Nothing is for granted:
Making wise decisions using real-time intelligence

Anastasia Ailamaki

AiAS
DATA-INTENSIVE APPLICATIONS AND SYSTEMS

EPFL

Change means trouble

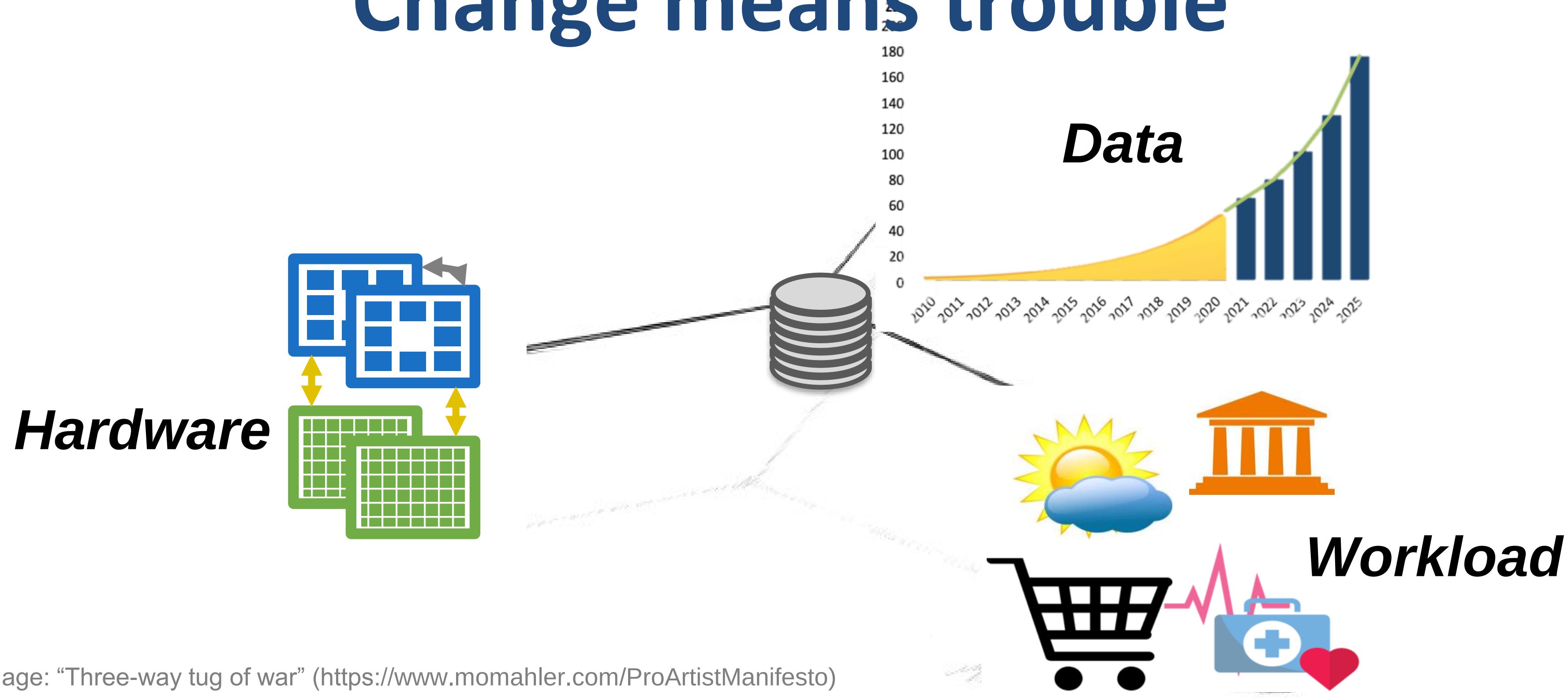


Image: “Three-way tug of war” (https://www.momahler.com/ProArtistManifesto)

Data management faces critical challenges

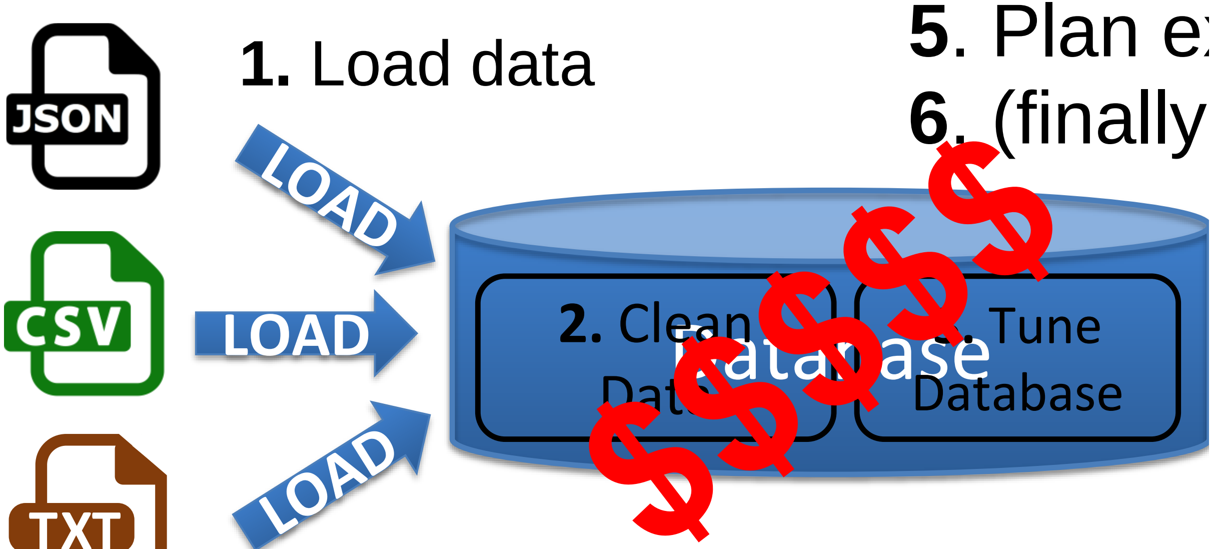
Preparation kills discovery

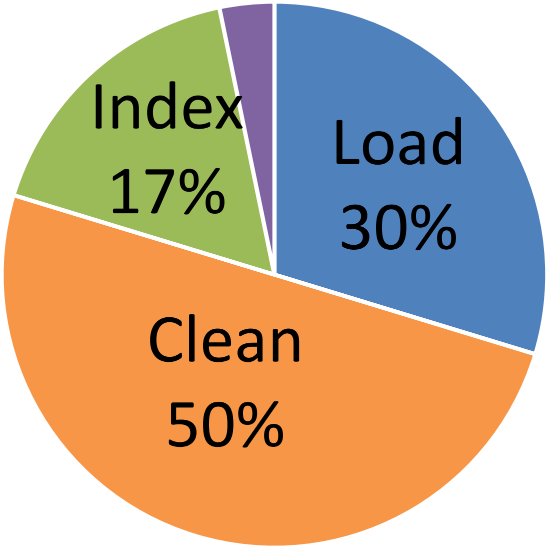
4. Ask analytical question

5. Plan e

6. (finally

1. Load data

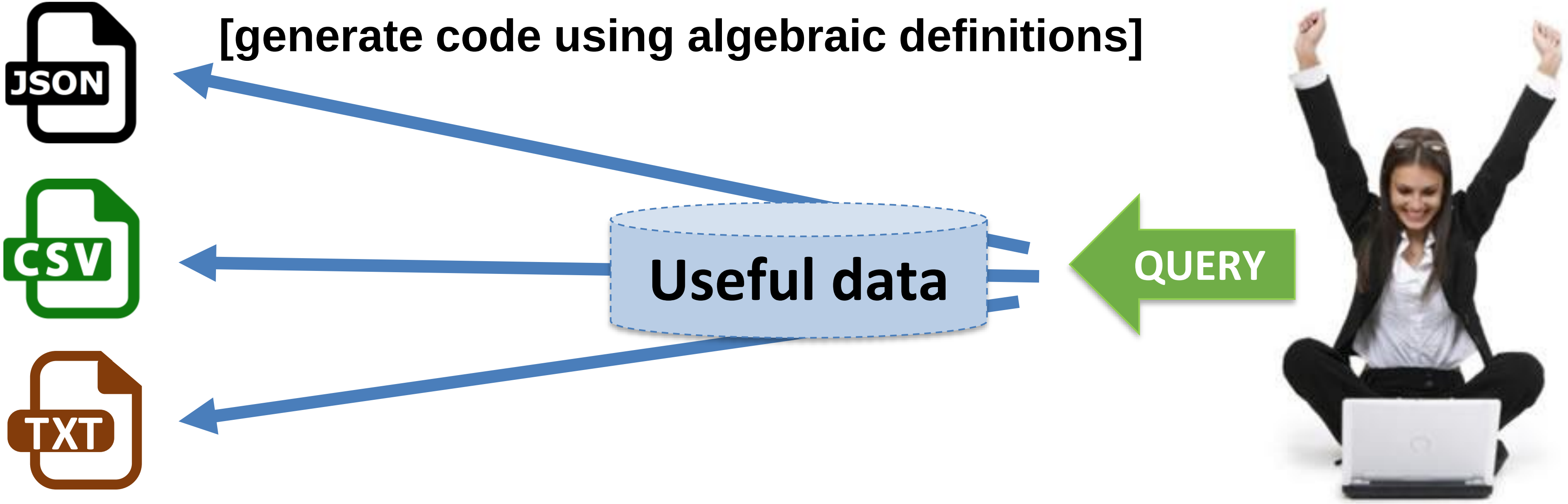




Cost grows with *owned* – not used! – data

Planning is expensive, often even wrong

“pay as you go”

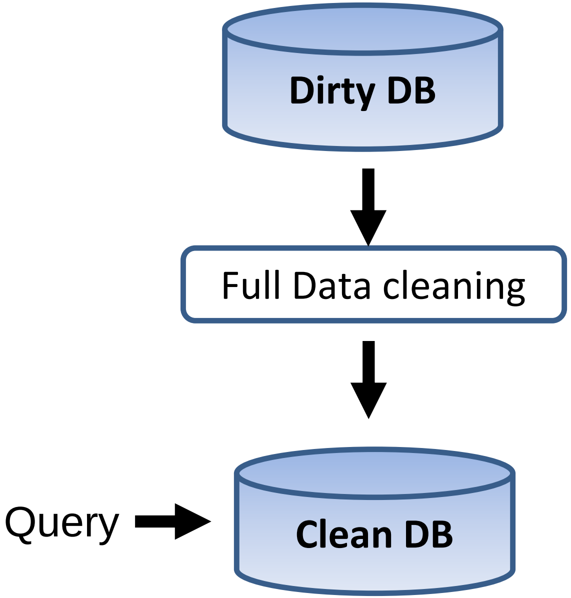


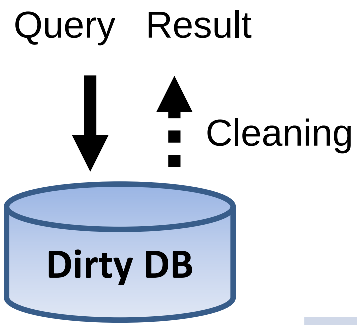
[generate code using algebraic definitions]

data-to-insight time = 0

prepare data *only when needed*

Query-driven data sanitization





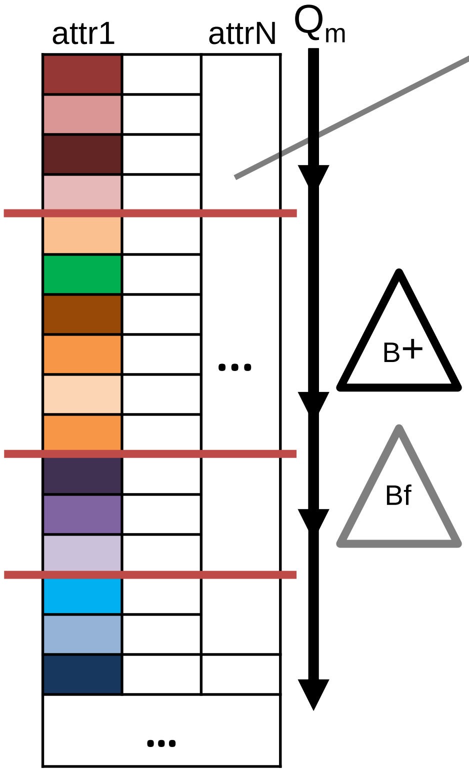
FD: Zip → City

Name	Zip	City
Jon	9001	Los Angeles
Jim	9001	San Francisco
Mary	10001	New York
Jane	10002	New York

Jon	9001	Los Angeles 50% San Francisco 50%
Jim	9001	San Francisco 50% Los Angeles 50%

Clean the useful subset of the data with probabilistic fixes

Evolving indexes



costs vs. gains
Should I build or not?

Data skipping

Fine-grained access path selection

Choose what to build & when

- Value-Existence (i.e., Bloom filters)
- Value-Position (i.e., B+ Trees)

Build / drop based on budget

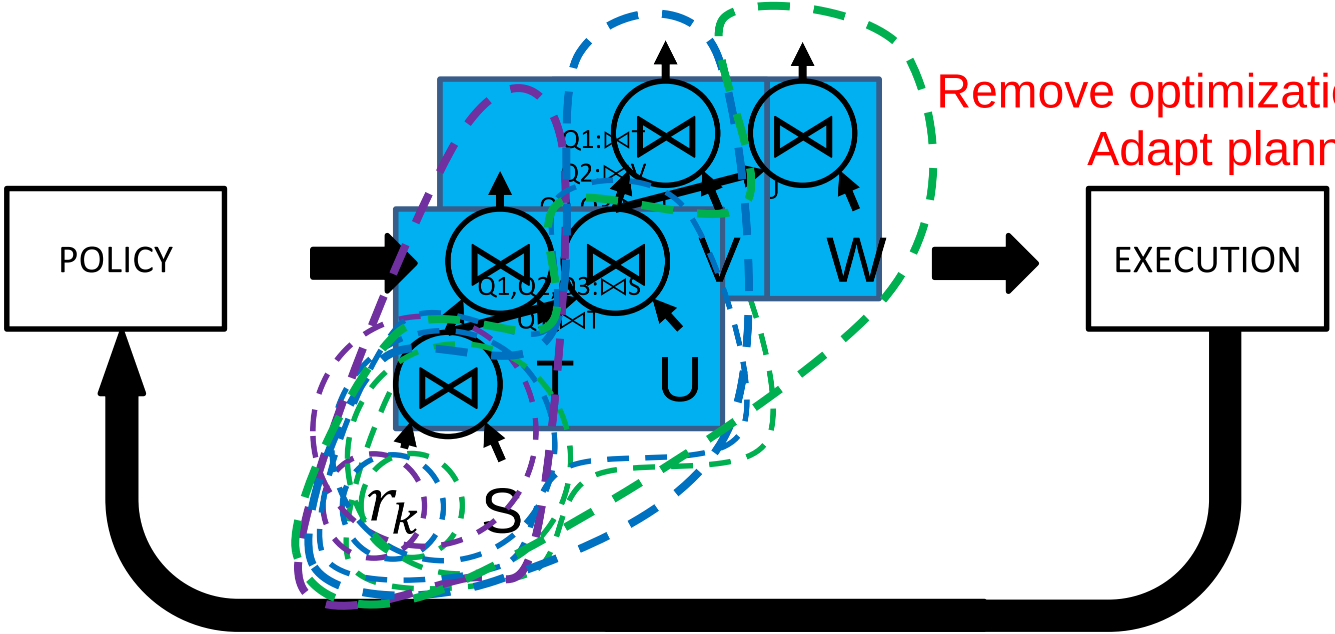
Max gain amortized by cost to build

Scaling sharing-aware optimization

Q1: select * from R, S, T where R.a=S.a and R.b=T.b

Q2: select * from R, S, U, V where R.a=S.a and S.e=U.e and S.f=V.f

Q3: select * from R, S, U, W where R.a=S.a and S.e=U.e and U.h=W.h



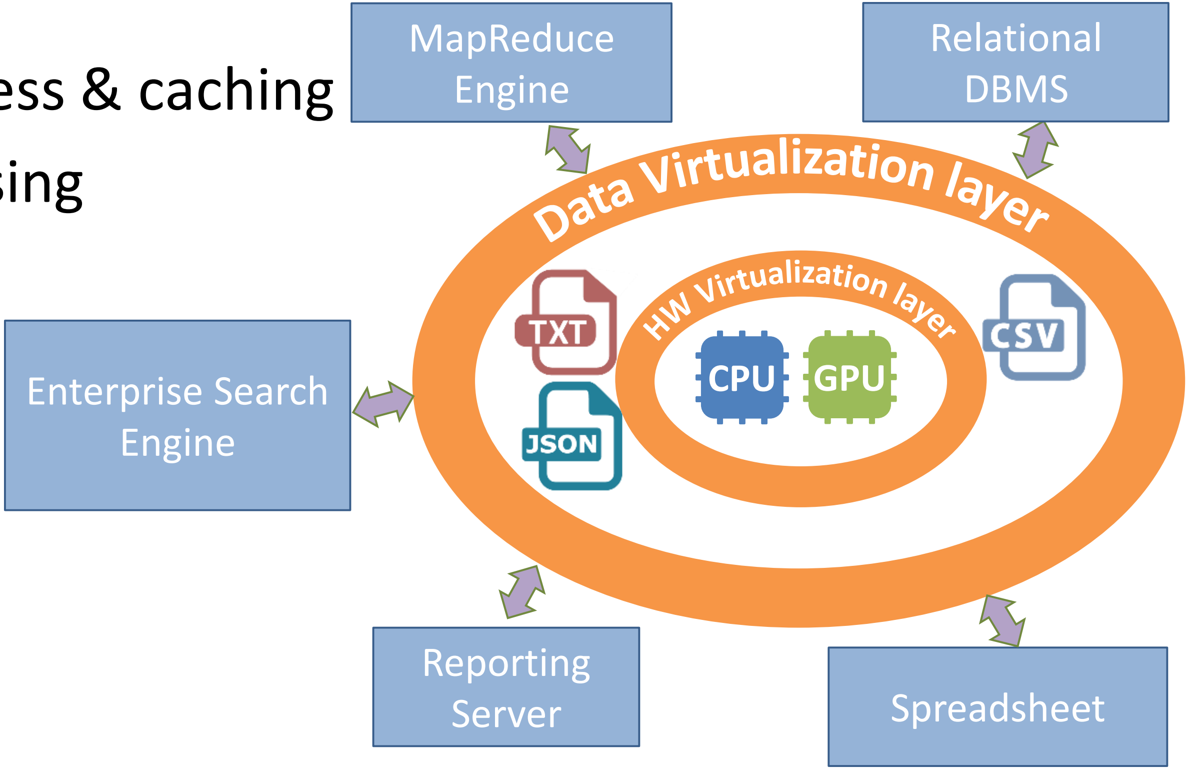
Remove optimization from critical path

Adapt planning heuristic

Timely planning through adaptation

Virtualization layers

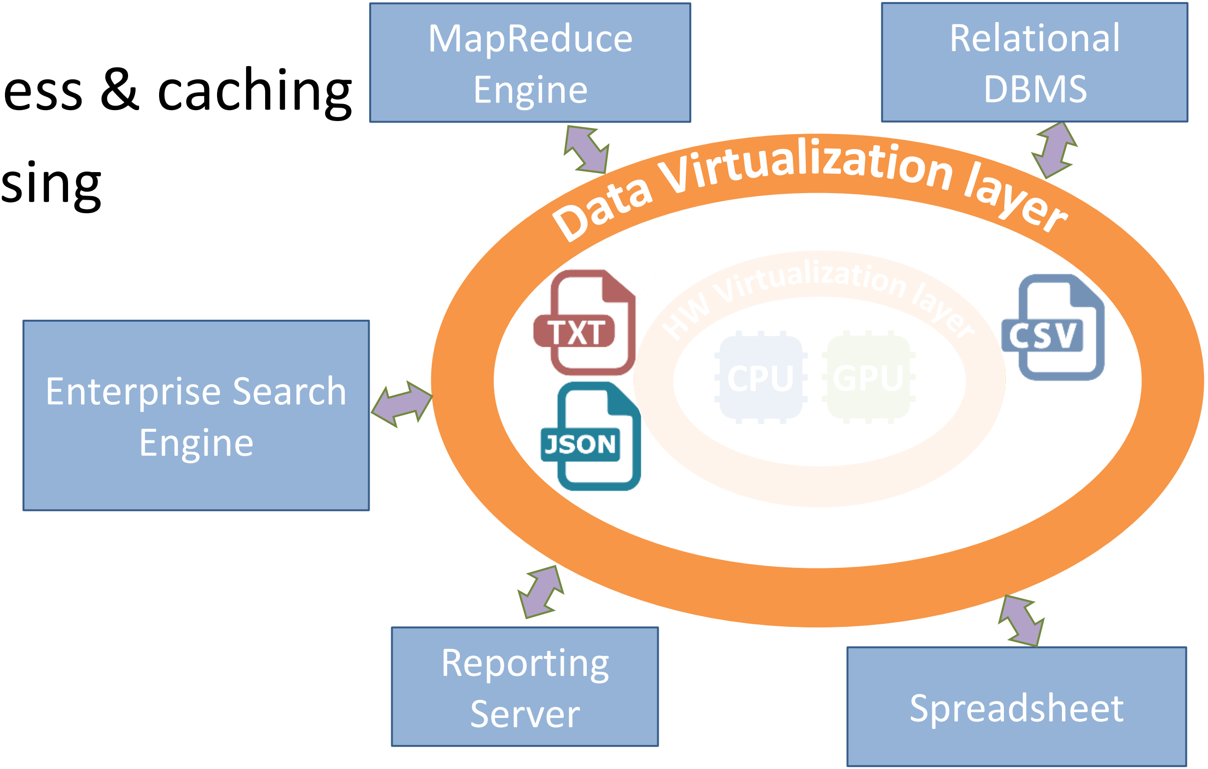
- Format impacts access & caching
- HW impacts processing



Virtualize format and hardware

Virtualization layers

- Format impacts access & caching
- HW impacts processing



MapReduce Engine

Relational DBMS

Enterprise Search Engine

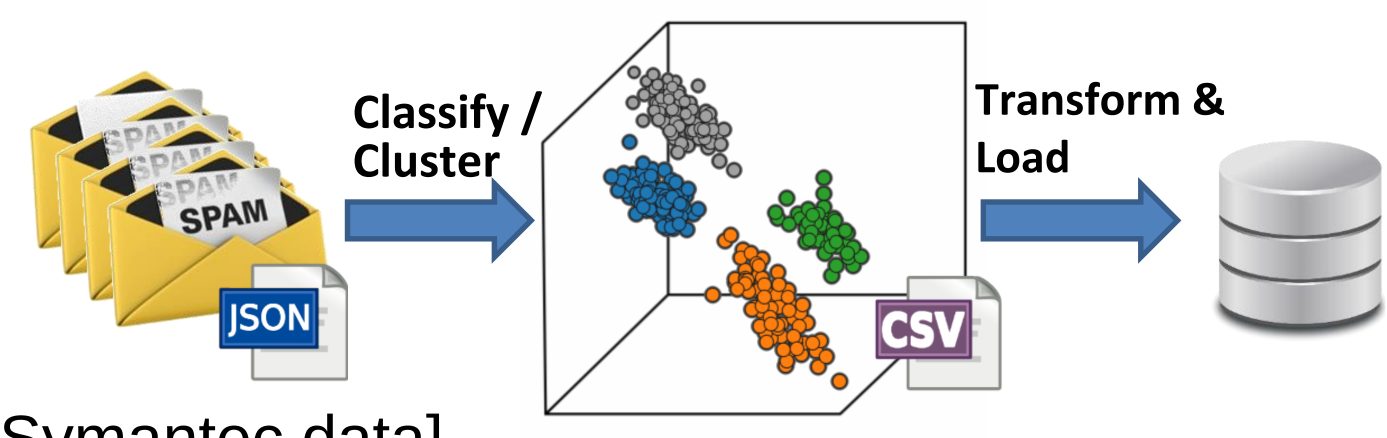
Reporting Server

Spreadsheet

Data Virtualization layer

Virtualize format and hardware

Detecting active spambots



Classify / Cluster

Transform & Load

[Symantec data]

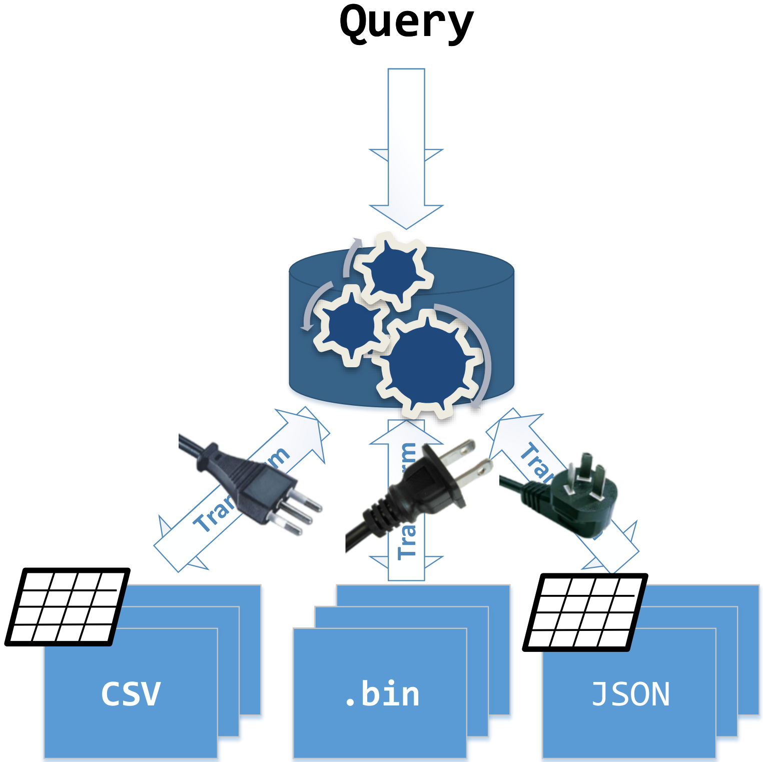
Flexibility

Ad-hoc queries
over diverse data formats

Performance

Fast queries
regardless of data format

Customizing data access layer



Traditional DBMS:
Data adapts to engine

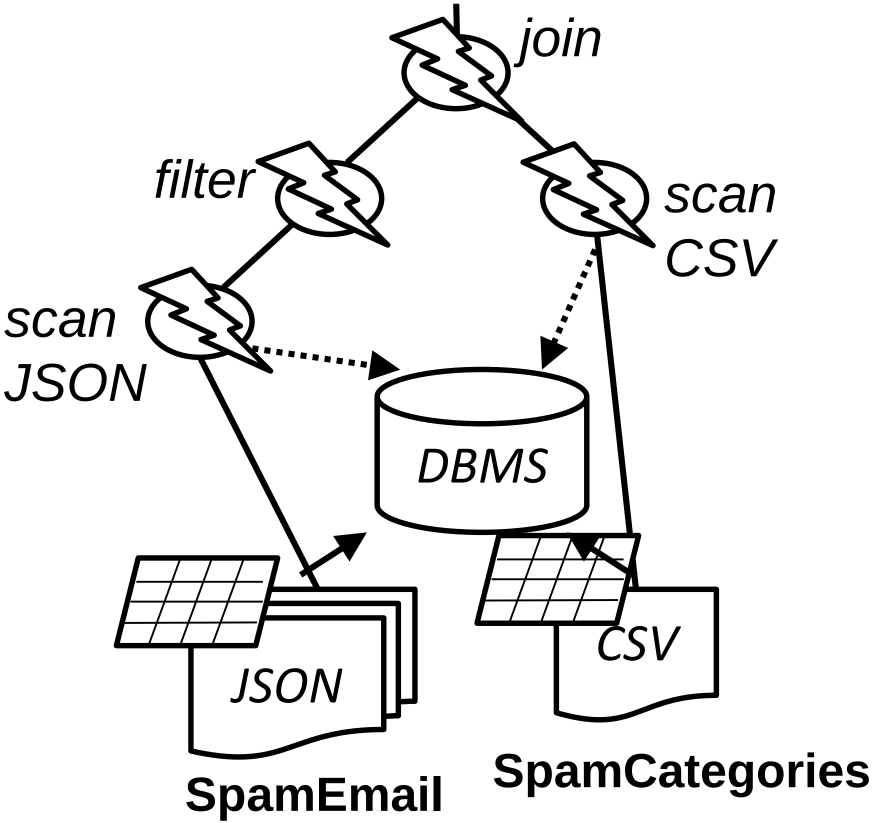
Proteus

Plug-in per data source
Build auxiliary structures

Treat each source as native storage format

How Proteus builds a just-in-time data base

```
SELECT bot, country, …
FROM SpamEmail e, SpamCategories c
WHERE e.id == c.id AND
            e.lang = ‘English’ AND …
```



⚡ Code Generate the Access Paths

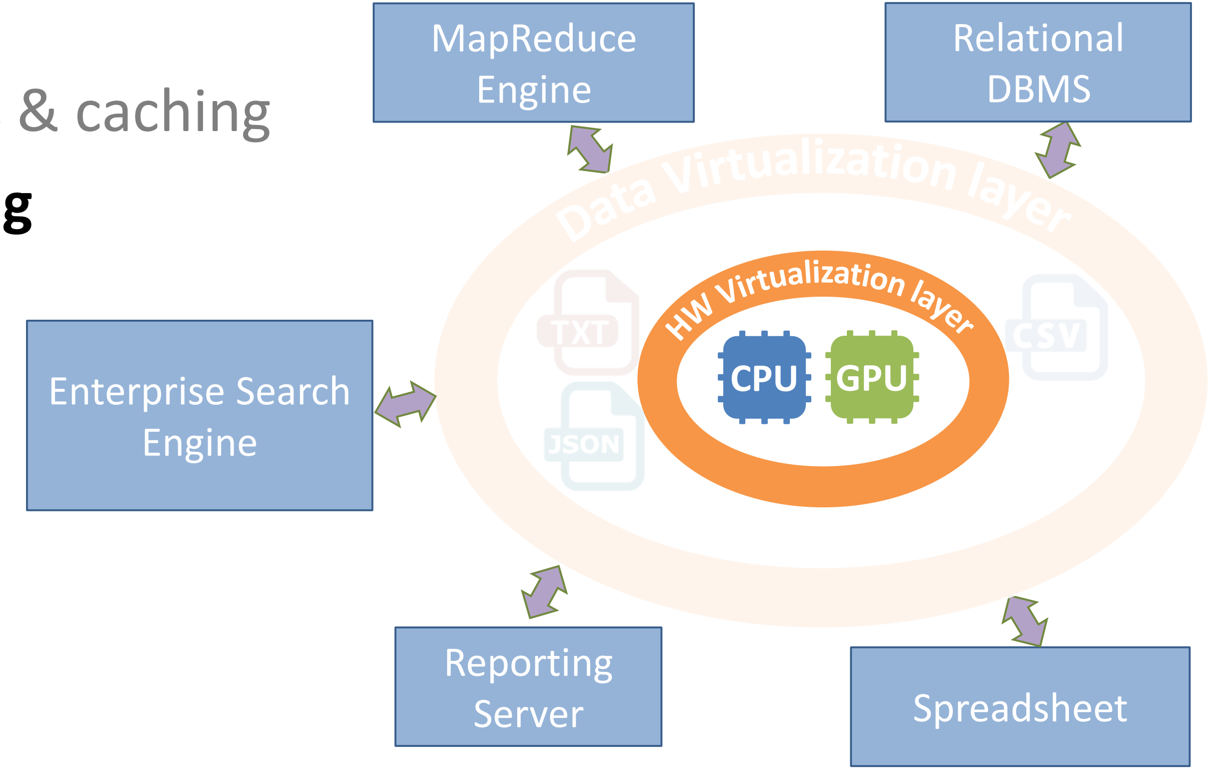
⚡ Code Generate the Query

▦ Build Position and Data Caches

Tame heterogeneity; Cache only useful data

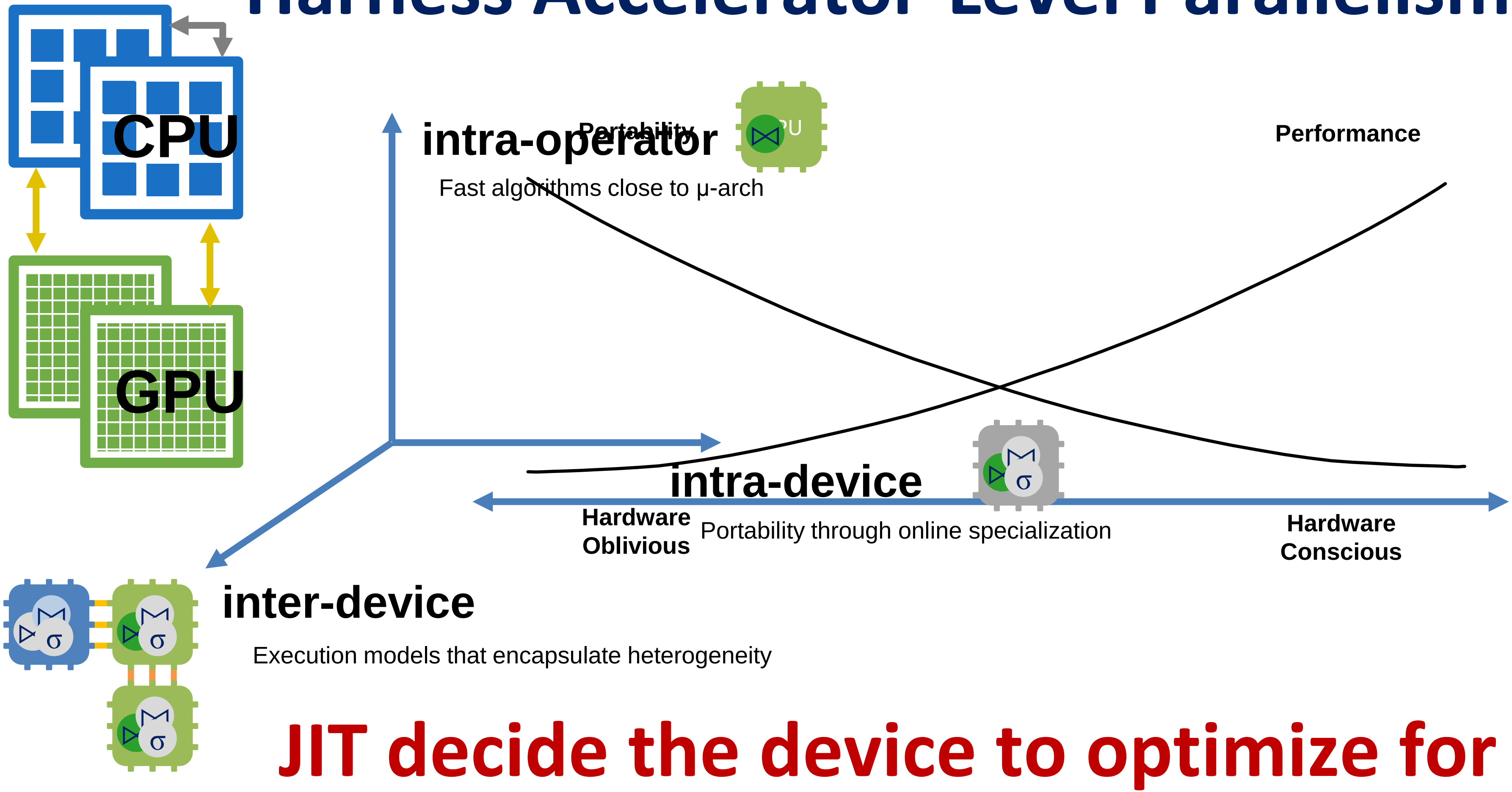
Virtualization layers

- Format impacts access & caching
- **HW impacts processing**



Virtualize format and hardware

Harness Accelerator-Level Parallelism



intra-operator

Portability

Fast algorithms close to μ-arch

Performance

intra-device

Hardware Oblivious

Portability through online specialization

Hardware Conscious

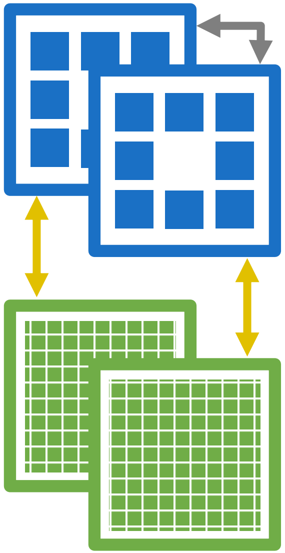
inter-device

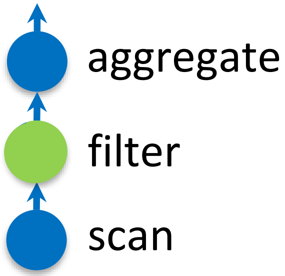
Execution models that encapsulate heterogeneity

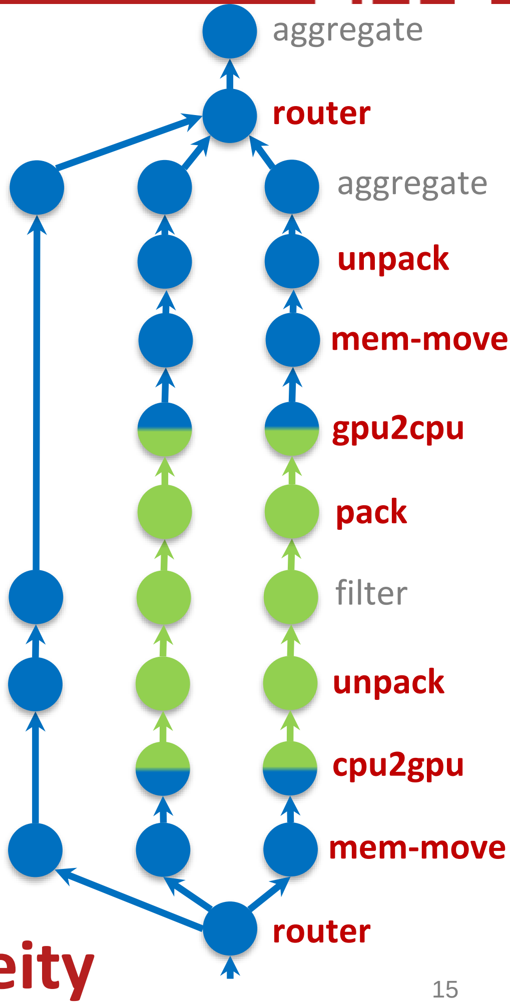
JIT decide the device to optimize for

Execution on CPU+GPU

- Decouple data- from control-flow
- Trait conversions



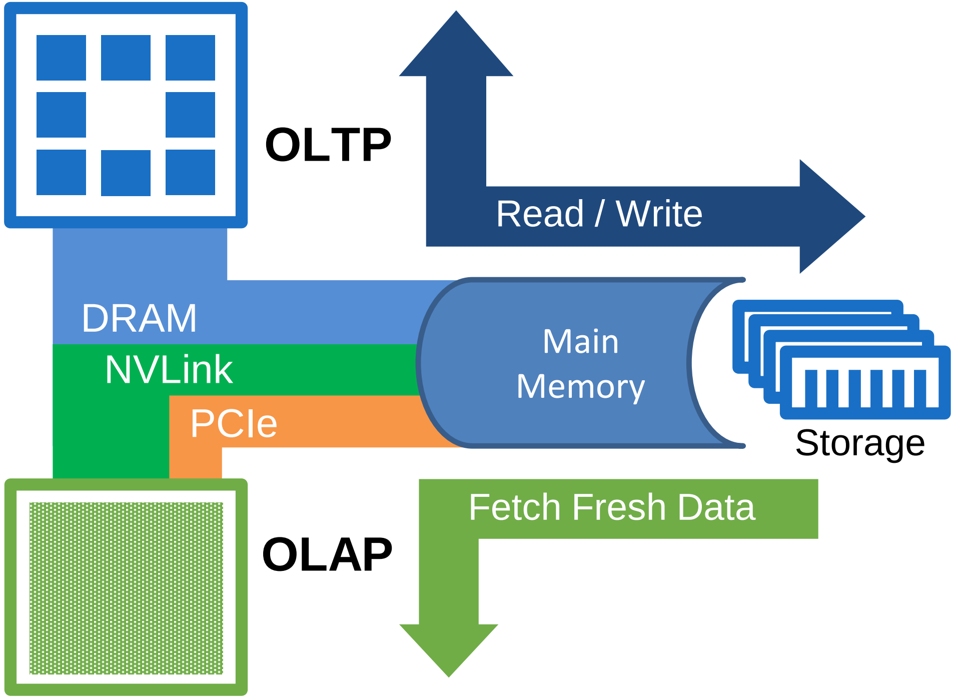




Operators encapsulate device heterogeneity

GPU Accesses Fresh Data from CPU Memory

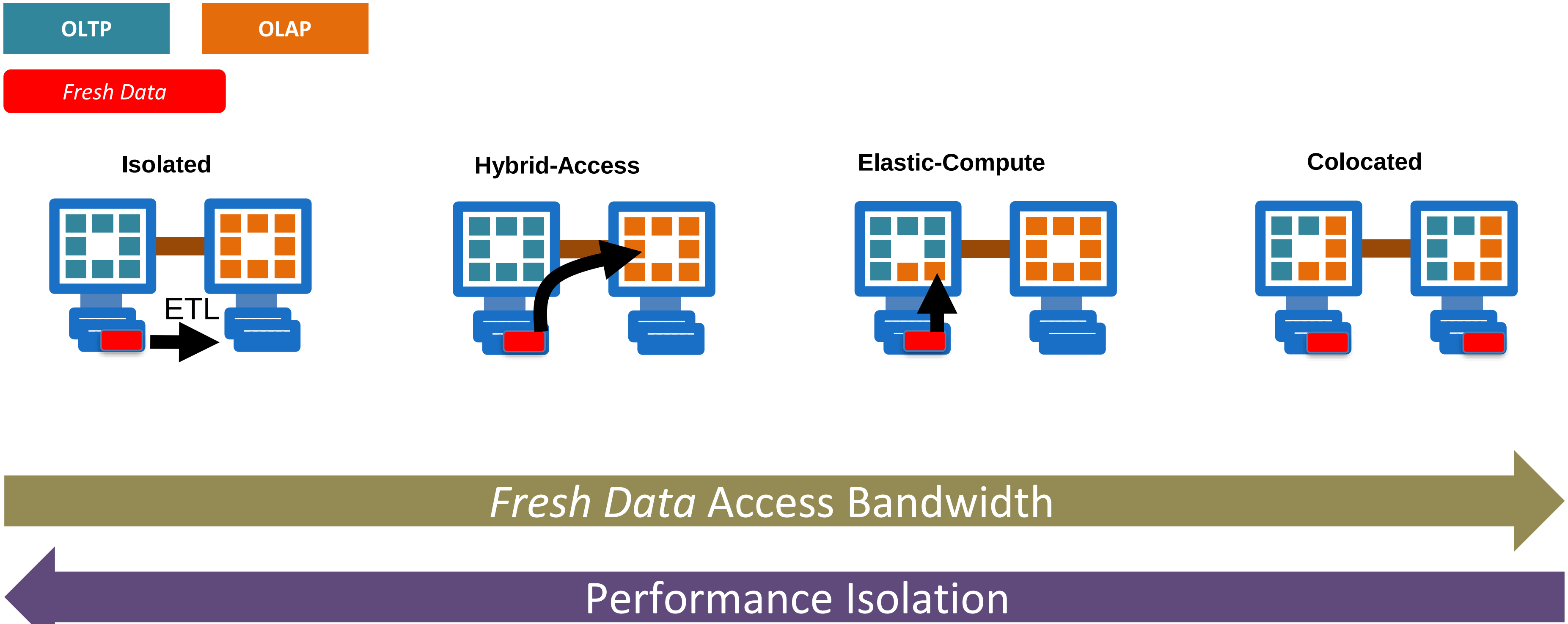
- OLTP generates fresh data on CPU Memory
- Data access protected by concurrency control
- OLAP needs to access fresh data over interconnect



Provide snapshot isolation for GPUs w/o CC overheads

Use shared main-memory bus efficiently

HTAP Design Spectrum [SIGMOD2020]



OLTP

OLAP

Fresh Data

Isolated

ETL

Hybrid-Access

Elastic-Compute

Colocated

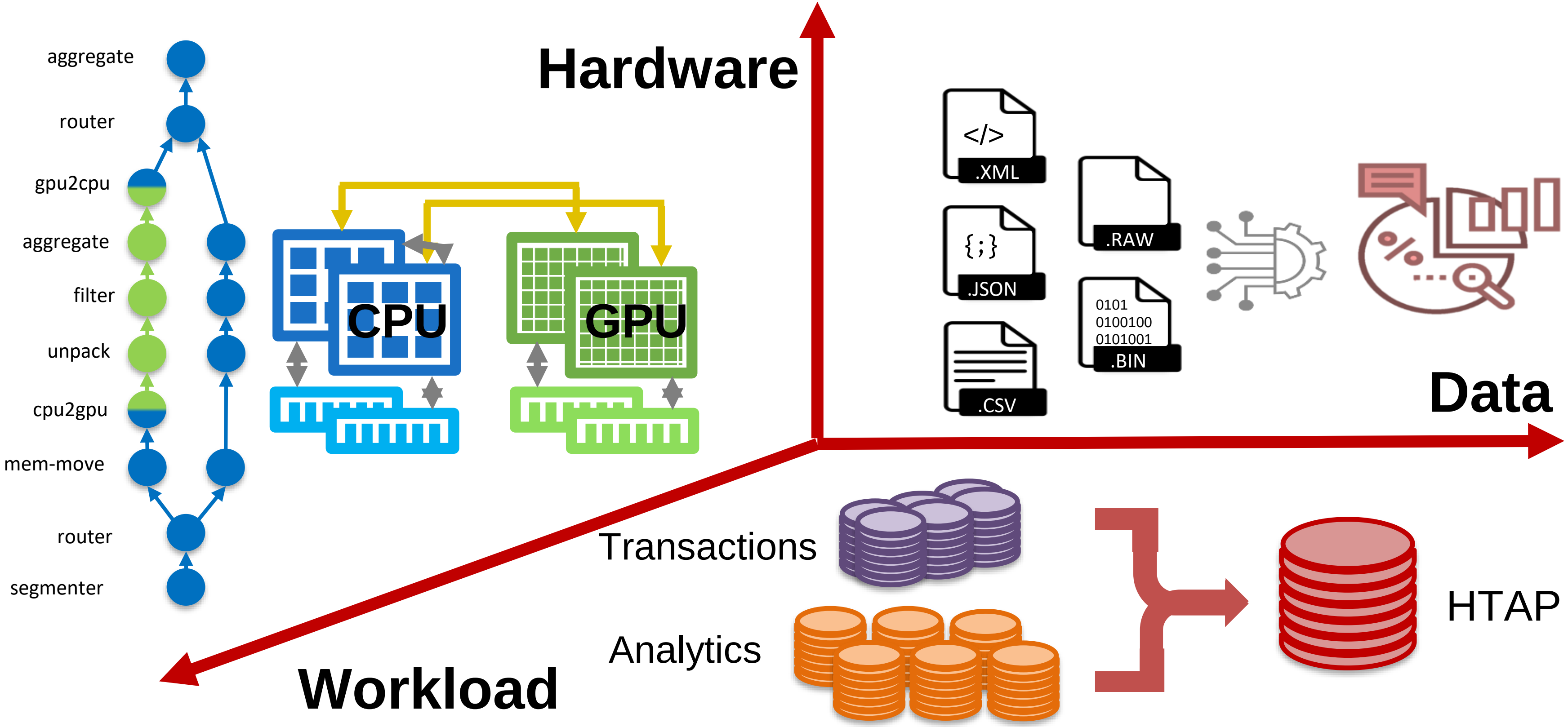
Fresh Data Access Bandwidth

Performance Isolation

Existing designs statically trades performance for isolation

Traverse HTAP spectrum based on *amount of fresh data*

Data virtualization and JIT engines



five old friends revisited

- ~~Data~~ variety → Operational environment variety
  - Unpredictable application requirements
- ~~Data~~ veracity → Inter-component veracity
  - Heterogeneous data & variable importance
- ~~Data~~ volume → Structural volume
  - Multi-layered system architectures
- ~~Data~~ value → Resource value
  - Broader, multi-featured analytics
- ~~Data~~ velocity → Technological velocity
  - Hardware heterogeneity & volatility

Intelligent systems to catch-up with an evolving landscape

Intelligent Real-time Systems

Incorporate change into native design.

Anticipate change and react, learning from errors.

A solution is only as efficient as its least adaptive component.

Reactive and optimistic
Imaginative and inspiring