Summarizing 5 years of my life in 10 minutes

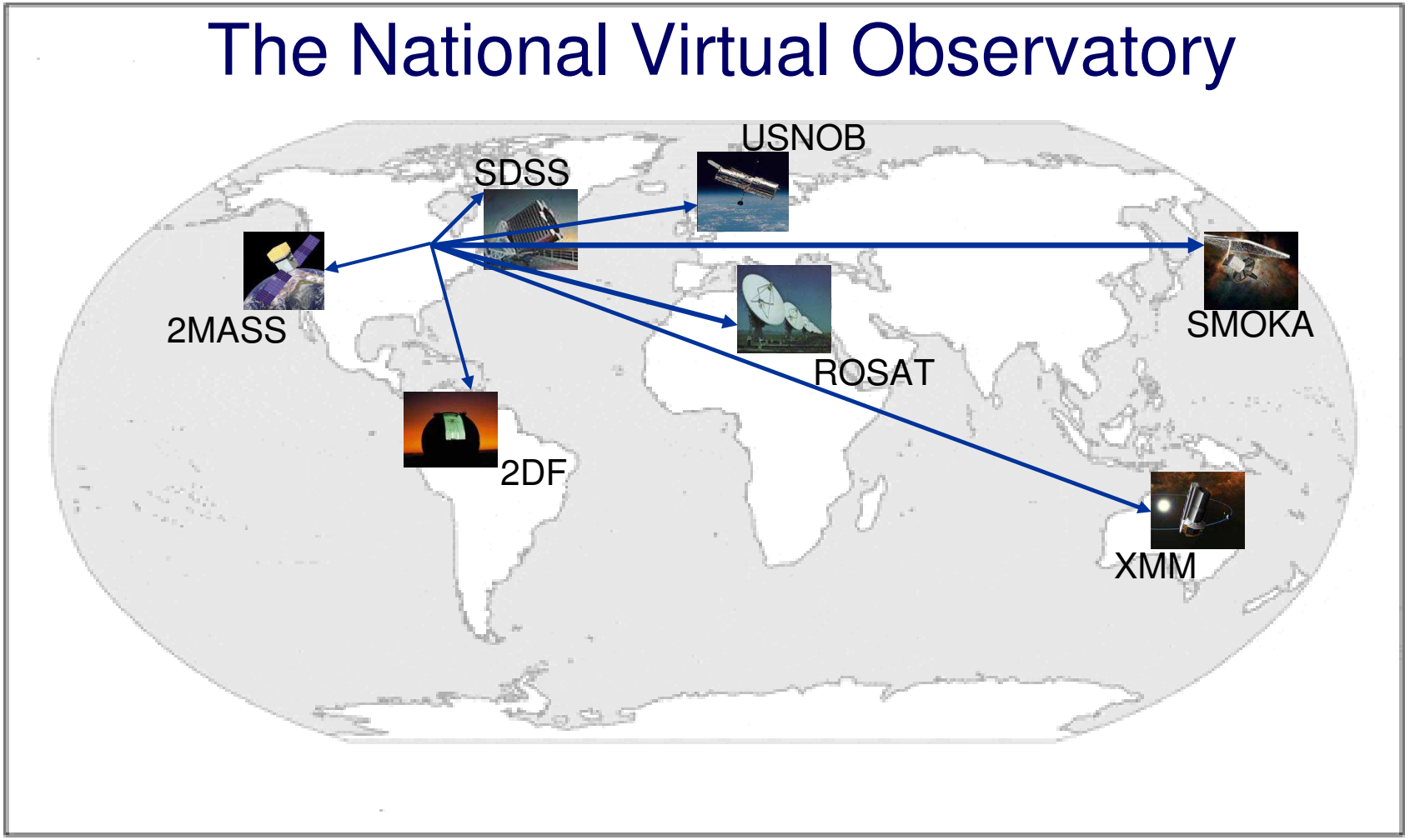
Tanu Malik

Dept. of Computer Science

Johns Hopkins University

Advisor: Dr. Randal Burns

The first years: Understanding the BIG problem



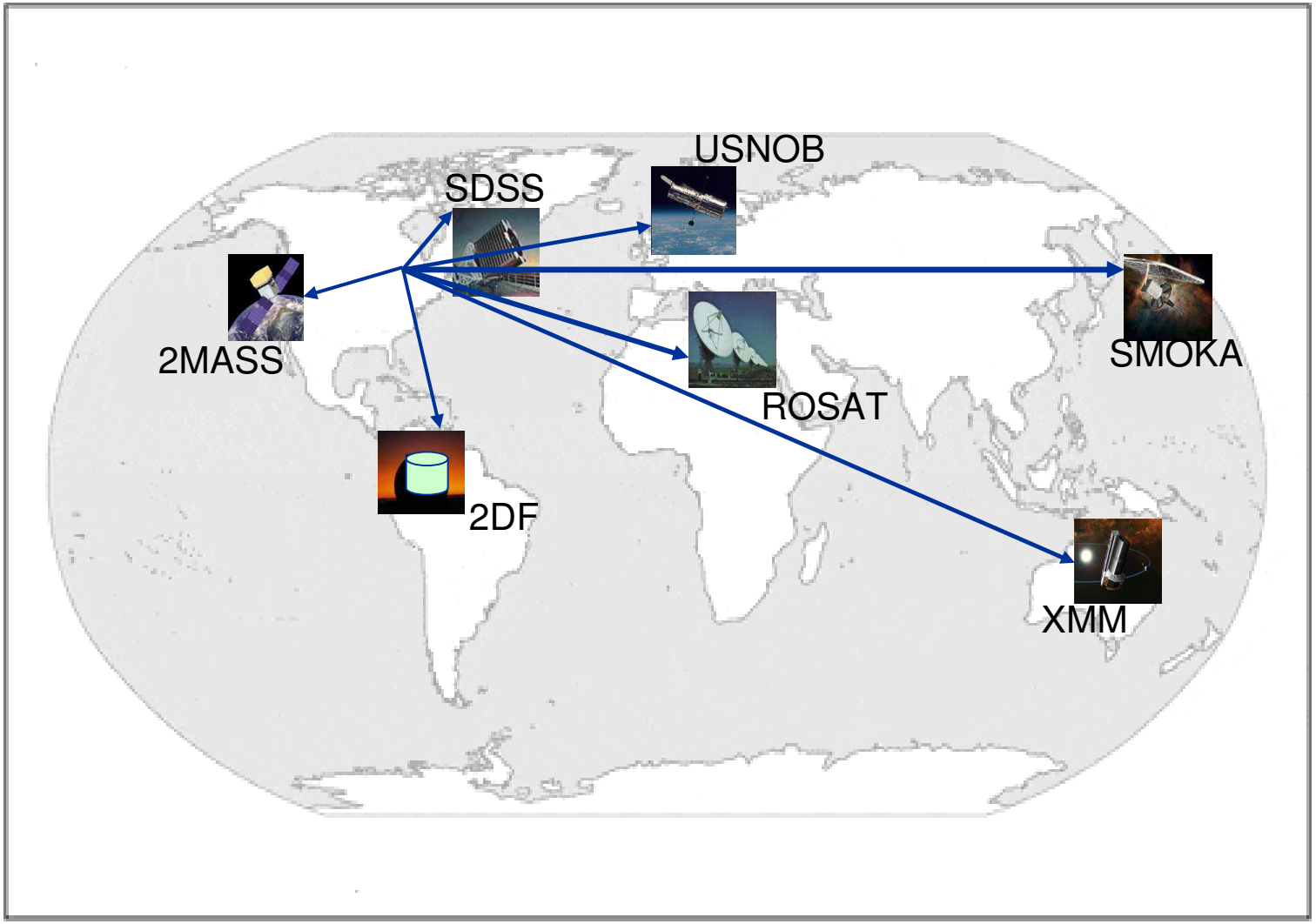
The BIG problem: Network limits scalability

```
SELECT o.objId, o.weight, t.lambda
FROM SDSS:PhotoObject o,
     TWOMASS:PhotoPrimary t
     FIRST: PrimaryObject p
WHERE
     AREA(181.3,-0.76,6.5) AND
     XMATCH(o,t,p)<3.5 AND
     type=GALAXY and (o.I - t.m_j)>2
```

- Area clause determines data
  - Gigabyte-size dataset
- Increasing size (current: 1
  - Astronomers typically jo

My BIG Question:
How to make these queries run efficiently?

Obvious Solution: Proxy caching

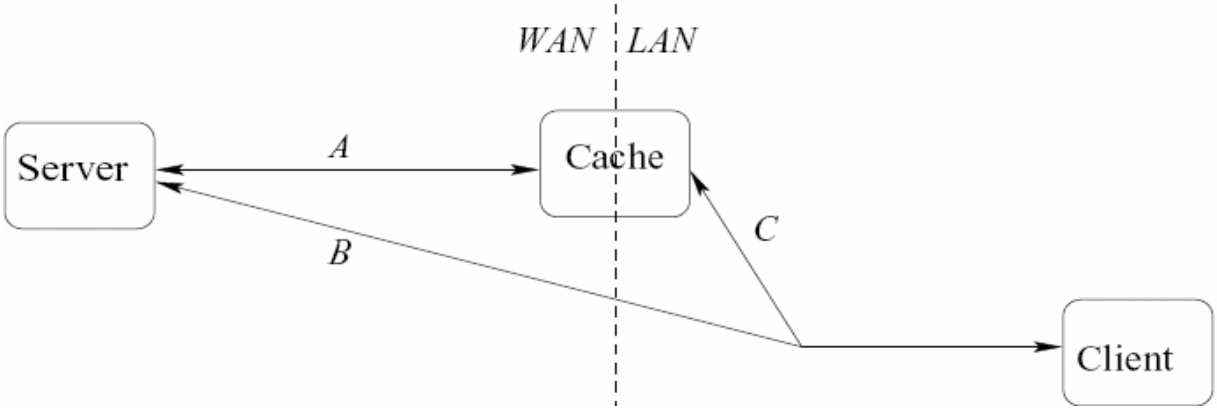


The Next Year: Caching the cacheable

- Primary Goal: Minimize network traffic
- What to cache?
  - x Query Results
  - x Materialized Views

    Schema/Database Objects (Columns, Tables)
- Observation 1 (From Workload Analysis)
  - Higher syntactic locality and low semantic locality
- Observation 2
  - Loading objects also causes network traffic
  - Cannot bring objects to cache on every query request

Bypass-Yield Caching

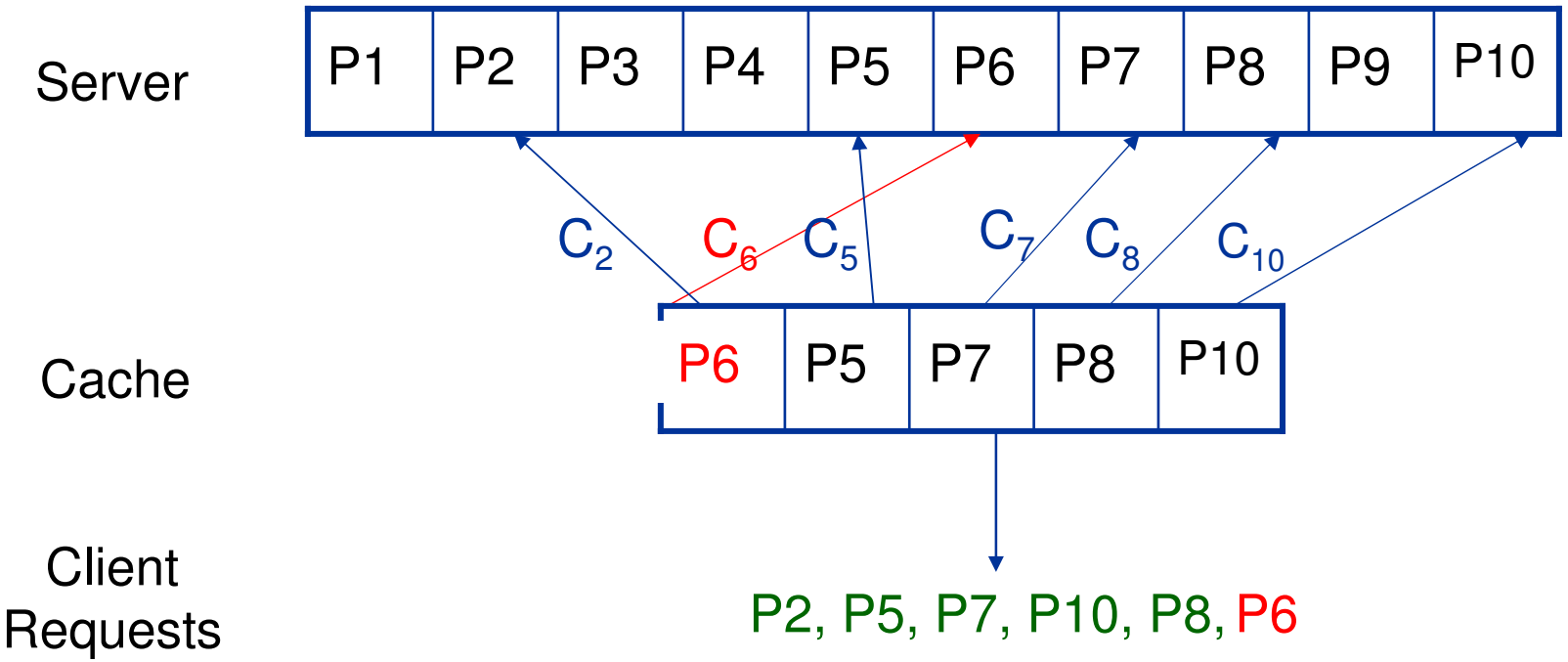


Goal: Minimize WAN traffic on A + B paths

- Idea 1:
  - Bypass queries which don't save network traffic
  - Load data objects into cache and serve query
- Idea 2:
  - Query result size/yield can decide between bypass and load

Page Caching

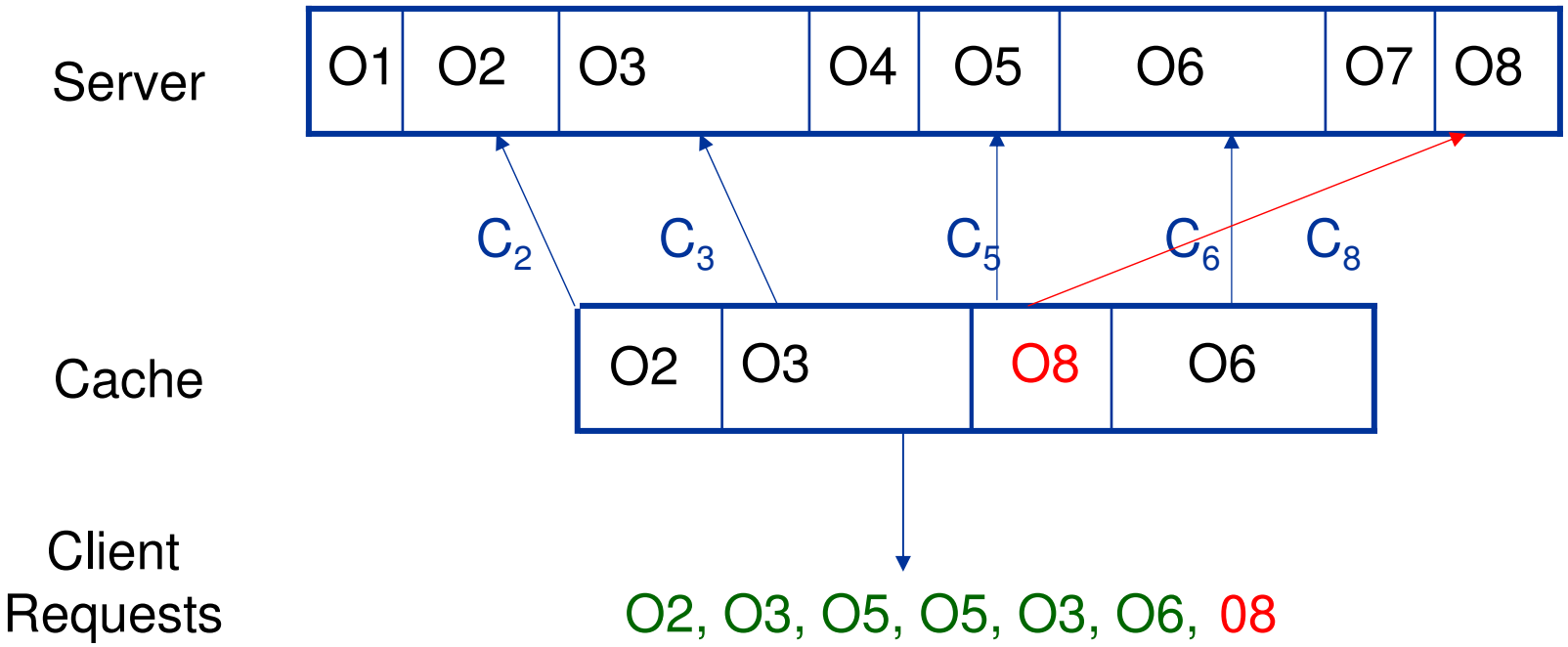
- Fixed size objects/pages, different fetch cost
- Cache hit is equivalent to an entire page being accessed
- Caches pages that have high fetch cost
- Used in operating systems



Goal: Minimize total fetch cost of pages

Object Caching

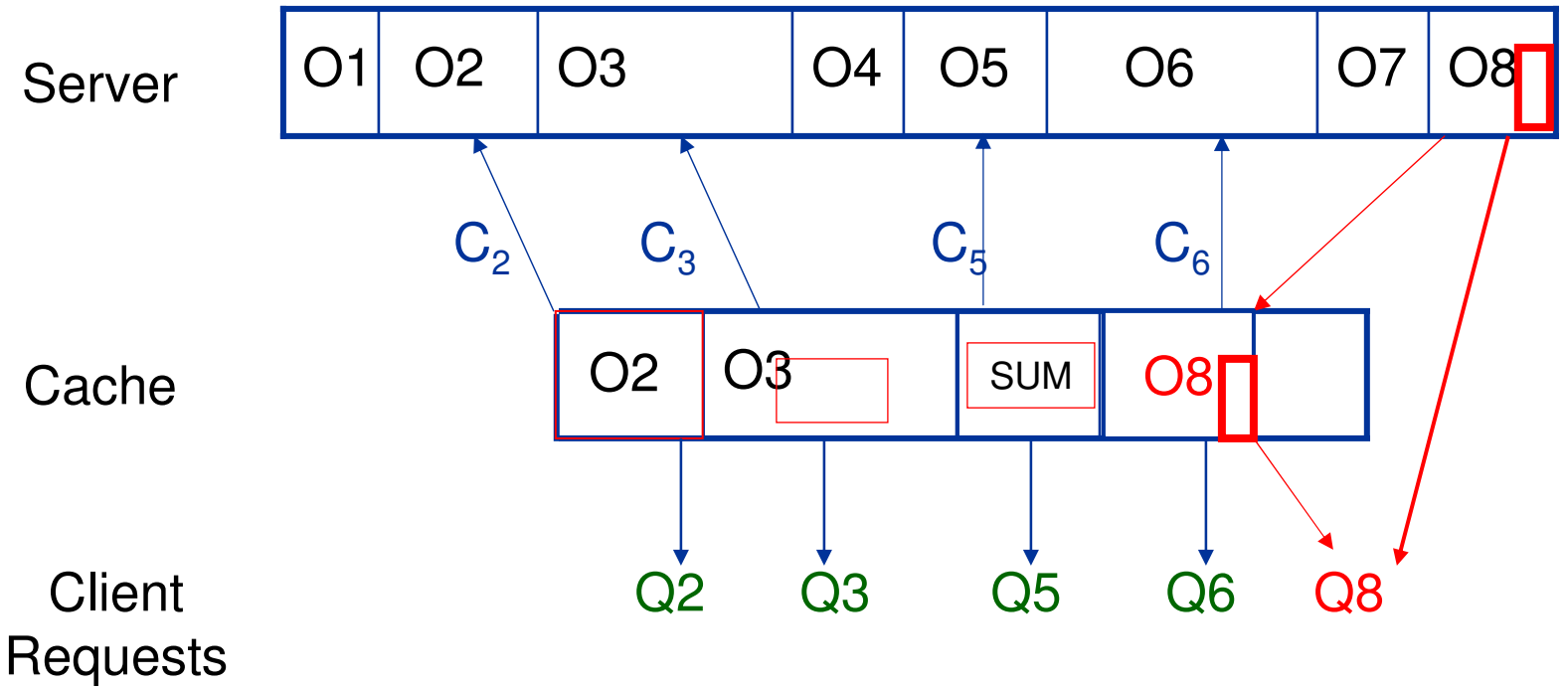
- Variable size objects, different fetch cost
- Cache hit is equivalent to accessing an entire object
- Caches objects that have high cost/size ratio.
- Used in proxy web caching



Goal: Minimize total fetch cost of objects

Bypass-Yield Caching

- Objects are of variable size
- For each object, cost of access varies
- Cache hit implies fetching an entire object, part of an object, or an aggregate computed over an object



Goal: Minimize total network cost (load and bypass)

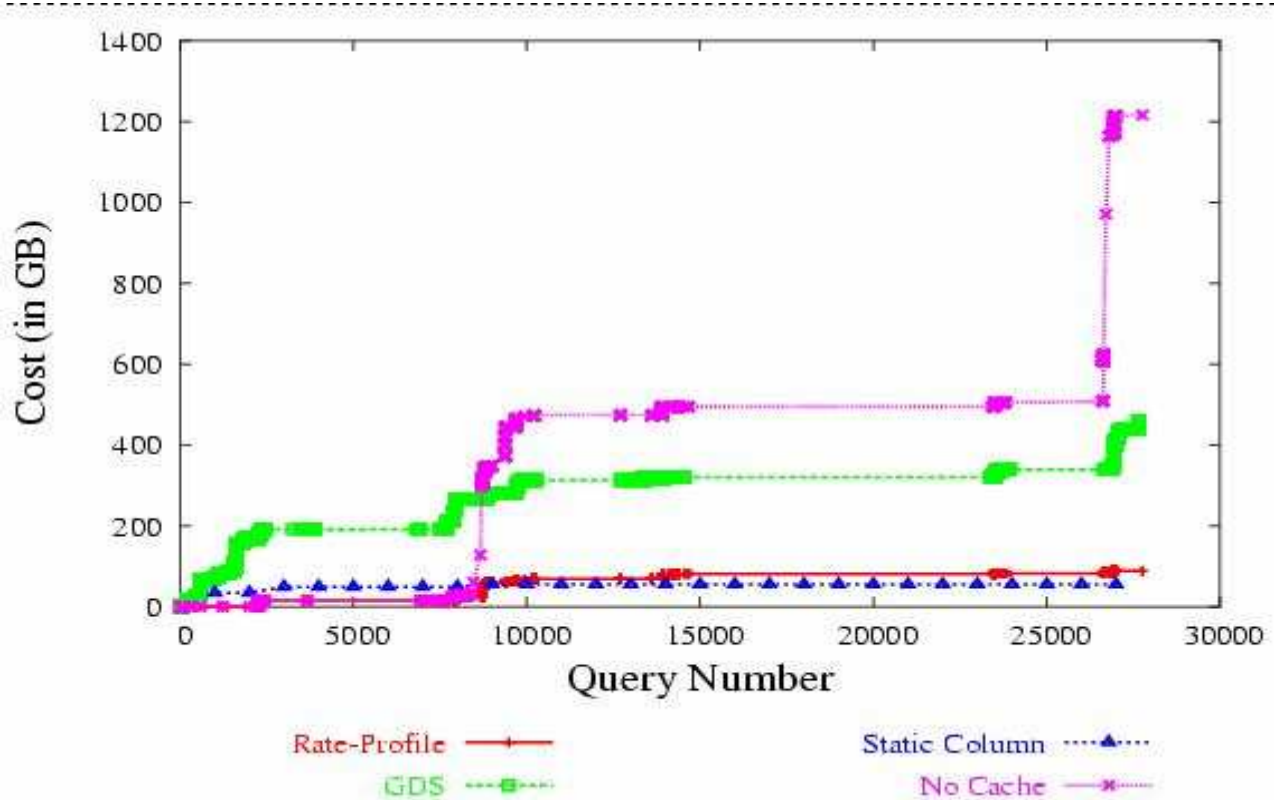
Byte-Yield Hit Rate

- Let s_i be the size and f_i be the fetch cost of object o_i
- Let $y_{i,j}$ be the size of the query-result of the jth query that accesses o_i
- Let $p_{i,j}$ be the probability of occurrence of the jth query above
- Define Byte-Yield Hit Rate (BYHR) as:

$$BYHR = \sum_j \frac{p_{i,j} y_{i,j} f_i}{s_i^2}$$

- BYHR has two components:
  - Expected benefit of caching o_i due to yield: $\sum_j \frac{p_{i,j} y_{i,j}}{s_i}$
  - Scaled by the cost/size ratio: $\frac{f_i}{s_i}$

Network Cost of a Trace (columns)



Contributions

- Developed metrics and caching algorithms that reduce WAN traffic
  - Heuristic and Competitive Online Algorithms
- Tested and evaluated performance within OpenSkyQuery
- Caching improves performance by **5x!**

T. Malik, R. Burns, and A. Chaudhary,
Bypass Caching: Making Scientific Databases Good Network Citizens, In ICDE 2005

What was next?: Relaxing assumptions

Cache performance requires accurate QRS estimation

Perfect, prior knowledge → 5x performance improvement

Commercial optimizer → 2.5x performance improvement

Designed a very light-weight, query-based estimation system. Reduces proxy cache traffic by 4.75x

[**Malik,Burns,Chawla,Szalay, QRS Estimation in Proxy Caching for Scientific DB Federations, SuperComputing, 2006. Best Student Paper Nominee**]

Other applications: Load Balancing, Replica Maintenance, Grid Computing, Web Caching

[Malik, Burns, Chawla, A Black-box Approach to Query Cardinality Estimation, CIDR 2007]

Challenges

- Constrained data access
  - Different management domains of cache and server
  - Stringent autonomy and privacy requirements
  - Collecting data distributions is I/O-intensive
- Strict space constraints
  - Large number of data sources
  - Limited cache metadata space
- Complex and a wide-variety of SQL queries
  - Multiple range clauses, joins, and user-defined functions
  - Cannot assume conditional independence between clauses