Hardware Accelerated Similarity Search

George Williams

Who Am I?

Director, GSI Technology

Previously, Chief Data Scientist

Senior Data Scientist

AI Research Scientist

Software Engineer

Recent Headlines

TC

A new ARM-based server processor challenges for the data center

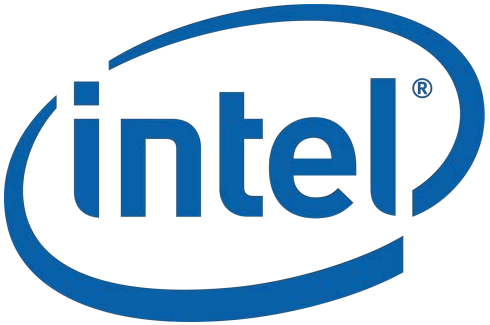
NBC

Facebook is looking for engineers to build its own chips

THENEXTPLATFORM

JUST HOW LARGE CAN NVIDIA'S DATACENTER BUSINESS GROW?

Convergence and Integration



That was then...

This is Now: Technology Disintegration

amazon web services™ | EC2

Google Cloud Platform

intel AI

facebook

Qualcomm

habana

arm

Movidius
an Intel company

Microsoft HoloLens

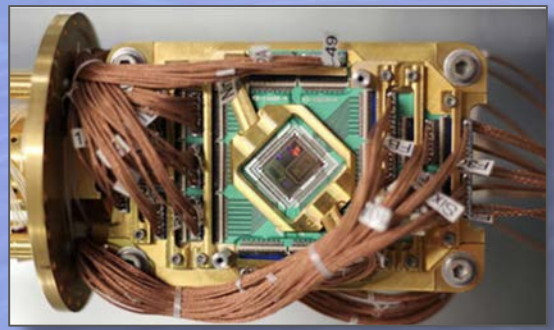
Microsoft Azure

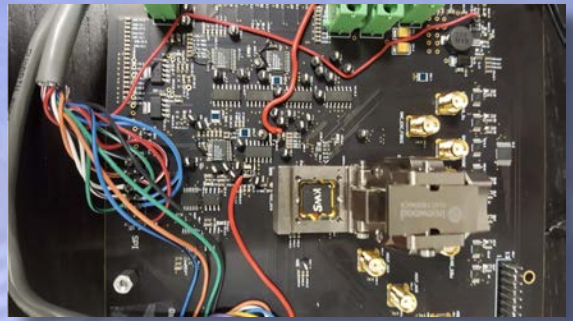
NVIDIA®

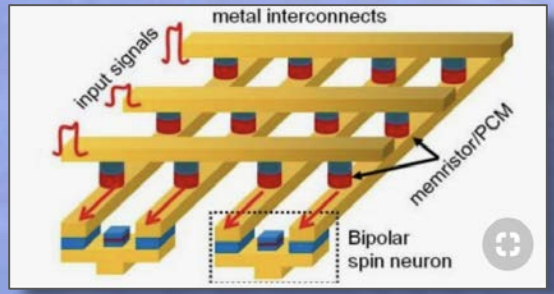
MYTHIC

Baidu 百度

More Innovation Around The Corner

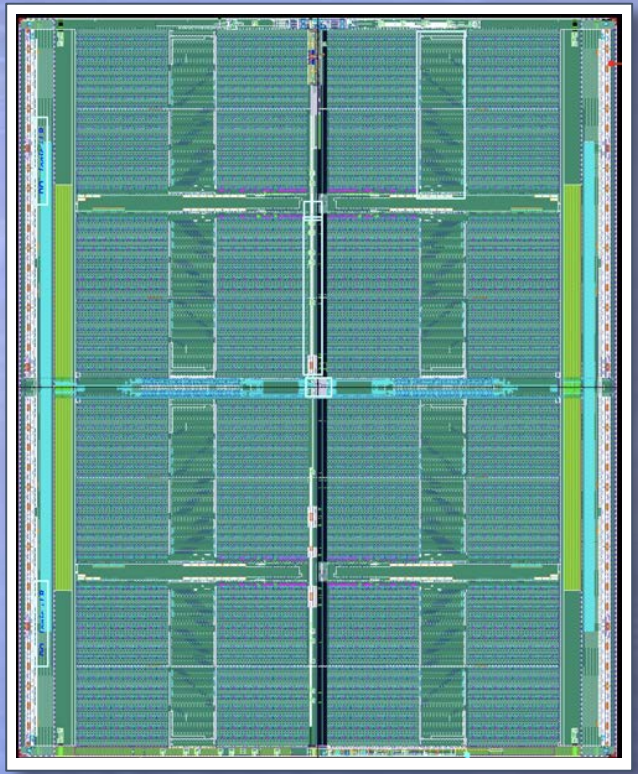






?

GSI's Similarity Search Accelerator

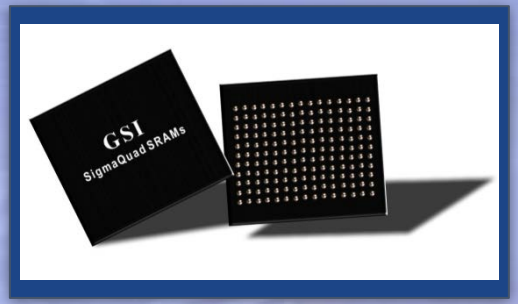


Agenda

- Chip Explosion
- GSI Technology
- What is Vector Similarity Search?
- GSI's Similarity Search Accelerator
- Integration Case Studies: Bio, Database
- Early Adopters Program

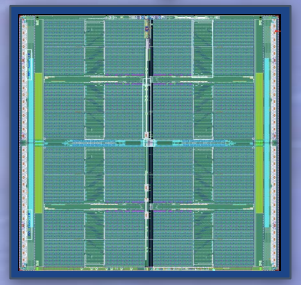
Who Is GSI Technology?

What We Do



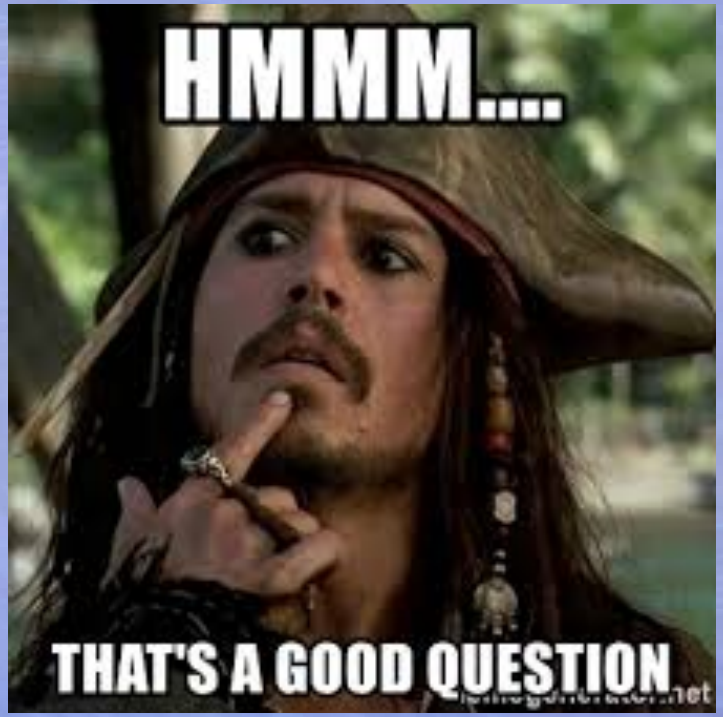
High Performance SRAM and DRAM

Aerospace, Government, R&D



GSI Vector Similarity Search Accelerator Chip

What Is Vector Similarity Search?



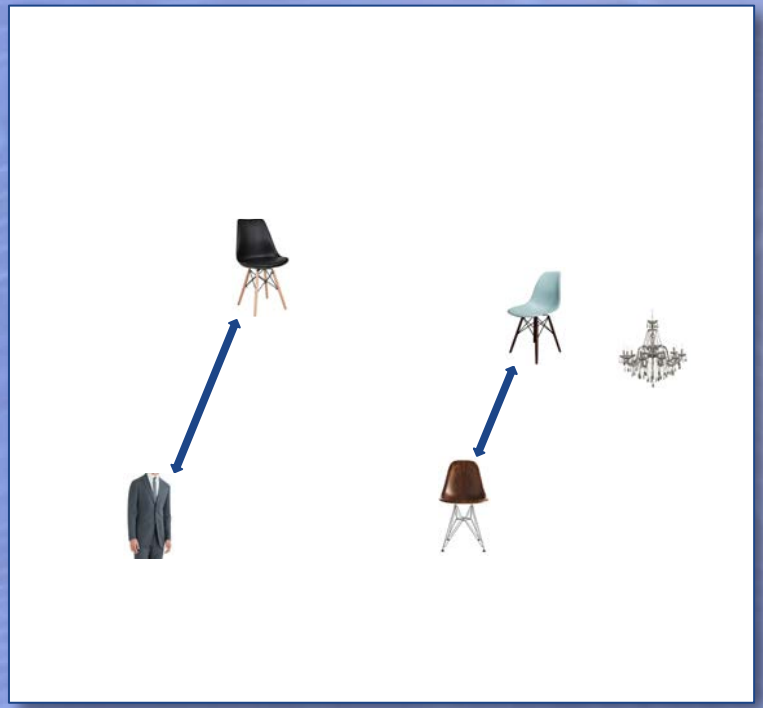
What is Vector Similarity Search?

- **Numeric Representation**

Bit-vector → 0110000100

Coordinates → (2.3, 5.6)

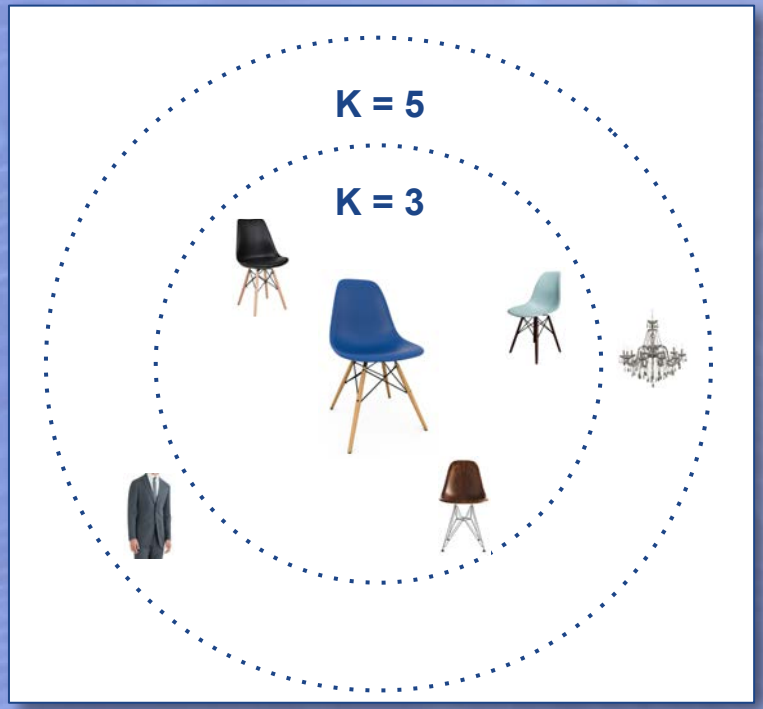
What is Vector Similarity Search?



- ✓ **Numeric Representation**
- ✓ **Simple "Distance" Function**

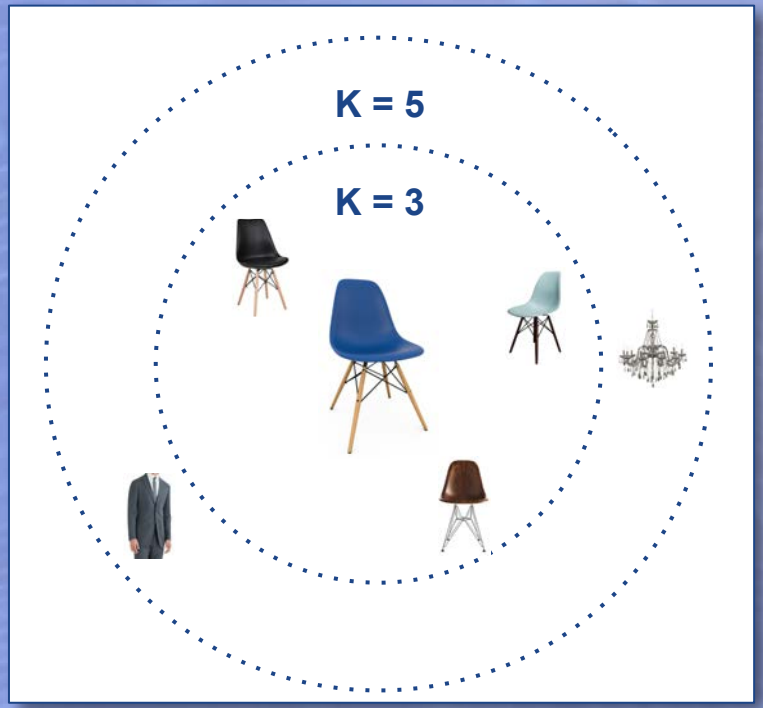
d = *Func* (a, b)

What is Vector Similarity Search?



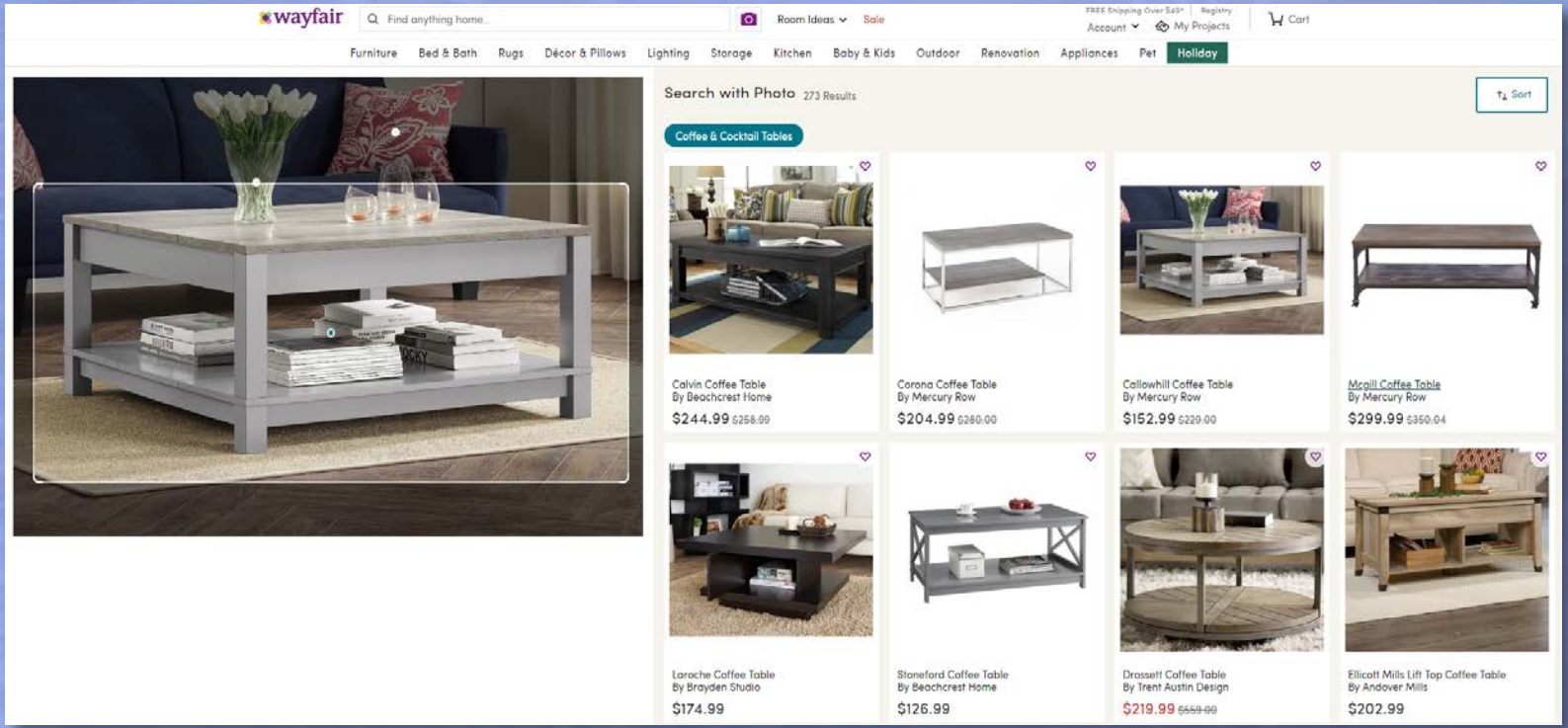
- Numeric Representation
- Simple “Distance” Function
- K Nearest Neighbor (Top-K)

What is Vector Similarity Search?



- Numeric Representation
- Simple “Distance” Functions
- K Nearest Neighbor (Top-K)
- Search is Computational
- E-Commerce, Bioinformatics

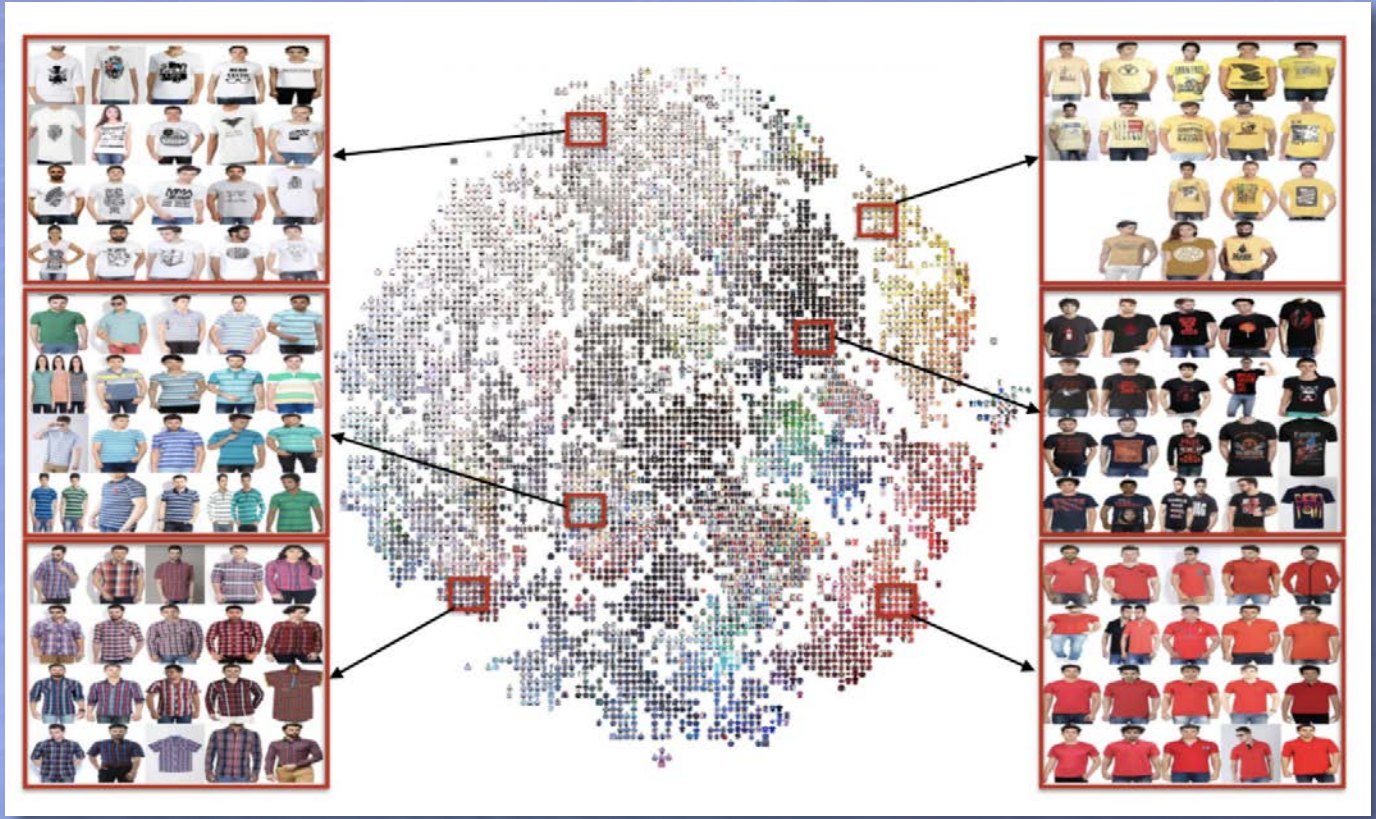
E-Commerce: Visual Search



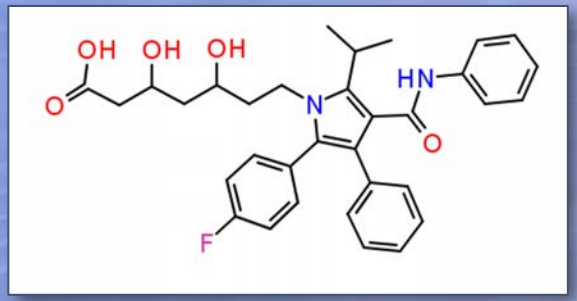
Visual Search

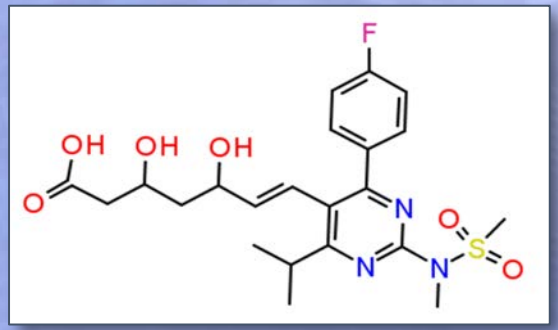
- Binary Codes, Continuous Embeddings
- Euclidean, L1, Hamming, Cosine
- >1 Billion Images

Visual Search: Embedding Space



Bioinformatics: Molecule Similarity

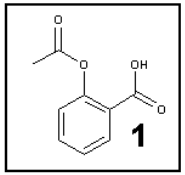


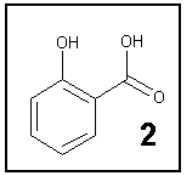


- Fingerprints
- Tanimoto
- Many Large DBs
- 100s GB

Molecule Similarity: Tanimoto

Similarity Searching





1	1	1	0	1	1	0	1	0
2	1	1	0	1	0	0	0	0
								

A = Number of bits set in both = 3
B = Number of bits set in (1), but not in (2) = 2
C = Number of bits set in (2), but not in (1) = 0

TANIMOTO COEFFICIENT = A / (A + B + C)
= 3 / (3 + 2 + 0) = 0.6 or 60%

- ✓ Jaccard
- ✓ Intersection / Union

Bioinformatics: Molecule Similarity

- Drug Discovery of Novel Molecules
- Virtual Screening
- Activity (Toxicity) Prediction

Programming Interfaces

```
CREATE TABLE features (
        id bigint(11) NOT NULL AUTO_INCREMENT,
        feature_vector binary(4096) DEFAULT NULL,
        KEY id (id) USING CLUSTERED COLUMNSTORE
)
```

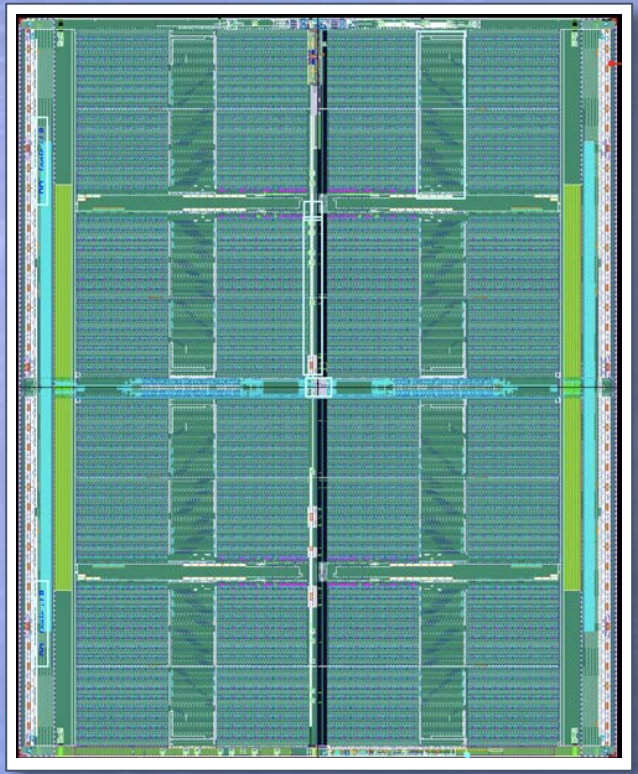
```
SELECT
        id
FROM
        features
WHERE
        DOT_PRODUCT(feature_vector, <Input>) > 0.9
```

- **Idiomatic SQL**
- **Integrate Into Data Pipelines**
- **Leverage Skills of Data Eng & Scientists**

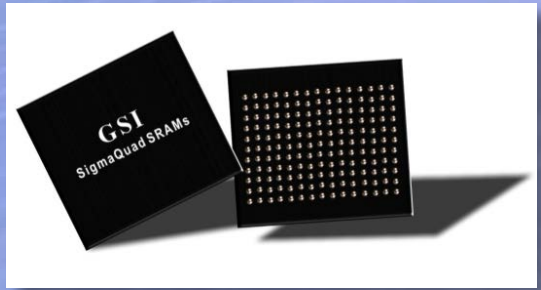
Many Domains and Applications

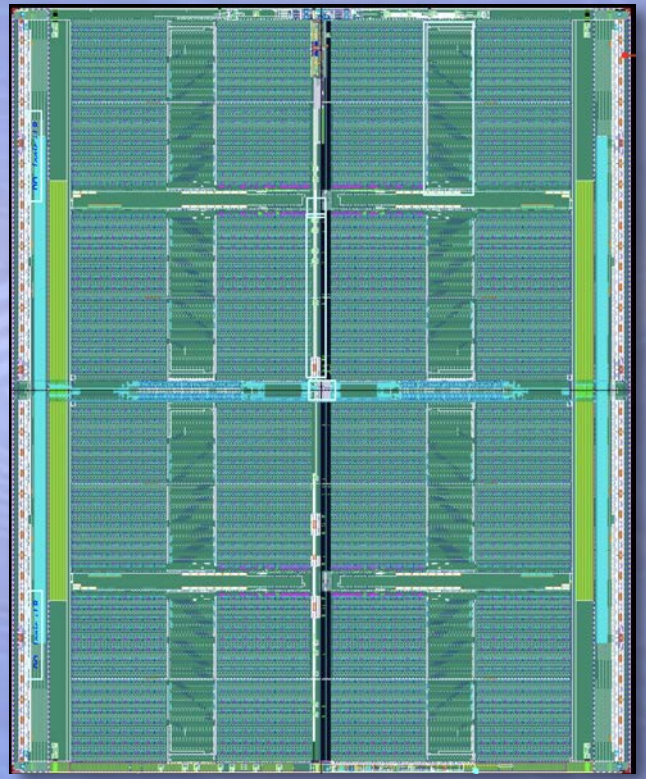
- E-Commerce / Recommendations
- Bioinformatics / Genomics
- Healthcare / Medical Records
- Cybersecurity / Malware Detection
- Computer Vision / Video Surveillance

GSI's Similarity Search Accelerator

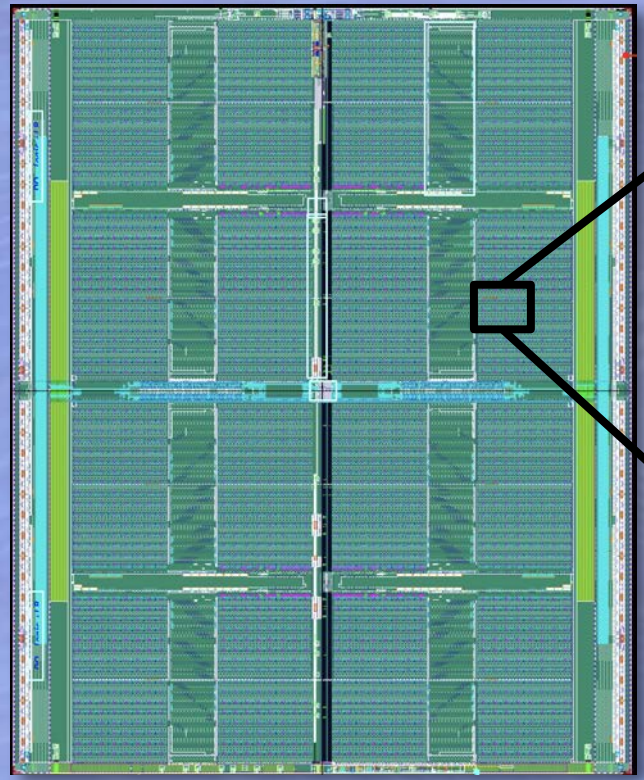


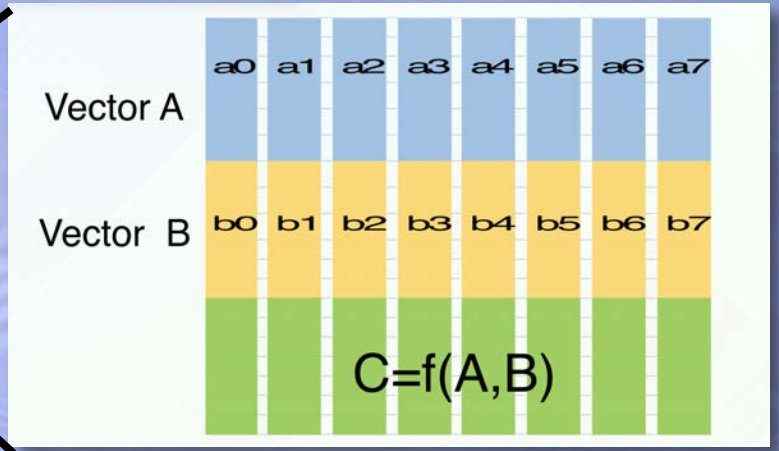
Computational Memory





“In-Place” Associative Processing

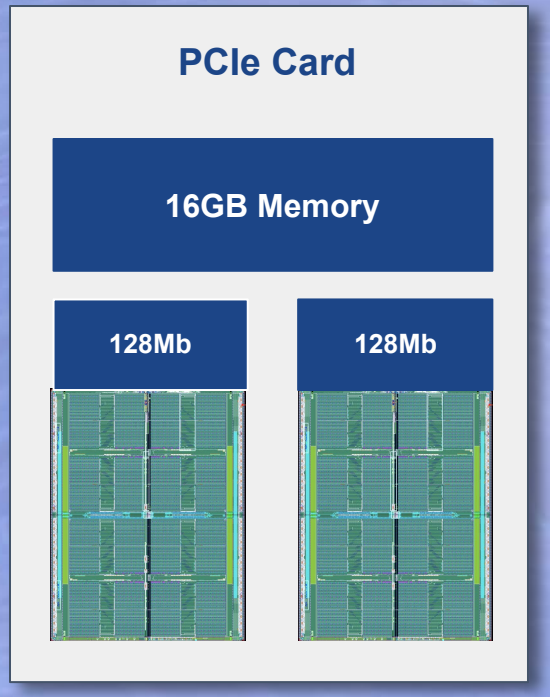




Bit Logic

- **Programmable**
- **2 million**

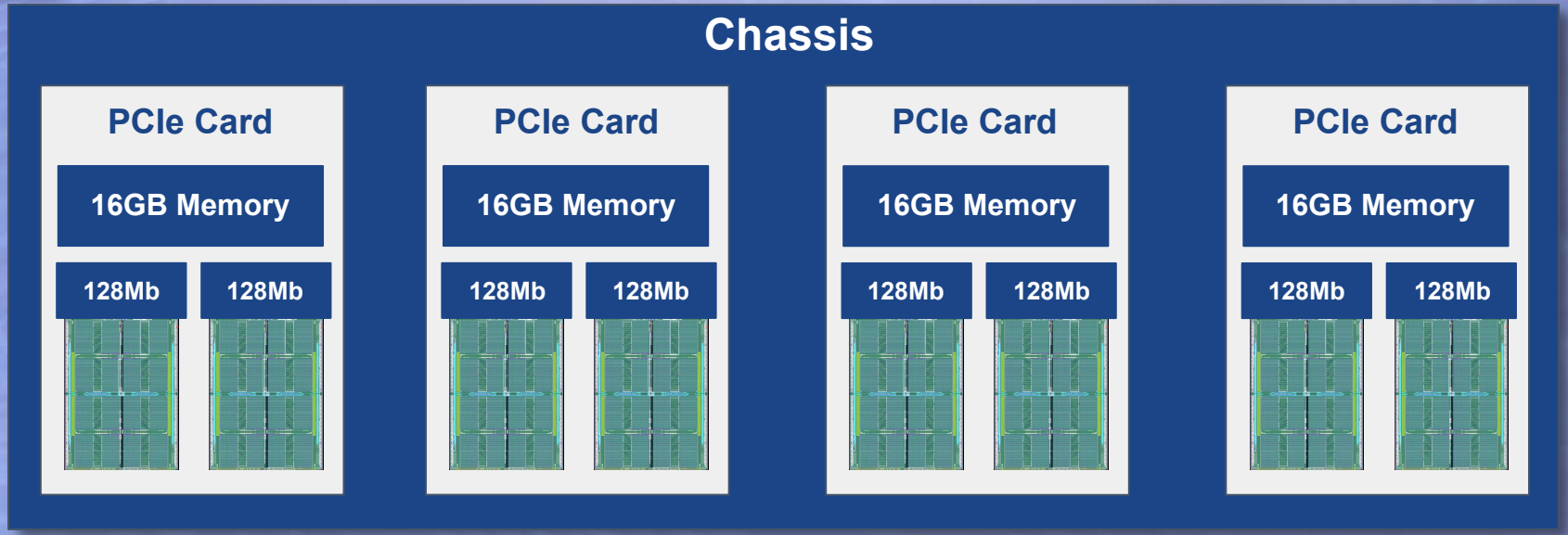
Consumer Board Solution



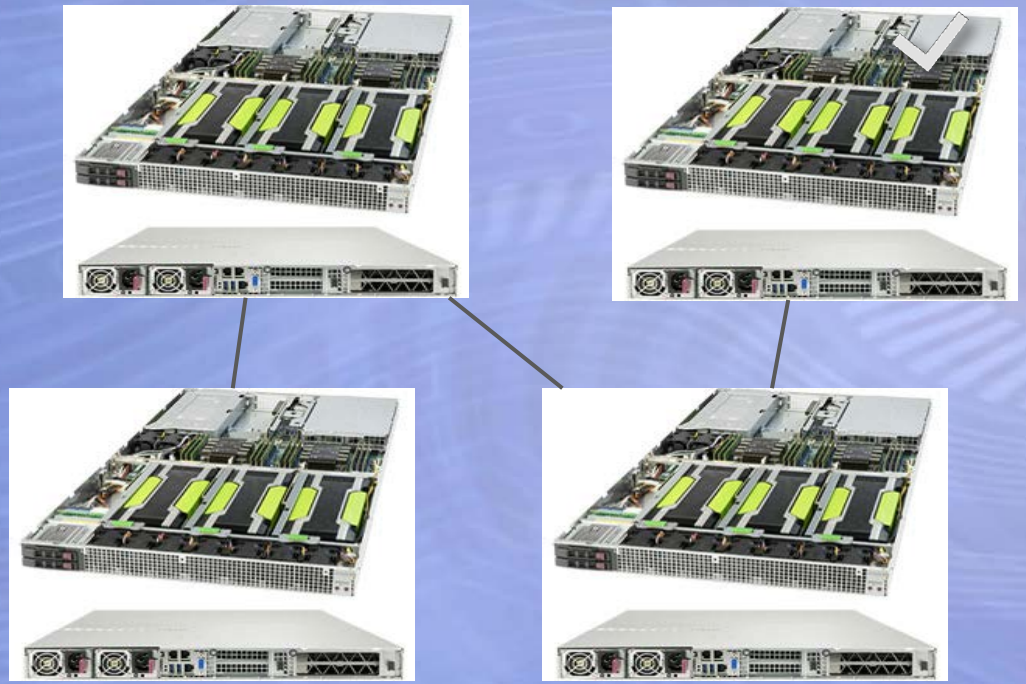
- 2 Chips Per Board
- On Board DDR4 Main Memory
- SRAM Cache Per Chip

1 Chassis (4U) Solution

4 Boards Per Chassis

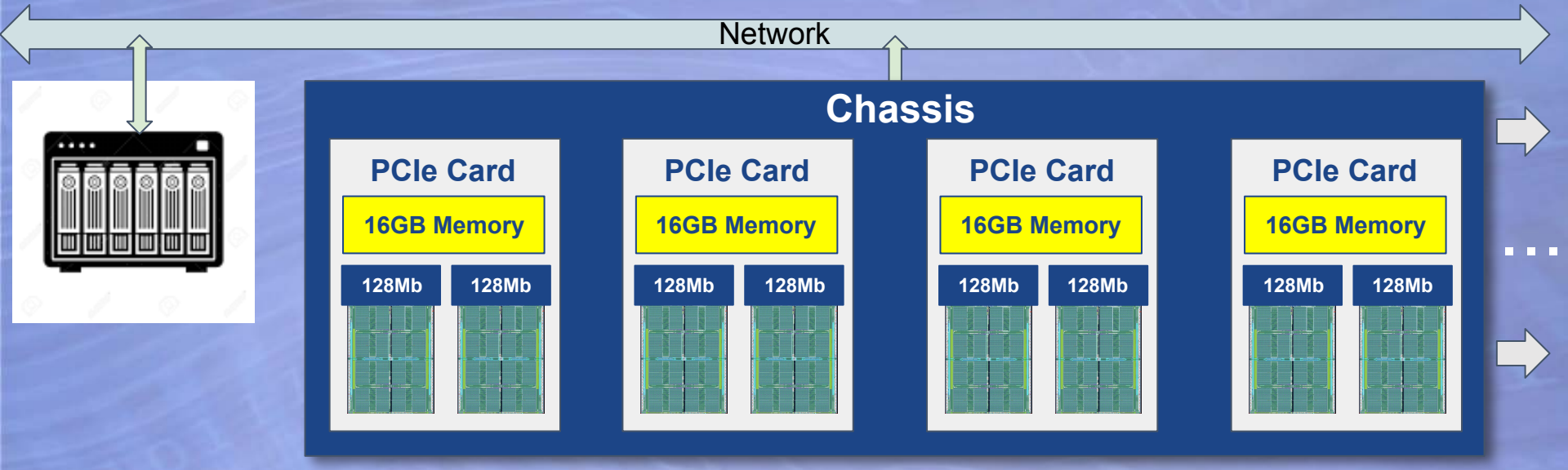


Multiple Chassis Solution



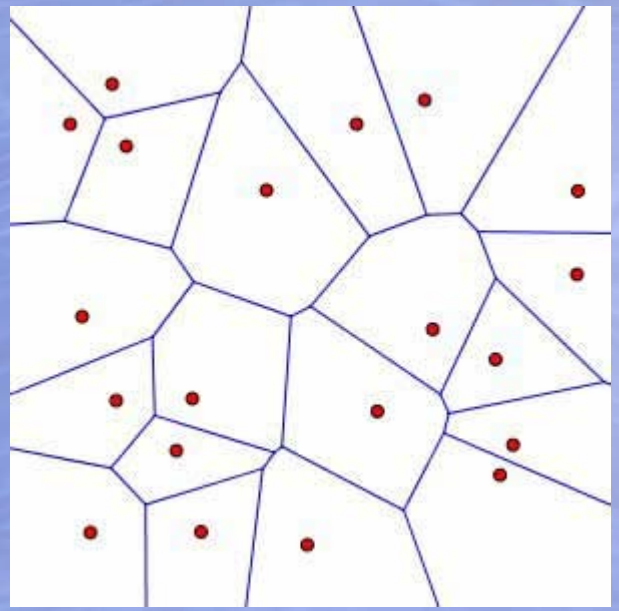
- One Chassis Is The Master

Network Attached Storage



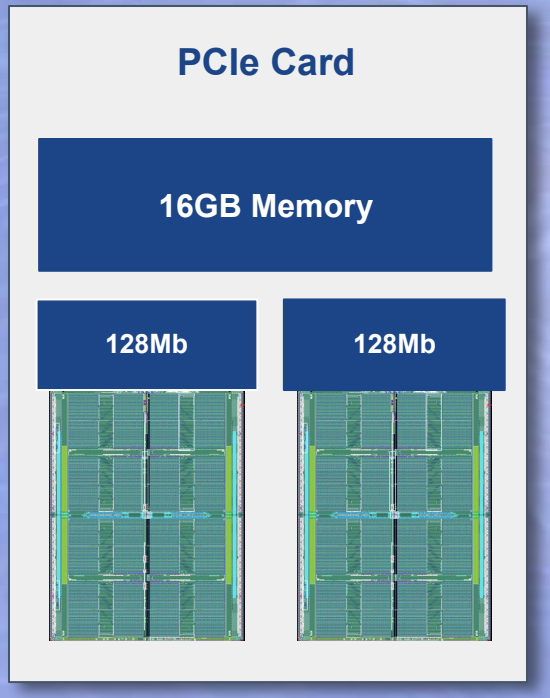
- ✓ **RDMA support (NAS As Data Source)**

Segmentation by Clustering



- Offline Clustering / K-Means
- Avoids Full DB “Scan”
- Faster Performance

Availability



chip → Q4, 2018

demo boards → Q1, 2019

mass production → Q2, 2019

Weizmann Institute Case Study

“GSI’s [Accelerator] can dramatically reduce the time required to search our small molecules database...”

- Dr. Efrat Ben-Zeev, Computational Chemist

מכון ויצמן למדע
WEIZMANN INSTITUTE OF SCIENCE

Weizmann Institute Case Study

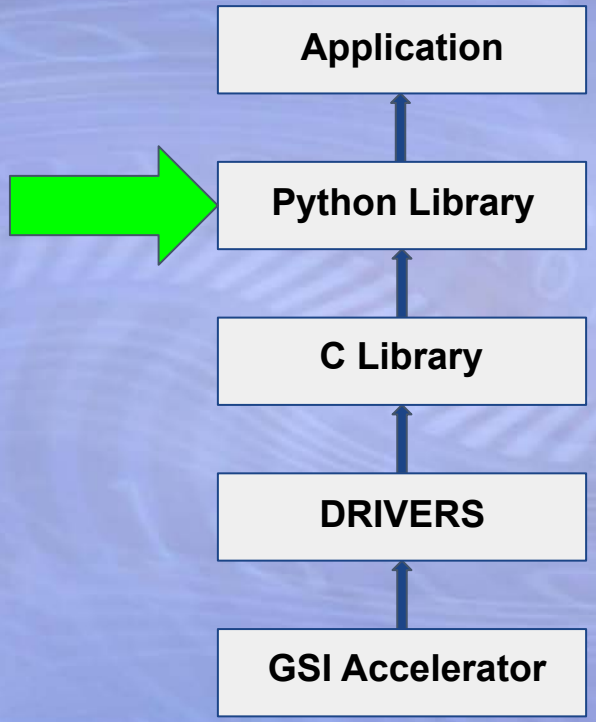
Molecule Similarity Search

- ✓ Biovia Pipeline Pilot Application
- ✓ Query of 34M Molecule DB Takes 10 Minutes !

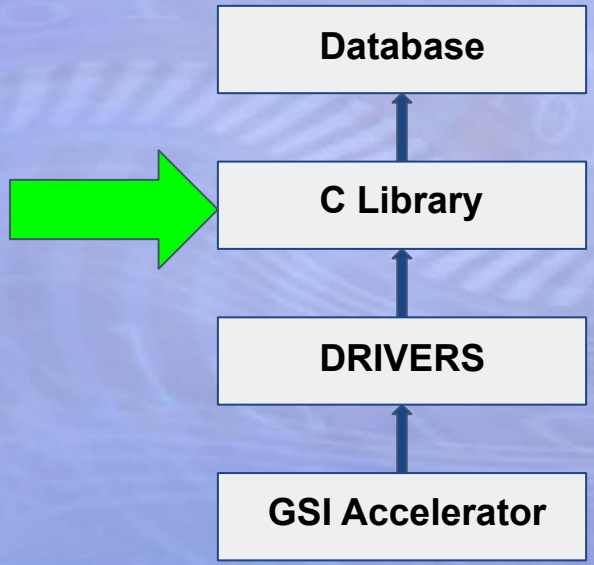
Using GSI Accelerator (estimated)

- ✓ Query Latency Reduced To 300ms
- ✓ 400 Queries In 1 Second

Biovia Application Integration



In-Memory Database Integration



In-Memory: Expected Performance

	Memory Speed	Vector Size	Throughput (imgs/ sec)
MemSQL	~50GB / sec	4 KB	12.5 million images/sec
GSI	~100GB / sec	4 KB	25 million images/sec

Early Adopters Program

- Consult With Our Hardware and AI Experts
- Co-Development and App Integration
- Access to simulator and test hardware
- Co-Marketing Opportunity

GSI Upcoming Events

- Nov, Open Data Science Panel, Visual Search
- Nov, PyData (Washington DC)
- Dec, GSI Similarity Search Accelerator Workshop
- Coming Soon, GSI's Tech Meetup
- 2019, First Chips and Boards Available

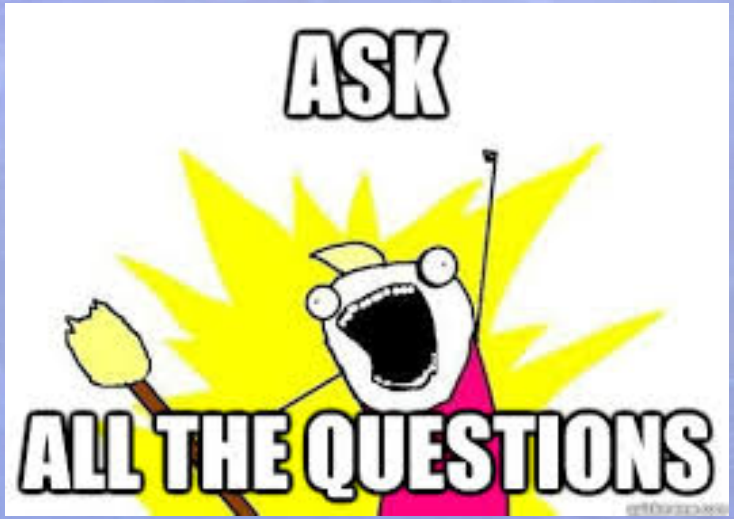
Contact Us

Twitter: @gsitechnology
@cgeorgewilliams

Blogs: gsitechnology.com
medium.com/gsitechnology

Email: associativecomputing@gsitechnology.com

The End. Thanks !



Query

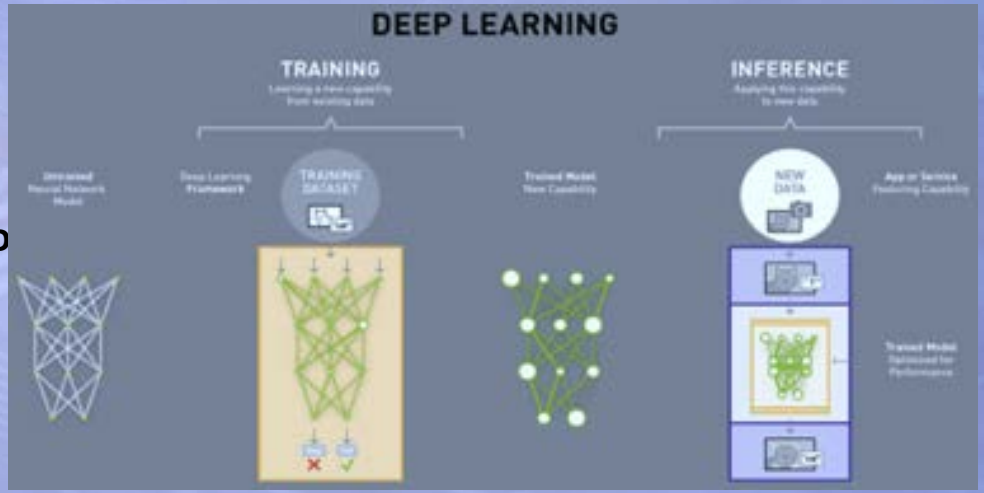
Option 1: INFERENCE IN PRODUCTION

- Image2Vec (VGG, Resnet)
- NLP
- LSTN

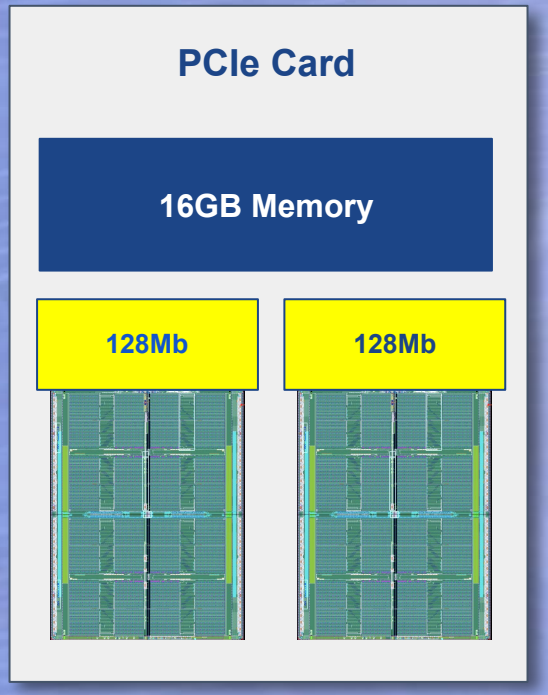
Option 2: Query Vector done by External App

- Fingerprint Query
- In Memory Vector (Cosine)

Option 3: 3rd Party Inference Vector

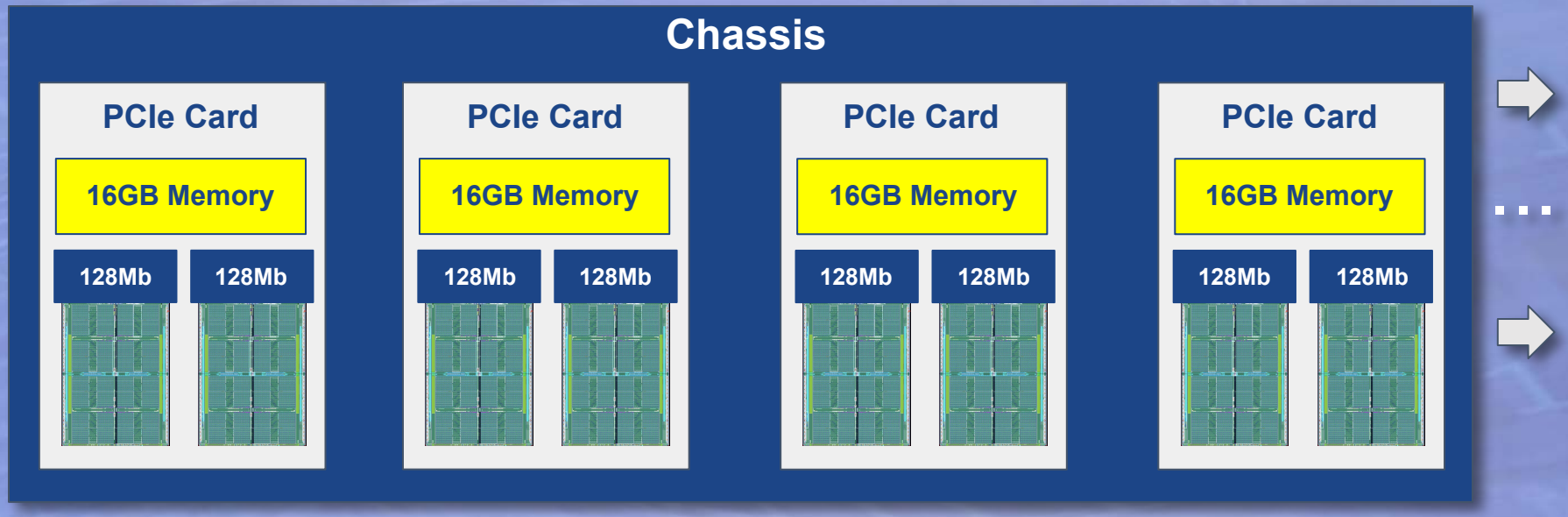


Single Board



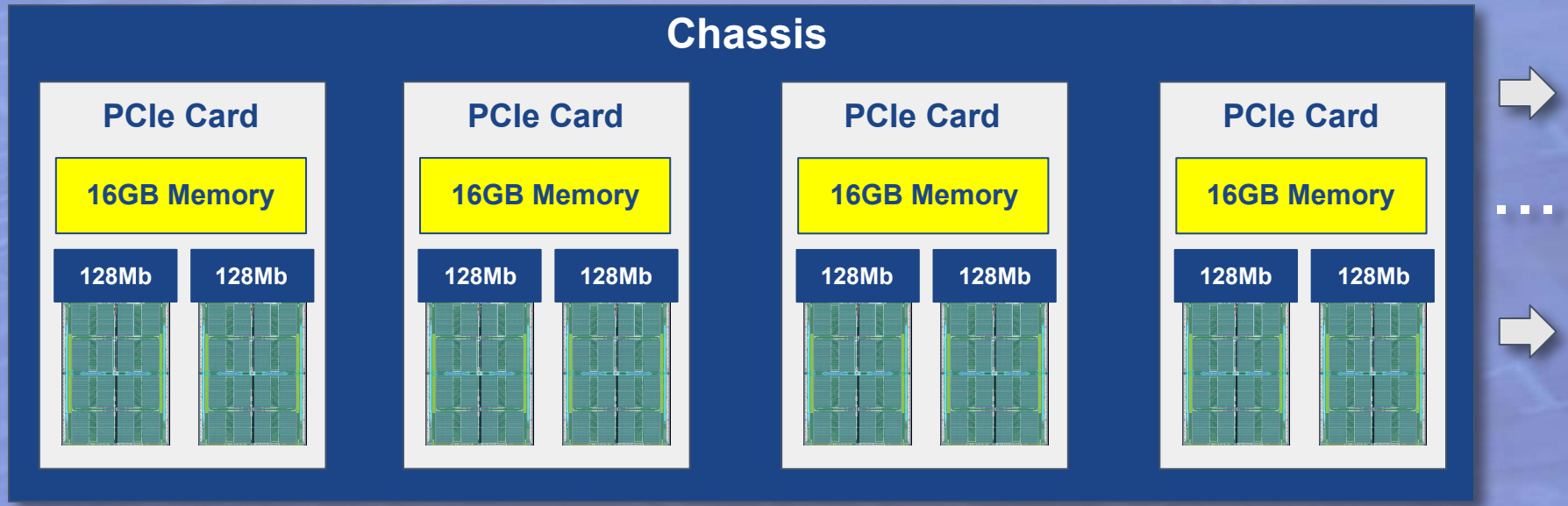
- Small Database
- Fit All Data Into Cache For Lowest Latency
- If Larger, Paging Occurs To Memory
- Cluster Techniques

Large Database



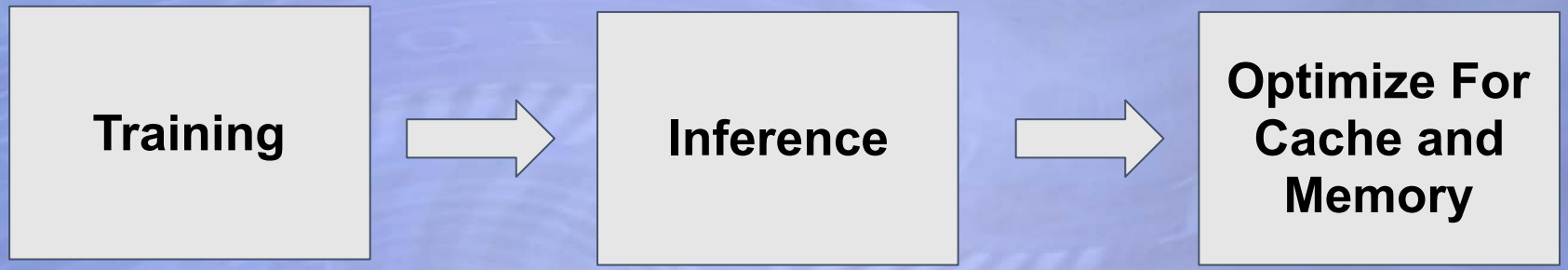
- Large Database (<1TB, Flat)
- Pharma, Drug Search, Weizmann Molecule Search, In-Memory

Multi Board Solutions

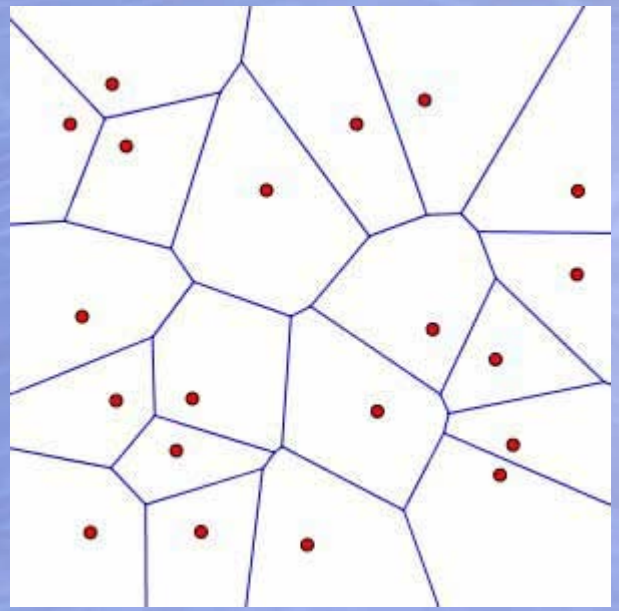


- For Huge Databases (~ 1TB)
- For Throughput: Batch Queries Split Across Boards
- Chassis Host Master Merges The Results

Offline Data Preparation

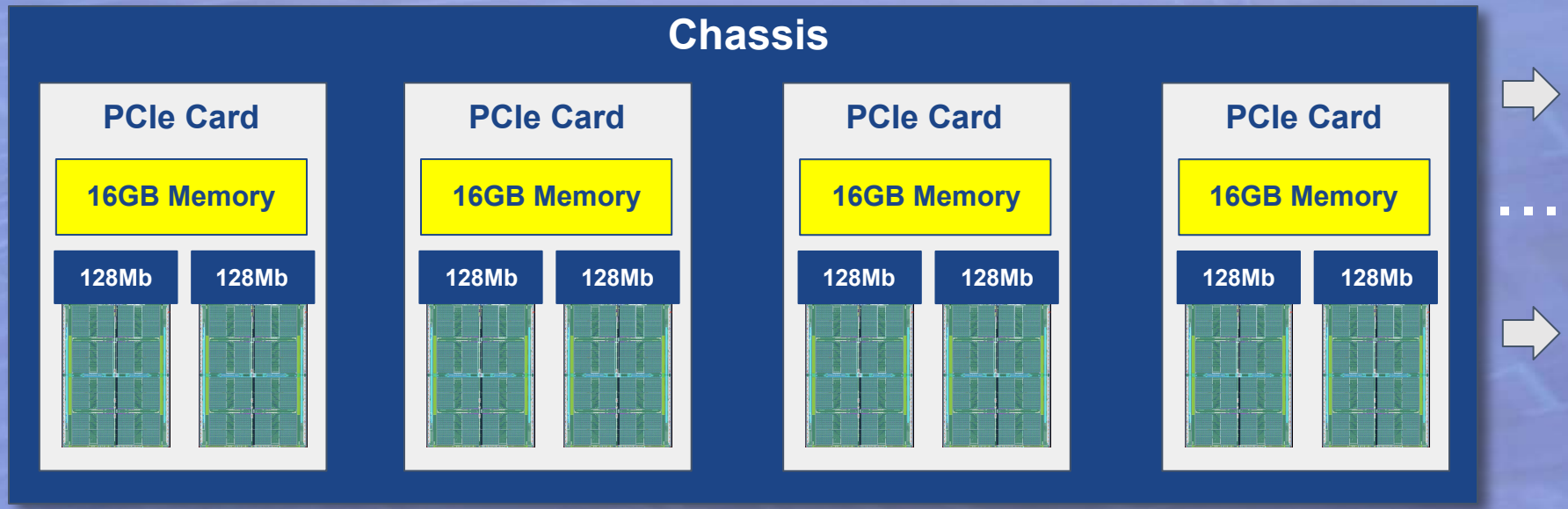


Clustering For Large Databases



- Offline Clustering
- Centroids List <16GB
- Reduces Storage
- Only Centroids Are Kept Local
- For Real-Time Performance

Huge Database

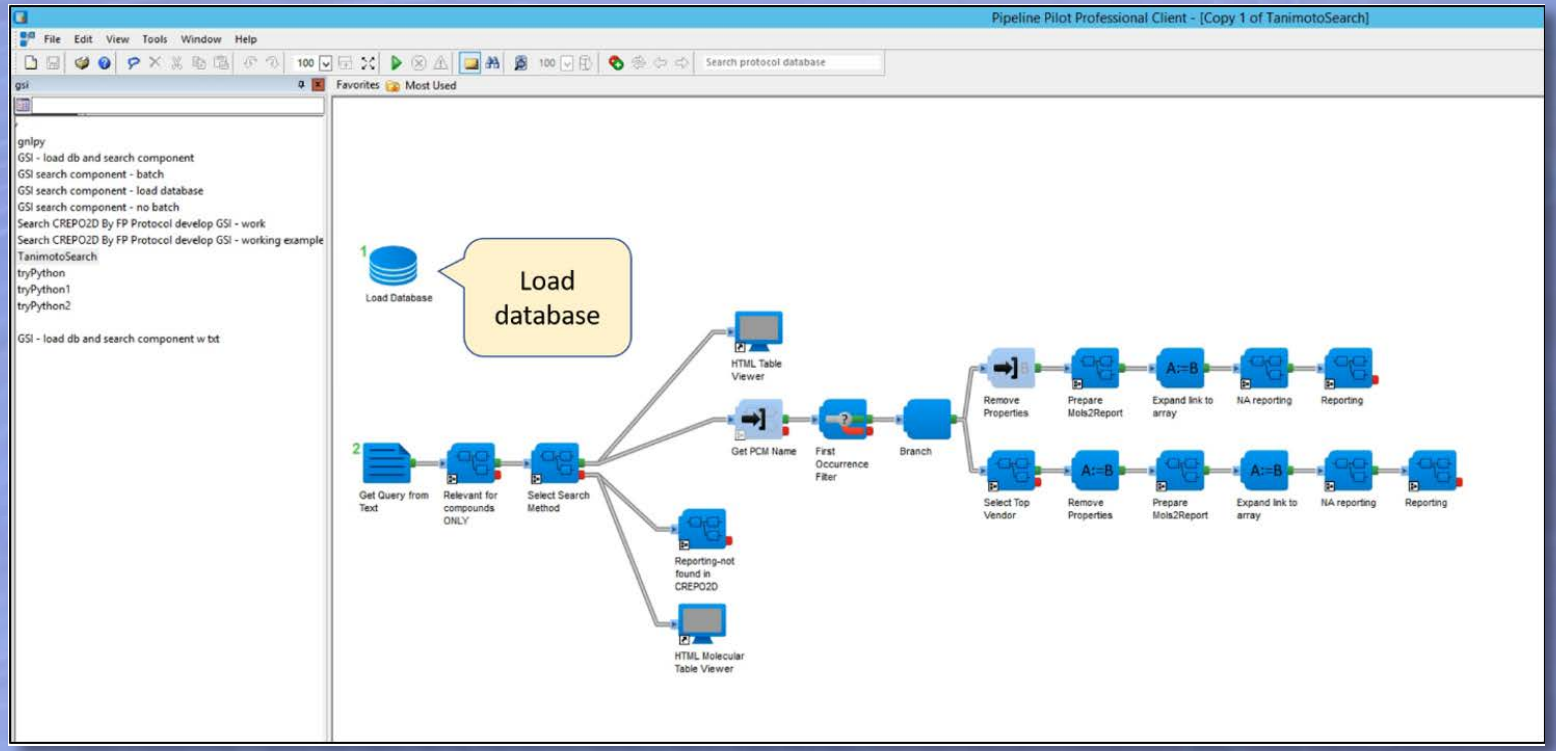


- Exact Nearest Neighbors For Small Vectors
- Approx For Large Vectors (Quantization)
- Large Scale Sim Search, FAISS

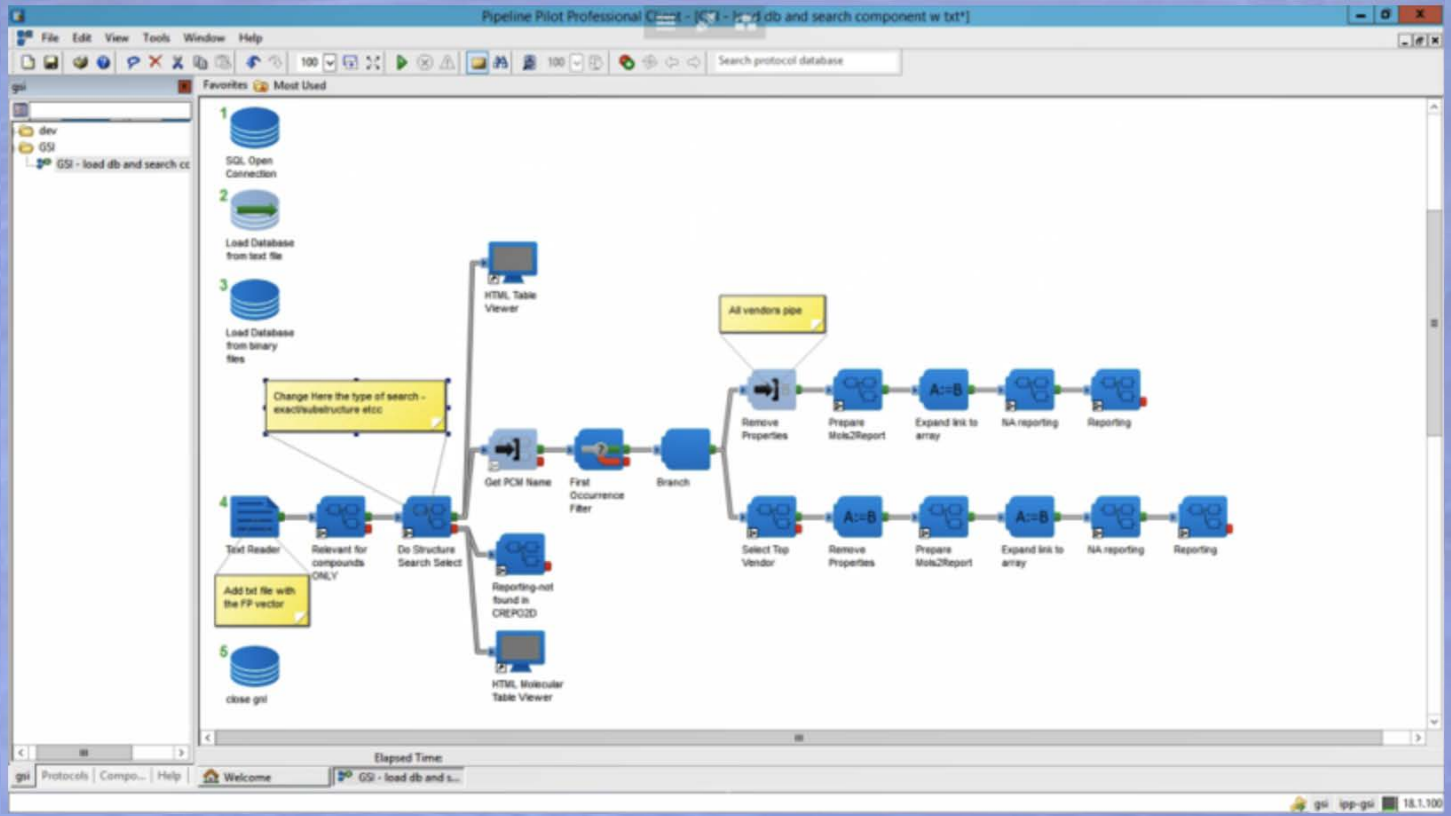
Biovia Application Integration

Biovia Application Used By Thousands of Bio-Tech Companies

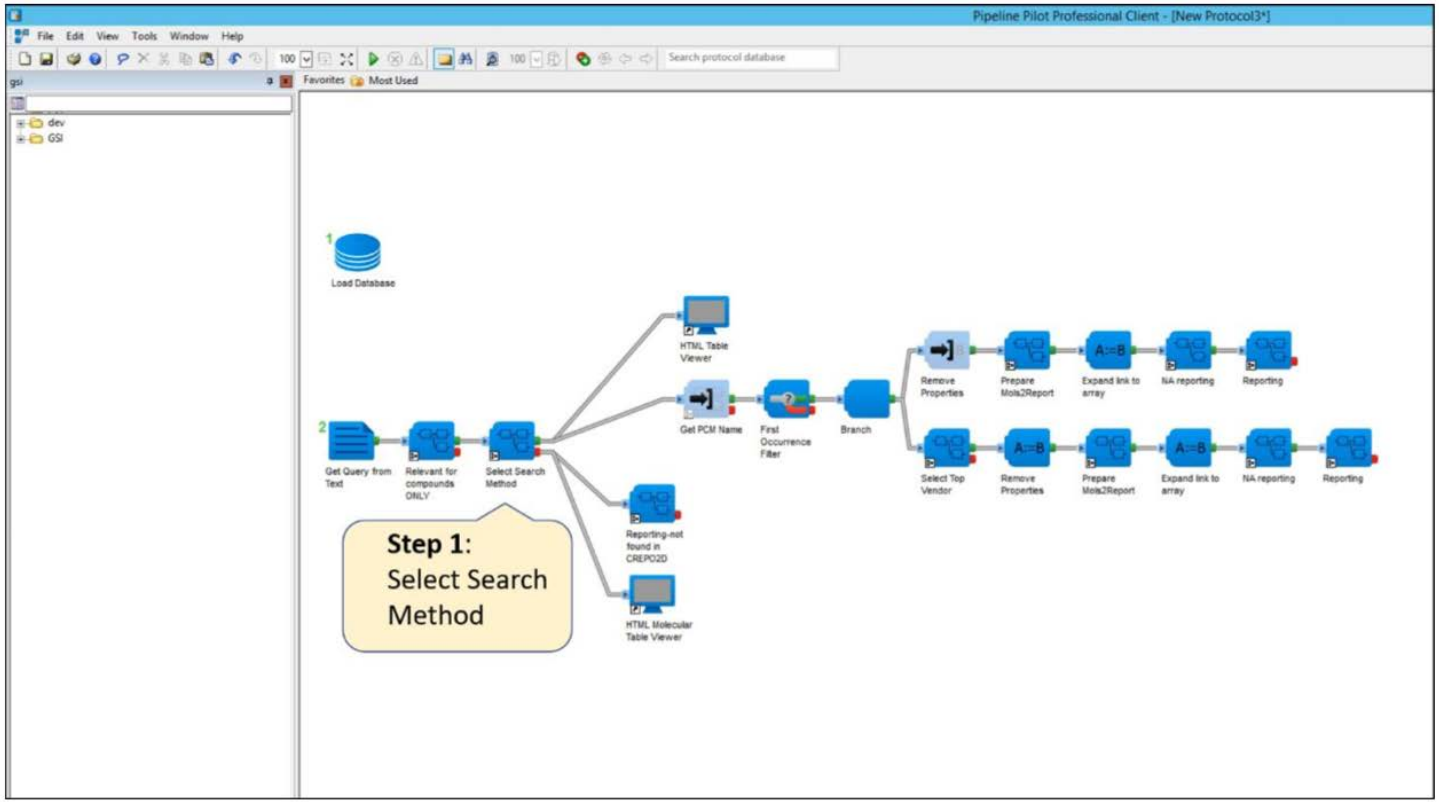
Weizmann: Load A Database



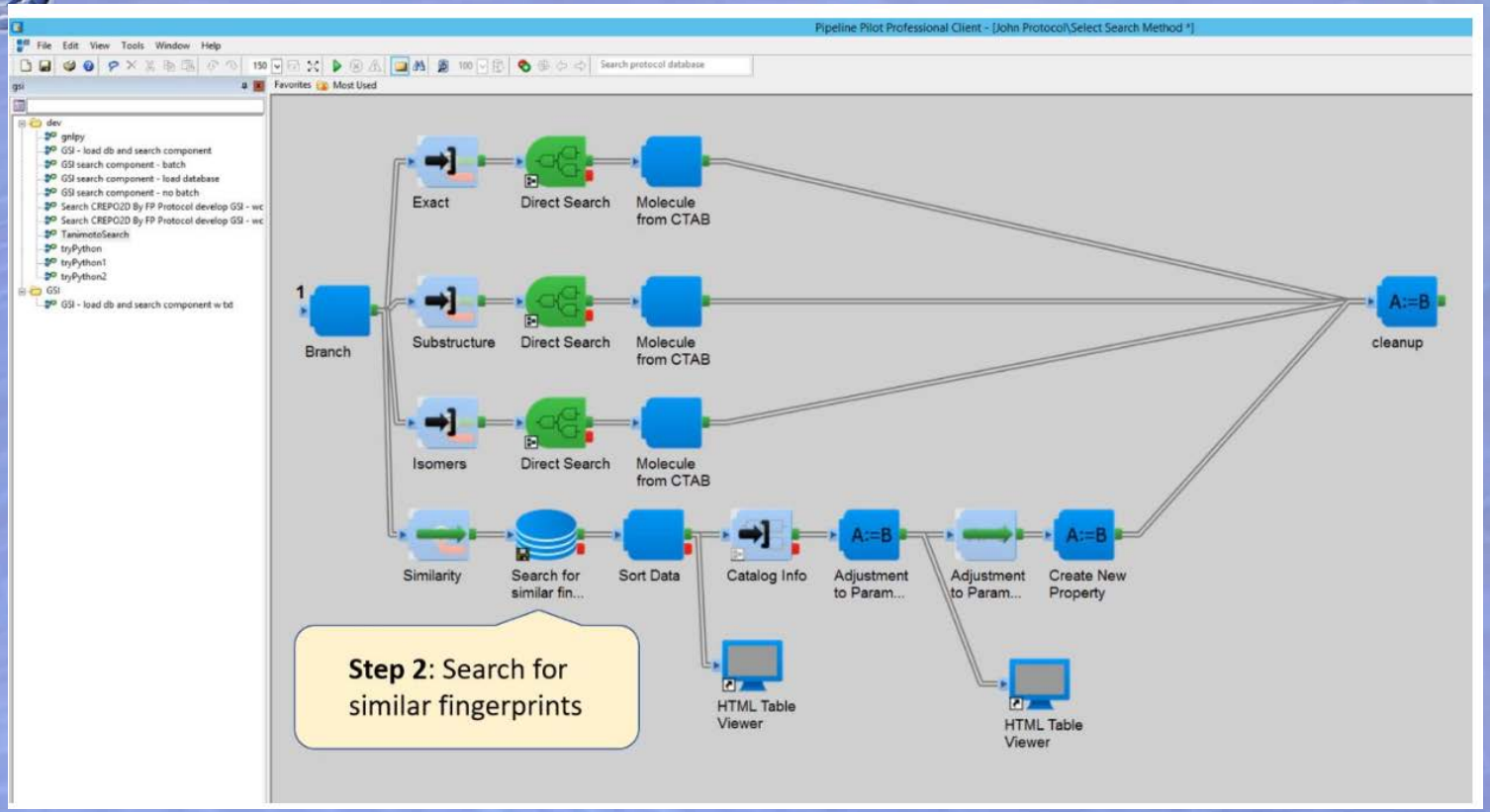
Weizmann: 3rd Party Search



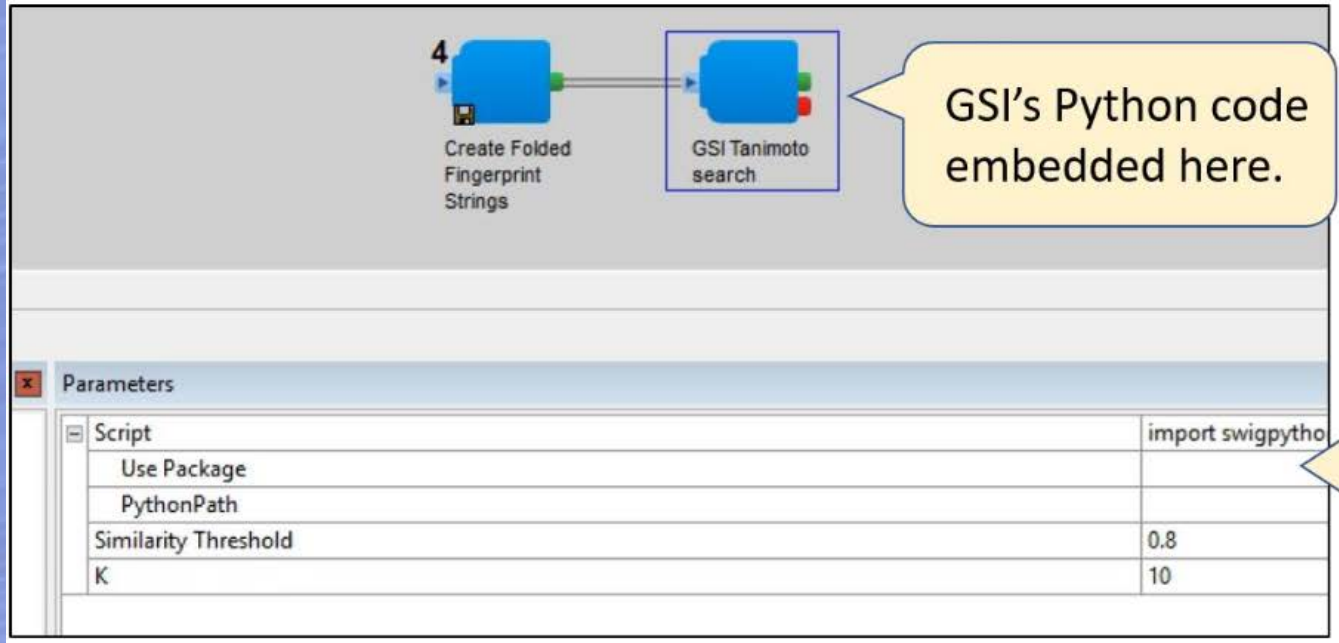
Weizmann: Select Search Method



Weizmann: Search



Weizmann: Define Parameters



GSI's Python code embedded here.

Step 3: Search parameters defined here, e.g. similarity threshold, K

Weizmann: Run Protocol