

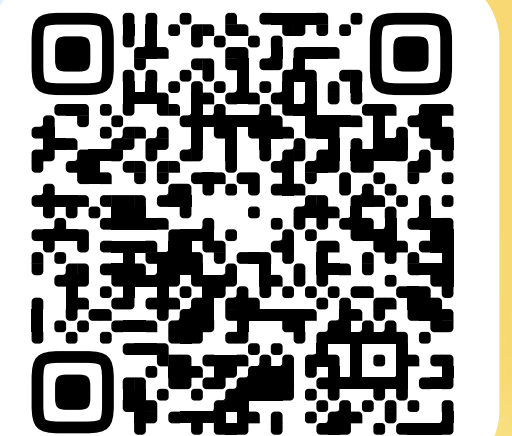


# Dealing with Big Data and moving towards AI

处理大数据，迈向人工智能

**Alexander Zevaykin, PhD**

**Group Leader at Yandex Infrastructure**



[ydb.tech/zh](https://ydb.tech/zh)

# Yandex consists of over 90 services, used by millions of people daily

## Yandex由90多个服务组成，每天有数百万人使用



### Search

Leading search engine in Russia



### Browser

Most popular non-native browser in Russia



### Direct

Leading ad service for placing contextual ads in Russia



### Maps and Navigation

Leading map and navigation app in Russia and CIS



### Market

A multi-category marketplace



### Lavka, Deli

Hyperlocal e-grocery delivery service



### Eats & Market Delivery<sup>2</sup>

Delivery of ready-to-eat food from restaurants and various goods from offline stores



### Delivery

Delivery solution for consumers and businesses

### MOBILITY



### Taxi

Leading ride-hailing operator in Russia with presence in other CIS and EMEA markets<sup>1</sup>



### Drive

Car-sharing service



### Scooter

Electric scooter rental service

### PLUS, ENTERTAINMENT SERVICES AND FINTECH



### Kinopoisk

Leading video streaming platform in Russia



### Music, Bookmate

Leading music streaming service in Russia; E-book and audiobook service



### Plus

Leading subscription program in Russia combining all key Yandex services via cashbacks and benefits for users



### Fintech

Retail lending focused digital financial services

### CLASSIFIEDS



### Auto.ru, Yandex Realty, Yandex Rent and Yandex Travel

Leading online classifieds in the auto, real estate and travel verticals

### OTHER



### Cloud, Yandex 360<sup>3</sup>

Full-fledged cloud platform for B2B and B2C



### Devices & Alice

A line of smart speakers and TV with an AI voice



### Practicum

The beginner-friendly online coding bootcamps with the



### SDG

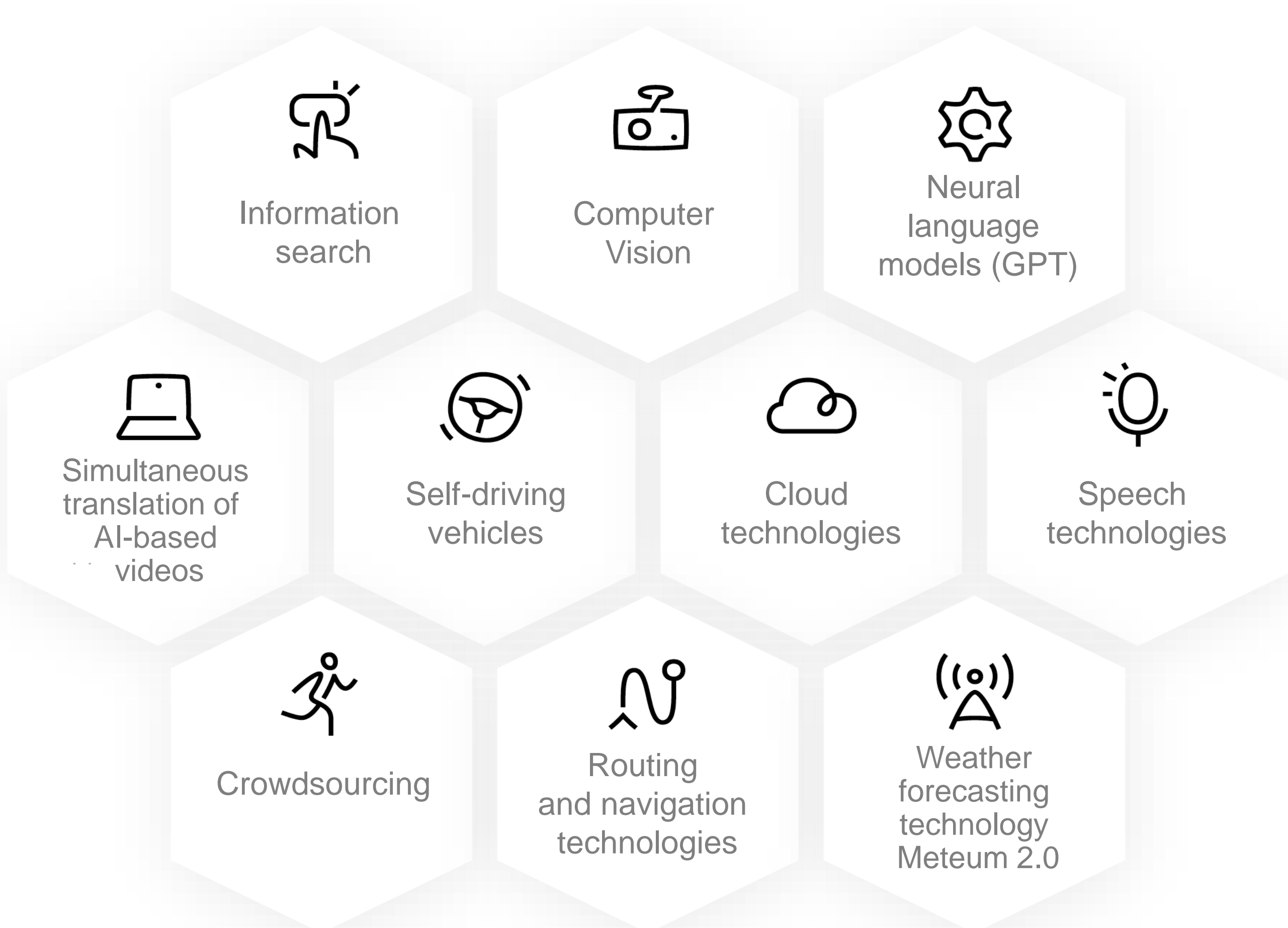
Self-driving vehicles and autonomous

# Yandex builds a lot of its infrastructure in-house

Yandex公司在内部建立了很多基础设施

**25700+**

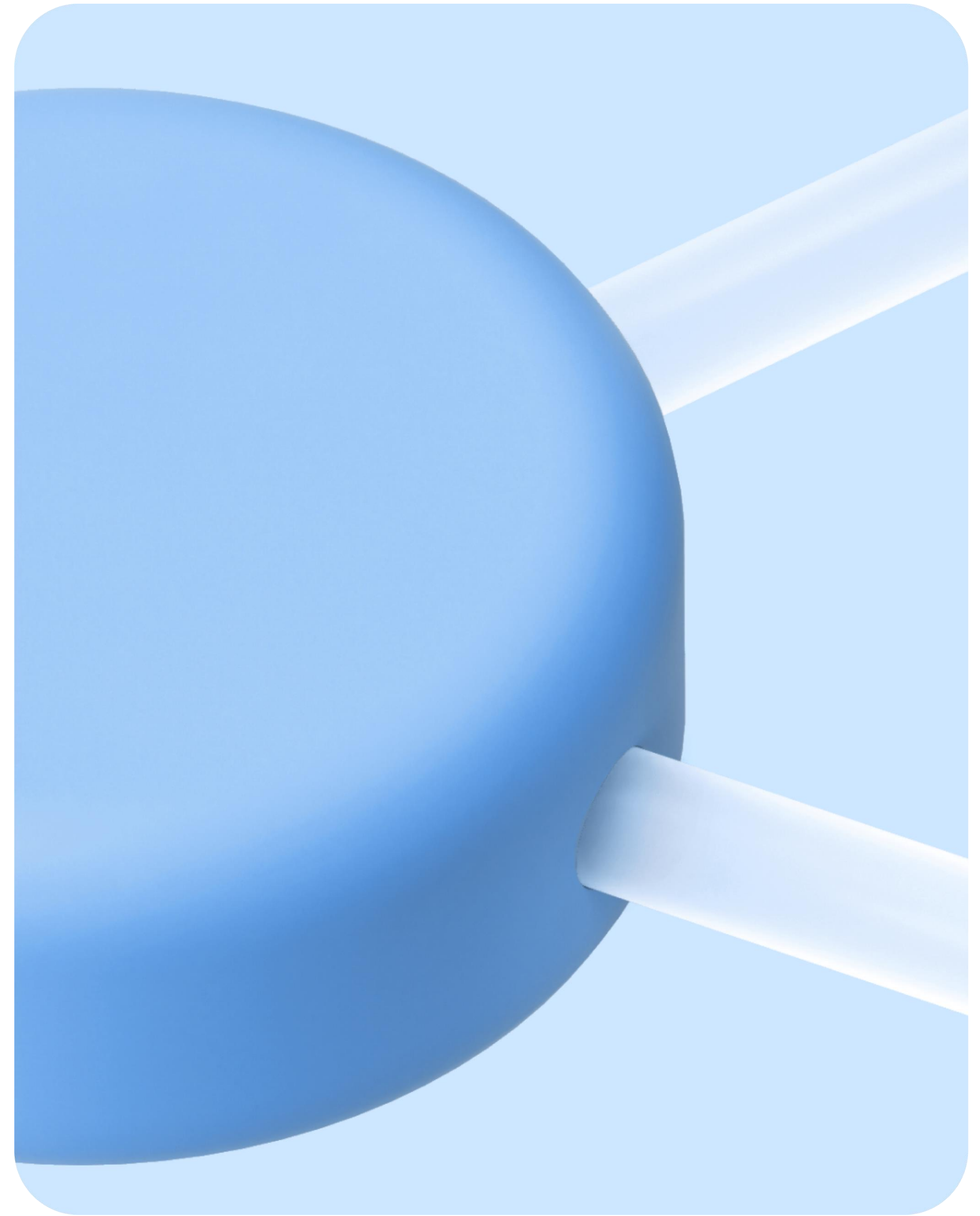
employees



# **Part1**

## **YDB: dealing with Big Data**

处理大数据



# What is YDB?

- Distributed SQL database for operational and analytical workloads
- YDB是一个开源、分布式、高容错的SQL数据库系统，能将高可用性、可扩展性与强一致性和ACID事务相结合
- 它可以同时处理事务性（OLTP）、分析性（OLAP）和流式工作负载

[ydb.tech/zh](https://ydb.tech/zh)



[github.com/ydb-platform/ydb](https://github.com/ydb-platform/ydb)

- Horizontal scaling  
横向扩展性
- ACID transactions in multiple AZ  
分布式环境保持ACID事务
- Operability and automatic recovery  
in case of failures  
故障时可操作性和自动恢复
- Scaling by millions of transactions  
per second and petabytes of data  
每秒可扩展数百万个事务和PB级数据
- Open-Source with Apache 2.0 license  
开源

# YDB: made in Yandex

**2014**

First commit

**2017**

Base for  
Yandex Cloud

**2022**

Open-Source  
github.com

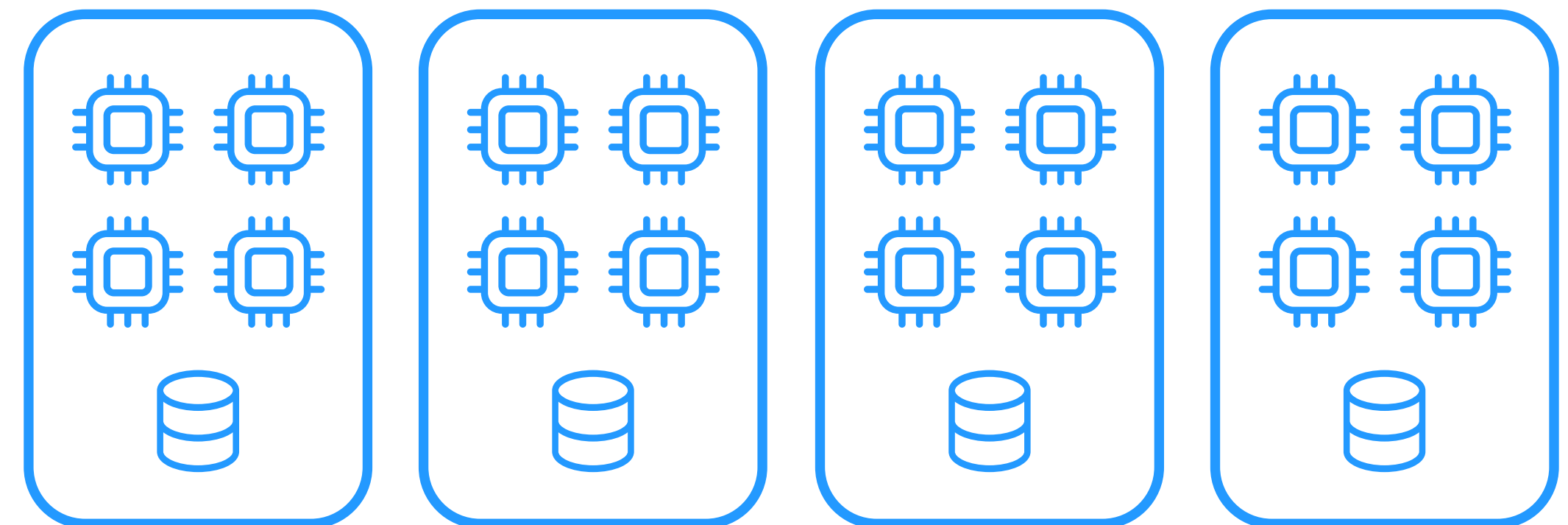
**2024**

**35000+** nodes  
**5000+** databases  
**70+** PB storage

YDB诞生于Yandex-俄罗斯最大的IT公司，  
我们已有十年发展历史。

# Shared Nothing

- 我们的基于无共享的架构
- Cluster of bare metal or virtual machines
- Shared nothing architecture
- Commodity hardware
- Cluster both stores the data and process user queries

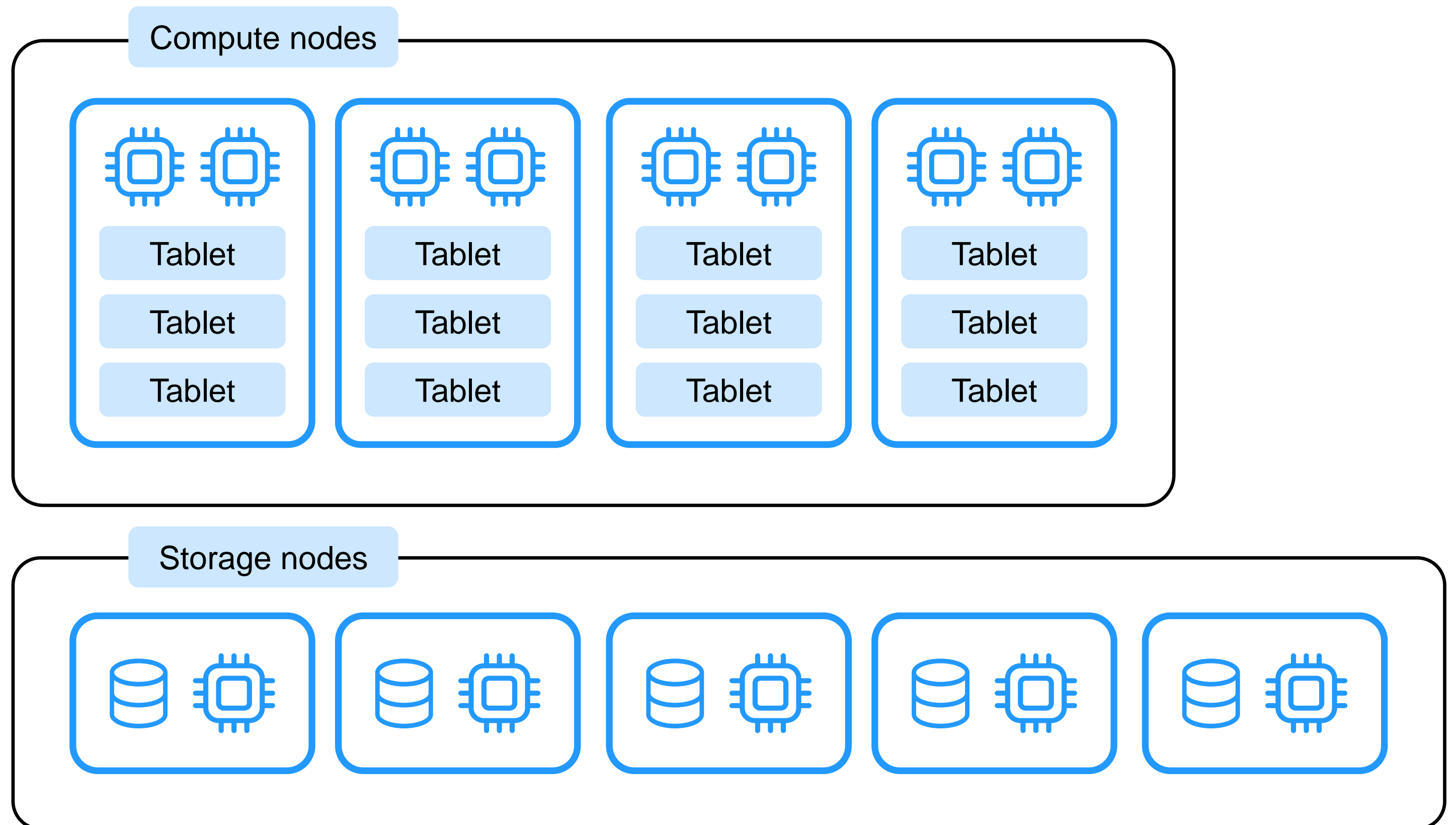


# Compute & Storage separation

## 计算和存储节点独立管理

Compute and storage nodes are managed independently

- Scalability
- Cost-efficiency
- Flexibility

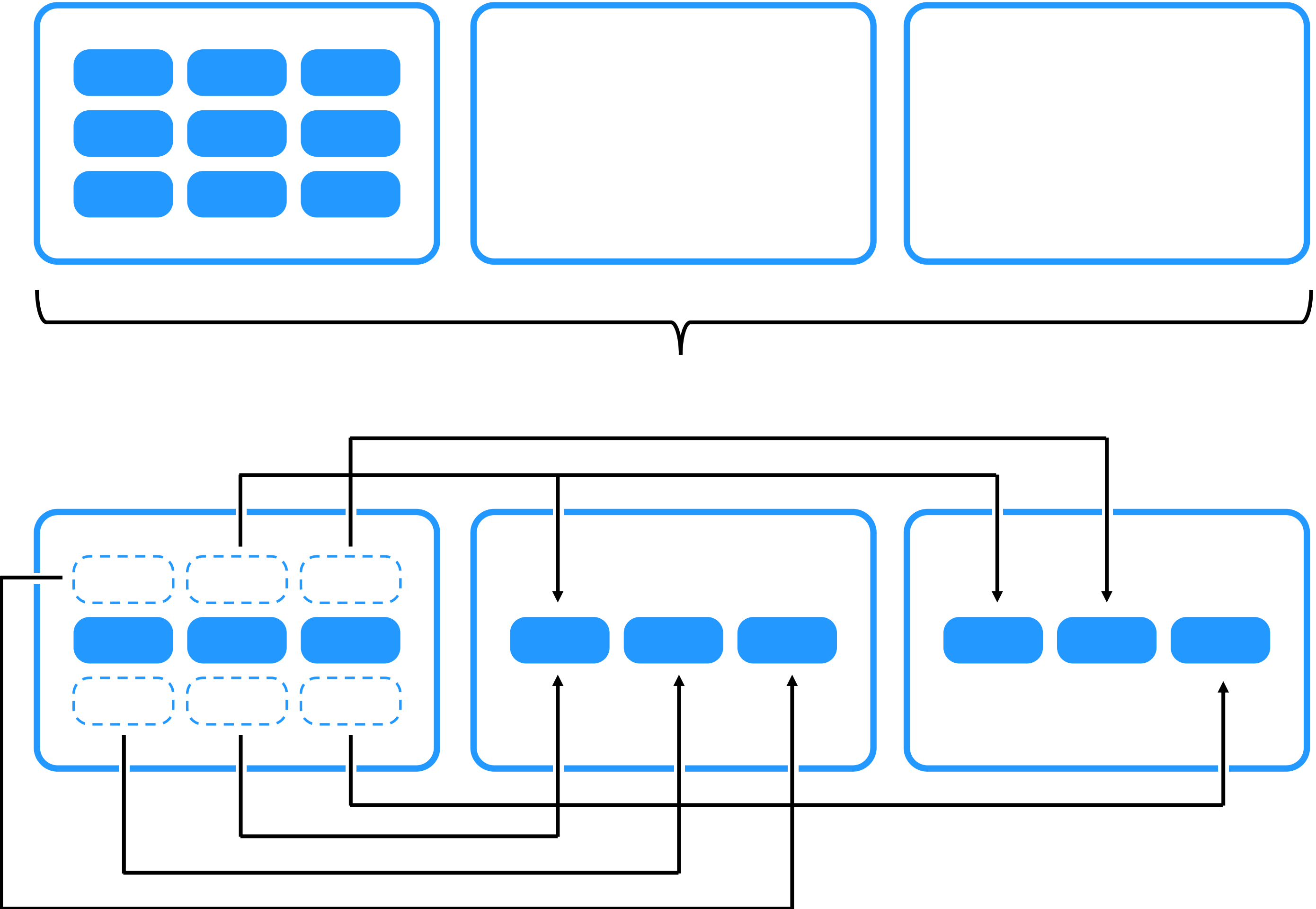




# Table Partitions Autosplit and Balancing

## 数据表自动拆分, 自动平衡

- Split by load
- Split by size
- YDB evenly distributes table partitions among the nodes



# Mirror-3-dc

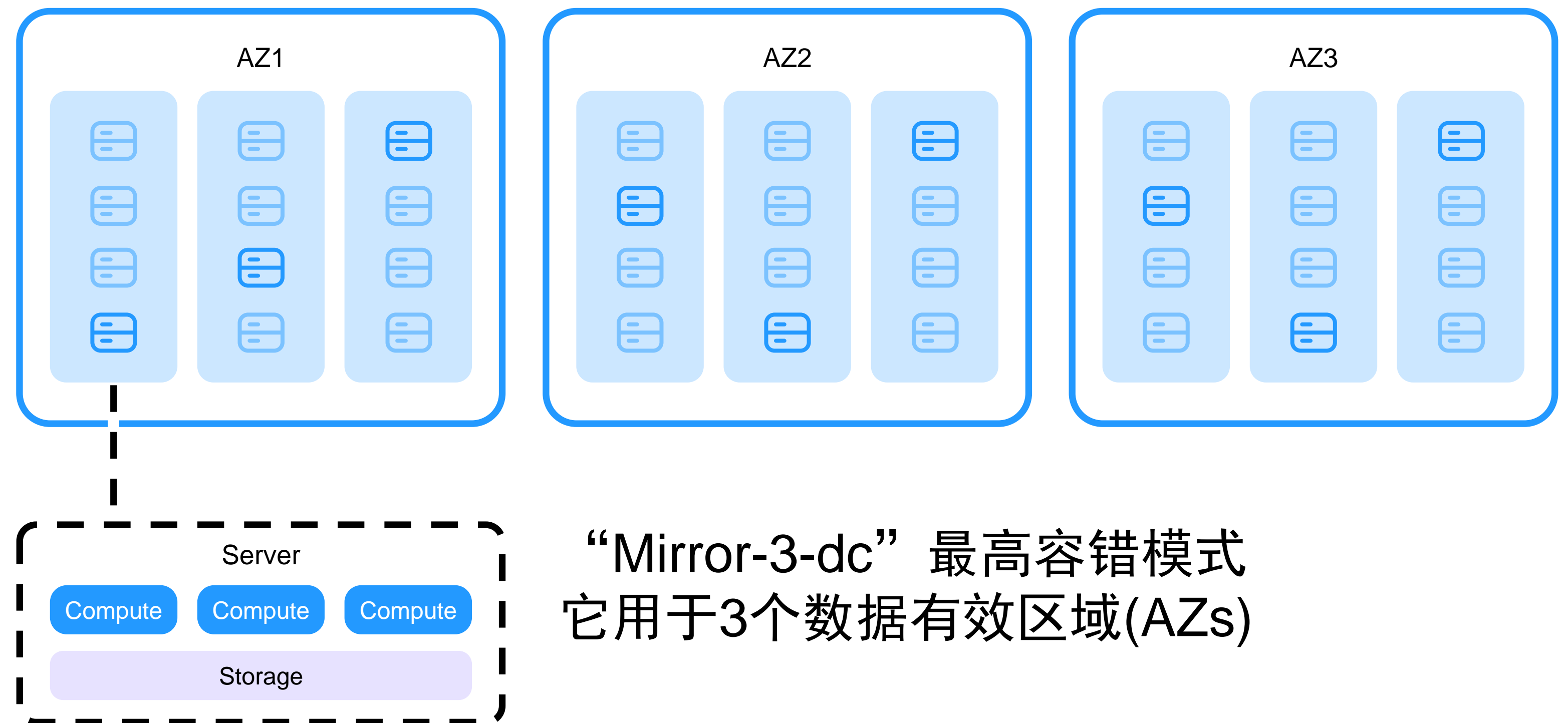
3

availability zones

×3

storage factor

cope with the loss of one AZ +  
one server rack in any other AZ



“Mirror-3-dc” 最高容错模式  
它用于3个数据有效区域(AZs)

# Block-4-2

## Erasure-encoding, Reed-Solomon

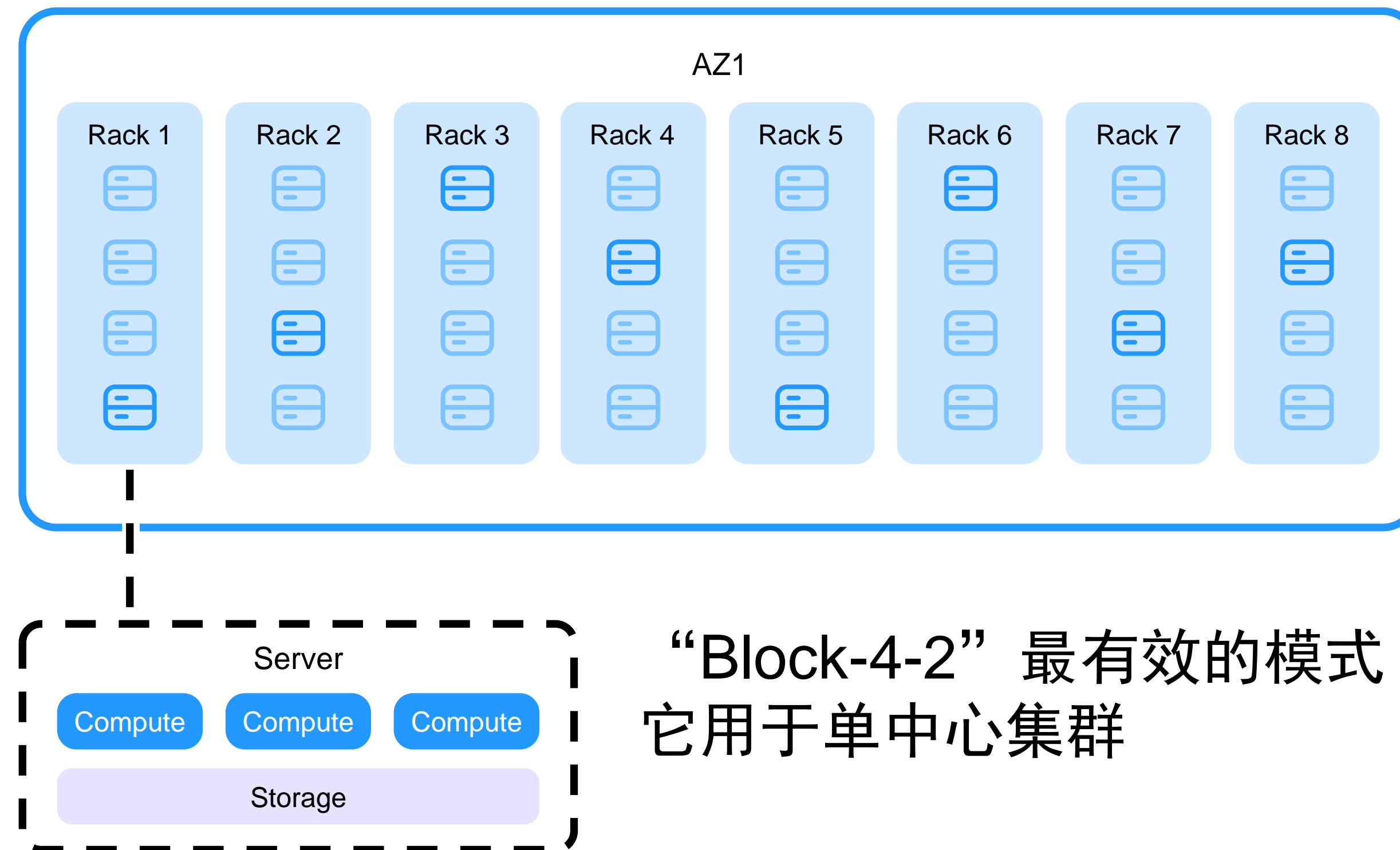
1

availability zone

×1,5

storage factor

copers with the loss  
of 1 server racks of 8



“Block-4-2” 最有效的模式  
它用于单中心集群



Shopping cart and checkout uses YDB

**100x**

Hundredfold load  
spikes

**99,99%**

Strict response time  
guarantees

**OLTP**

Workload

Yandex市场是俄罗斯最大的电子商务服务之一



**>1.5 mln**

Transactions per second

**>1 PB**

Data

**OLAP**

Workload

**Among the largest**

Web and mobile analytics platform in the world

Yandex Metrika是世界上最大的网络和移动分析平台之一



## Alice

It can recognize speech, hold a conversation, answer questions, and much more

Alice users per month

Requests to the voice Assistant per month

## Devices

Russia's widest range of devices models with a virtual assistant. We hold a 90% share of the smart speaker market

**5 countries**

Where to buy Devices with Alice

**9.6 million**

Total number of Yandex Stations sold since the end of 2018

**3.2 million**

Active devices with Yandex TV per month



**Lite**  
Basic level



**Mini with digital watch**  
Compact



**Midi**  
Compact with powerful sound



**Station 2**  
Middle segment



**Max with Zigbee**  
With 3-way speakers



**Duo Max**  
With a display



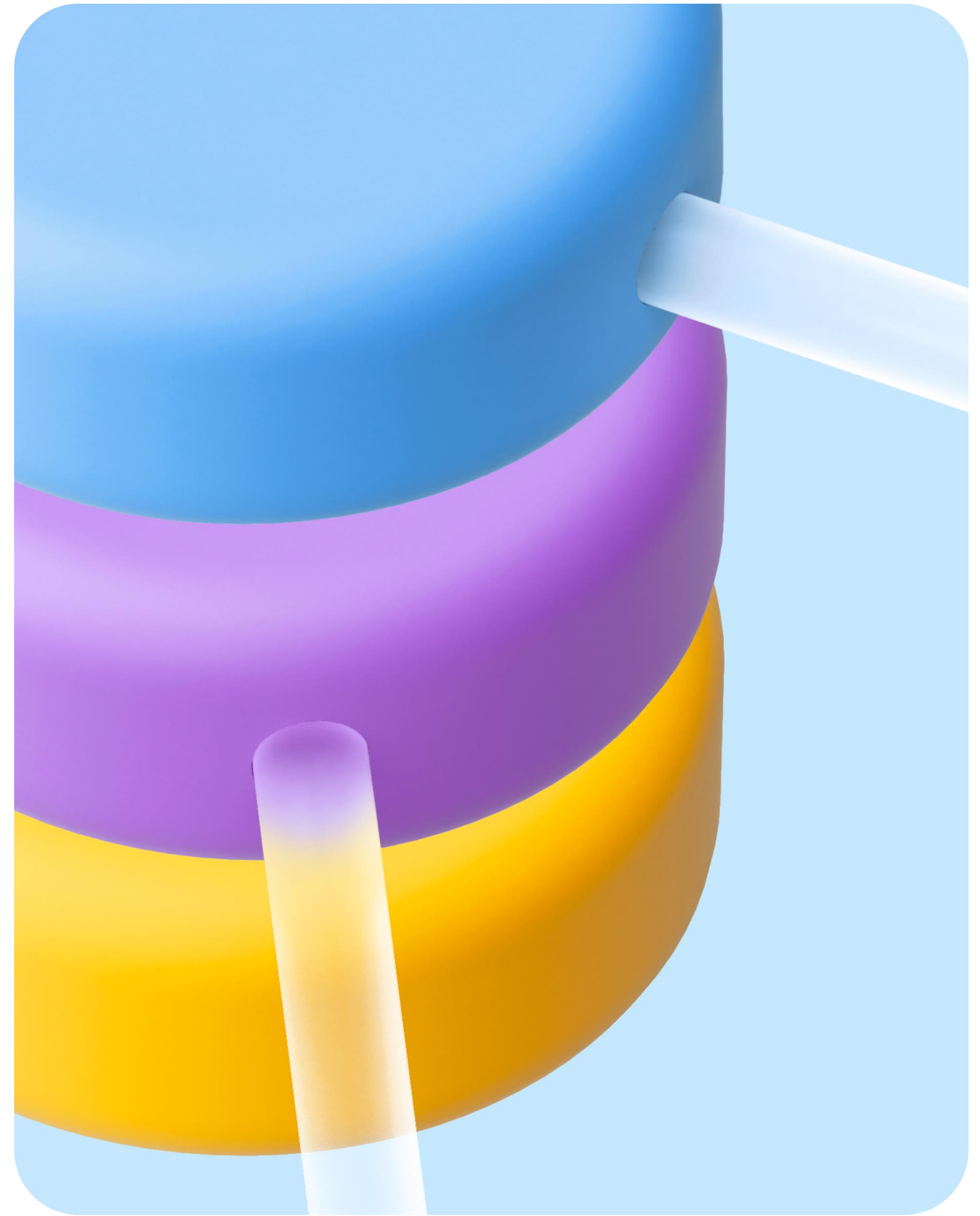
**Smart TV**  
Has all the features of a smart speaker

Alice是语音助手和智能家居生态系统

# Part 2

## YDB: moving towards AI

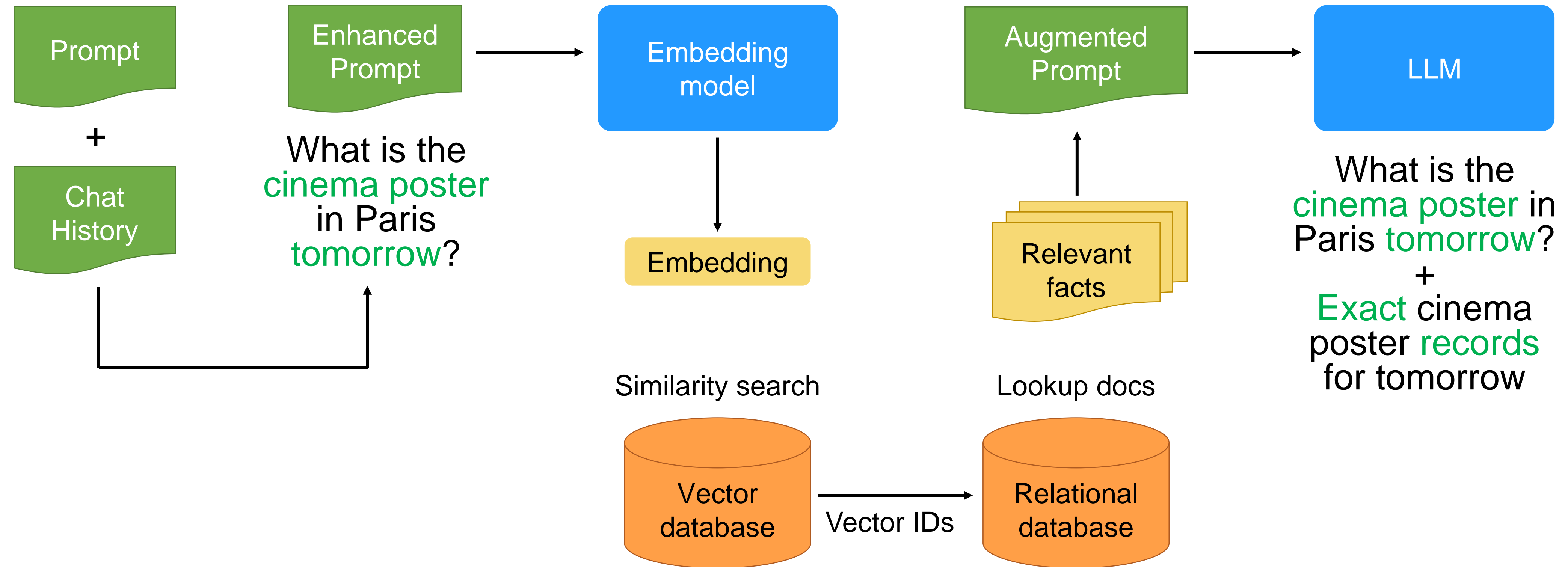
迈向AI



# Retrieval Augmented Generation

## 检索增强生成

What can I see in Paris?



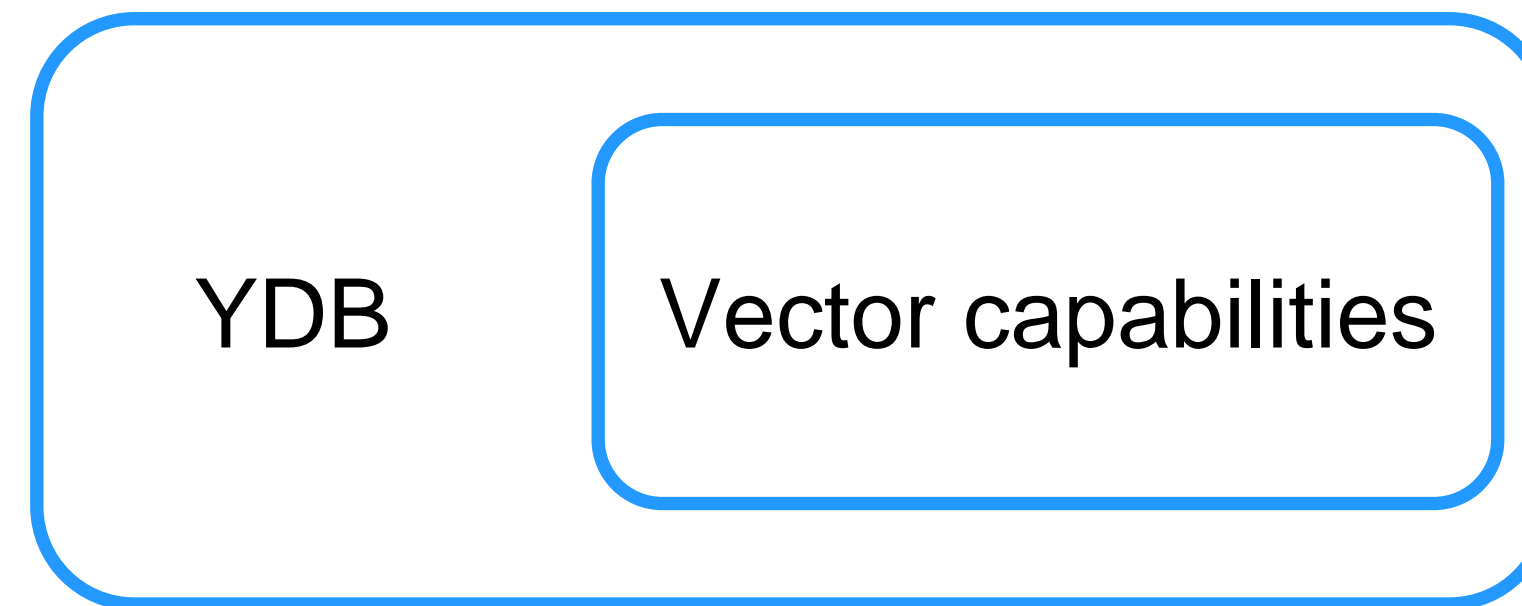
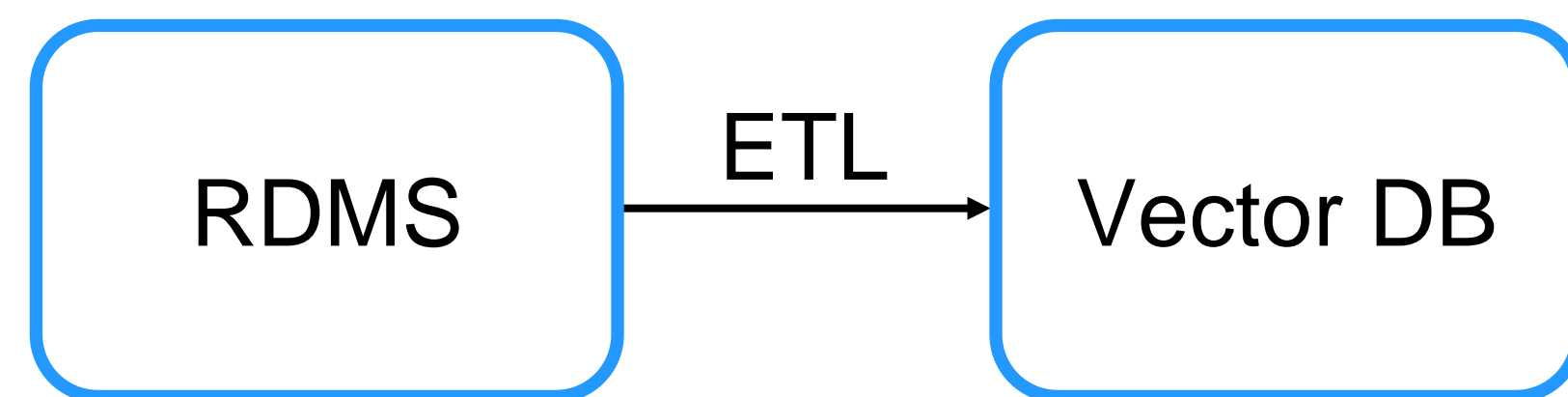


# YDB intrinsic advantages: logical layer

## YDB 固有优势：逻辑层

- Data fragmentation
- Data copy
- Data divergence

- Zero copy
- No storage overhead
- Data consistency



# YDB intrinsic advantages: infrastructure layer

YDB固有优势:基础设施层

## Scale

Sharding

Replication

Multitenancy

Spiky workload

Cross-datacenter

## Production readiness

Fault-tolerance

Rolling update

Persistence

Consistency

Alerting

Support

Monitoring

Herculean tasks

# Vector search in modern databases

## 现代数据库的向量搜索

Database	Release of vector search
PostgreSQL	2021
Lucene	2021
OpenSearch	2022
Redis	2022
Cassandra	2023
Clickhouse	2023
MongoDB	2023
Oracle	2023
MariaDB	2023

# YDB as a Platform

Distributed storage

ACID transactions

OLAP-tables

OLTP-tables

Unified query language

Federated queries

Topics

Key-Value

Vector search

# Vector search use cases

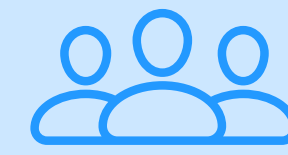
## 向量搜索使用场景



### **Exact** search of user facts

- User has tiny amount of facts
  - He has 3 children
  - Dog name is Bella

Vector search can afford scan all the user facts



### **Approximate** search of all facts

- World cinema poster
- Private organization has billions of its confidential facts

Vector search can't use brute force scan

# Methods of vector search

## 向量搜索的方法

### Exact methods

#### 精确搜索

- Linear search (brute force)
- Space partitioning

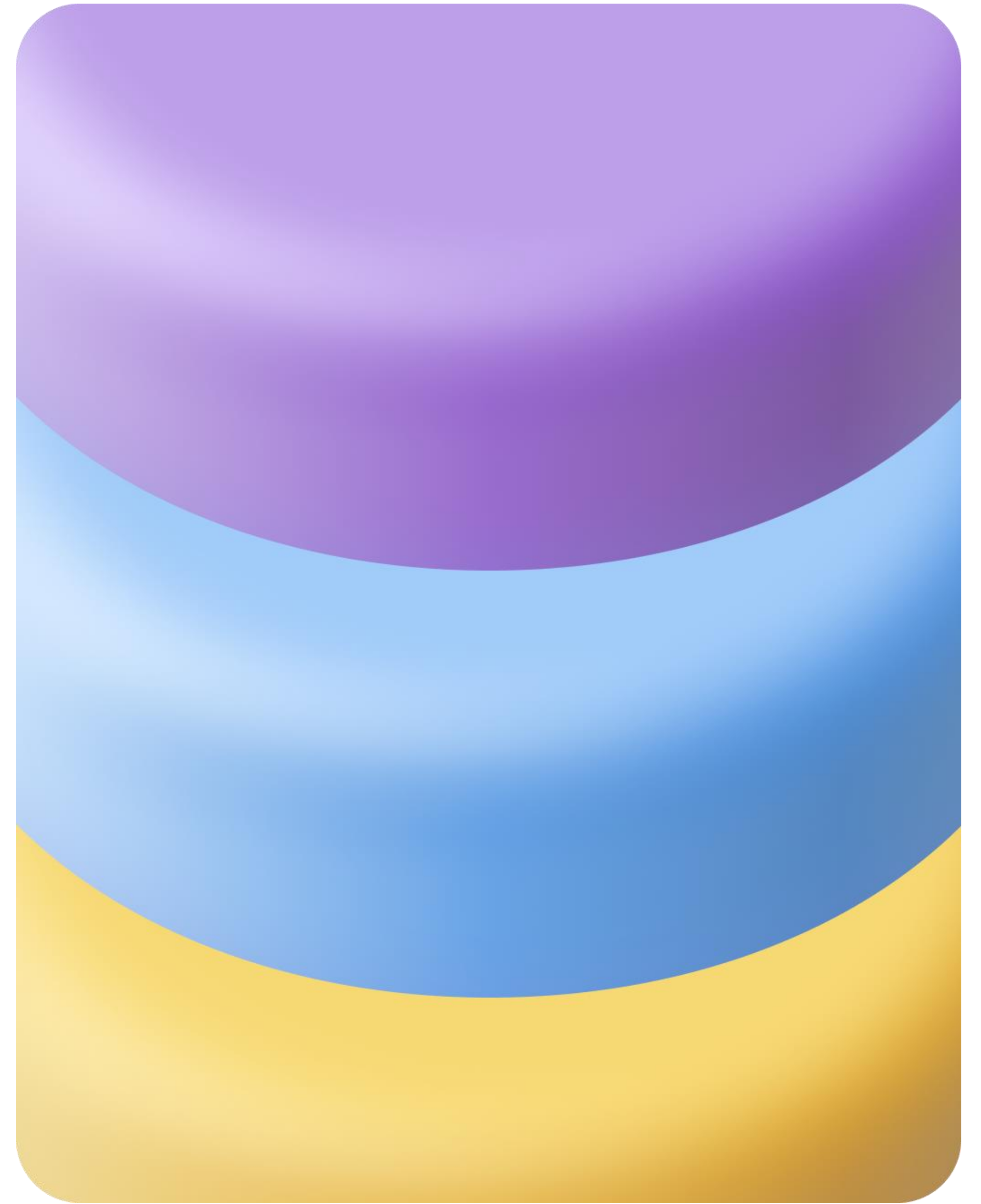
### Approximation methods

#### 近似搜索

- Random Projections
- Locality sensitive hashing
- Faiss
- Hierarchical Navigable Small World

# YDB: Exact method of vector search

精确向量搜索的使用方法



# Create table

```
CREATE TABLE facts (  
    id UInt64,  
    text String,  
    user_id UInt64,  
    vector Bytes,  
    PRIMARY KEY (id)  
)
```



# Brute force

```
SELECT id, text FROM facts
WHERE user_id = 1
ORDER BY CosineDistance(vector, $TargetVector)
LIMIT 10
```

# **YDB: approximation methods of vector search**

近似向量搜索方法

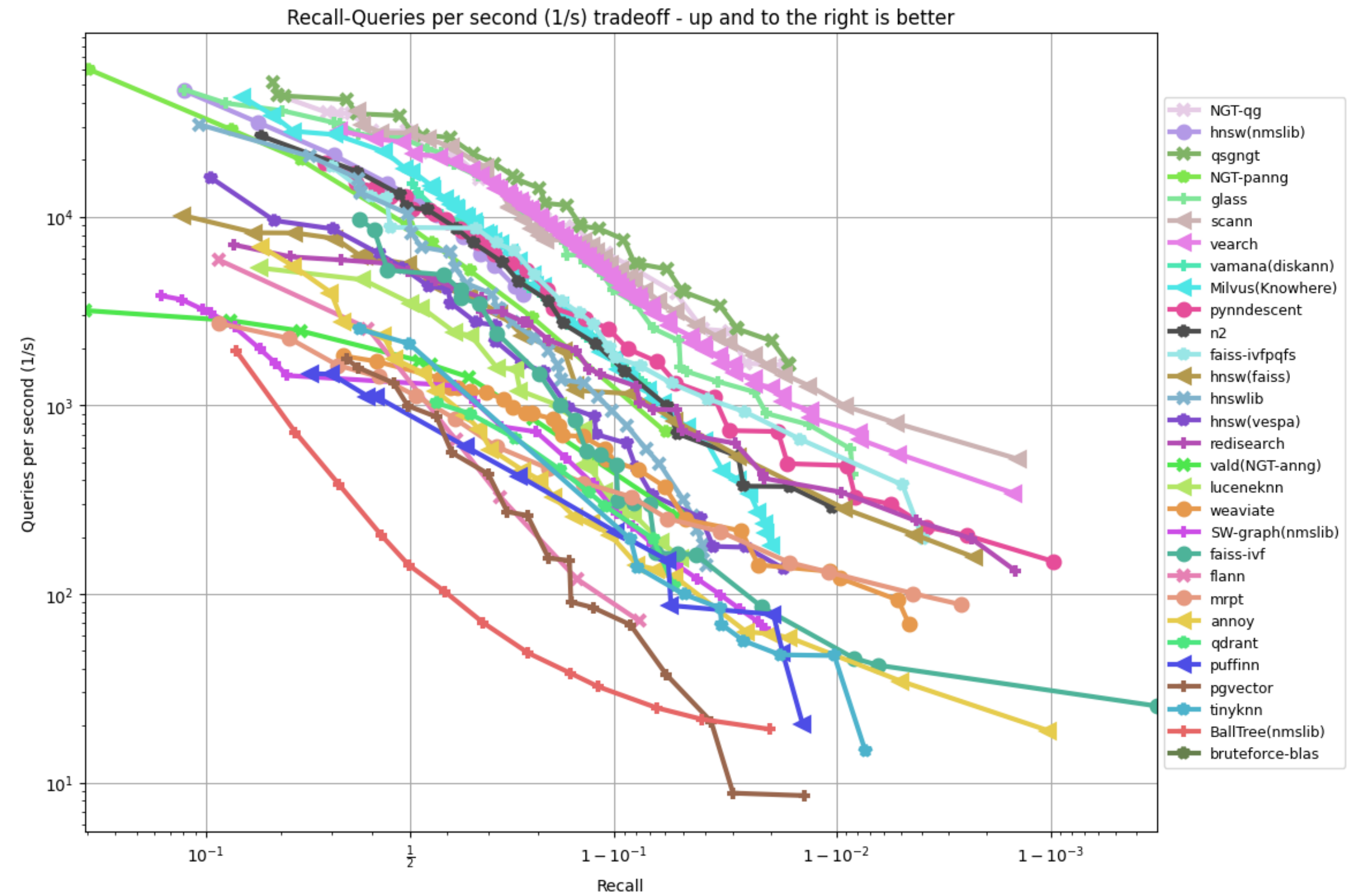


# Benchmarks

10+  
methods

30+  
libraries

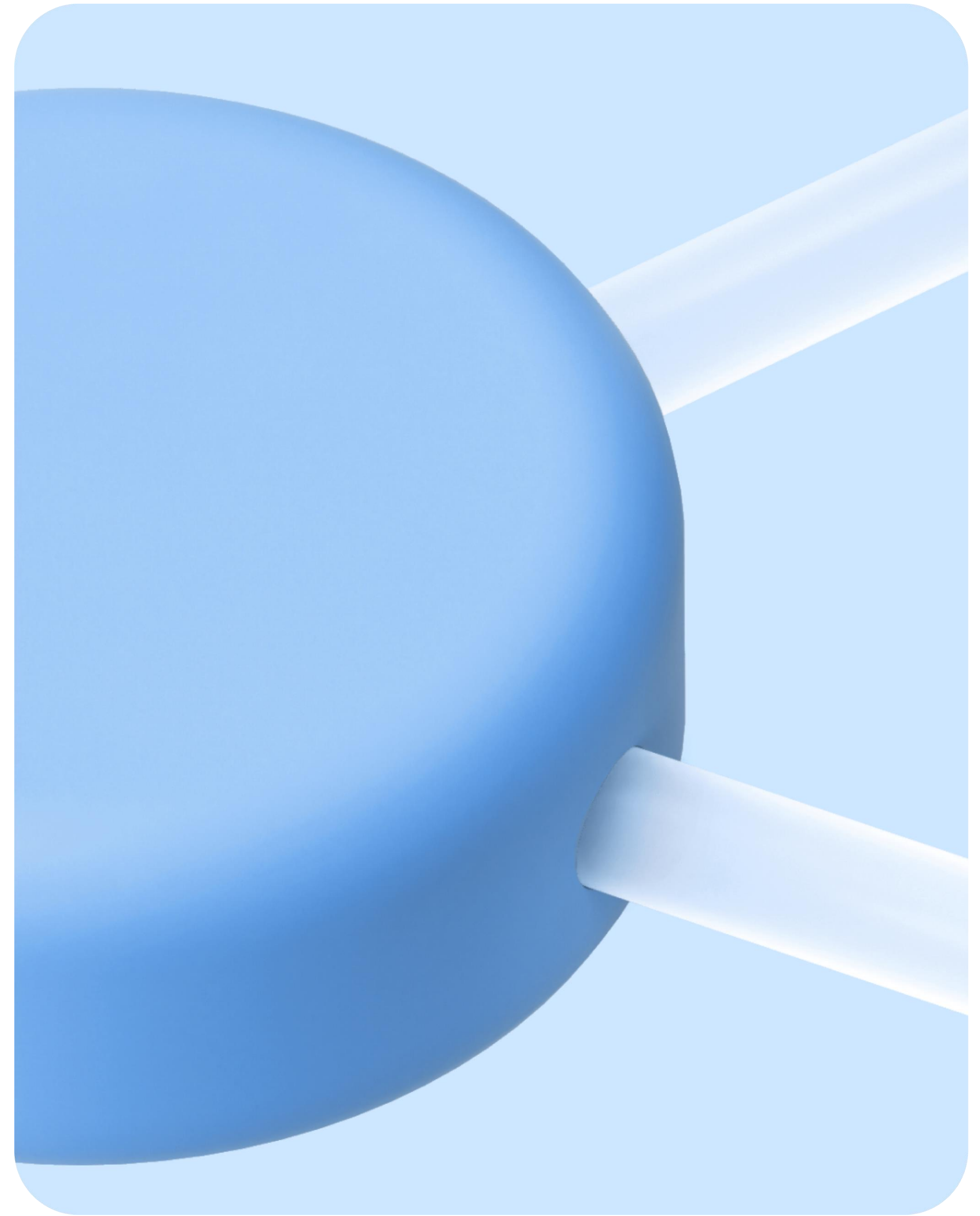
10+  
data sets



# 3 popular methods

	<b>Annoy</b>	<b>Faiss</b>	<b>HNSW</b>
Type	Random projections	Inverted index	Graph
Used by	Clickhouse	<ul style="list-style-type: none"><li>• PostgreSQL</li><li>• Oracle</li></ul>	<ul style="list-style-type: none"><li>• Clickhouse</li><li>• PostgreSQL</li><li>• Oracle</li><li>• MongoDB Atlas</li><li>• Redis Stack</li><li>• Lucene<ul style="list-style-type: none"><li>• Elasticsearch</li><li>• OpenSearch</li><li>• Cassandra</li></ul></li></ul>

**YDB:  
Random  
Projections  
method  
随机投影方法搜索**

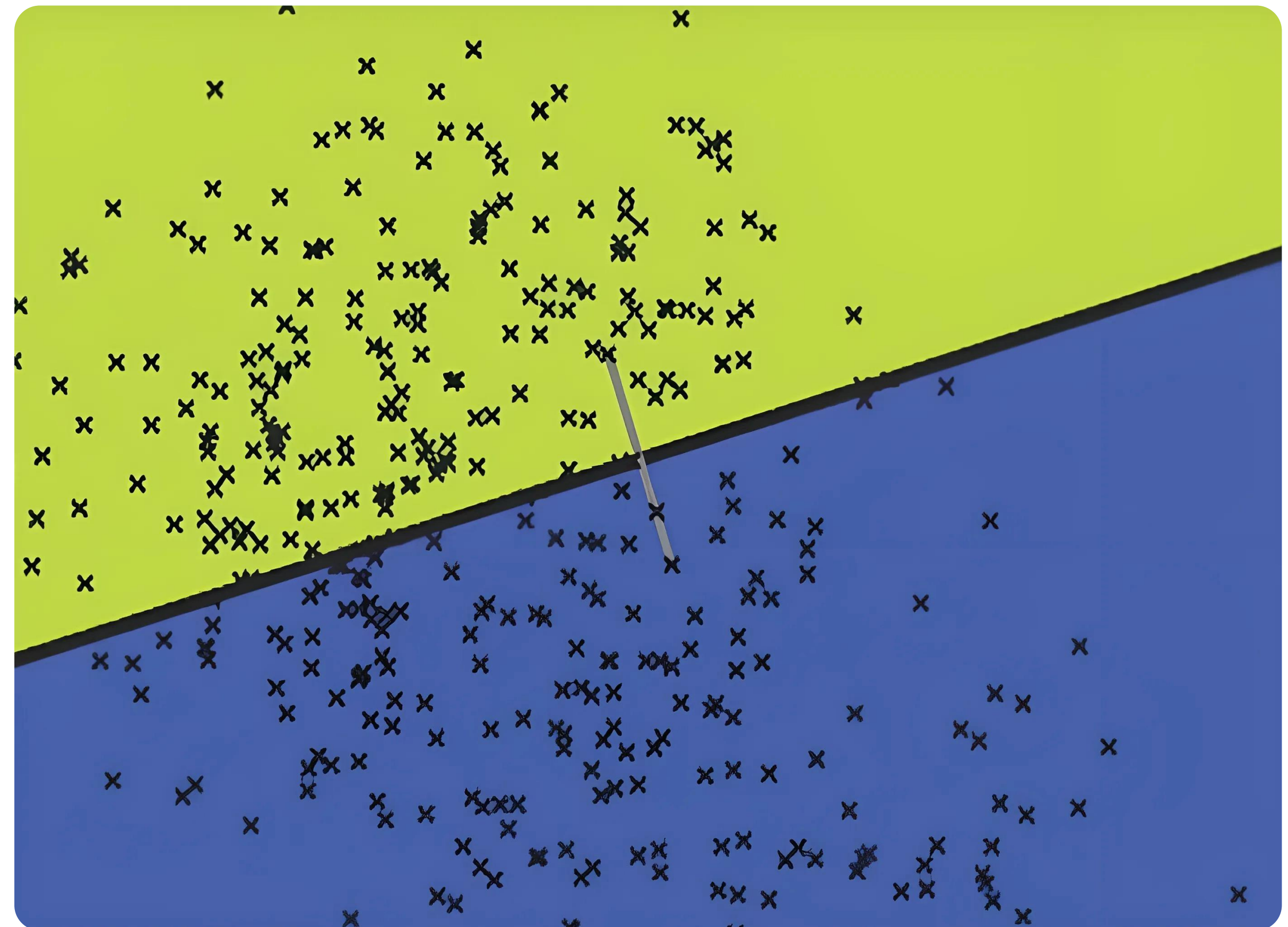


# Random Projections

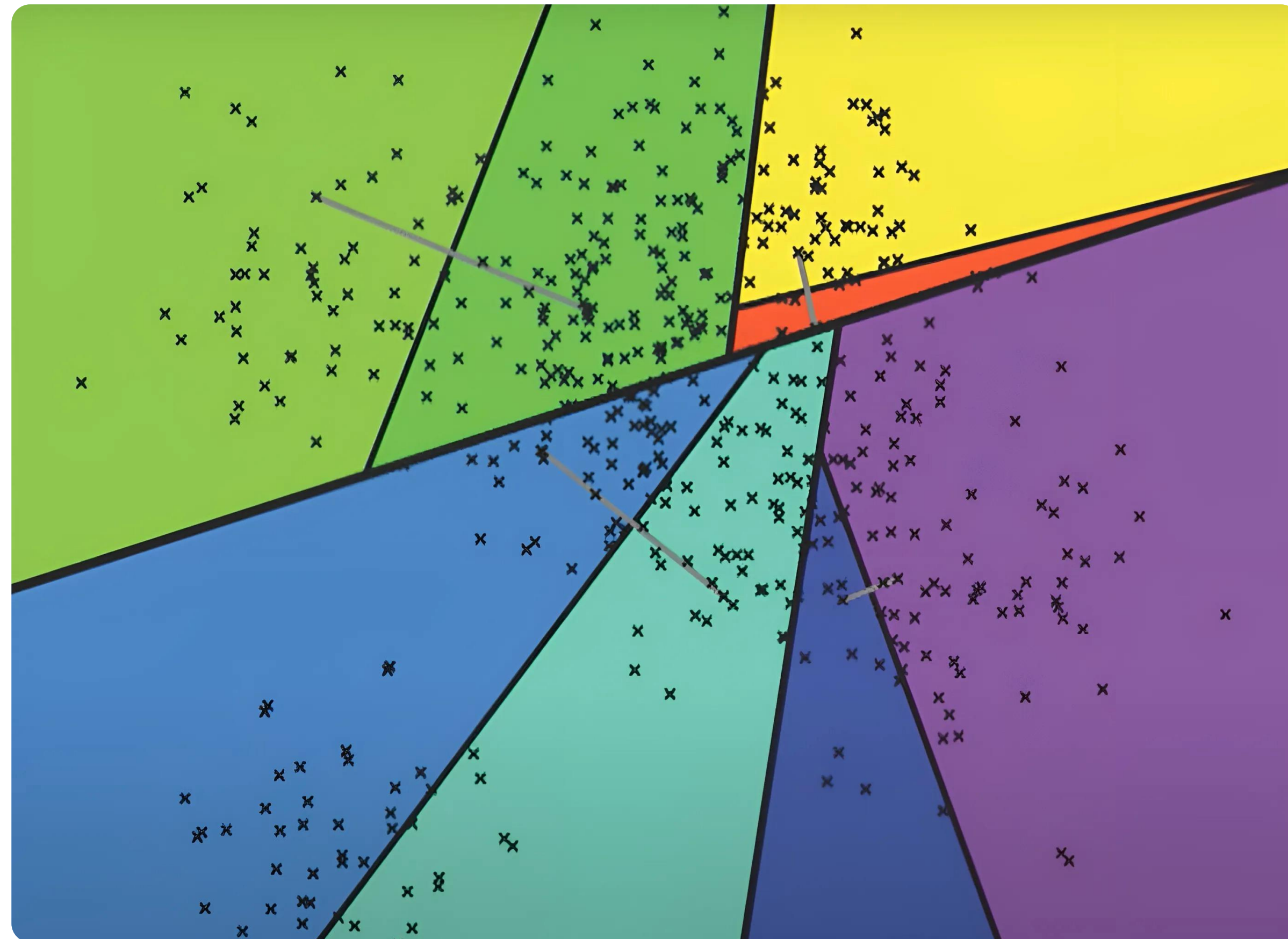
**Partition the space by  $N$  random hyperplanes**

**How to choose hyperplanes?**

- Take a pair of random points from the dataset
- Take a hyperplane separating these two points, orthogonal to the vector between them



# Split again recursively

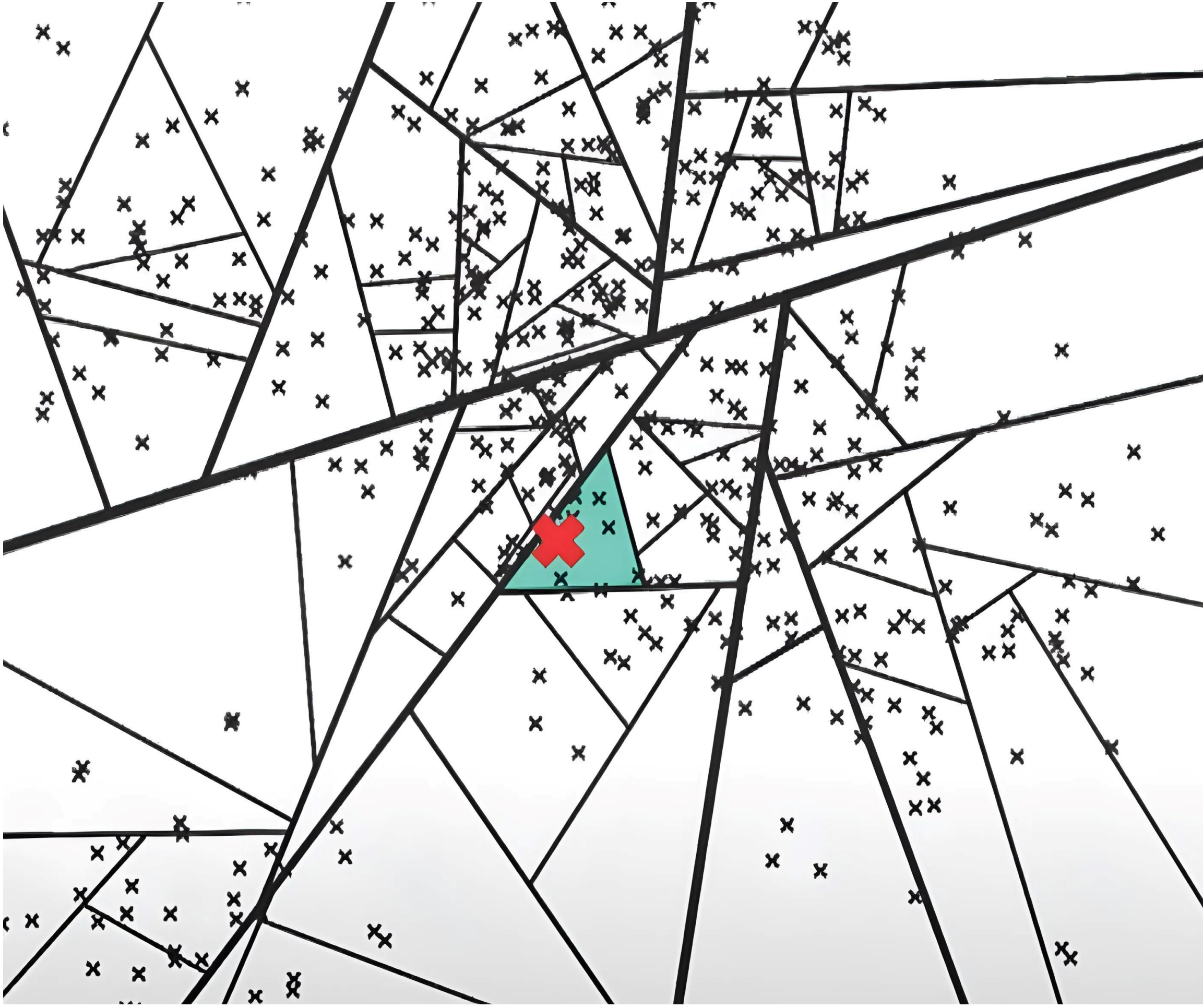


# And again



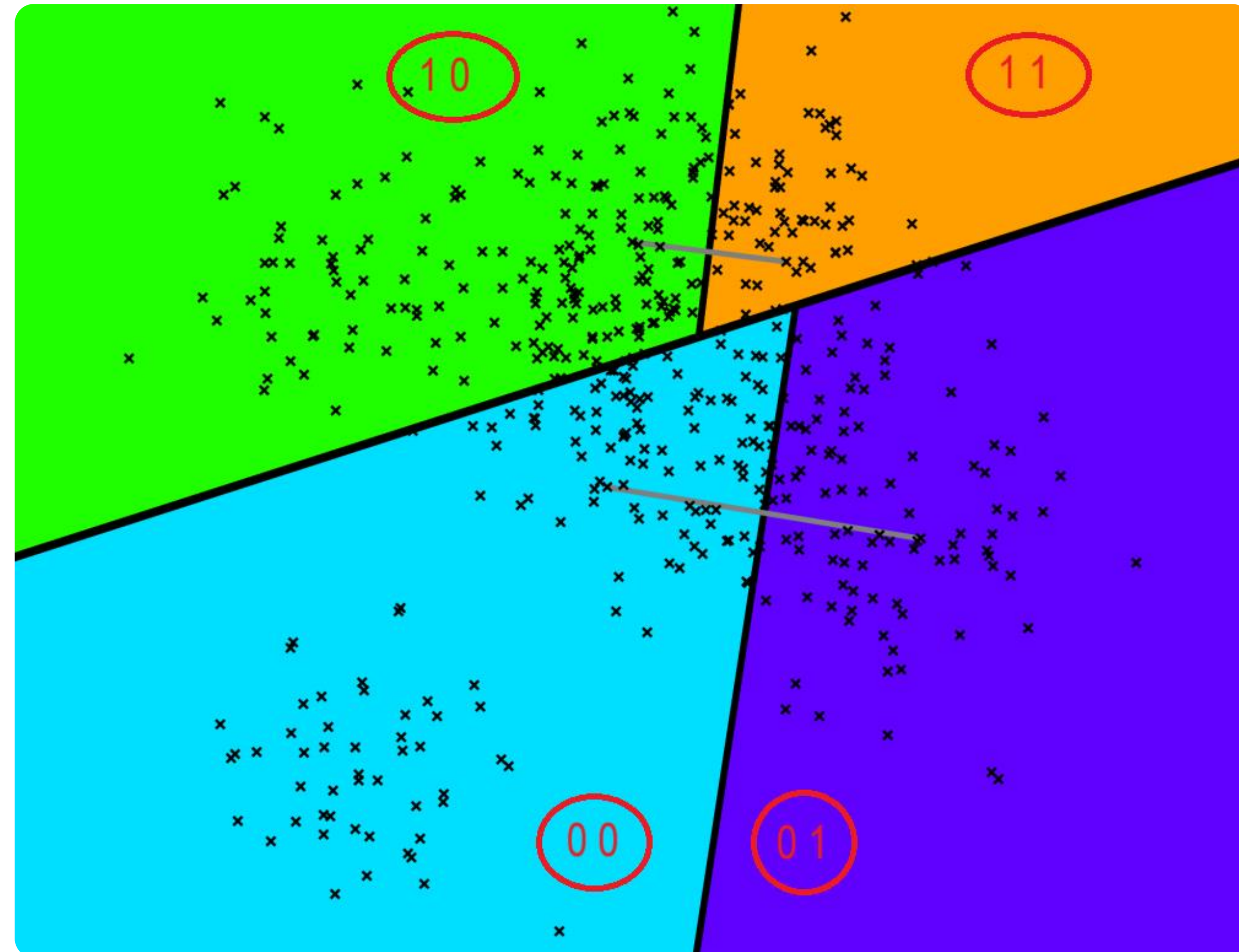


# Random projection search

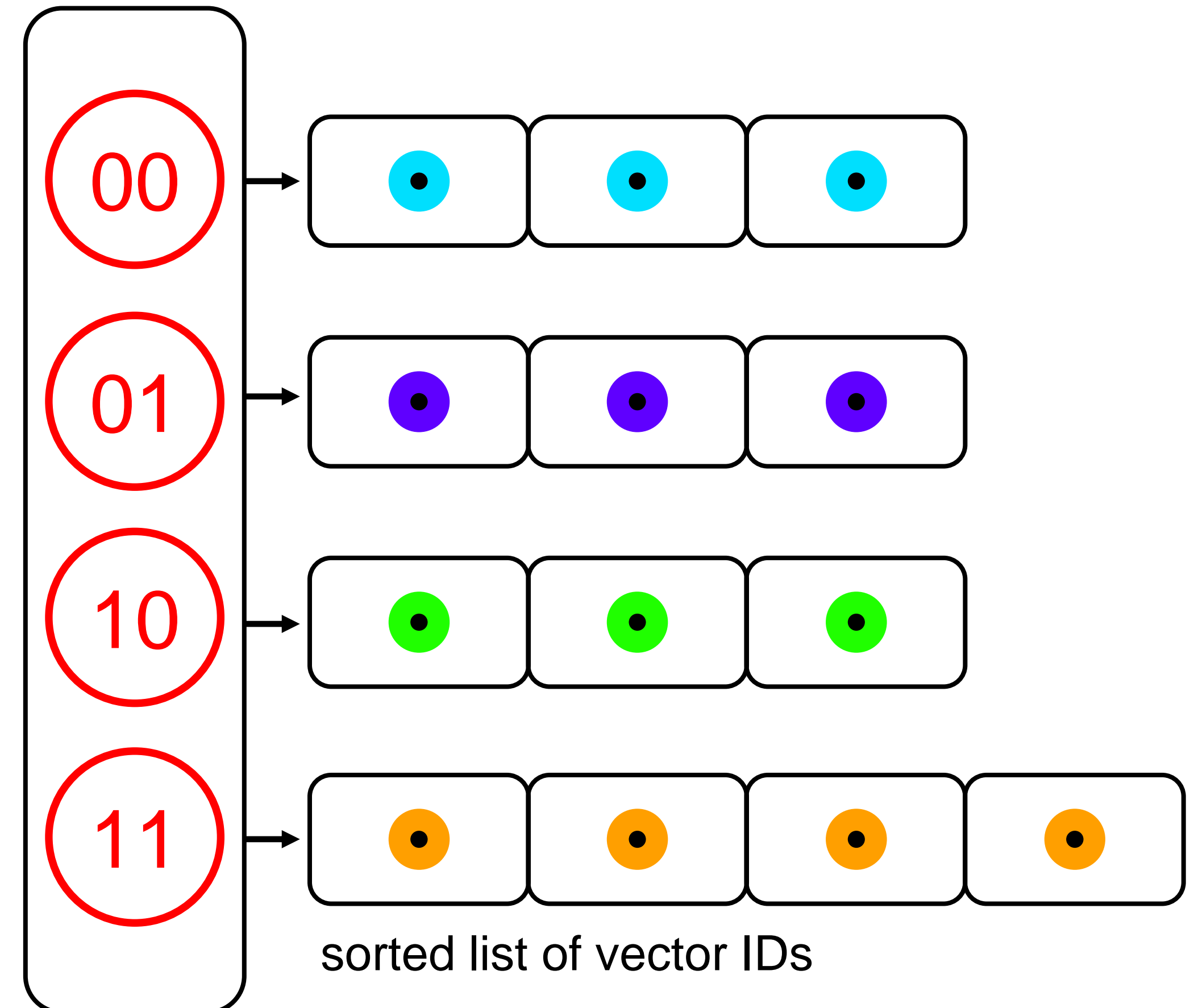
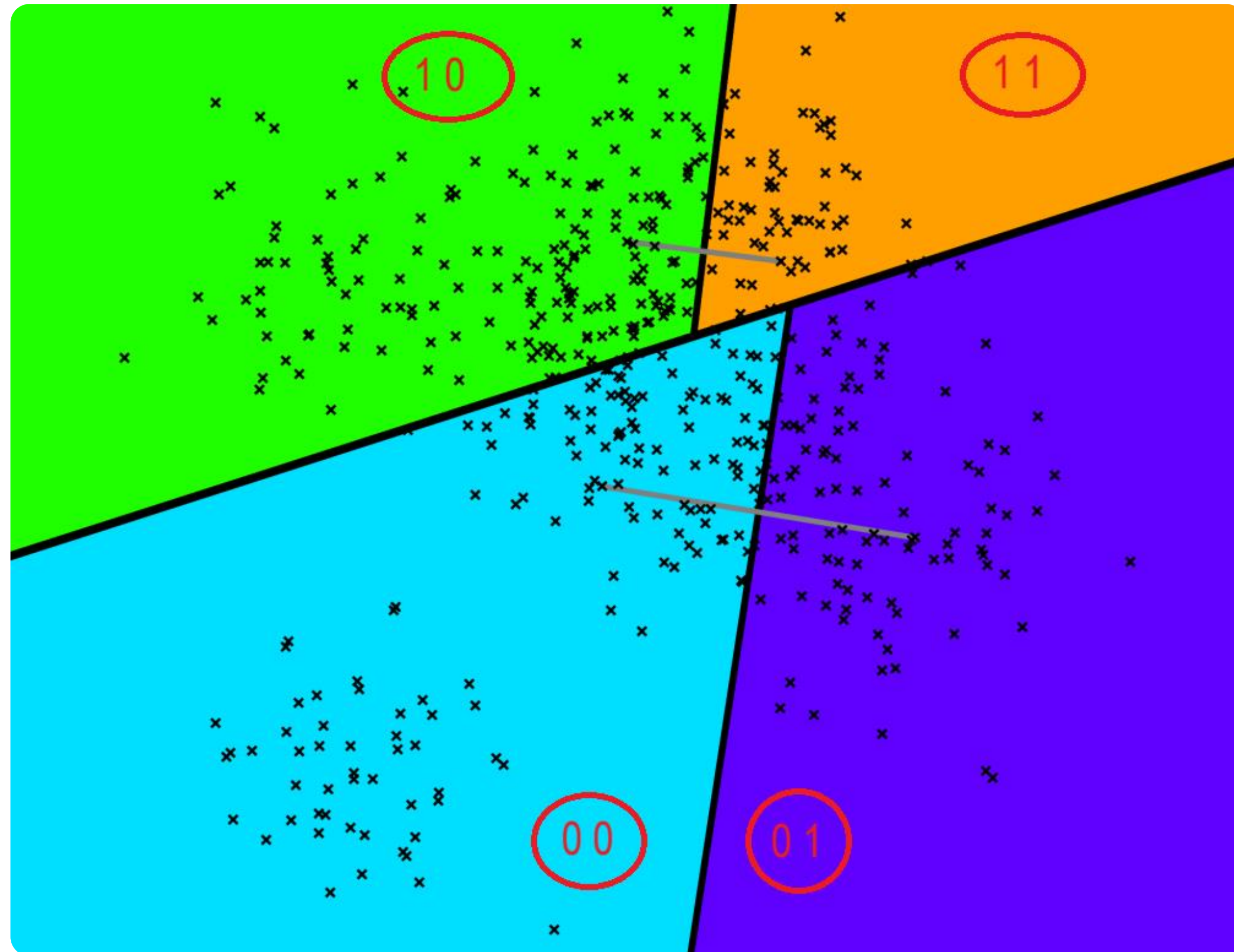


# Random projection encoding

hyperplane  $\leftrightarrow$  bitmask  
(of N bits)

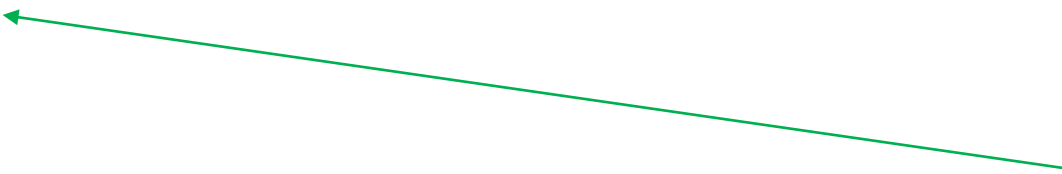


# Random projection encoding



# Random projections scheme

```
CREATE TABLE vectors (  
    id UInt64,  
    text String,  
    vector Bytes,  
    PRIMARY KEY (id)  
)  
  
CREATE TABLE polygons (  
    bits Bytes, // bit set  
    ids Bytes, // packed Sorted List  
    PRIMARY KEY (bits)  
)
```

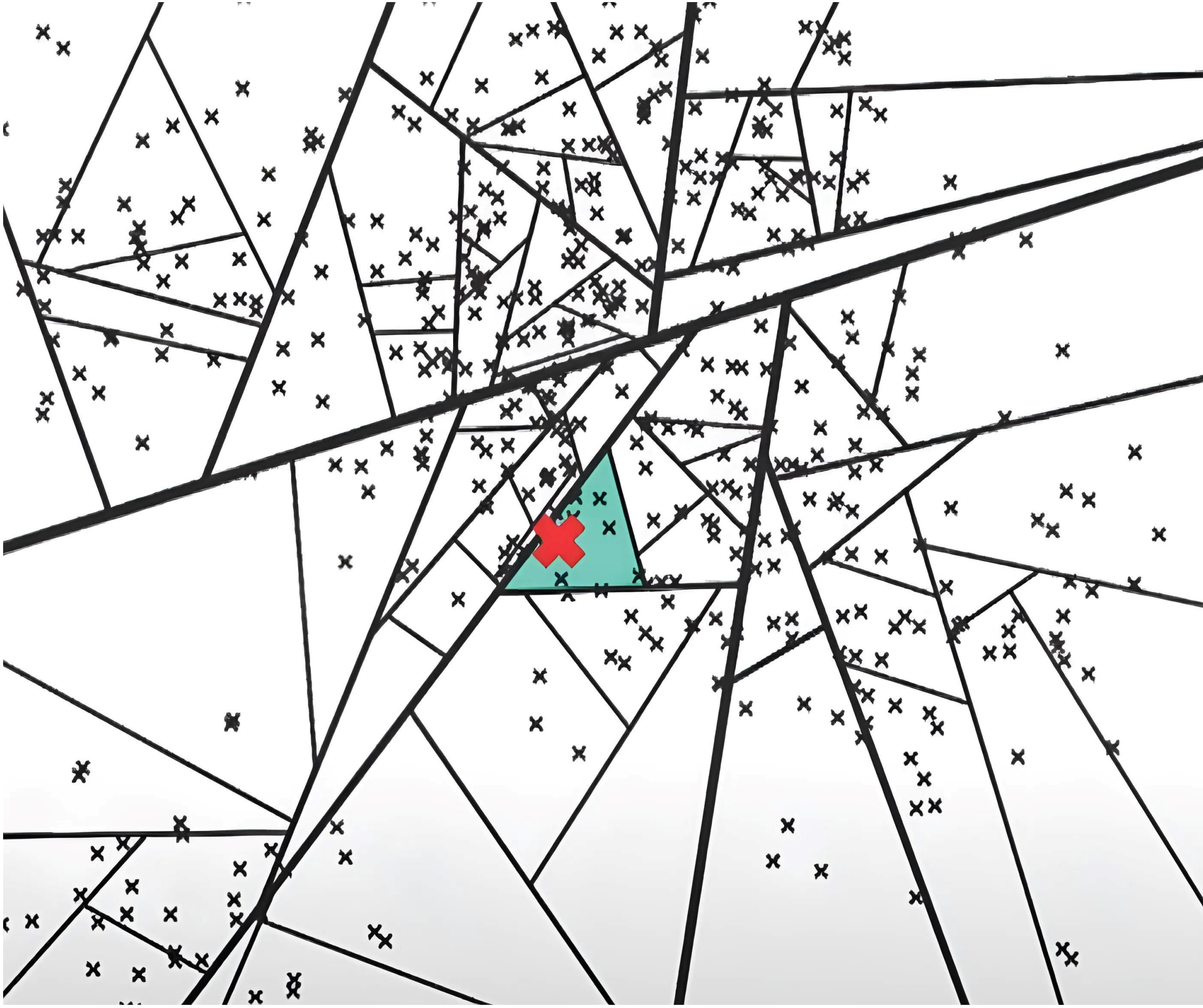


# Random projections search

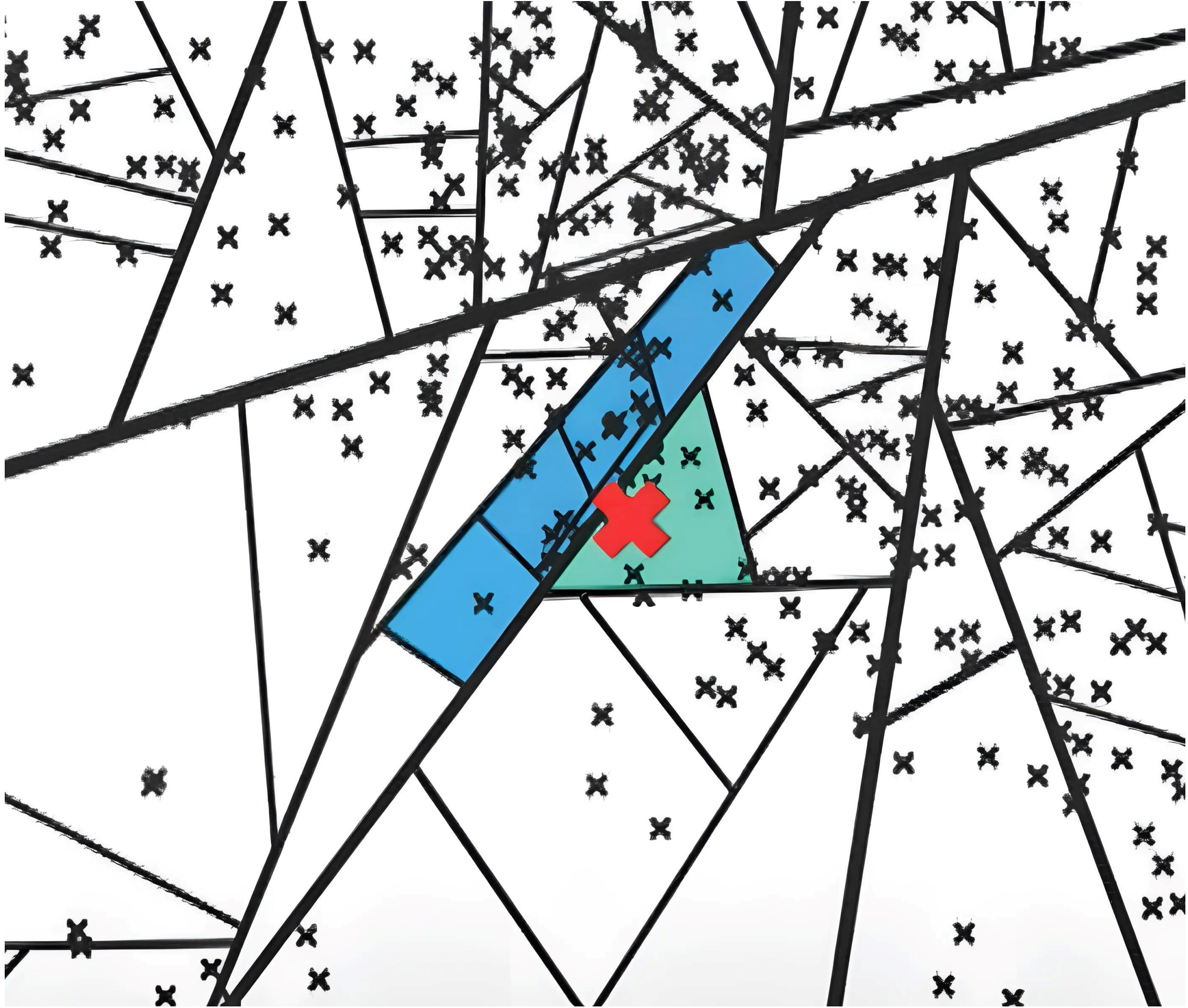
```
// approximate search for polygon
$approximate_ids = (
    SELECT ids FROM polygons WHERE
    bits = $TargetBits
)

// exact vectors search in the polygon
SELECT id, text FROM vectors
WHERE id IN $approximate_ids
ORDER BY CosineDistance(vector, $TargetVector)
LIMIT 10
```

# Random projection search



# Recall problem



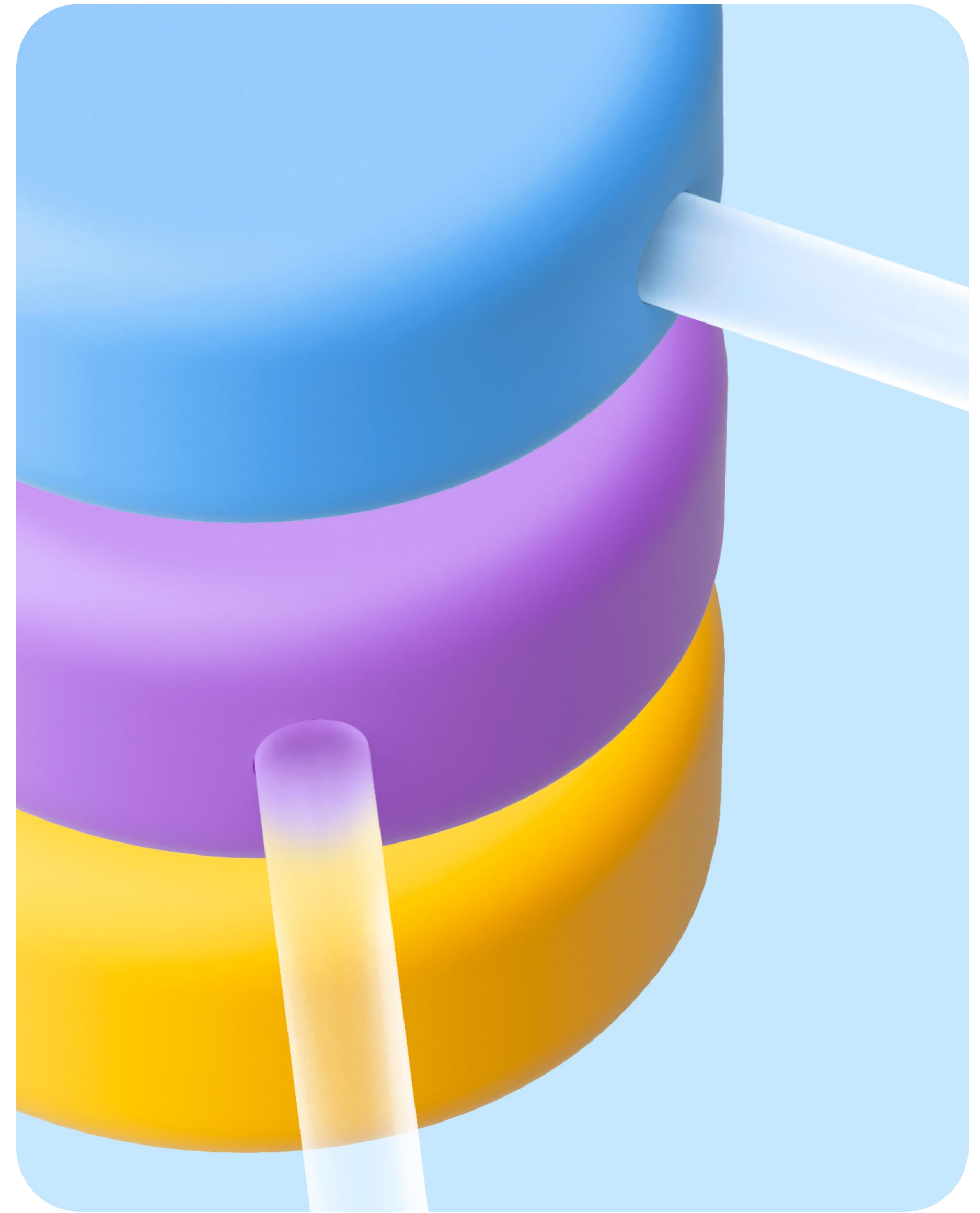
# **Solution: forest of trees**

## **Inspired by random forest classifier**

- Construct several random projections
- Search all of them
- Take a union
- Compute distance
- Return the K nearest

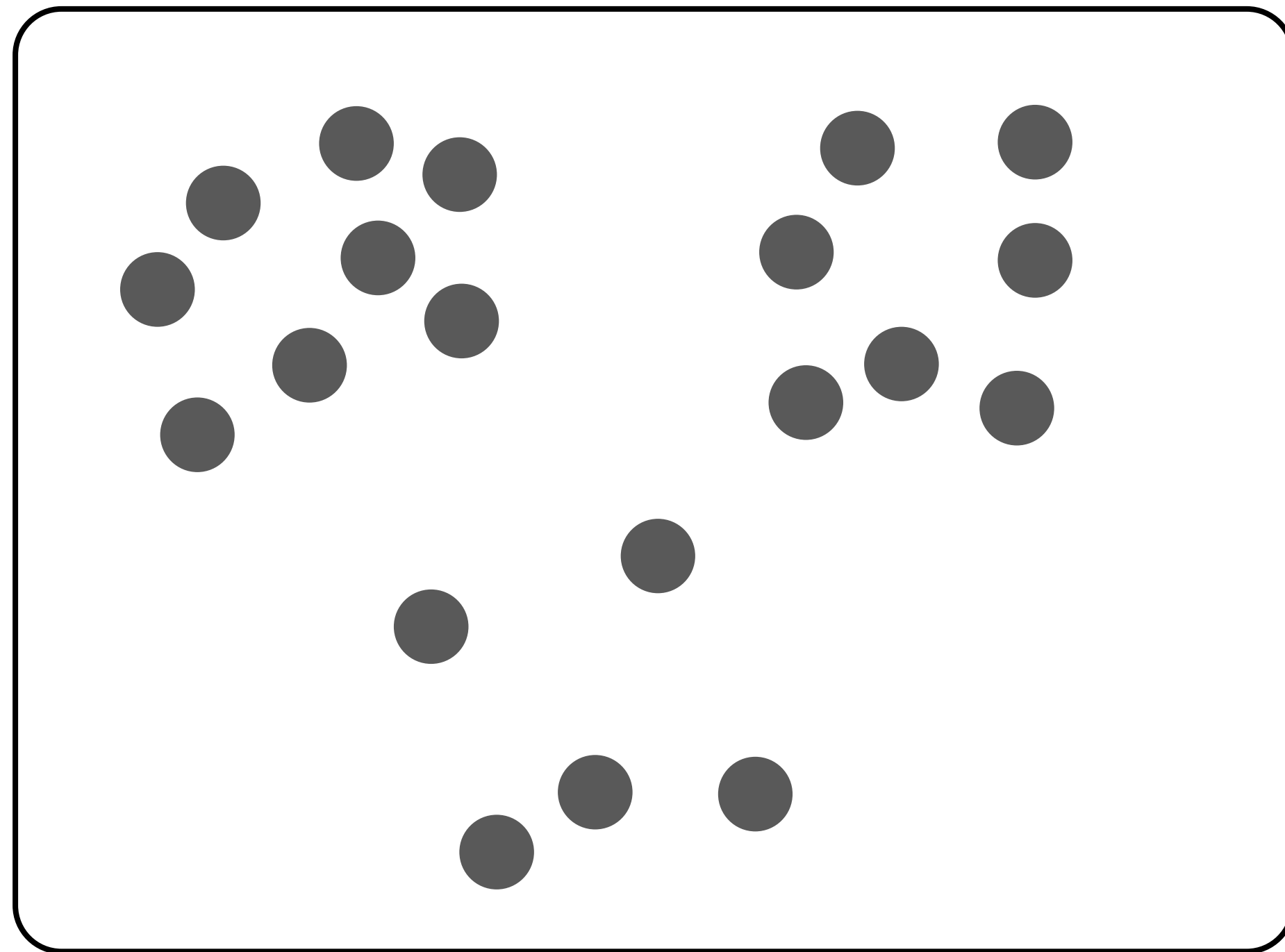


# YDB: FAISS inverted index



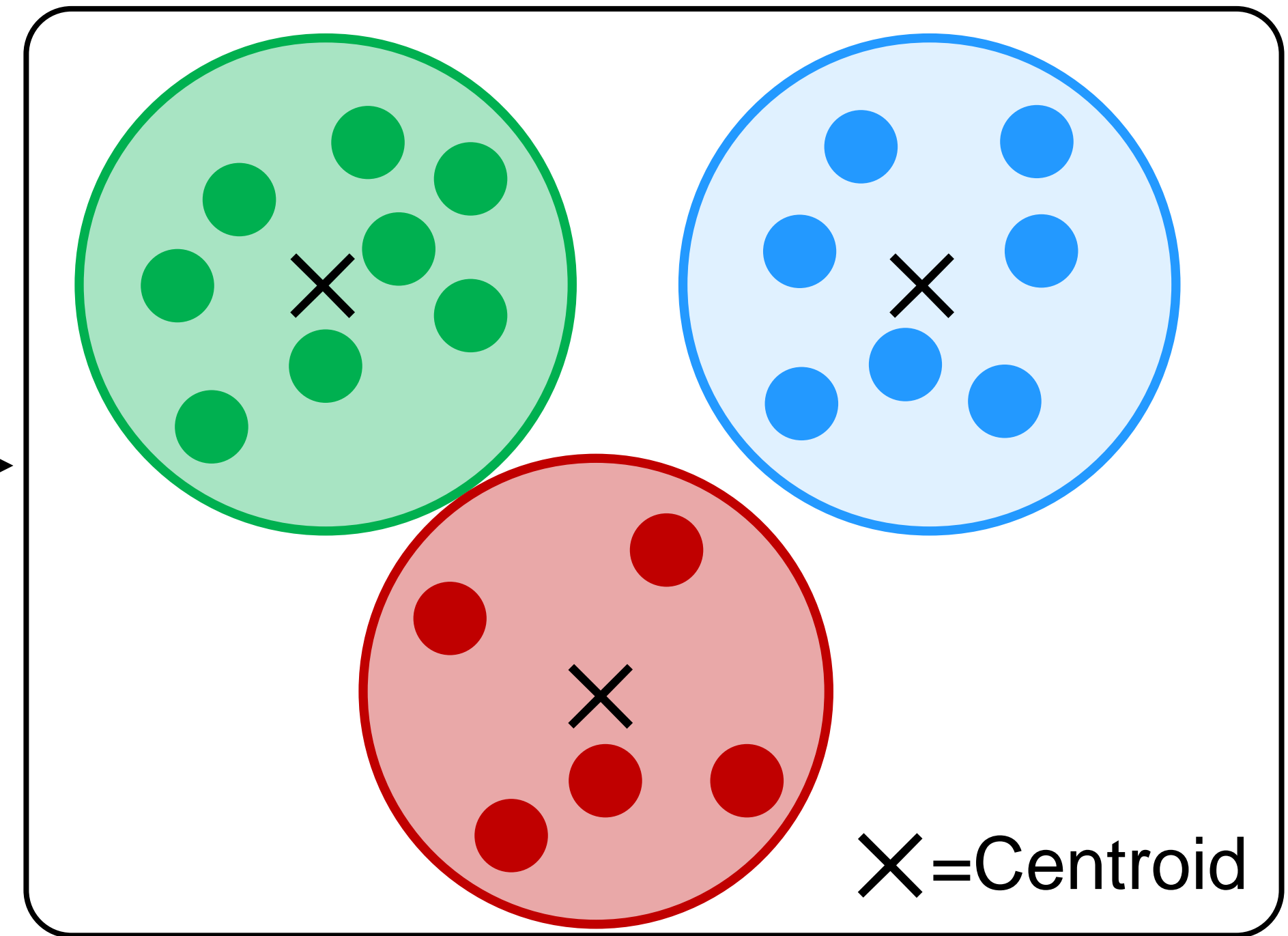
# K-means

Unlabelled Data



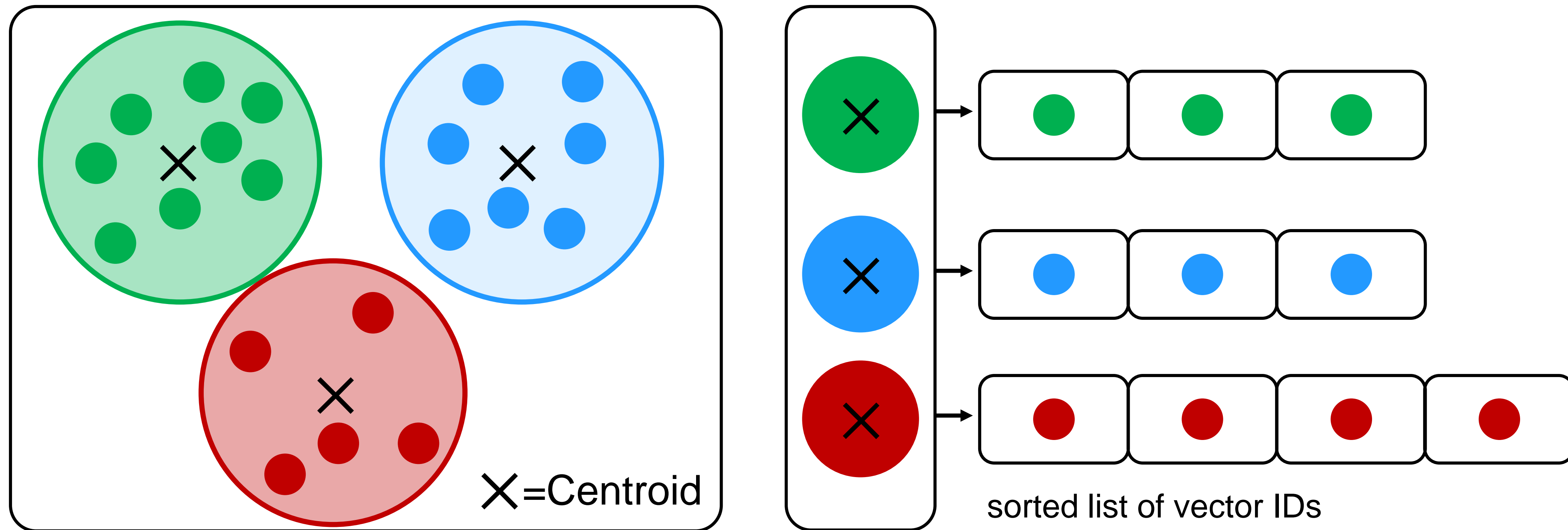
K-means

Labelled Clusters



# FAISS Inverted index

Labelled Clusters



# Resilience to incremental changes

## When a new vector is added:

- The only one ID is added to the sorted list of IDs
- The only one polygon/centroid is affected

# Random projections and Faiss vector indexes are preferred for YDB



Inverted indexes  
can be easy built  
on simple database  
scheme

简单的数据库方案



Search SQL  
queries are simple  
简单的SQL查找方法



Index can  
be global  
全局索引



Resilient to  
incremental  
changes  
适应增量变化

# YDB doesn't consider HNSW index

**Yes, it's the fastest one**

**But this is a **graph** with corresponding disadvantages:**

- requires more memory (nodes + edges)
- should fit the RAM
- difficult to distribute on shards, most implementations use local indexes
- data updates require index rebuilds

**Let's stay  
in touch**

**How to try YDB?**

**Why does it scale so well?**

**Why is it so robust?**

**What client utilities/  
languages are supported?**

# Conclusion

- Brute force exact search works well

精确搜索效果一流

- Approximate vector indexes are suitable for Big data

近似向量索引适用于大数据

- We are combining Big Data and AI

我们正在结合大数据和人工智能



[ydb.tech/zh](https://ydb.tech/zh)