Distributed Summarization of Dynamic Data

Emma Alexander and Eric Balkanski

CS 262 Project Presentation

April 25th, 2016

Summarizing Crime Data

Dataset

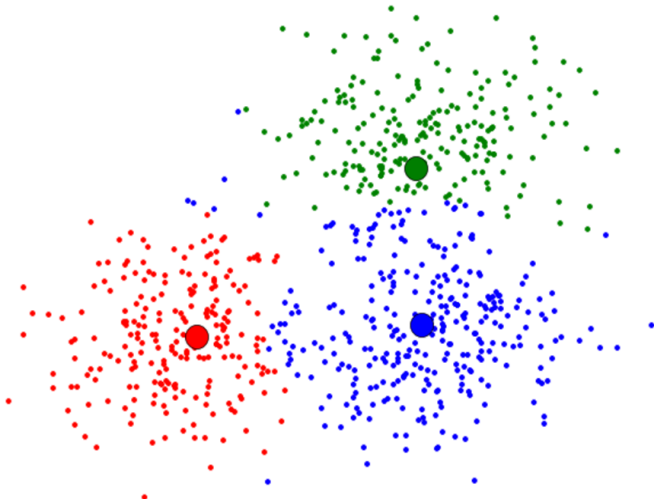
City	Type	Time	Arrest
NY	Assault	Night	Yes
NY	Theft	Night	Yes
NY	Assault	Night	No
LA	Theft	Morning	Yes
LA	Theft	Evening	No

Summary

NY	Assault	Night	Yes
LA	Theft	Morning	Yes

Data Summarization as Optimization

- Input:
  - **set of elements** V (e.g., set of crimes)
  - **function** $f(S)$ measures how well S summarizes V
    - via **clustering**

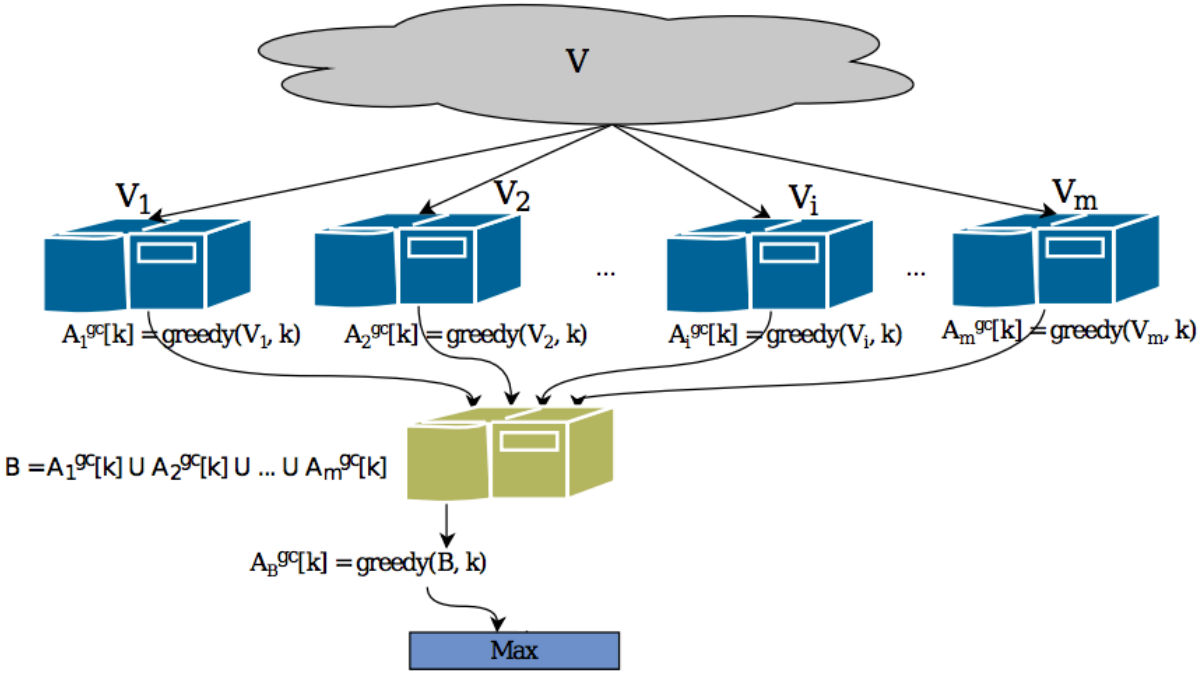


- Optimization problem:

$$\max_{\substack{|S| \leq k \\ S \subseteq V}} f(S)$$

Distributed Data Summarization

- What if V is too large to: to efficiently compute summary? fit on single machine?
- Distributed approach: [Mirzasoleiman et al. '13 and '15]



Tradeoffs

As number of machines m ↗ …

- Runtime ↘
- Memory per machine ↘
- Approximation ratio $\alpha = f(A^*) / f(A)$
  (quality of A compared to optimal solution A^*) ↗
- Communication complexity ↗

Our Contribution

- Real world datasets are **dynamic** with
  - **Insertions** (e.g., new crimes happen everyday)
  - **Deletions** (e.g., erroneous or outdated records)
- Naïve solution: rerun entire algorithm for every insertion and deletion
- We can achieve **better communication complexity** …

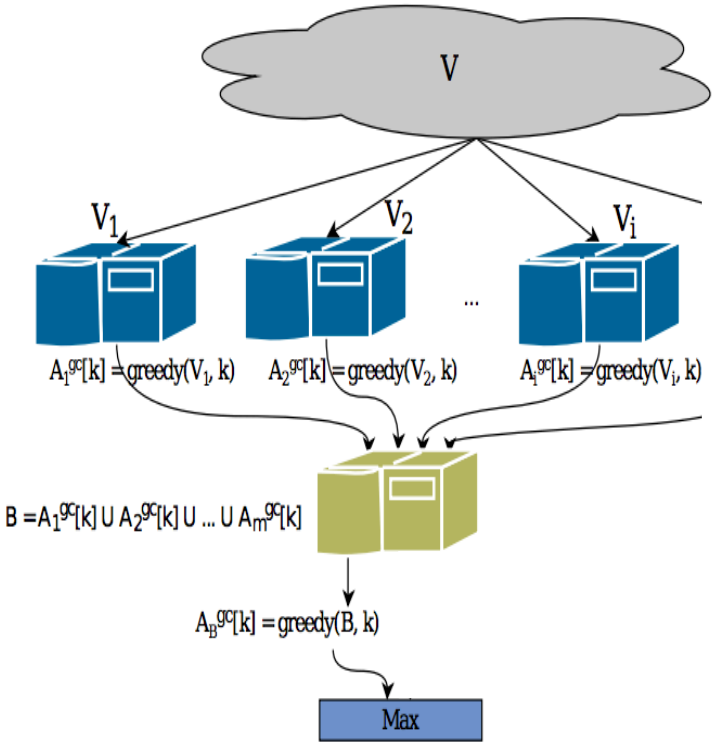
Our Solution

- Insertion or deletion for V_i
- A_i^{old} := last set sent from machine i

1. Update local solution A_i

2. If $d(A_i, A_i^{old}) \geq t$:
   - Send A_i to central machine

3. Update central solution A when receive updated A_i



Tradeoffs

- Trigger for sending a new local solution: $d(A_i, A_i^{old}) \geq t$
- As threshold t ↗ :
  - Approximation ratio ↗
  - Communication complexity ↘

Failure

- On local machine, either

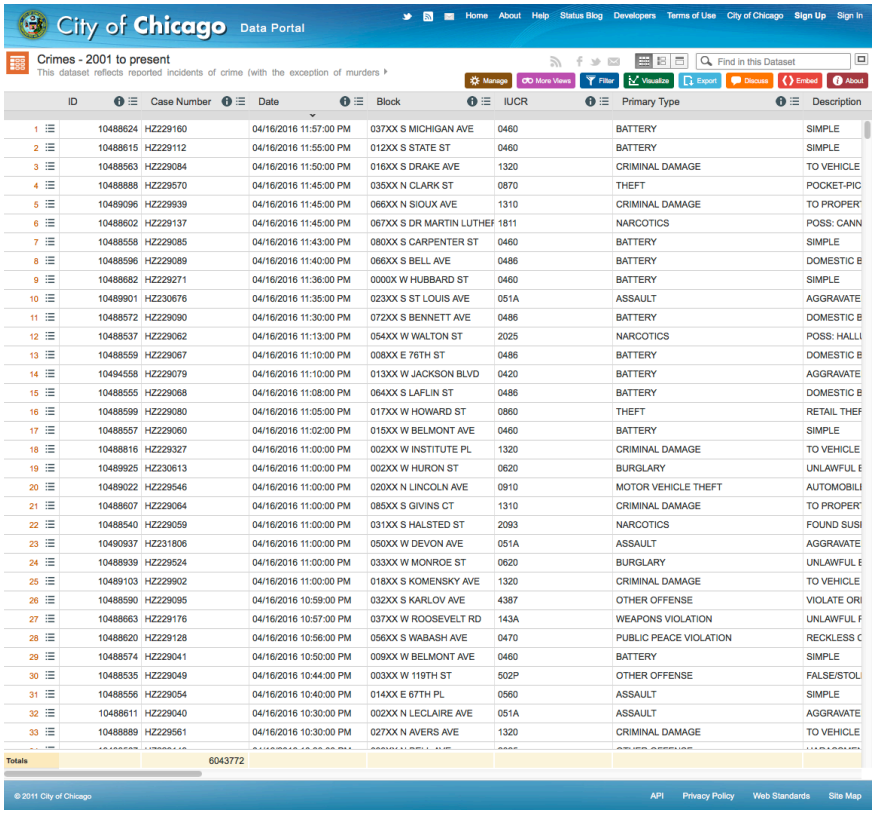
1. central machine waits for $m - T$ local solutions A_i
    - T fault tolerance
    - Approximation ratio
   – $T+1$ replicas of each element on different machines
    - T fault tolerance
    - Memory per machine

- On central machine, either
  – leader election
  – $T+1$ replicas of central machine

Implementation...

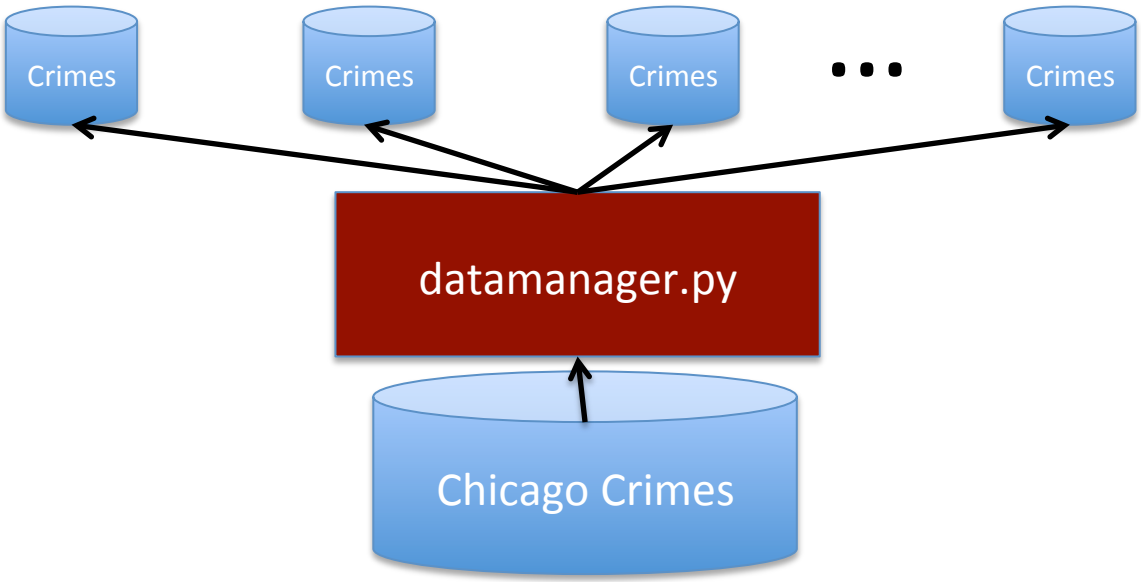
Dataset

- City of Chicago data portal
- 6 million+ crimes, 22 features
- We keep 6:
  - time of day
  - crime description
  - location description
  - arrest (boolean)
  - domestic (boolean)
  - district

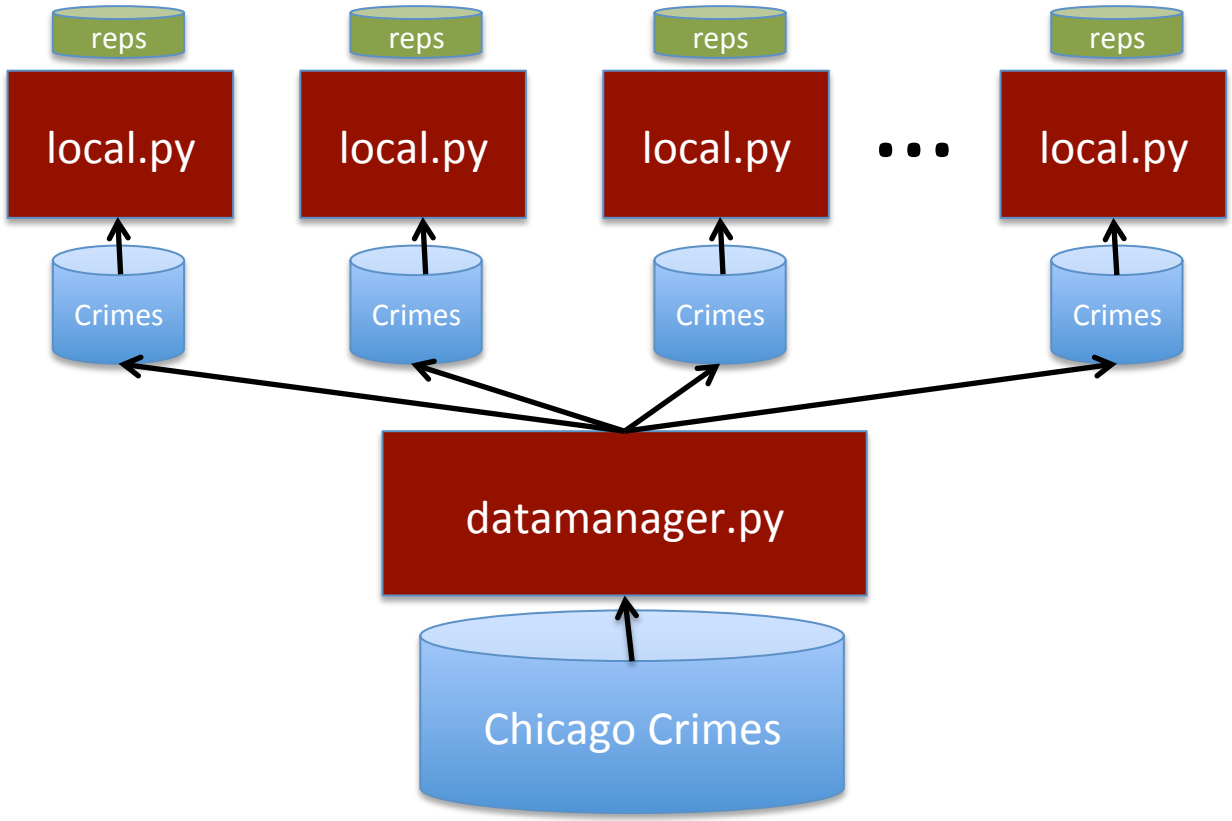


- Ex.) NIGHT,BURGLARY,"RESIDENCE",false,false,006

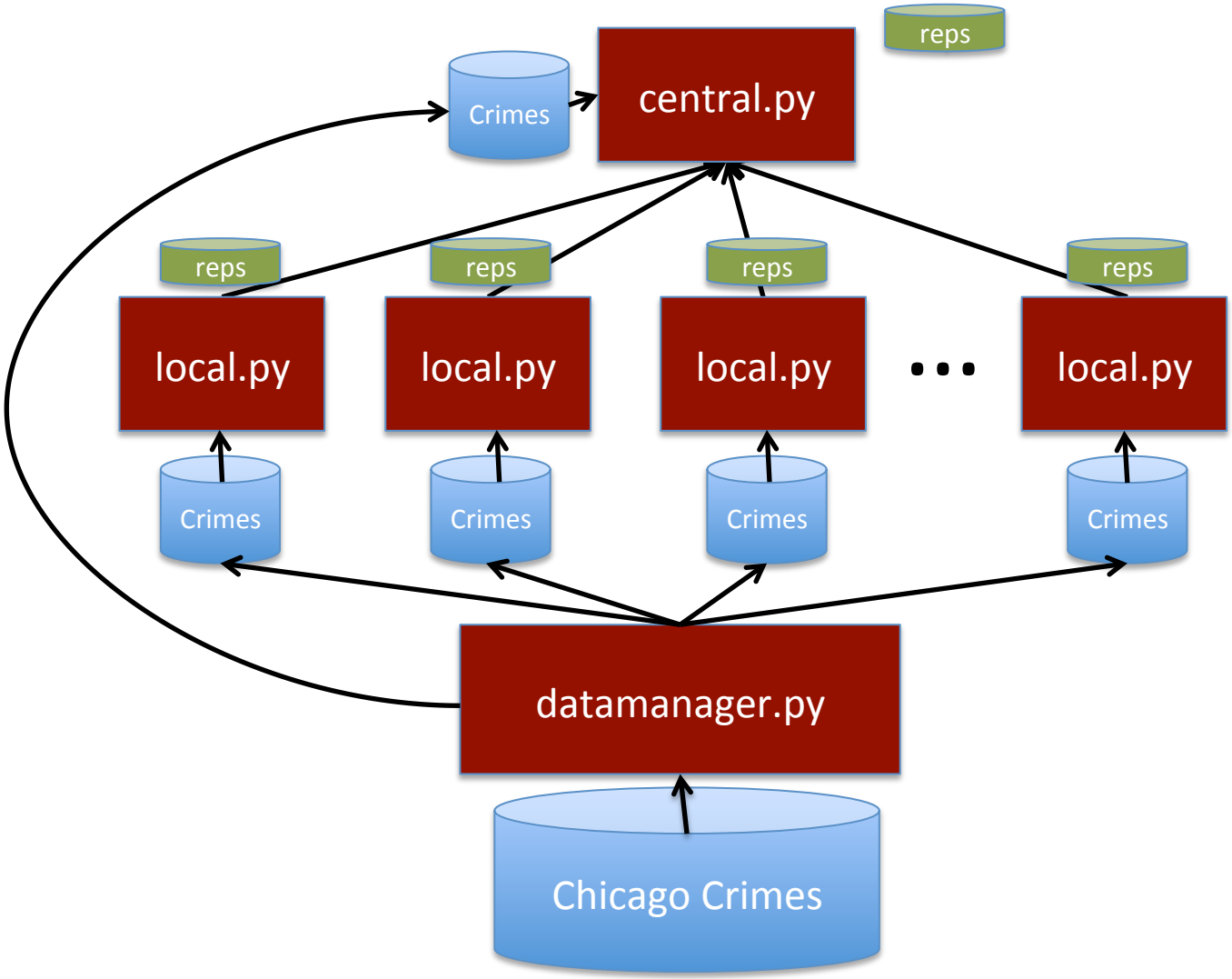
Simulation


Crimes
Crimes
Crimes
...
Crimes
datamanager.py
Chicago Crimes

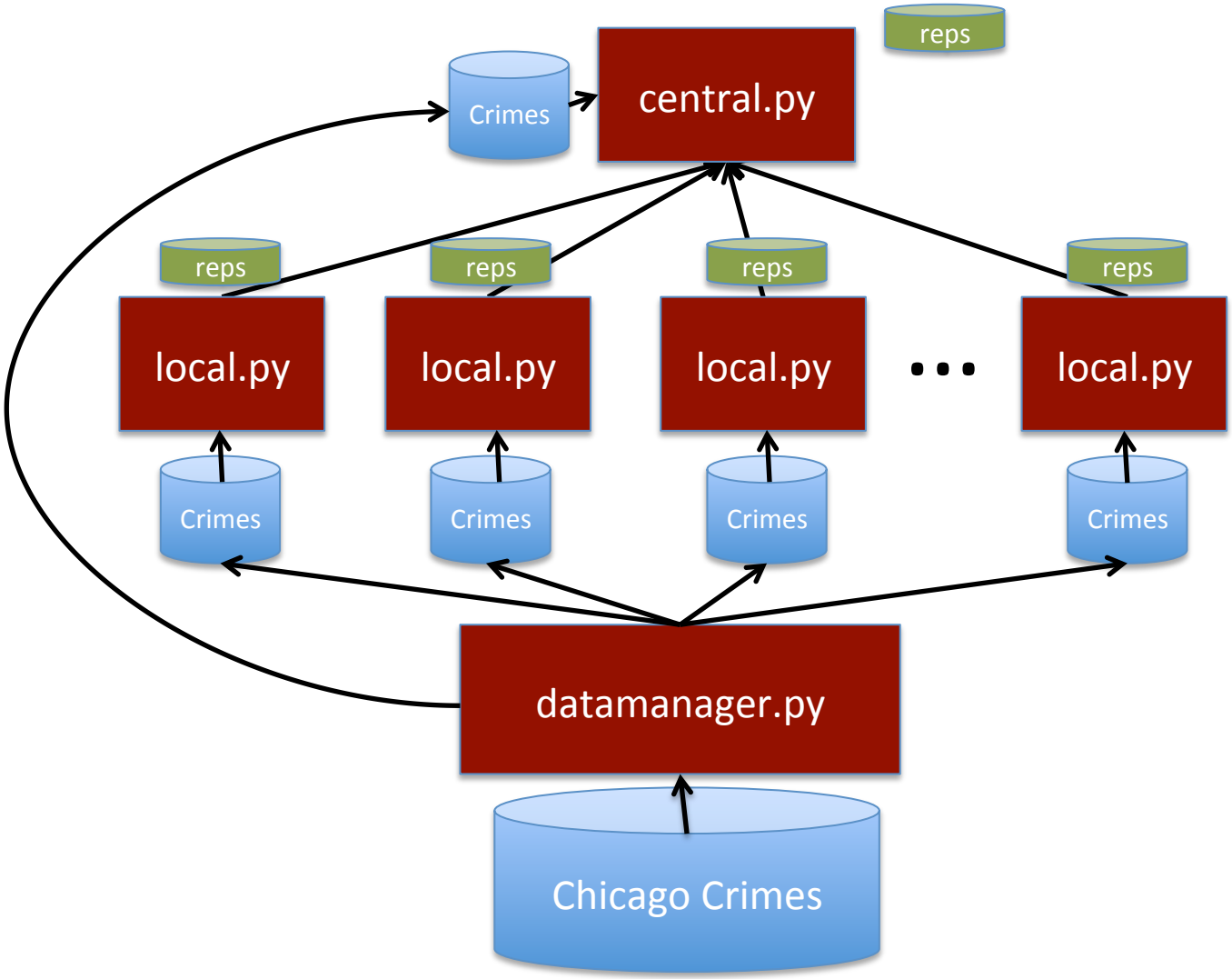
Simulation


reps
reps
reps
reps
local.py
local.py
local.py
local.py
Crimes
Crimes
Crimes
Crimes
datamanager.py
Chicago Crimes

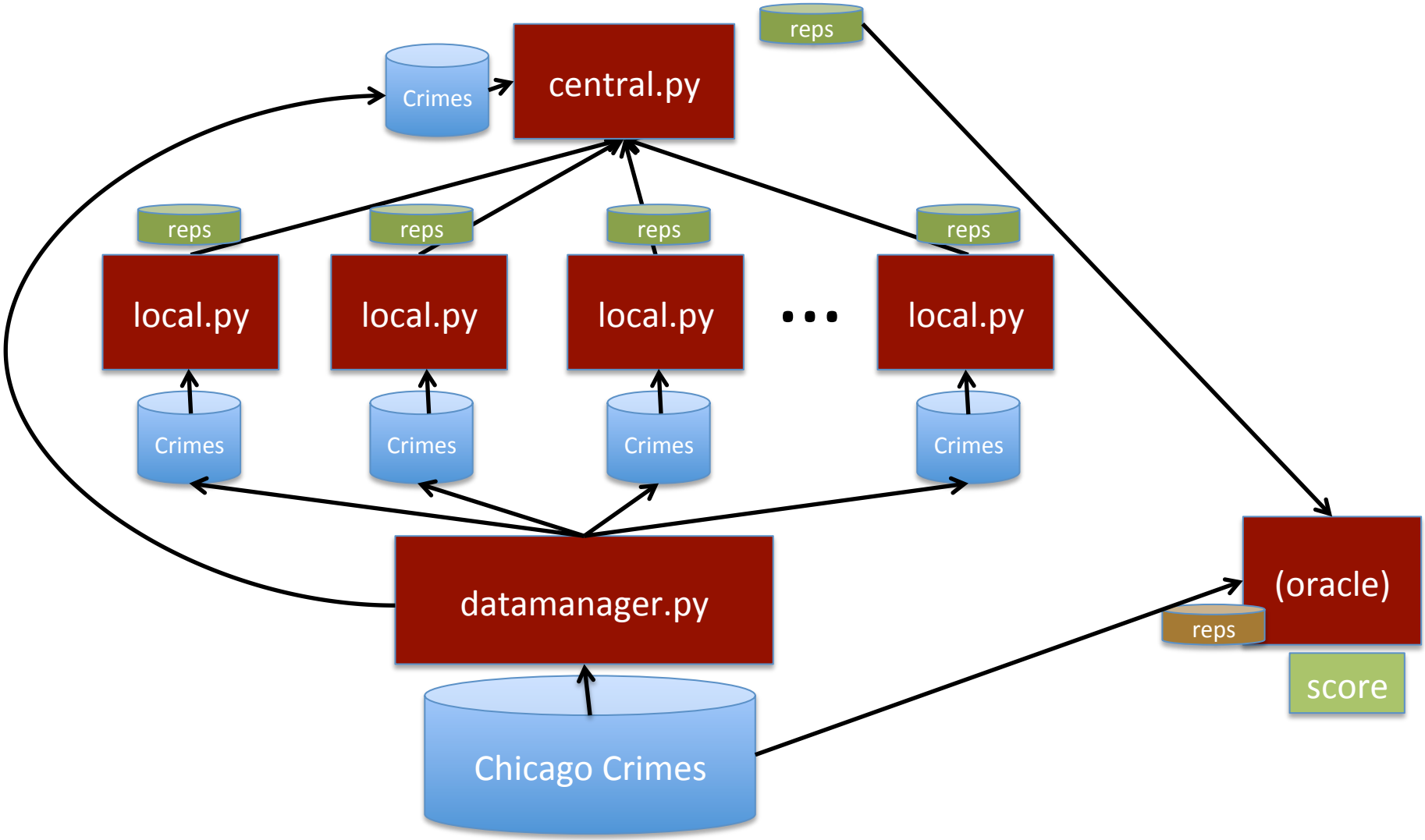
Simulation


reps
Crimes
central.py
reps
reps
reps
reps
local.py
local.py
local.py
local.py
Crimes
Crimes
Crimes
Crimes
datamanager.py
Chicago Crimes

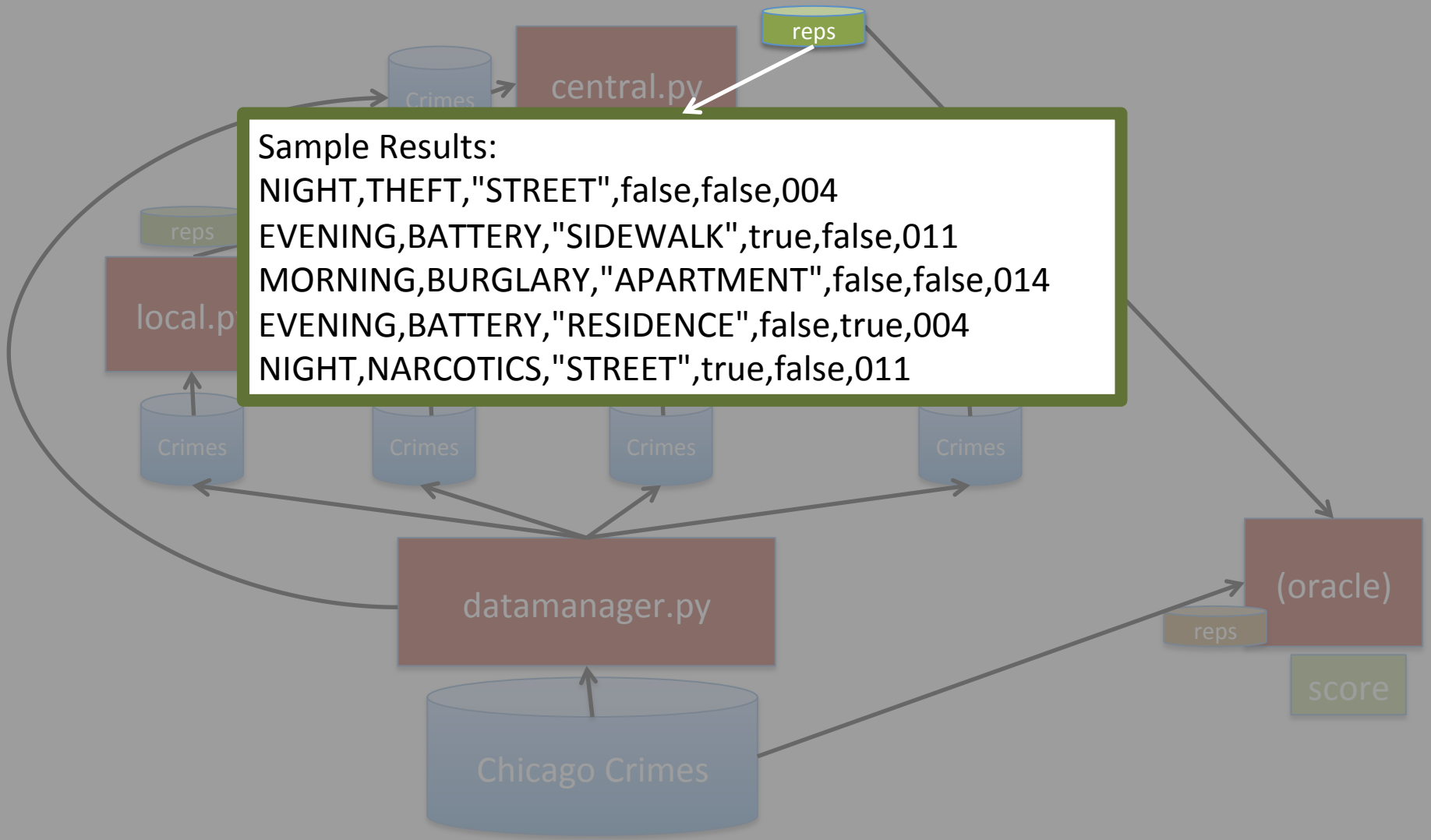
Simulation


reps
Crimes
central.py
reps
reps
reps
reps
local.py
local.py
local.py
local.py
Crimes
Crimes
Crimes
Crimes
datamanager.py
Chicago Crimes

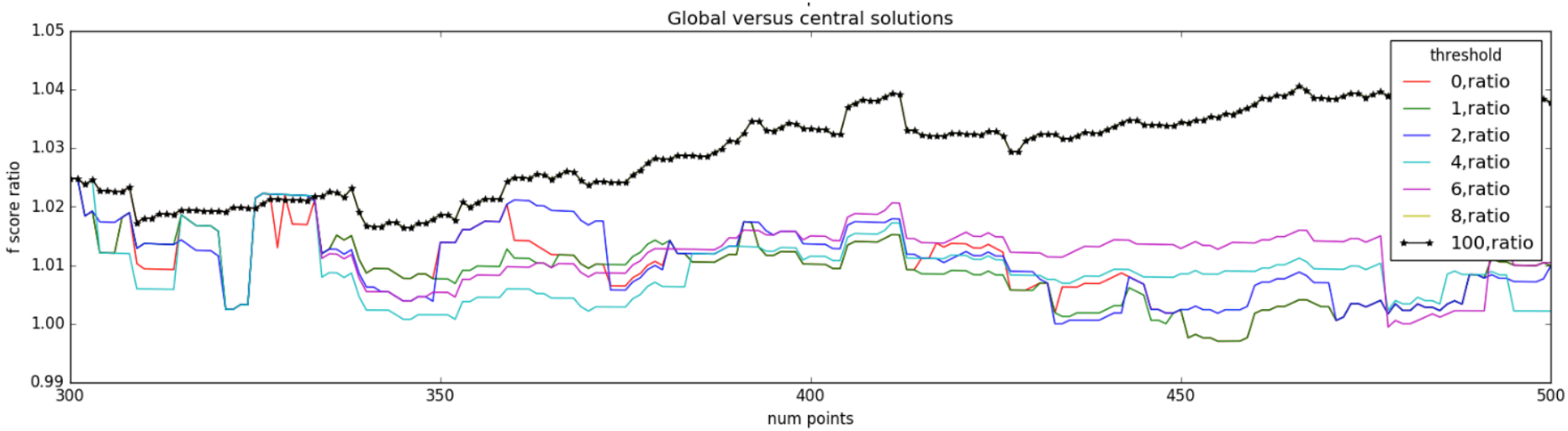
Simulation


reps
Crimes
central.py
reps
reps
reps
reps
local.py
local.py
local.py
local.py
Crimes
Crimes
Crimes
Crimes
datamanager.py
(oracle)
reps
score
Chicago Crimes

Simulation


reps
Crimes
central.py
Sample Results:
NIGHT,THEFT,"STREET",false,false,004
EVENING,BATTERY,"SIDEWALK",true,false,011
MORNING,BURGLARY,"APARTMENT",false,false,014
EVENING,BATTERY,"RESIDENCE",false,true,004
NIGHT,NARCOTICS,"STREET",true,false,011
reps
local.p
Crimes
Crimes
Crimes
Crimes
datamanager.py
(oracle)
reps
score
Chicago Crimes

Results



2 local processes, 10 representatives, 300 to 430 points

Next Steps

- Immediate:
  - Characterize tradeoff
  - Simulate failures
- General:
  - Theoretical bounds
  - Civic applications