# Physical Design

CHIDATA

CHIDATA

# Physical Design

How do we organize a "real" collection of data

**Last few lectures**: Single data type, simple list

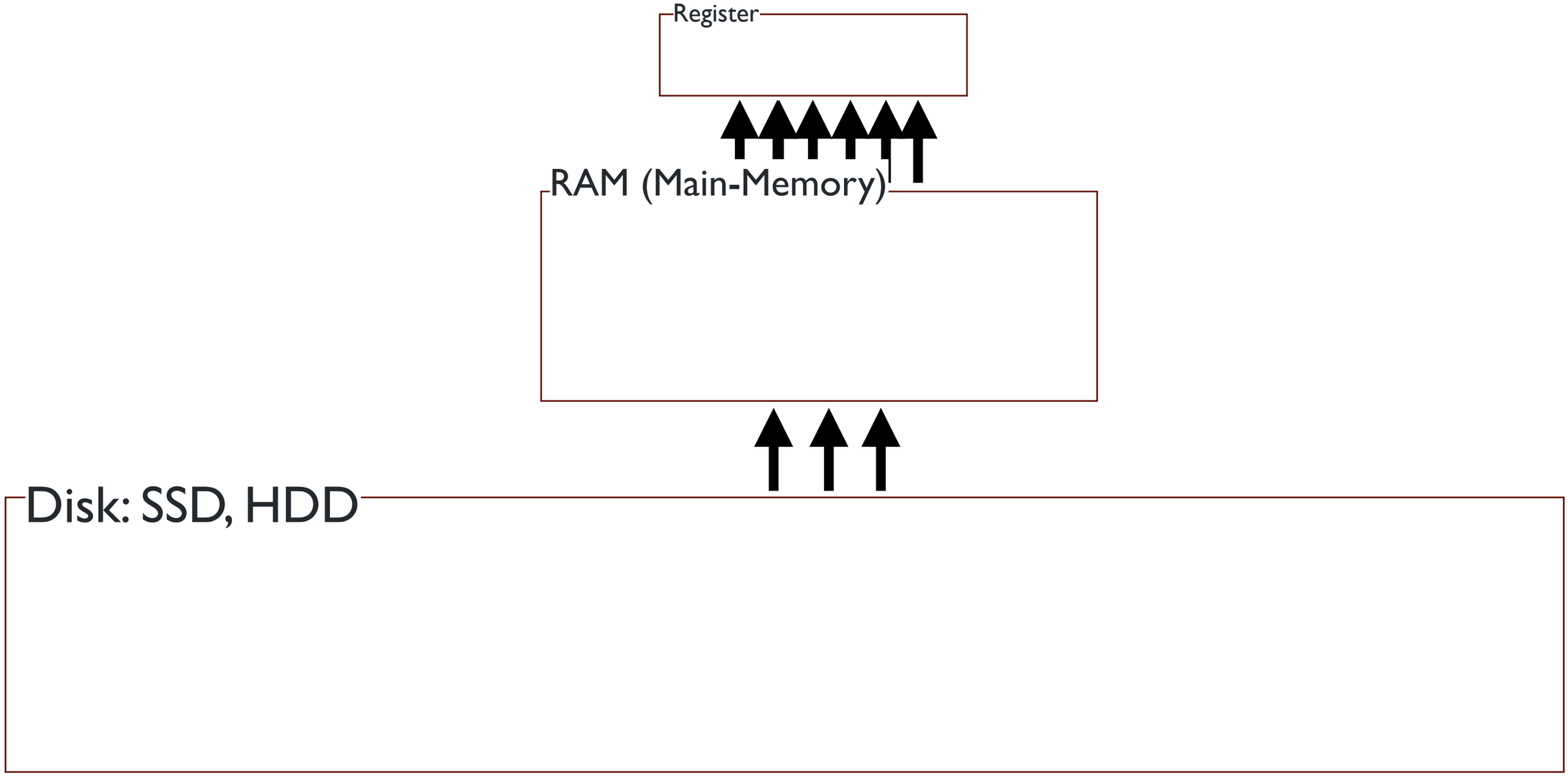
```
data = ['red', 'blue', 'red', 'red']
```

**This lecture**: Collection of "records"

| **Car_id** | **Make** | **Model** |
|---|---|---|
| 1 | Toyota | Camry |
| 2 | Ford | Fusion |
| 3 | Toyota | Corolla |
| 4 | Honda | Civic |

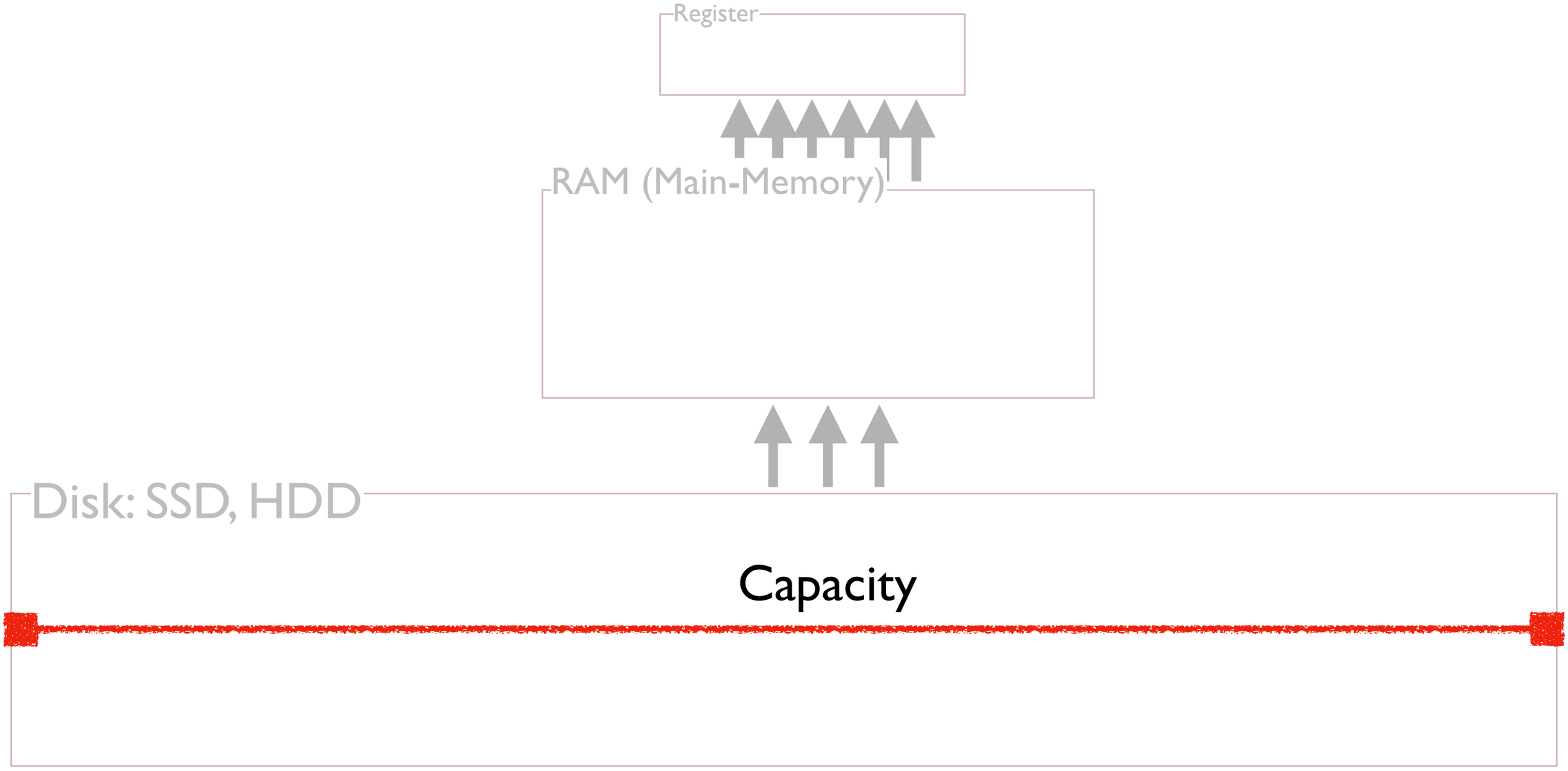
CHIDATA

# Schematic of Computer Storage



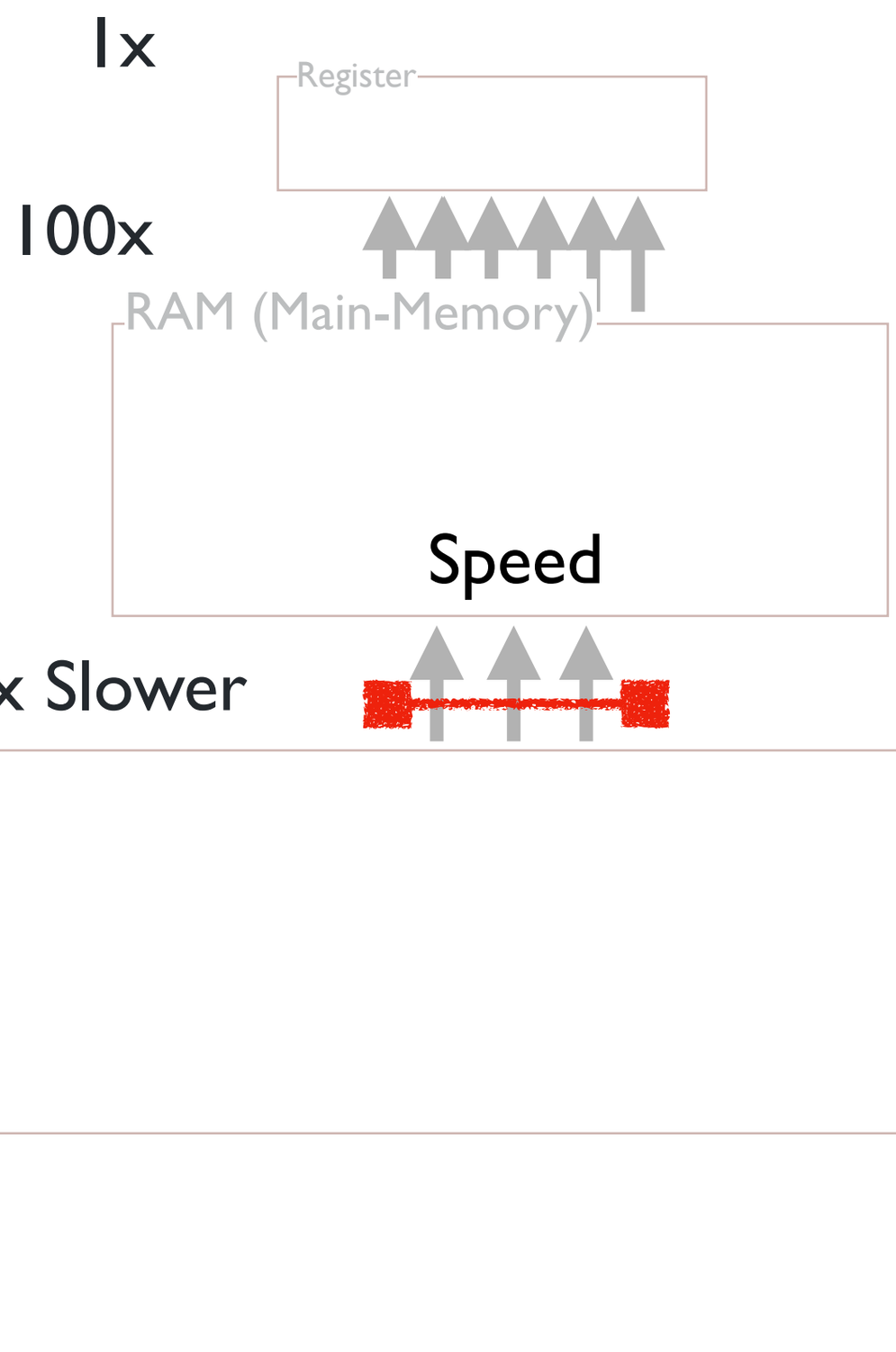
CHIDATA

# Schematic of Computer Storage



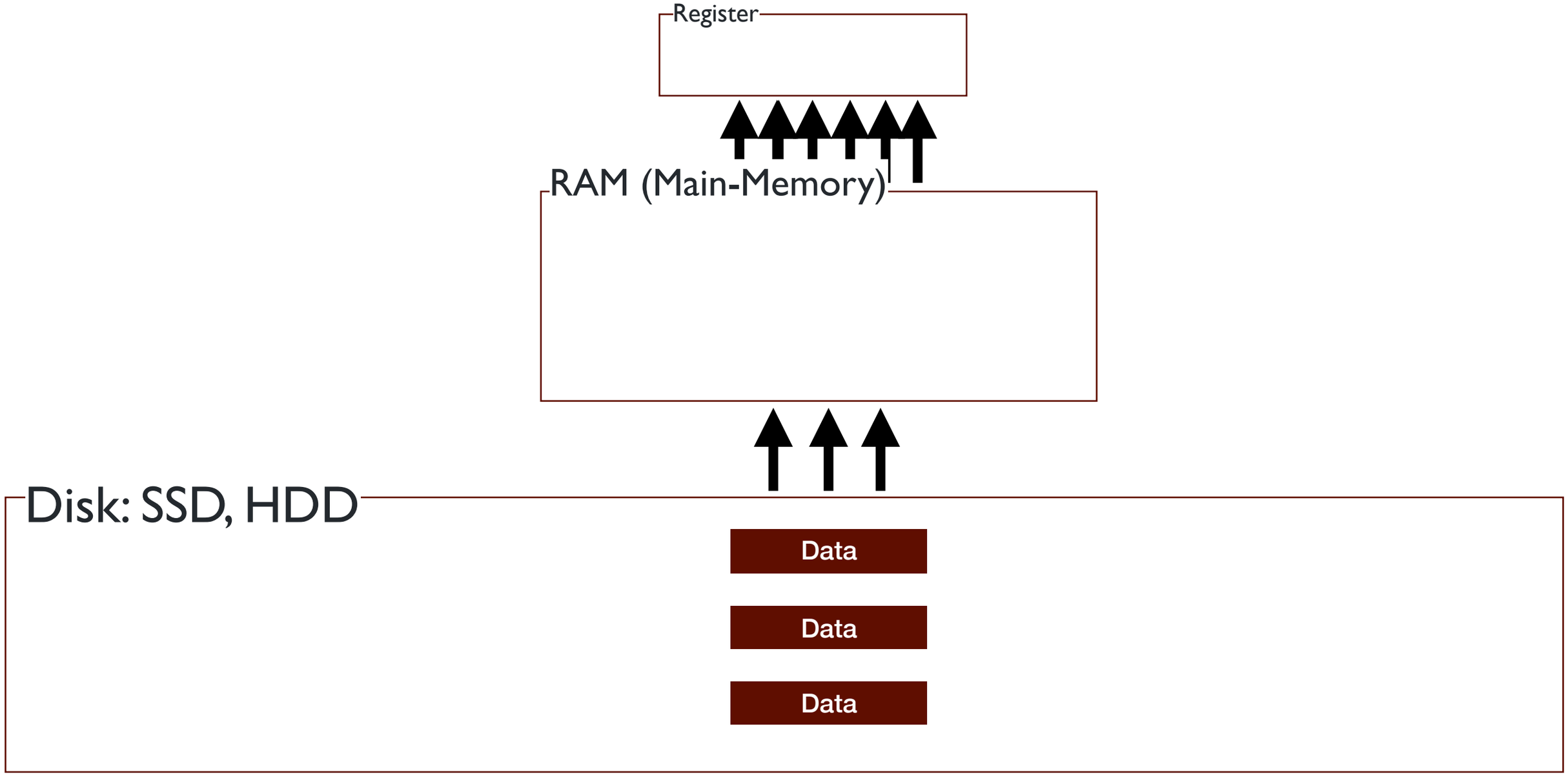
CHIDATA

# Schematic of Computer Storage



CHIDATA

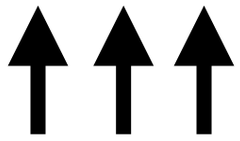
# Data Must Move For Analysis



CHIDATA

# Physical Design Objective

Organize data so you avoid moving data that you don't need.



Disk: SSD, HDD

Block 1 | Block 2 | ... | Block N-1 | Block N

Think about data as stored in blocks

CHIDATA

# Row-Oriented Storage

Store data as a collection of "records"

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 2 | Ford | Fusion |
| 3 | Toyota | Corolla |
| 4 | Honda | Civic |

## Disk: SSD, HDD

**Block 1**

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 2 | Ford | Fusion |

*Min: 1, Max: 2*

**Block 2**

| Car_id | Make | Model |
|---|---|---|
| 3 | Toyota | Corolla |
| 4 | Honda | Civic |

*Min: 3, Max: 4*

# Row-Oriented Storage

Task: read record with id 2

I/O Cost: **Size of a block**

Disk: SSD, HDD

**Block 1**

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 2 | Ford | Fusion |

*Min: 1, Max: 2*

**Block 2**

| Car_id | Make | Model |
|---|---|---|
| 3 | Toyota | Corolla |
| 4 | Honda | Civic |

*Min: 3, Max: 4*

CHIDATA

# Row-Oriented Storage

Task: count records with make = 'Toyota'

I/O Cost: **Size of all blocks**

Disk: SSD, HDD

**Block 1**

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 2 | Ford | Fusion |

**Block 2**

| Car_id | Make | Model |
|---|---|---|
| 3 | Toyota | Corolla |
| 4 | Honda | Civic |

CHIDATA

# Row-Oriented Storage

Task: read record with id 2

I/O Cost: **Size of a block**

Pretty much as good as it gets

Task: count records with make = ‘Toyota’

I/O Cost: **Size of all blocks**

Hmm…..

CHIDATA

Sort the data on “Make”

| Car_id | Make | Model |
|---|---|---|
| 2 | Ford | Fusion |
| 4 | Honda | Civic |
| 1 | Toyota | Camry |
| 3 | Toyota | Corolla |

## Disk: SSD, HDD

**Block 1**

| Car_id | Make | Model |
|---|---|---|
| 2 | Ford | Fusion |
| 4 | Honda | Civic |

*Min: Ford, Max: Honda*

**Block 2**

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 3 | Toyota | Corolla |

*Min:Toyota, Max:Toyota*

CHIDATA

# Let's try that again

Task: count records with make = 'Toyota'

| Car_id | Make | Model |
|---|---|---|
| 2 | Ford | Fusion |
| 4 | Honda | Civic |
| 1 | Toyota | Camry |
| 3 | Toyota | Corolla |

## Disk: SSD, HDD

**Block 1**

| Car_id | Make | Model |
|---|---|---|
| 2 | Ford | Fusion |
| 4 | Honda | Civic |

*Min: Ford, Max: Honda*

**Block 2**

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 3 | Toyota | Corolla |

*Min: Toyota, Max: Toyota*

CHIDATA

# Uh-oh!

Task: read record with id 2

| Car_id | Make | Model |
|---|---|---|
| 2 | Ford | Fusion |
| 4 | Honda | Civic |
| 1 | Toyota | Camry |
| 3 | Toyota | Corolla |

## Disk: SSD, HDD

**Block 1**

| Car_id | Make | Model |
|---|---|---|
| 2 | Ford | Fusion |
| 4 | Honda | Civic |

*Min: Ford, Max: Honda*

**Block 2**

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 3 | Toyota | Corolla |

*Min: Toyota, Max: Toyota*

CHIDATA

# Sort on "Make"

Task: read record with id 2

I/O Cost: **Size of all blocks (expected to see half blocks)**

Hmm….

Task: count records with make = 'Toyota'

I/O Cost: **Size of all blocks that contain Toyota**

Pretty much as good as it gets

CHIDATA

# Row-Oriented Layouts

Blocks are collections of roughly the same amount of records

Sorting allows us to filter on particular attributes: need to select these before hand.

CHIDATA

# Insertions

Sort the data on "Make"

| Car_id | Make | Model |
|---|---|---|
| 2 | Ford | Fusion |
| 4 | Honda | Civic |
| 1 | Toyota | Camry |
| 3 | Toyota | Corolla |
| 5 | Ford | Escape |

## Disk: SSD, HDD

**Block 1**

| Car_id | Make | Model |
|---|---|---|
| 2 | Ford | Fusion |
| 4 | Honda | Civic |

*Min: Ford, Max: Honda*

**Block 2**

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 3 | Toyota | Corolla |

*Min:Toyota, Max:Toyota*

CHIDATA

# Insertions

Sort the data on "Make"

| Car_id | Make | Model |
|---|---|---|
| 2 | Ford | Fusion |
| 4 | Honda | Civic |
| 1 | Toyota | Camry |
| 3 | Toyota | Corolla |
| 5 | Ford | Escape |

## Disk: SSD, HDD

**Block 1**

| Car_id | Make | Model |
|---|---|---|
| 2 | Ford | Fusion |
| 5 | Ford | Fusion |

*Min: Ford, Max: Ford*

**Block 2**

| Car_id | Make | Model |
|---|---|---|
| 4 | Honda | Civic |
| 1 | Toyota | Camry |

*Min: Honda, Max: Toyota*

**Block 3**

| Car_id | Make | Model |
|---|---|---|
| 3 | Toyota | Corolla |

*Min: Toyota, Max: Toyota*

CHIDATA

# Row-Oriented Layouts

Blocks are collections of roughly the same amount of records

Sorting allows us to filter on particular attributes: need to select these before hand.

Sorting: insertions are bad*

* appends are ok (know all new records are >= max val).

CHIDATA

# An Alternate Approach?

Task: count records with make = 'Toyota'

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 2 | Ford | Fusion |
| 3 | Toyota | Corolla |
| 4 | Honda | Civic |

## Disk: SSD, HDD

**Block 1**

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 2 | Ford | Fusion |

**Block 2**

| Car_id | Make | Model |
|---|---|---|
| 3 | Toyota | Corolla |
| 4 | Honda | Civic |

CHIDATA

# An Alternate Approach?

Task: count records with make = 'Toyota'

Disk: SSD, HDD

**Block 1**

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 2 | Ford | Fusion |

**Block 2**

| Car_id | Make | Model |
|---|---|---|
| 3 | Toyota | Corolla |
| 4 | Honda | Civic |

Note: Each block contains 4 strings

CHIDATA

# Columnar Storage

Task: count records with make = 'Toyota'

| Car_id | Make | Model |
|---|---|---|
| 1 | Toyota | Camry |
| 2 | Ford | Fusion |
| 3 | Toyota | Corolla |
| 4 | Honda | Civic |

## Disk: SSD, HDD

**Block 1**

| Make |
|---|
| Toyota |
| Ford |
| Toyota |
| Honda |

**Block 2**

| Model |
|---|
| Camry |
| Fusion |
| Corolla |
| Civic |

CHIDATA

# Columnar Storage

Task: count records with make = 'Toyota'

I/O Cost: **Size of all "Make" blocks**

Disk: SSD, HDD

**Block 1**

| **Make** |
|---|
| Toyota |
| Ford |
| Toyota |
| Honda |

**Block 2**

| **Model** |
|---|
| Camry |
| Fusion |
| Corolla |
| Civic |

Can dictionary encode and compress each block to fit more!

# Agressive Compression

## Dictionary Encoding

Disk: SSD, HDD

**Block 1**

| Make |
|---|
| “00” |
| “01” |
| “00” |
| “00” |
| “11” |

**Block 2**

| Model |
|---|
| “00” |
| “01” |
| “10” |
| “10” |
| “11” |

~~Task: count records with make = ‘Toyota’~~

Task: count records with make = ’00’

Can fit much more relevant data (effectively) in one block.

# Columnar Storage

Task: read record with id 2

I/O Cost: **Size of all blocks**

Disk: SSD, HDD

**Block 1**

| **Make** |
|---|
| Toyota |
| Ford |
| Toyota |
| Honda |

**Block 2**

| **Model** |
|---|
| Camry |
| Fusion |
| Corolla |
| Civic |

Makes sense when you are interested in slicing or aggregating along columns!

CHIDATA

# Column-Oriented Layouts

Blocks are collections of columns of data

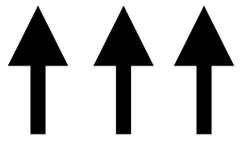
Efficient aggregation along certain columns

Enables aggressive compression (similar to our previous lectures)

CHIDATA

# Physical Design Objective

Organize data so you avoid moving data that you don't need.



Disk: SSD, HDD

Many modern systems use a hybrid of row and columnar storage

CHIDATA

# Evaluation Metrics

**Workload:** Which filters are fast? Which are slow?

**Storage size:** How big is the stored data?

**Maintenance:** How much effort does it take to support insertions or updates to the data?