# Range Query Processing for Monitoring Applications over Untrustworthy Clouds

Hoang Van Tran, Tristan Allard, Laurent D'Orazio, and Amr El Abbadi

{van-hoang.tran, laurent.dorazio}@univ-rennes1.fr,
tristan.allard@irisa.fr, amr@ucsb.edu

## Introduction

- Use case: Real-time monitoring of the spatial spread of seasonal epidemics
  - ∗ Outsourced data in untrusted clouds
  - ∗ Range queries
  - ∗ High rate data arrivals
- Challenge: Performance problems (e.g., bottlenecks and overload)
- Proposal: PINED-RQ++, an extension of PINED-RQ [2]

## PINED-RQ [2]

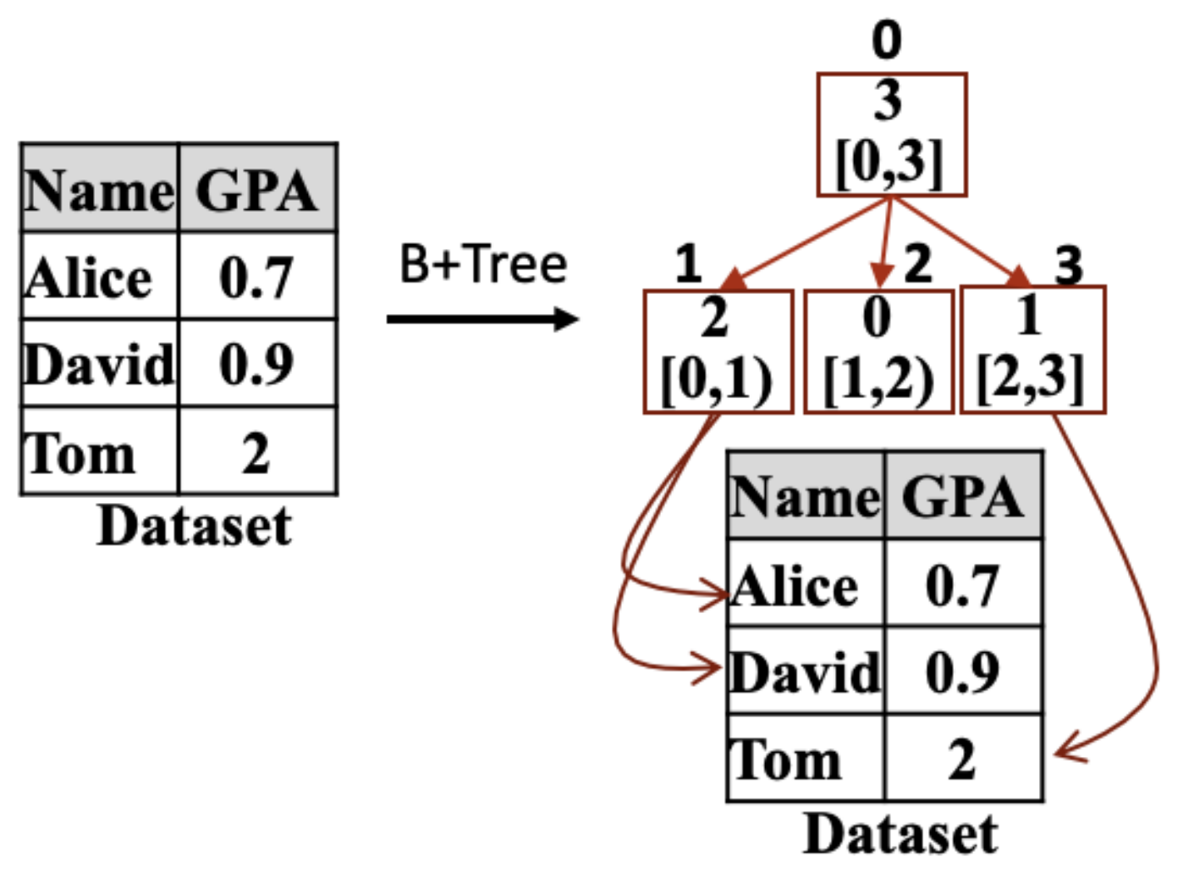
**Objective:** Efficient range query processing in untrusted clouds via secure and low-storage consumption indexes

**Privacy model:** Probabilistic relaxation of a computational variant of differential privacy

**PINED-RQ index:** Built in 2 steps

### Step 1: Construction

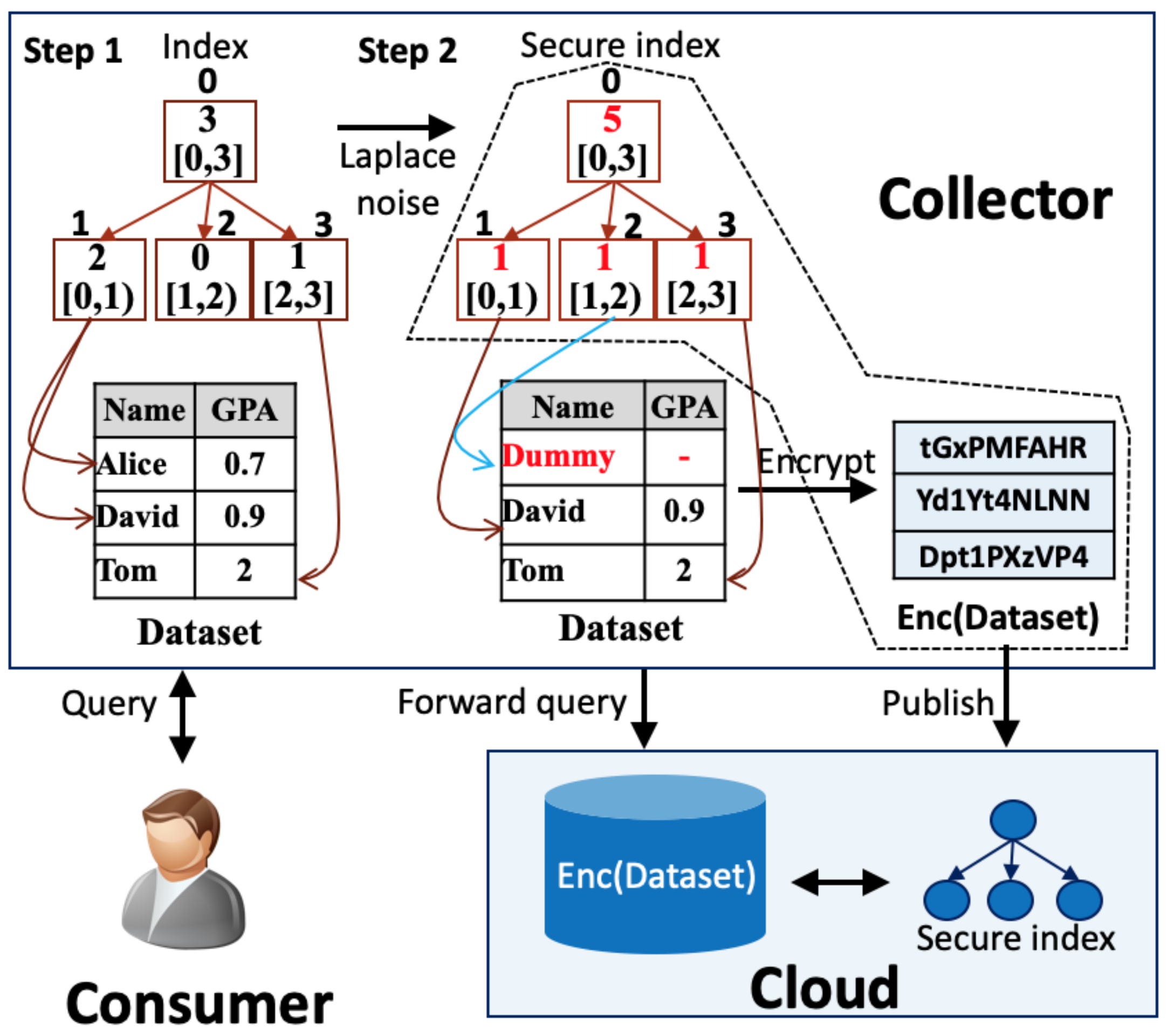
- An index is constructed based on a B+Tree
- The set of all nodes is defined as a histogram covering the domain of an indexed attribute



| Name | GPA |
|---|---|
| Alice | 0.7 |
| David | 0.9 |
| Tom | 2 |

### Step 2: Perturbation

- Laplace noise [1] are added to all counts
- The (positive/negative) noise makes the counts changing
- To keep the consistency between a leaf node's noisy count and the number of pointers it holds:
  - ∗ Positive noise: Add dummy tuples to the dataset
  - ∗ Negative noise: Remove real tuples from the dataset
  - ∗ Conceal removed tuples in overflow arrays



| Name | GPA |
|---|---|
| Alice | 0.7 |
| David | 0.9 |
| Tom | 2 |

| Name | GPA |
|---|---|
| Dummy | - |
| David | 0.9 |
| Tom | 2 |

| |
|---|
| tGxPMFAHR |
| Yd1Yt4NLNN |
| Dpt1PXzVP4 |

### Limitations

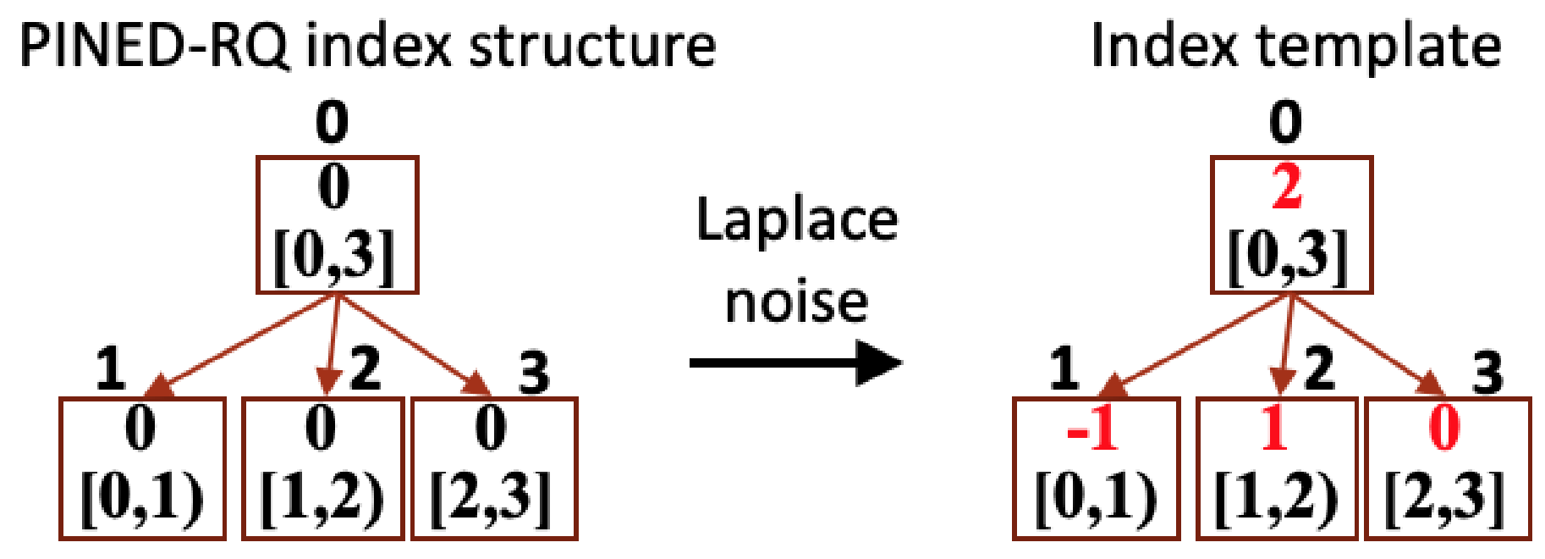
- Publish data in batches
- Partially process queries at the collector

⇒ Potential bottlenecks

## PINED-RQ++

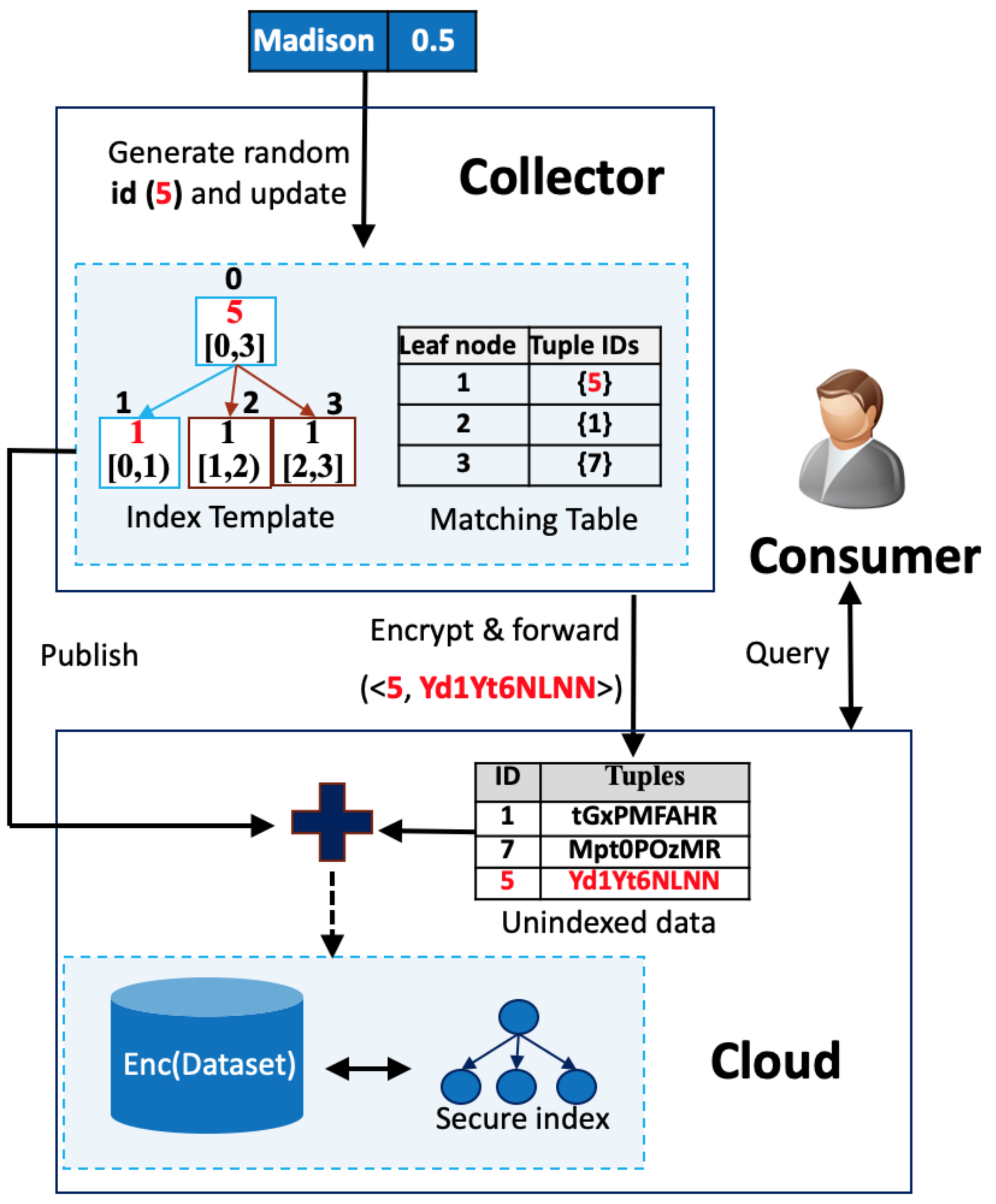
**Intuition:** Send a new tuple to the cloud immediately when it arrives at the collector

**Proposal:** Index template



### Index template updating

- Initiate an index template
- Update the index template as a new tuple arrives
- Encrypt and forward the tuple to the cloud
- Associate the index template with unindexed data to produce a secure index
- Use unique random numbers and a matching table to keep the relationship between published tuples and its index



| Leaf node | Tuple IDs |
|---|---|
| 1 | {5} |
| 2 | {1} |
| 3 | {7} |

| ID | Tuples |
|---|---|
| 1 | tGxPMFAHR |
| 7 | Mpt0POzMR |
| 5 | Yd1Yt6NLNN |

### Challenges

How to manage dummy and removed tuples?

- Dummy tuples: The collector sends dummy data according to the actual distribution of the sending time of the real tuples
- Removed tuples: If a leaf node receives negative noise *c*, move its first *c* tuples to the corresponding overflow array

### Query processing

A query is only processed at the cloud holding indexed and unindexed data at time

### Parallel PINED-RQ++

- To enhance the collector's throughput
- Multiple clones of the original template

**Assumption:** An individual has a single tuple

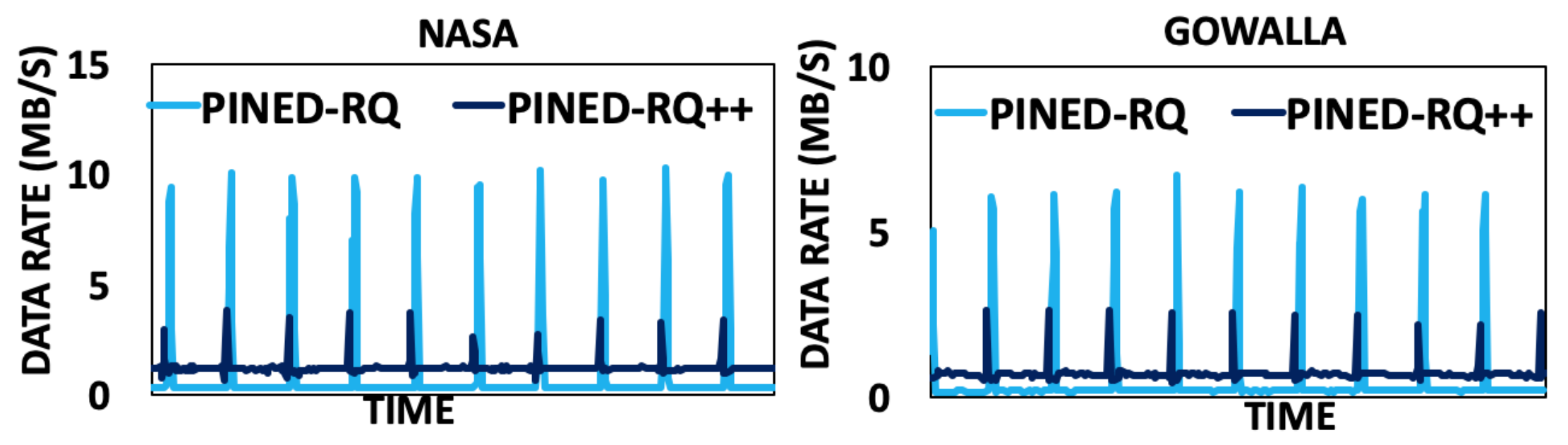
## Experimental Results

### Benchmark Environment

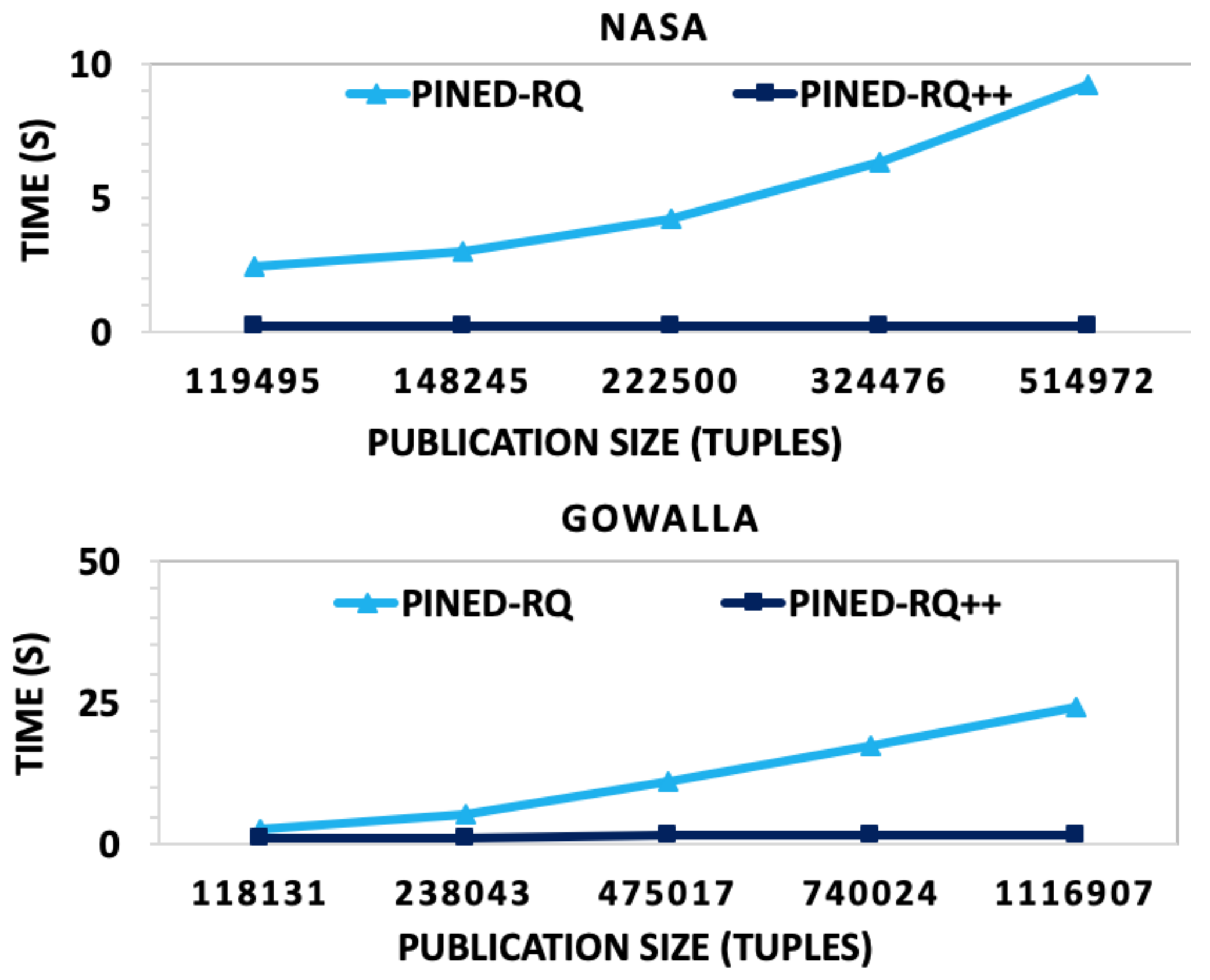
The proposal was evaluated on NASA log [3] and Gowalla [4]

### Results

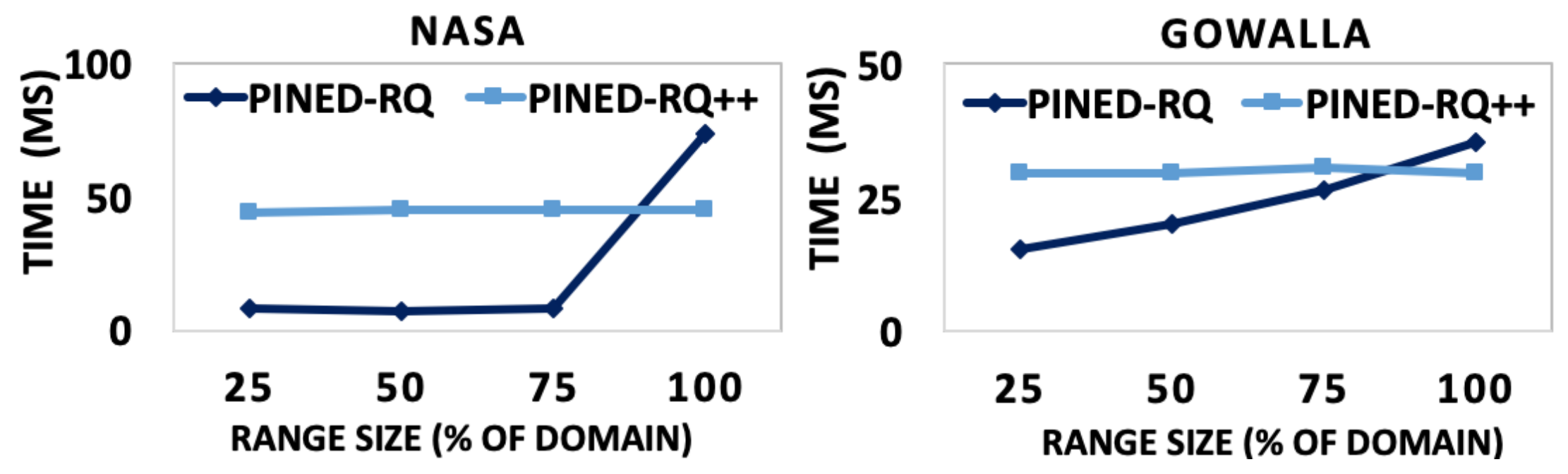
- Network traffic (Maximum data rate)



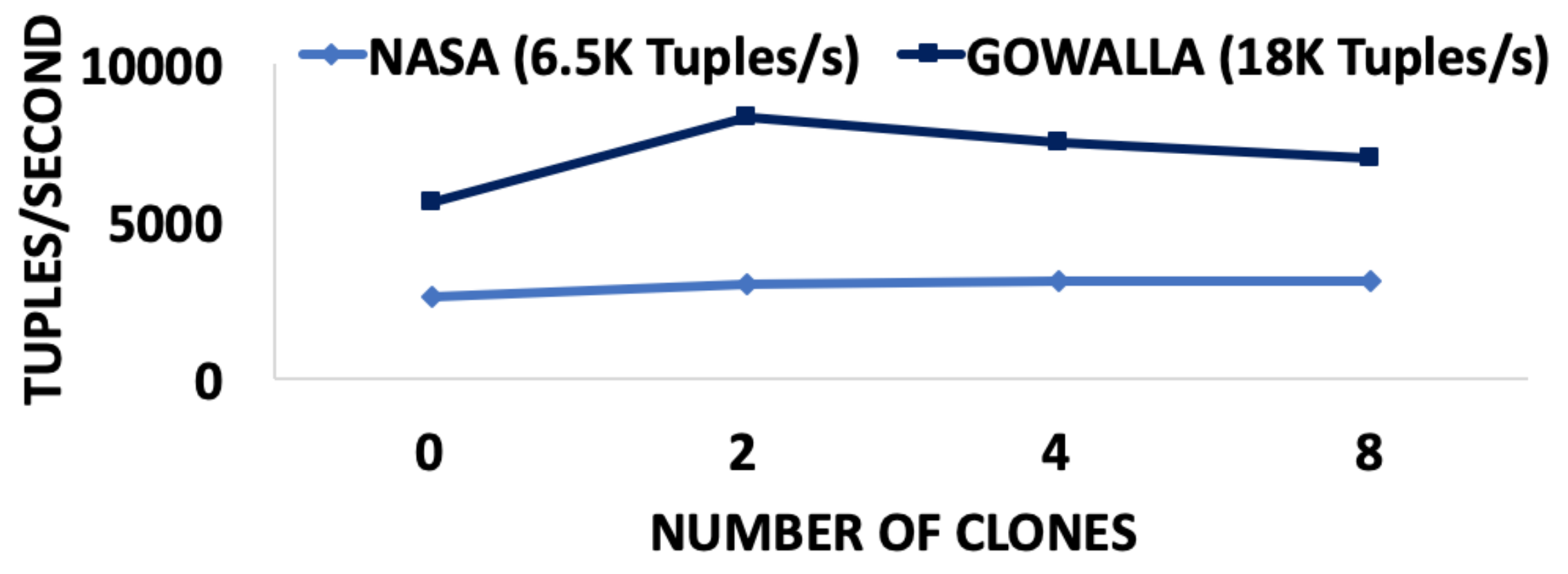
- Publishing time



- Response time latency



- Throughput at the collector



## Future Works

- Query performance improvement (caching)
- Dynamic adaptation to query workload

## References

[1] C. Dwork, F. McSherry, K. Nissim, and A. Smith Calibrating Noise to Sensitivity in Private Data Analysis. In *TCC '06*. Pages 265–284.

[2] C. Sahin, T. Allard, R. Akbarina, A. El Abbadi, and E. Pacitti. A Differentially Private Index for Range Query Processing in Clouds. In *ICDE'18*. Pages 857–868.

[3] NASA Log (1996). http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html

[4] J. Leskovec and A. Krevl. SNAP Datasets: Stanford Large Network Dataset Collection. June, 2014. http://snap.stanford.edu/data.

## Acknowledgement

This work was partially supported by ARED Pôle d'Excellence Cyber/Lannion-Trégor Communauté.