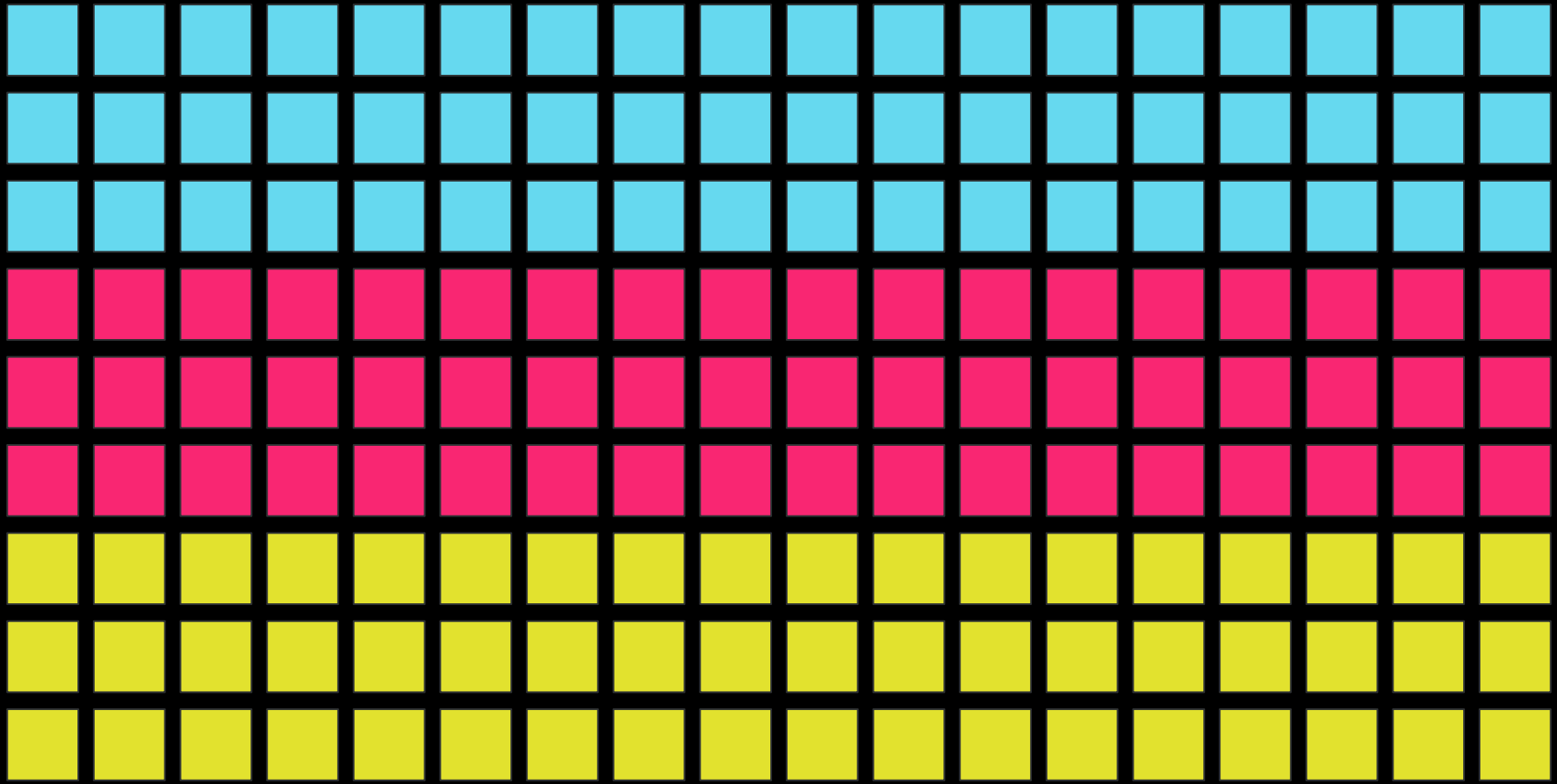
Customer Success Engineer @ Anyscale

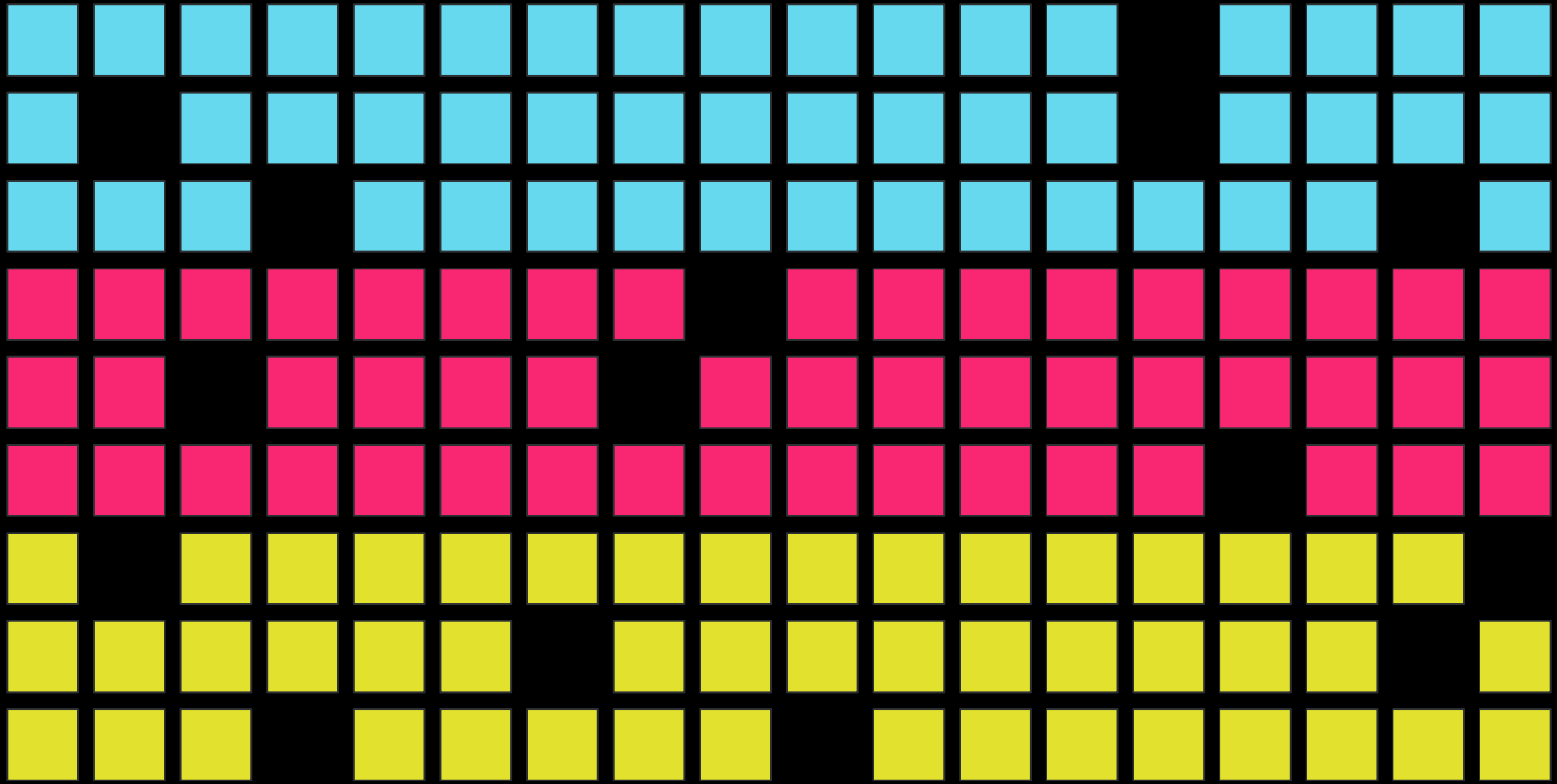
www.robdewit.nl

Let’s get to coding

From single machine to cluster

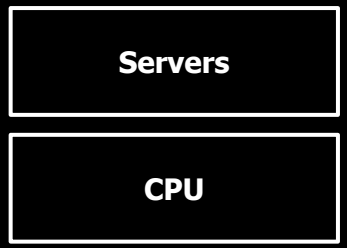




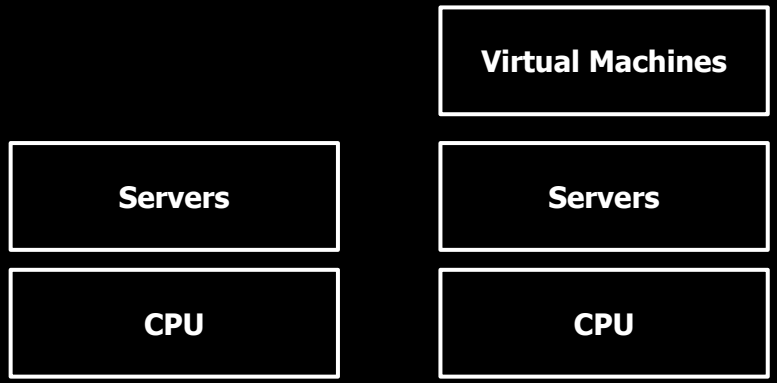


Distributed computing turns out to be difficult

Increasing complexity
means we invent new tech

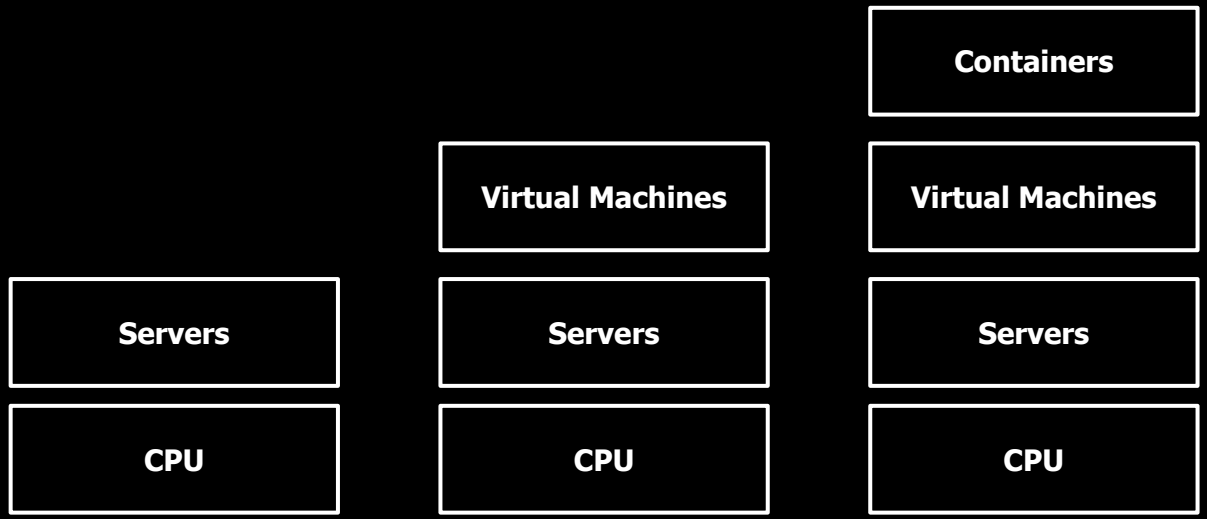


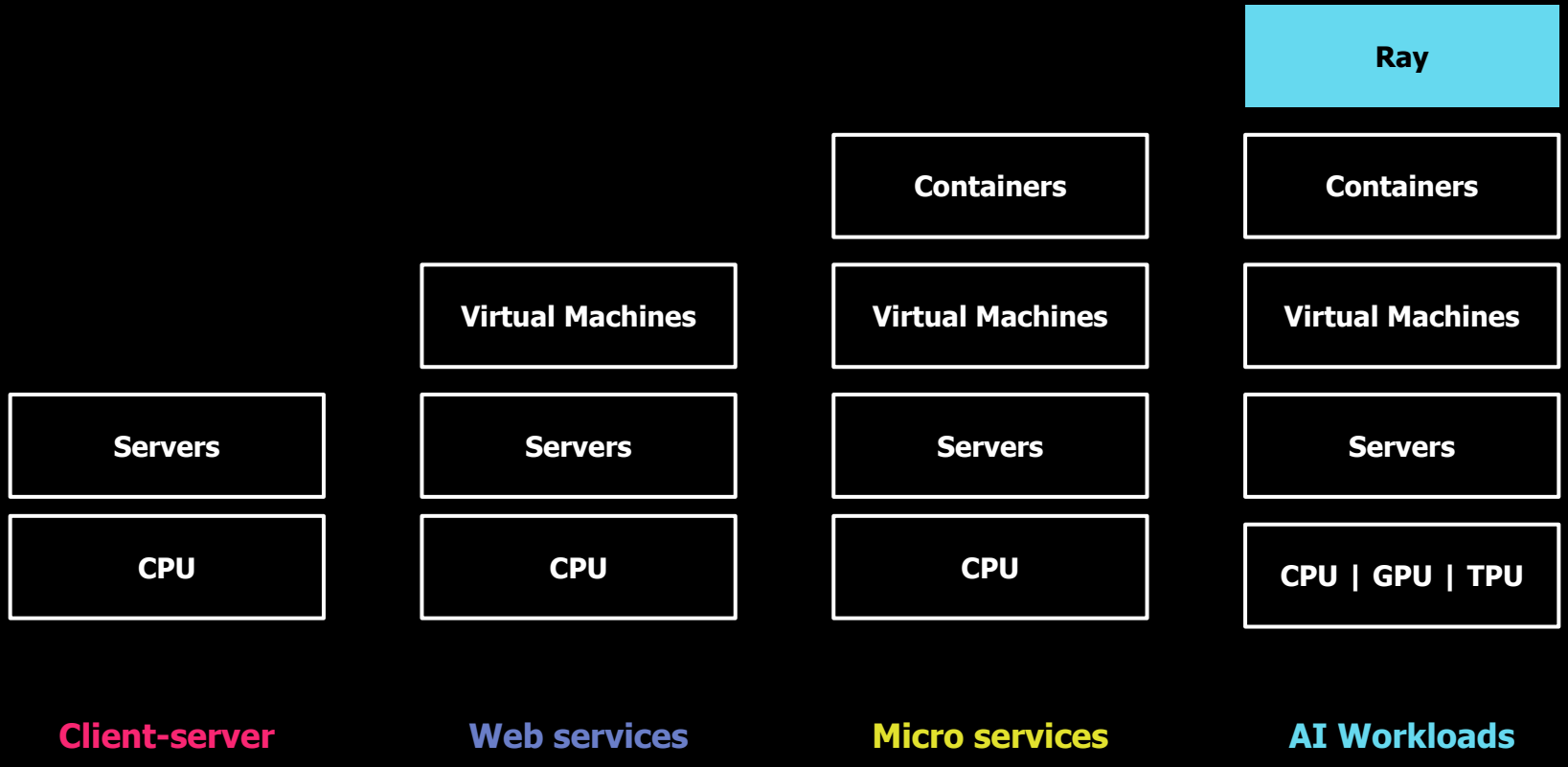
Client-server



Client-server

Web services


Containers
Virtual Machines
Virtual Machines
Servers
Servers
Servers
CPU
CPU
CPU
Client-server
Web services
Micro services

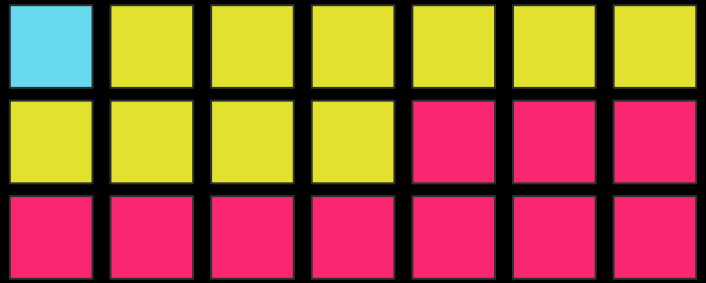

Ray
Containers
Containers
Virtual Machines
Virtual Machines
Virtual Machines
Servers
Servers
Servers
Servers
CPU
CPU
CPU
CPU | GPU | TPU
Client-server
Web services
Micro services
AI Workloads

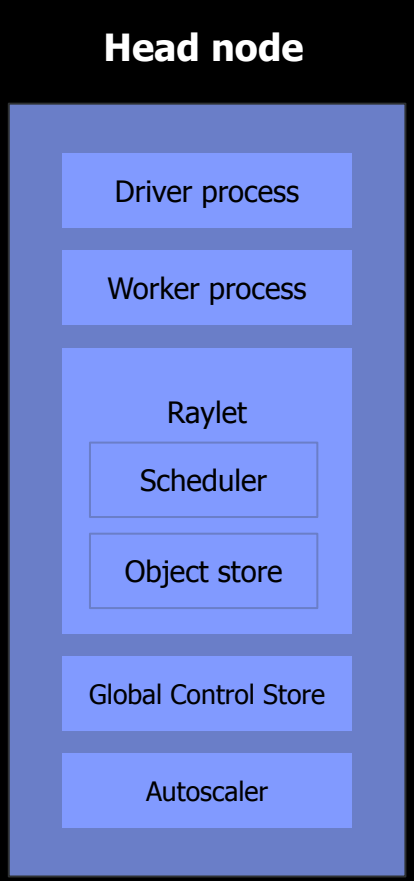
Ray clusters

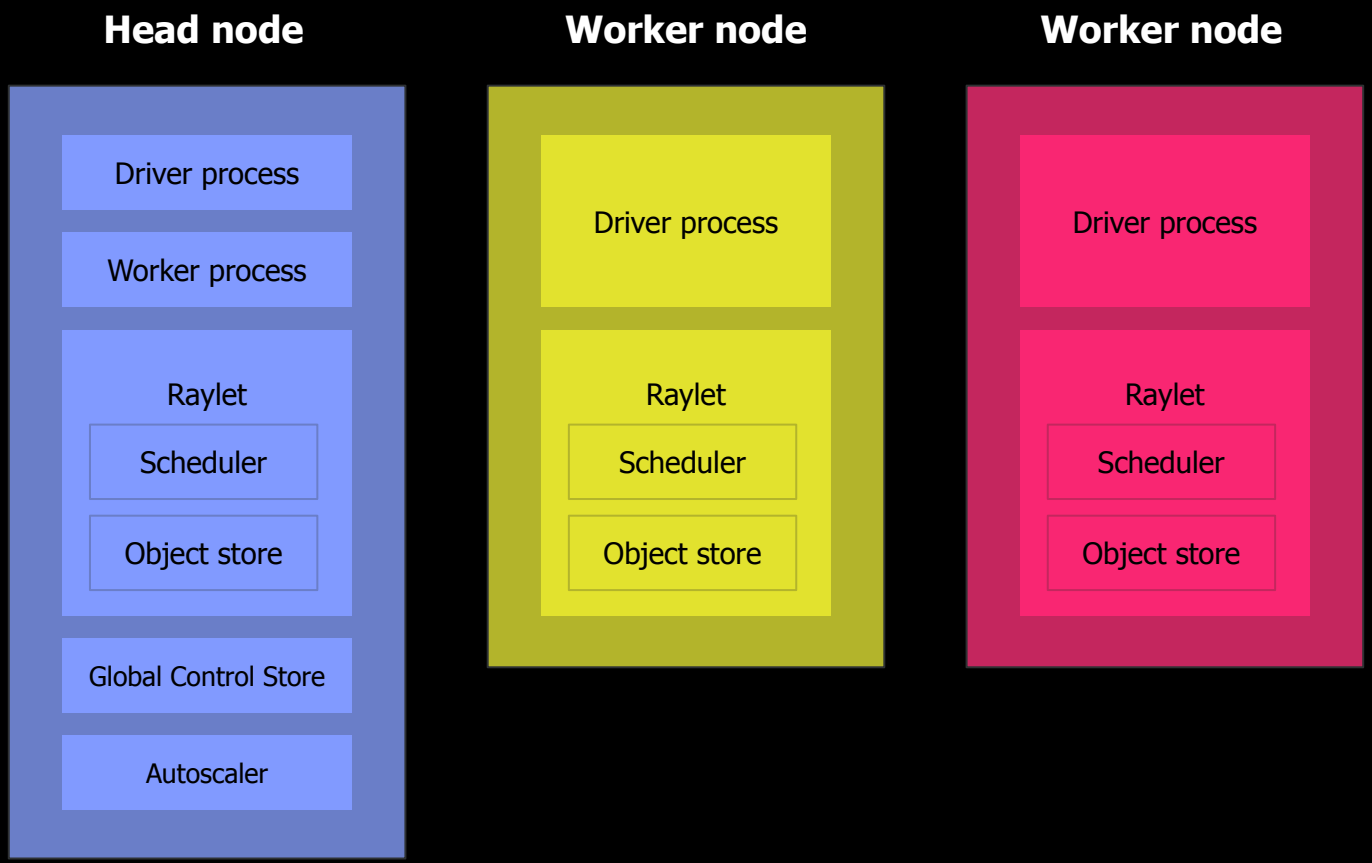
```yaml
cloud: rob-aws-uswest2
head_node:
  instance_type: m5.2xlarge
worker_nodes:
  - instance_type: m5.8xlarge
    min_nodes: 5
    max_nodes: 10
    market_type: ON_DEMAND
  - instance_type: g5.xlarge
    min_nodes: 0
    max_nodes: 10
    market_type: SPOT
idle_termination_minutes: 120
```

```yaml
cloud: rob-aws-uswest2
head_node:
  instance_type: m5.2xlarge
worker_nodes:
  - instance_type: m5.8xlarge
    min_nodes: 5
    max_nodes: 10
    market_type: ON_DEMAND
  - instance_type: g5.xlarge
    min_nodes: 0
    max_nodes: 10
    market_type: SPOT
idle_termination_minutes: 120
```

```yaml
cloud: rob-aws-uswest2
head_node:
  instance_type: m5.2xlarge
worker_nodes:
  - instance_type: m5.8xlarge
    min_nodes: 5
    max_nodes: 10
    market_type: ON_DEMAND
  - instance_type: g5.xlarge
    min_nodes: 0
    max_nodes: 10
    market_type: SPOT
idle_termination_minutes: 120
```




Head node
Driver process
Worker process
Raylet
Scheduler
Object store
Global Control Store
Autoscaler


Head node
Driver process
Worker process
Raylet
Scheduler
Object store
Global Control Store
Autoscaler
Worker node
Driver process
Raylet
Scheduler
Object store
Worker node
Driver process
Raylet
Scheduler
Object store

Tasks & actors

```python
import ray
import time



characters = ["Rand", "Mat", "Perrin", "Egwene", "Nynaeve"]


def retrieve(item):
    time.sleep(10)
    return item, characters[item]

indices = [retrieve(i) for i in range(5)]

print(indices)
# -> [(0, 'Rand'), (1, 'Mat'), (2, 'Perrin'), (3, 'Egwene'), (4, 'Nynaeve')]
# Time: ~50s
```

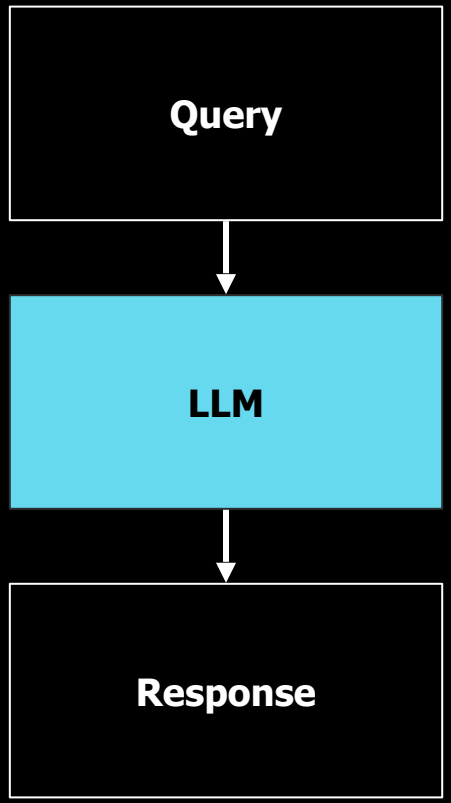
```python
import ray
import time

ray.init()

characters = ["Rand", "Mat", "Perrin", "Egwene", "Nynaeve"]

@ray.remote
def retrieve(item):
    time.sleep(10)
    return item, characters[item]

indices = [retrieve.remote(i) for i in range(5)]

print(ray.get(indices))
# -> [(0, 'Rand'), (1, 'Mat'), (2, 'Perrin'), (3, 'Egwene'), (4, 'Nynaeve')]
# Time: ~10s
```

```python
class AesSedai:
    def __init__(self, name, power):
        self.name = name
        self.power = power
        self.oaths = 0
    def get(self):
        return (self.name, self.power, self.oaths)
    def take_oath(self):
        if self.oaths >= 3:
            raise ValueError("Already taken 3 oaths")
        self.oaths += 1
        return self.oaths
    def still(self):
        self.power = 0
        self.oaths = 0
```

```python
import ray

ray.init()

@ray.remote
class AesSedai:
    def __init__(self, name, power):
        self.name = name
        self.power = power
        self.oaths = 0
    def get(self):
        return (self.name, self.power, self.oaths)
    def take_oath(self):
        if self.oaths >= 3:
            raise ValueError("Already taken 3 oaths")
        self.oaths += 1
        return self.oaths
    def still(self):
        self.power = 0
        self.oaths = 0
```

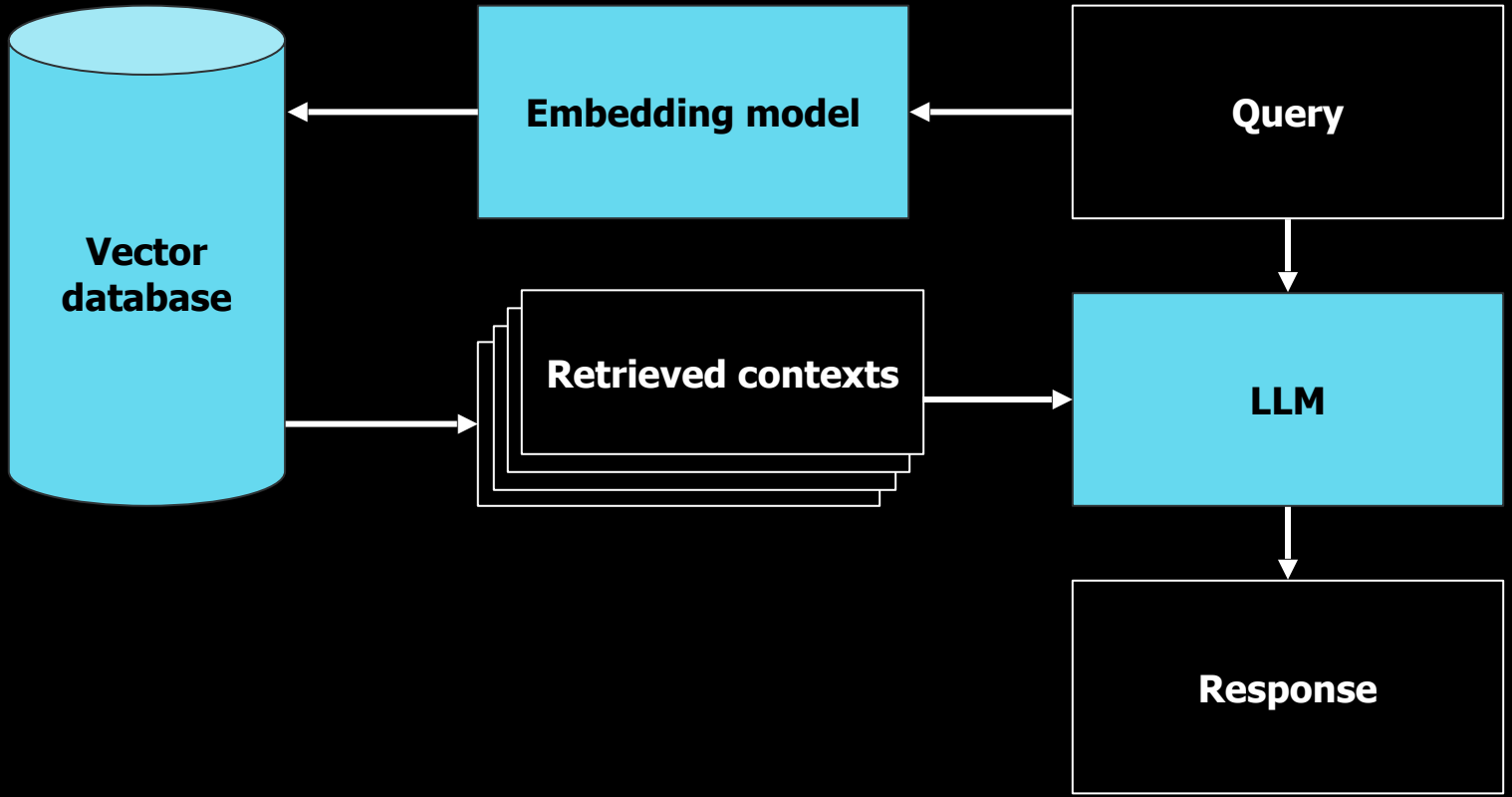
```python
import ray

ray.init()

@ray.remote
class AesSedai:
    ...

siuan = AesSedai.remote("Siuan", 13)

for _ in range(3):
    ray.get(siuan.take_oath.remote())

print(ray.get(siuan.get.remote()))
# -> ('Siuan', 13, 3)

ray.get(siuan.still.remote())

print(ray.get(siuan.get.remote()))
# -> ('Siuan', 0, 0)
```

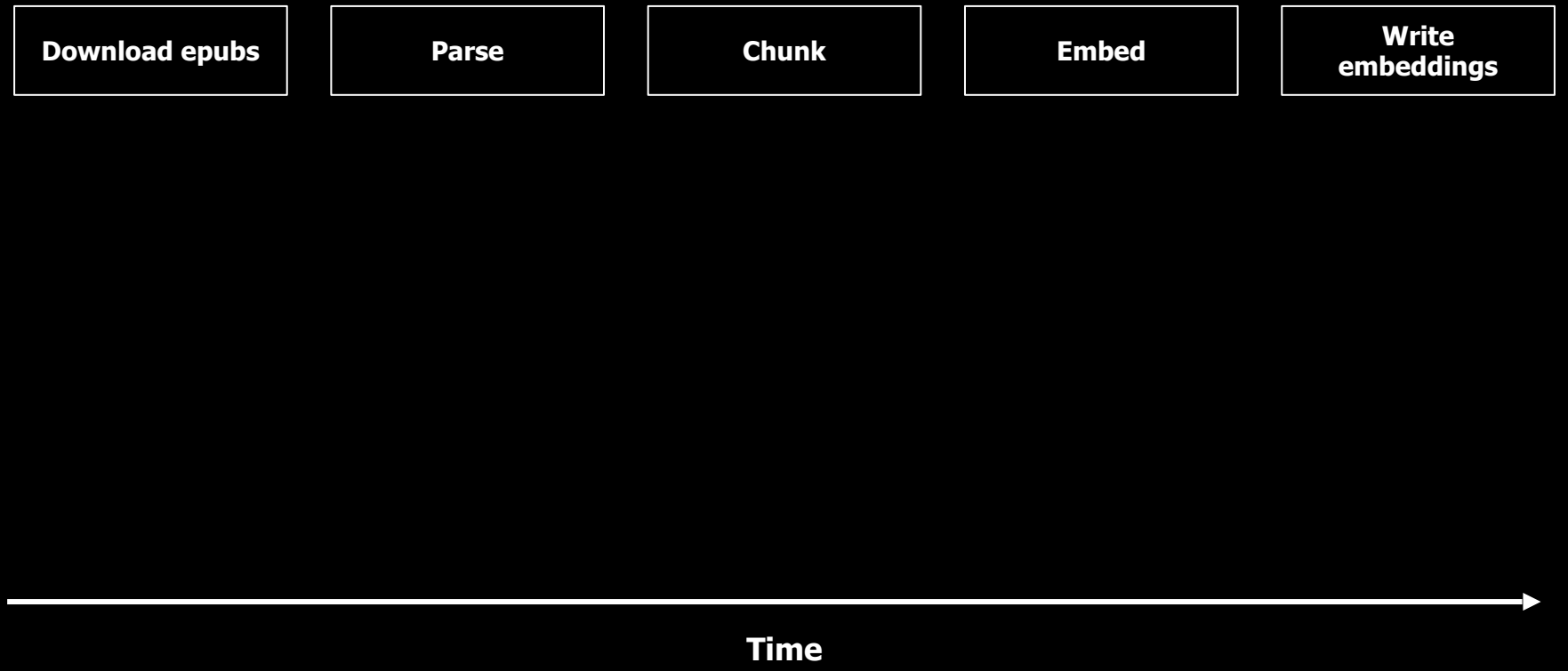
RAG pipeline

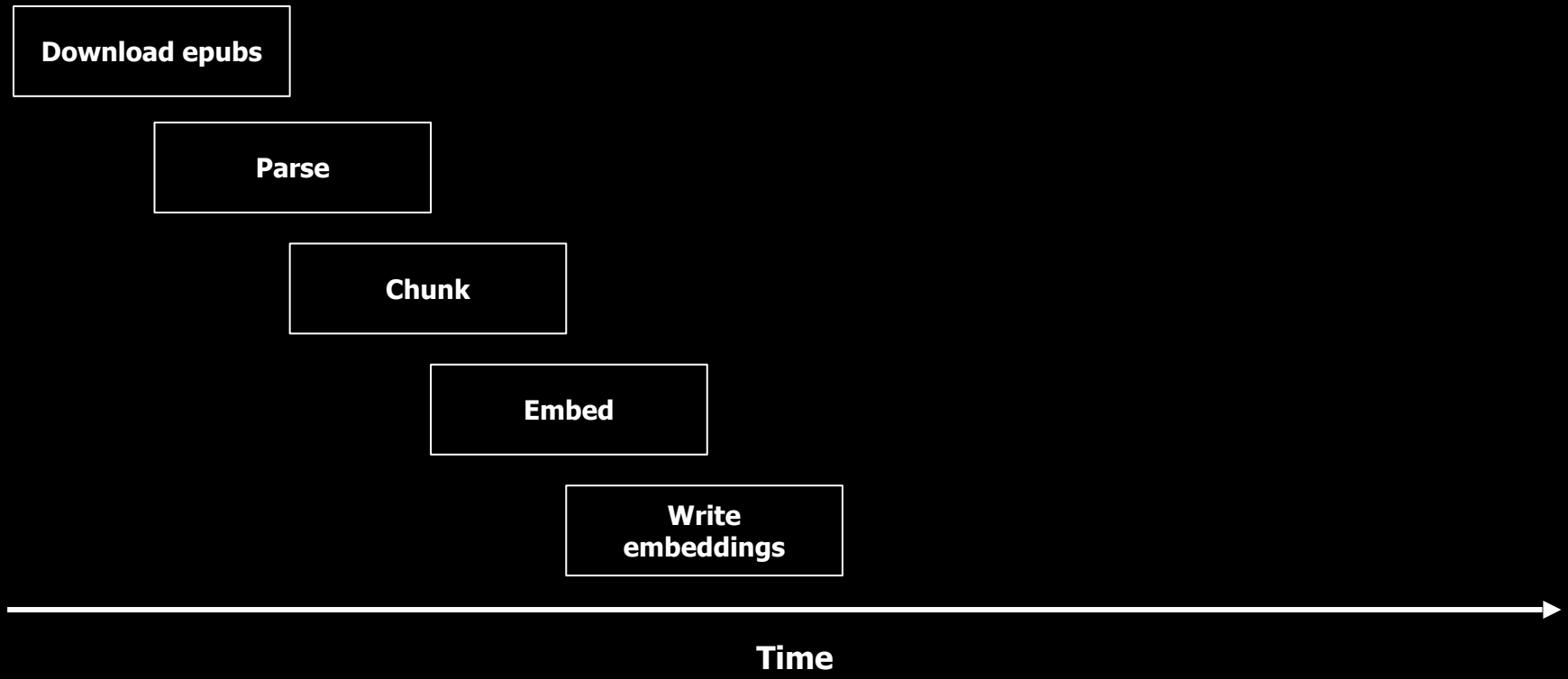

Query
LLM
Response

“Who is Tylin Quintara Mitsobar?”

```
(default) robdewit@Robs-MacBook-Pro-2 ray-wheel-of-time % curl -X POST "https://rag-llm-service-
wwcnm.cld-rcnjlw42qd461rmr.s.anyscaleuserdata.com/v1/chat/completions" \
    -H "Authorization: Bearer M9K-6G8YIdo3DSpYZfc8njHM9WWzH8-IXKin5QjGfZw" \
    -H "Content-Type: application/json" \
    -d '{
      "model": "mistral-7b-instruct",
      "messages": [
        {"role": "user", "content": "Tylin Quintara Mitsobar?"}
      ],
      "temperature": 0,
      "max_tokens": 500
    }'
{"id":"chatcmpl-a1461c65-25b2-4b12-
b876-6d8e5e58a25a","object":"chat.completion","created":1765229372,"model":"mistral-7b-
instruct","choices":[{"index":0,"message":{"role":"assistant","content":" It seems like you've
provided a name in a fictional language, possibly from the \"A Song of Ice and Fire\" series by
George R.R. Martin, as it resembles the Dothraki language. However, I don't have the specific
knowledge of this language to break down the name \"Tylin Quintara Mitsobar.\" I recommend checking
out resources dedicated to the Dothraki language or the series for a more accurate
interpretation.","refusal":null,"annotations":null,"audio":null,"function_call":null,"tool_calls":
[],"reasoning_content":null},"logprobs":null,"finish_reason":"stop","stop_reason":null}],"service_ti
er":null,"system_fingerprint":null,"usage":
{"prompt_tokens":13,"total_tokens":112,"completion_tokens":99,"prompt_tokens_details":null},"prompt_
logprobs":null,"kv_transfer_params":null}%
```


Vector database
Embedding model
Query
Retrieved contexts
LLM
Response


Download epubs
Parse
Chunk
Embed
Write embeddings
Time


Download epubs
Parse
Chunk
Embed
Write embeddings
Time

Ray Data

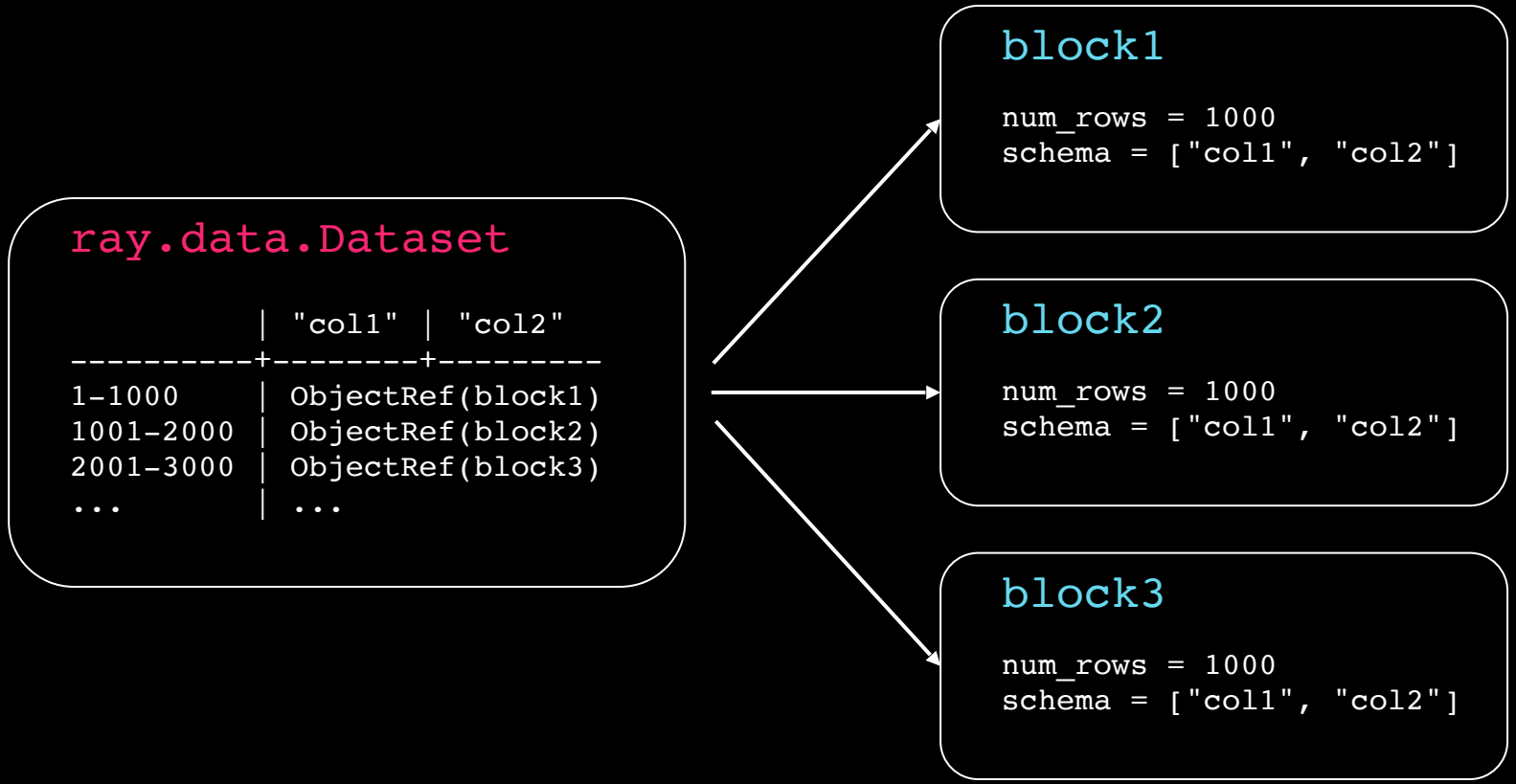
Ray Data

Streaming execution

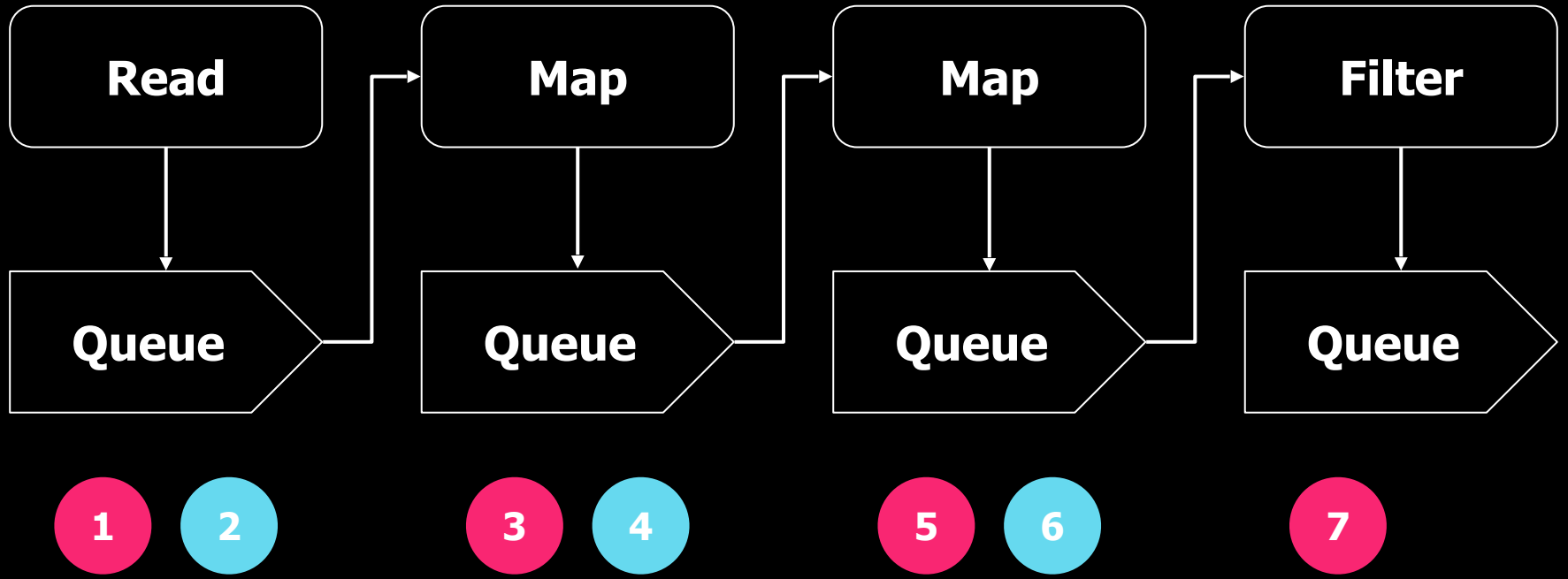
Core concepts: datasets and blocks

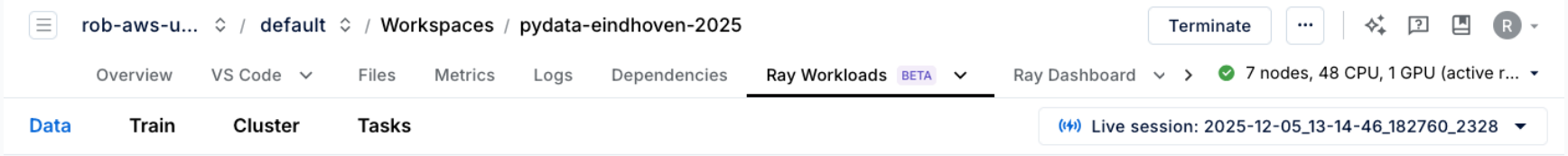
Ideal for AI workloads:

- Batch inference
- Data preprocessing
- Ingest for ML training
- …


ray.data.Dataset
| "col1" | "col2"
1-1000 | ObjectRef(block1)
1001-2000 | ObjectRef(block2)
2001-3000 | ObjectRef(block3)
... | ...
block1
num_rows = 1000
schema = ["col1", "col2"]
block2
num_rows = 1000
schema = ["col1", "col2"]
block3
num_rows = 1000
schema = ["col1", "col2"]

Streaming topology





Datasets

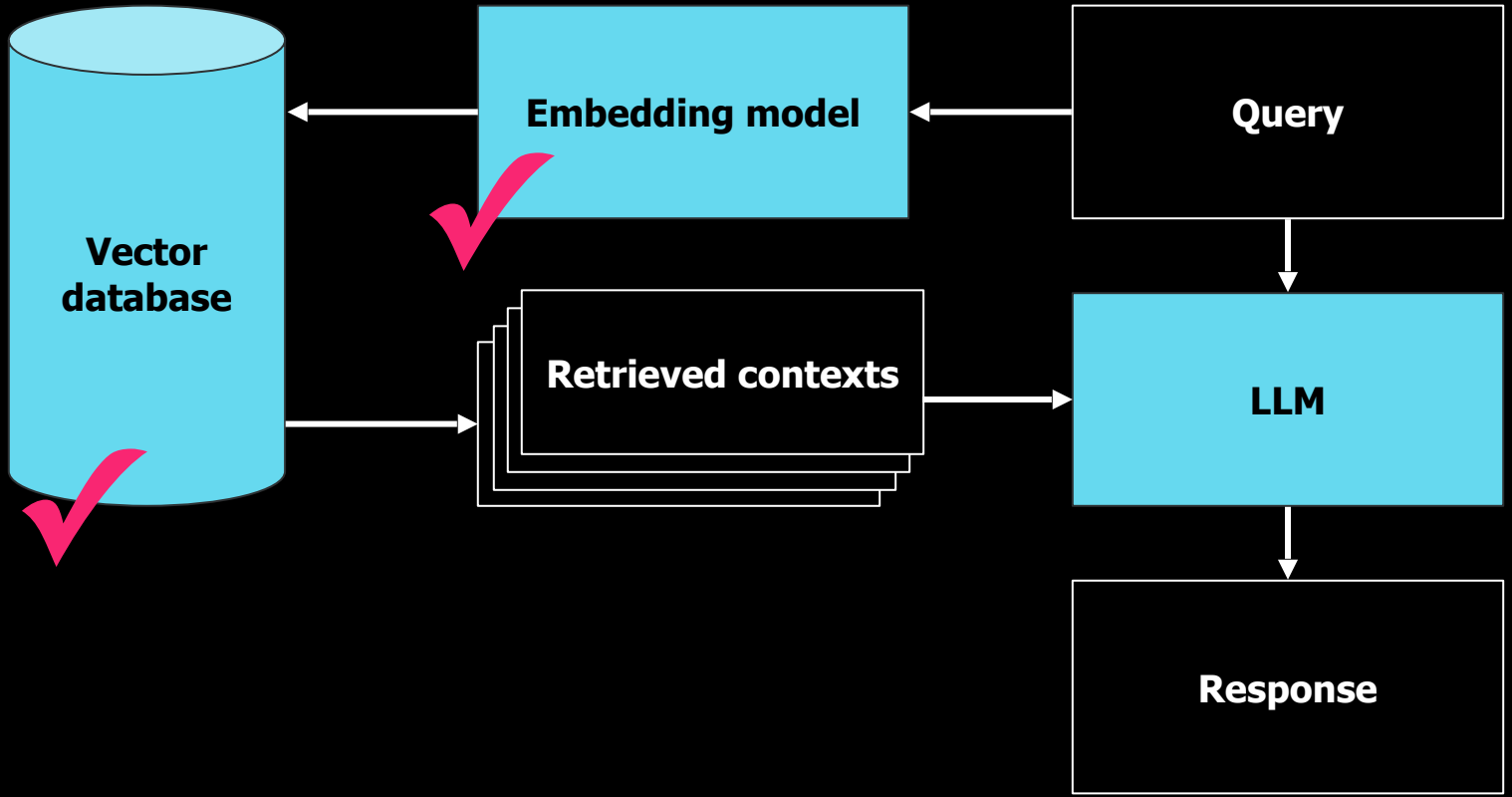
dataset_5_0 0% Outputted / Estimated total rows: 0 / - Active / Allocated resources: 2 CPU, 1 GPU, 286.21 MB object store Started at 5 Dec 2025, 14:46:03

Tree view | DAG view | Logs

Job ID	Entrypoint	Runtime environment
02000000	-	-

Operator	Status	Outputted / Estimated total rows	Outputted / Total blocks	Queued blocks	Active tasks / actors	Throughput (current / avg)	Active / Allocated resourc
MapBatches(ChromaWriter)	Running	0 / -	0 / -	0	0 / 0	0 / 0 row/s	1 / 18.5 CPU, 0 Bytes / 9.6
MapBatches(Embedder)	Running	0 / -	0 / -	8	4 / 0	0 / 0 row/s	0 / 29.5 CPU, 1 / 1 GPU, 2
FlatMap(<lambda>)→FlatMap(Chunker)	Running	12.66K / 13.57K	14 / 15	0	1 / 4	73.32 / 40.98 row/s	1 / 15.56 CPU, 28.32 MB /
Map(parse_epub_content)	Running	15 / 15	15 / 15	0	0 / 0	0.07 / 0.05 row/s	0 / 10.25 CPU, 1.89 MB / 5
ReadBinary→SplitBlocks(2)	Finished	15 / 15	15 / 15	0	-	- / 0.05 row/s	-
Input	Finished	15 / -	15 / 15	0	-	- / 0.56 row/s	-

Rows per page: 20 1 - 1 of 1


Vector database
Embedding model
Query
Retrieved contexts
LLM
Response

Ray Serve

Ray Serve

Run live inference, provide HTTPS endpoint

Scale cluster up/down to match number of requests

Particularly suited for model composition and multi-model serving

Core concepts: deployments and replicas

- Deployments contain replicas
- Replicas map to a Ray Actor

```python
from ray import serve

@serve.deployment(ray_actor_options={"num_gpus": 0.1}, autoscaling_config={"min_replicas": 1})
class QueryEncoder:
    ...

@serve.deployment(ray_actor_options={"num_cpus": 1})
class VectorStore:
    ...

@serve.deployment(ray_actor_options={"num_cpus": 0.1})
class Retriever:
    ...

@serve.deployment(ray_actor_options={"num_cpus": 0.1})
class LLMClient:
    ...
```

```python
@serve.deployment(autoscaling_config=dict(min_replicas=1, max_replicas=3),)
class QA:
    """
    Main QA engine combining retrieval and generation.

    This deployment orchestrates the complete RAG pipeline:
    1. Retrieve relevant context using Retriever
    2. Augment the query with context
    3. Generate answer using LLMClient
    4. Format response with sources
    """
    ...

app = FastAPI(
    title="RAG QA Service",
    description="Retrieval-Augmented Generation Question Answering Service",
    version="1.0.0"
)

@serve.deployment(autoscaling_config=dict(min_replicas=1, max_replicas=3))
@serve.ingress(app)
class QAGateway:
    ...
```

“Who is Tylin Quintara Mitsobar?”

```
(base) ray@ip-10-0-167-218:~/default$ uv run main.py query "Who is Tylin?"
Query: Who is Tylin?
Service: http://localhost:8000/rag

2025-12-05 15:58:16,620 - urllib3.connectionpool - DEBUG - Starting new HTTP connection (1):
localhost:8000
2025-12-05 15:58:17,328 - urllib3.connectionpool - DEBUG - http://localhost:8000 "GET /rag/answer?
query=Who+is+Tylin%3F&top_k=3&include_sources=True HTTP/1.1" 200 198
Answer:
--------------------------------------------------------------------------------
 I don't know.

Sources:
rag-wheel-of-time/library/07 - A Crown of Swords.epub (page 45, book: A Crown of Swords)
rag-wheel-of-time/library/07 - A Crown of Swords.epub (page 45, book: unknown)
```

```
(base) ray@ip-10-0-167-218:~/default$ uv run main.py query "Who is Rand al Thor?"
Query: Who is Rand al Thor?
Service: http://localhost:8000/rag

2025-12-05 16:04:55,402 - urllib3.connectionpool - DEBUG - Starting new HTTP connection (1):
localhost:8000
2025-12-05 16:04:58,987 - urllib3.connectionpool - DEBUG - http://localhost:8000 "GET /rag/answer?
query=Who+is+Rand+al+Thor%3F&top_k=3&include_sources=True HTTP/1.1" 200 321
Answer:
--------------------------------------------------------------------------------
 Rand al Thor is a character in the book series "The Wheel of Time" by Robert Jordan. He is the son of
Odin and the heir to the throne of Asgard. He is also known as the Dragon Reborn and is a powerful
warrior and leader.

Sources:
rag-wheel-of-time/library/02 - The Great Hunt.epub (page 20, book: The Great Hunt)
--------------------------------------------------------------------------------
```

Whoops…

Fiddling around

- Increase chunk size
- Increase overlap
- Increase top_k

Max context at 32k characters (~15 pages)

How about now?

```
(base) ray@ip-10-0-47-53:~/default$ uv run main.py query "Who is Tylin Quintara Mitsobar?"

Query: Who is Tylin Quintara Mitsobar?
Service: https://rag-qa-service-wwcnm.cld-rcnjlw42qd461rmr.s.anyscaleuserdata.com/rag

Querying service (first query may take 2-3 minutes while downloading vector store from S3)...
2025-12-08 15:40:37,550 - urllib3.connectionpool - DEBUG - Starting new HTTPS connection (1): rag-qa-service-wwcnm.cld-rcnjlw42qd461rmr.s.anyscaleuserdata.com:443
2025-12-08 15:40:42,813 - urllib3.connectionpool - DEBUG - https://rag-qa-service-wwcnm.cld-rcnjlw42qd461rmr.s.anyscaleuserdata.com:443 "GET /rag/answer?query=Who+is+Tylin+Quintara+Mitsobar%3F&top_k=5&include_sources=True HTTP/1.1" 200 691
Answer:
--------------------------------------------------------------------------------
 Tylin Quintara Mitsobar is the Queen of Ebou Dar and the mother of Beslan. She is described as a beautiful woman with long black hair and sharp eyes. She is also described as being possessive of Mat Cauthon, and she wants to marry him. She is a powerful figure in the story, and she has a significant impact on the events that unfold.

Sources:
rag-wheel-of-time/library/09 - Winter's Heart.epub (page 24, book: Winter's Heart)
--------------------------------------------------------------------------------
```

Better…

Crossing the t’s

- Use newer Mistral model that allows for more chunks
- Add re-ranker to select best chunks instead of closest

Max context at 128k characters (~60 pages)

```
(default) robdewit@Robs-MacBook-Pro-2 ray-wheel-of-time % uv run main.py query "Who is Tylin
Quintara Mitsobar?"
Query: Who is Tylin Quintara Mitsobar?
Service: https://rag-qa-service-wwcnm.cld-rcnjlw42qd461rmr.s.anyscaleuserdata.com/rag

Answer:
--------------------------------------------------------------------------------
 Tylin Quintara Mitsobar is the Queen of Altara. She is a widow and has four sons and one daughter.
She is known for her imposing presence and is considered to have little real power, as a man could
ride beyond her writ in two or three days and still have a lot of Altara ahead. She is also a member
of House Mitsobar, which has ruled Altara for five generations. She is initially hesitant to allow
Mat Cauthon into her presence, but later becomes more accommodating towards him. She is also willing
to make compromises with the Seanchan Empire, and eventually swears allegiance to the Daughter of
the Nine Moons and becomes a member of the High Blood.

Sources:
rag-wheel-of-time/library/07 - A Crown of Swords.epub (page 29, book: A Crown of Swords)
rag-wheel-of-time/library/07 - A Crown of Swords.epub (page 24, book: A Crown of Swords)
rag-wheel-of-time/library/09 - Winter's Heart.epub (page 23, book: Winter's Heart)
rag-wheel-of-time/library/07 - A Crown of Swords.epub (page 45, book: A Crown of Swords)
rag-wheel-of-time/library/09 - Winter's Heart.epub (page 24, book: Winter's Heart)
rag-wheel-of-time/library/06 - Lord of Chaos.epub (page 56, book: Lord of Chaos)
rag-wheel-of-time/library/09 - Winter's Heart.epub (page 26, book: Winter's Heart)
rag-wheel-of-time/library/09 - Winter's Heart.epub (page 25, book: Winter's Heart)
rag-wheel-of-time/library/09 - Winter's Heart.epub (page 36, book: Winter's Heart)
rag-wheel-of-time/library/07 - A Crown of Swords.epub (page 37, book: A Crown of Swords)
rag-wheel-of-time/library/10 - Crossroads of Twilight.epub (page 12, book: Crossroads of Twilight)
rag-wheel-of-time/library/12 - The Gathering Storm.epub (page 31, book: The Gathering Storm)
rag-wheel-of-time/library/07 - A Crown of Swords.epub (page 25, book: A Crown of Swords)
--------------------------------------------------------------------------------
```

Conclusions

Takeaways

Ray: orchestrate AI workloads across distributed, heterogeneous compute

Basics: tasks (stateless) and actors (stateful)

Ray Data & Serve: dedicated libraries for data processing and model serving

Wheel of Time: masterpiece, but has pacing issues and too many characters

Thank you!

robdewit.nl

Useful links

- https://docs.ray.io
- https://docs.anyscale.com
- https://www.anyscale.com/examples
- https://courses.anyscale.com/
- https://github.com/RCdeWit/ray-wheel-of-time