DNSStream: Towards Unifying and Streaming DNS Data

Alfred Arouna - Ioana Livadariu - Mattijs Jonker

alfred@simula.no

simulamet   UNIVERSITY OF TWENTE.
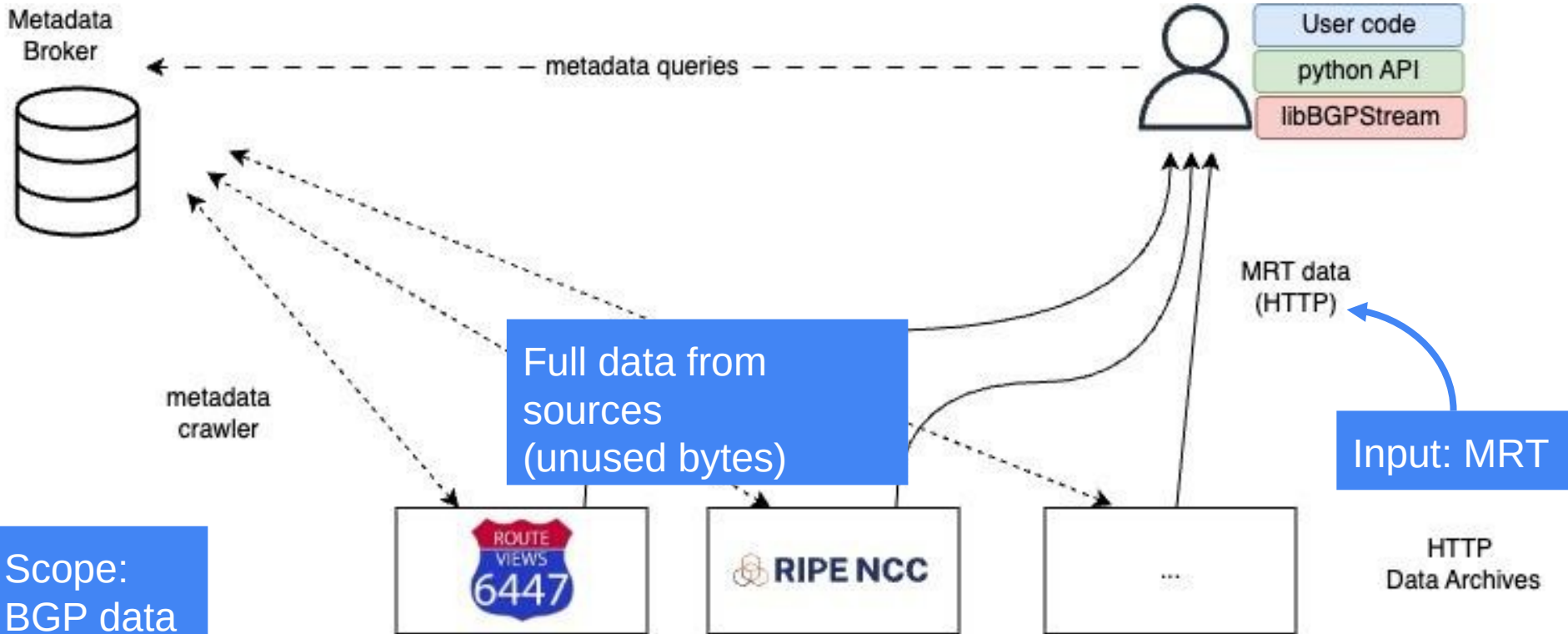
GMI-AIMS-5 / 12.02.2025

BGPStream

output: stream of BGP records

Metadata Broker

metadata queries

User code

python API

libBGPStream

Full data from sources (unused bytes)

MRT data (HTTP)

Input: MRT

metadata crawler

Scope: BGP data

ROUTE VIEWS 6447

RIPE NCC

...

HTTP Data Archives

DNS data: native client

Current/naive approach.

- Input: metadata (query)
- Challenges:
  - **Various (meta)data query.**
  - **Various formats (HTTP/S3).**
  - **Various data/time partitioning.**
  - **Unrelated data:**
    - **Files.**
    - **Rows.**
- Output: **full data on all files from sources** (unused bytes)
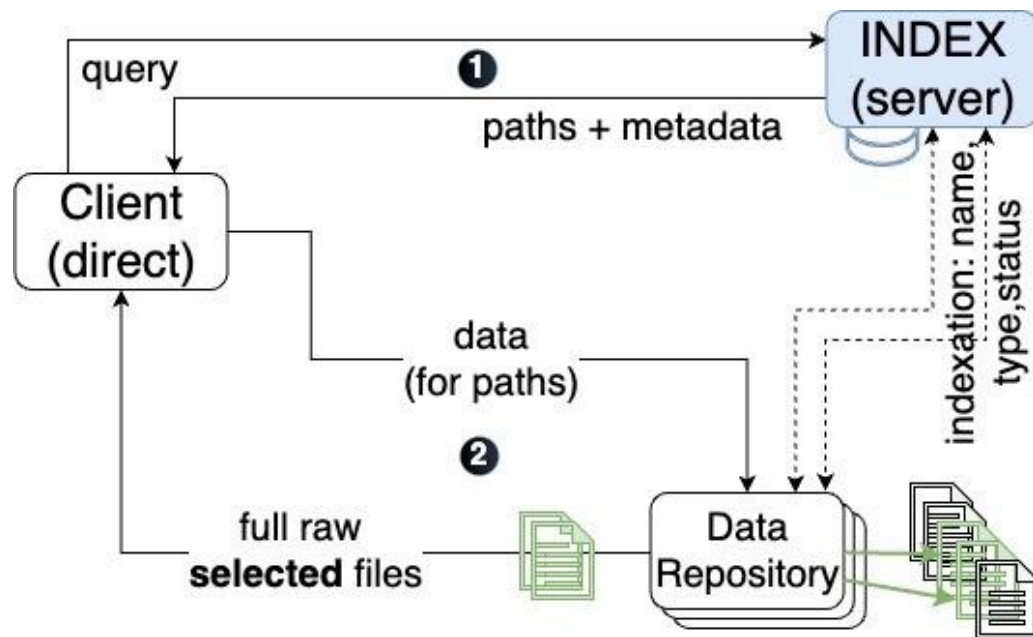- **Post-filtering**: **no files reduction and no data reduction.**
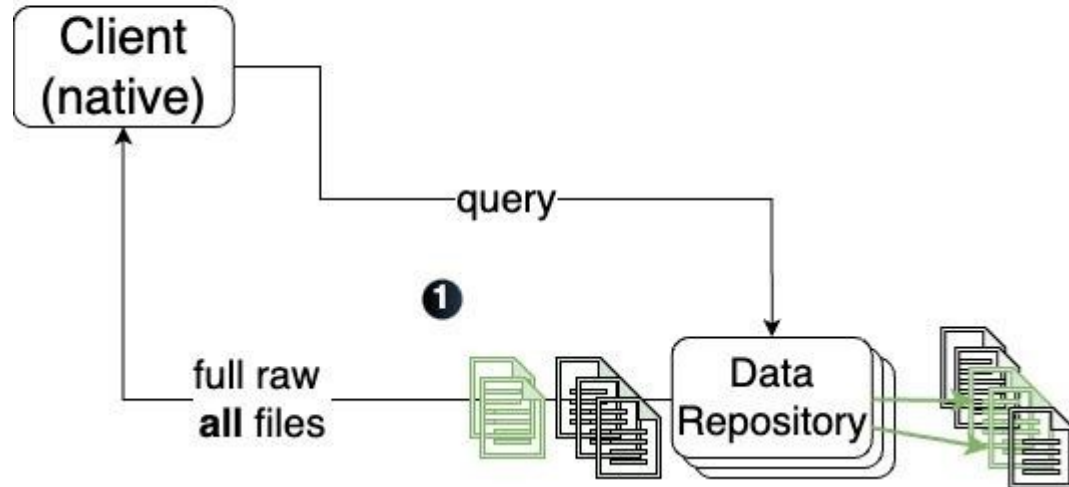
DNSStream (direct mode)

New approach.

- Input: **common** metadata (query)
- Challenges:
  - ~~Various (meta)data query.~~
  - ~~Various formats (HTTP/S3).~~
  - ~~Various data/time partitioning.~~
  - ~~Unrelated data:~~
    - ~~Files.~~
    - **Rows**.
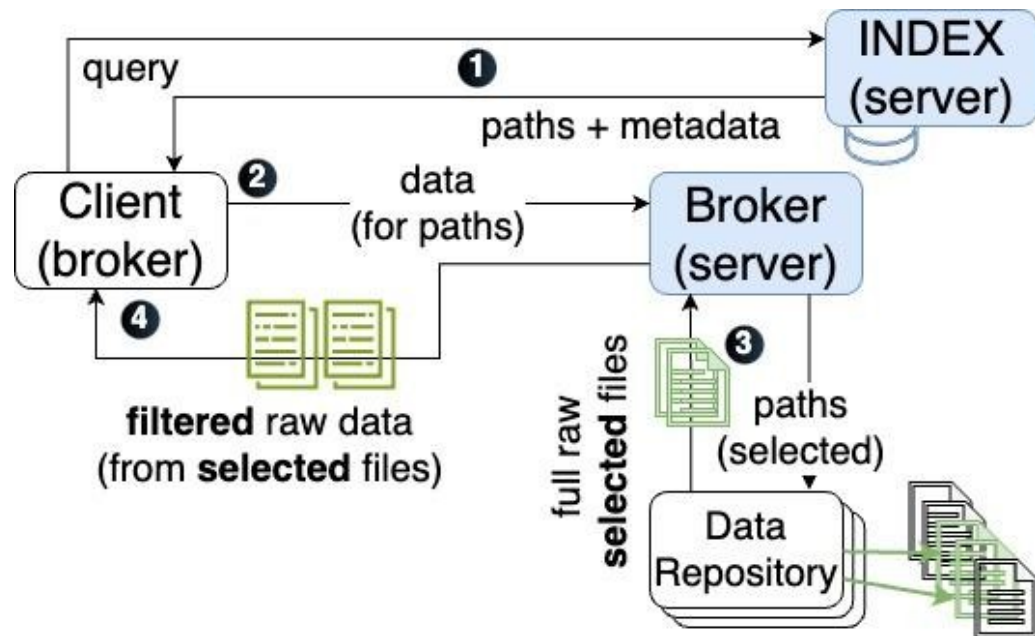- Output: list of matching file paths from the Indexer (datastore).

Naive approach:

- Input: list of paths
- Output: full data on **matching** files from sources (unused bytes)
- **Post-filtering**: **files reduction but no data reduction.**

DNSStream (broker mode)

New approach (extended direct approach).

- Input: **common** metadata (query)
- Challenges:
  - ~~Various (meta)data query~~
  - ~~Various formats (HTTP/S3)~~
  - ~~Various data/time partitioning.~~
  - ~~Unrelated data~~
    - ~~Files.~~
    - ~~Rows.~~
- Output: Stream of **matching** data from sources (based on the list of matching file paths)
- **Pre-filtering**: **Files reduction and data reduction.**

Future plans

Performance evaluation of the prototype.

Case studies