

# Portable In-Browser Data Cube Exploration

Kareem El Gebaly, Lukasz Golab, and Jimmy Lin

David R. Cheriton School of Computer Science  
University of Waterloo, Ontario, Canada



## Data exploration for everyone

From data democratization to analytics democratization

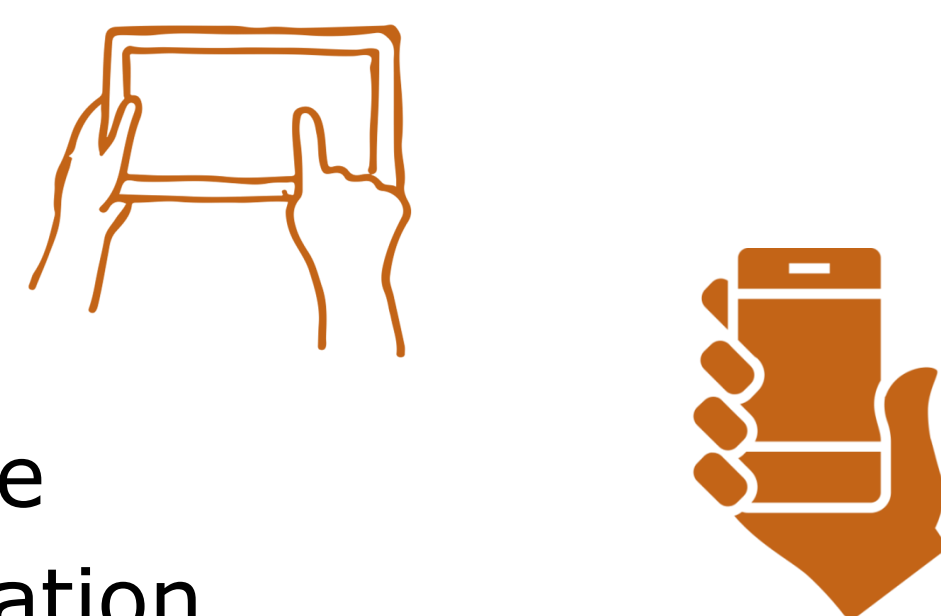
**Who?**

- Data scientists
- Data analysts
- Data journalists
- And may be their audience!



**How?**

- Easy to use
- Easy to interpret
- Does not require specialized infrastructure
- Does not require specialized pre-configuration



## Explanation tables:

- Information theoretic approach highlights the most informative parts of the data cube
- Iterative scaling finds maximum entropy estimates
- Sample based approach for pruning the datacube

| item      | season | location     | count | expires? |
|-----------|--------|--------------|-------|----------|
| *         | *      | *            | 14    | 7/14     |
| Chocolate | *      | *            | 5     | 0/5      |
| *         | *      | Winter House | 4     | 3/4      |
| *         | *      | Summer House | 3     | 2/3      |



Why do items go bad?

| id | item      | season | location     | expires? |
|----|-----------|--------|--------------|----------|
| 1  | Cheese    | Winter | Kitchen      | No       |
| 2  | Cherries  | Summer | Summer house | Yes      |
| 3  | Chocolate | Summer | Summer house | No       |
| 4  | Chocolate | Spring | Bedroom      | No       |
| 5  | Chocolate | Winter | Office       | No       |
| 6  | Chocolate | Summer | Basement     | No       |
| 7  | Chocolate | Fall   | Winter house | No       |
| 8  | Eggs      | Fall   | Kitchen      | Yes      |
| 9  | Eggs      | Winter | Winter house | Yes      |
| 10 | Juice     | Spring | Office       | No       |
| 11 | Milk      | Spring | Office       | Yes      |
| 12 | Milk      | Summer | Winter house | Yes      |
| 13 | Veggies   | Spring | Summer house | Yes      |
| 14 | Veggies   | Winter | Winter house | Yes      |



Something going wrong with the items?  
.. the locations?  
.. the seasons?

.. is it a combination of these?



Where do I start?



| item | season | location | count | expires? |
|------|--------|----------|-------|----------|
| *    | *      | *        | 14    | 7/14     |



| item      | season | location | count | expires? |
|-----------|--------|----------|-------|----------|
| Cheese    | *      | *        | 1     | 0/1      |
| Cherries  | *      | *        | 1     | 1/1      |
| Chocolate | *      | *        | 5     | 0/5      |



| item | season | location | count | expires? |
|------|--------|----------|-------|----------|
| *    | Winter | *        | 4     | 2/2      |
| *    | Summer | *        | 4     | 2/2      |
| *    | Spring | *        | 4     | 2/2      |



| item | season | location | count | expires? |
|------|--------|----------|-------|----------|
| *    | *      | Kitchen  | 2     | 1/2      |
| *    | *      | Bedroom  | 1     | 0/1      |
| *    | *      | Office   | 3     | 1/3      |



Explanation tables – Highlight the most informative parts of the cube



Afterburner – Explore the data cube in the browser

## Afterburner:

An in browser SQL engine that uses Code Generation that almost matches the state of the art SQL engines running native on the same machine.

Afterburner exploits two JavaScript features:

### JavaScript typed arrays:

- Contiguous in memory storage
- Predefined types using typed views
- Similar storage efficiency to C arrays

### Asm.js:

- Statically-typed subset of JavaScript
- Amenable to AOT optimization
- On average ~1.5× slower than native code

## Missed the talk?

Check out our live demo at:

<https://afterburnerdb.github.io/afterburner/explore.html>

Code is open source at:

<https://github.com/afterburnerdb/afterburner>