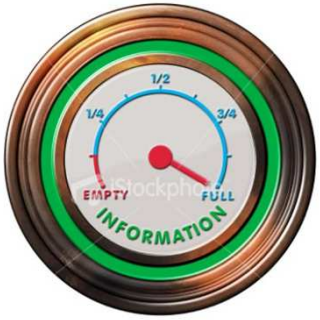
Sedic: Privacy-Aware Data Intensive Computing on Hybrid Clouds

K. Zhang, X. Zhou, Y. Chen, X. Wang, Y. Ruan

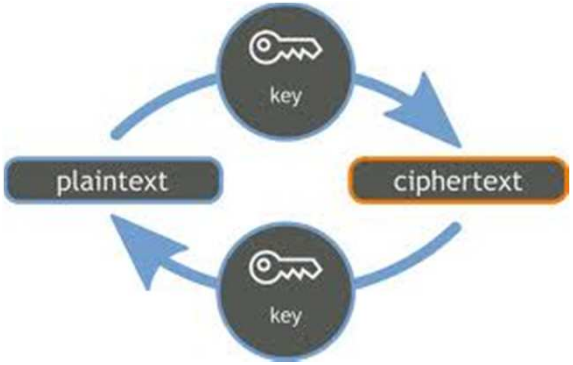
Motivation

- Rapid growth of information ⇒ High processing demand



- Commercial cloud providers can meet demand
  - Amazon EC2, EMR, *etc.*

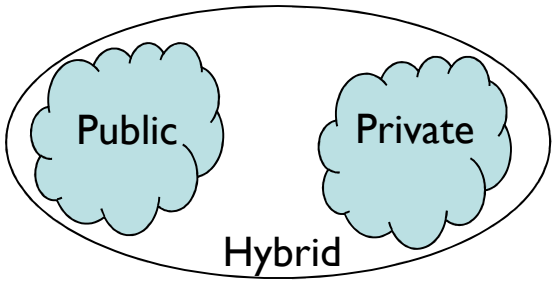
- Large privacy risks with outsourcing processing – HIPAA



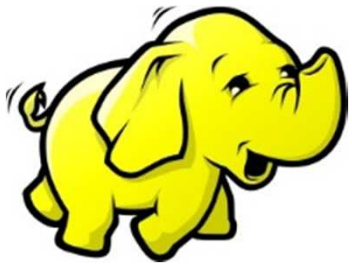
- Are cryptographic techniques a solution??
  - Prohibitively expensive
  - Hard to scale

Motivation

- Are Hybrid Clouds a solution??
  - Split computations
  - Send computations over non-sensitive info to public cloud
  - Send computations over sensitive info



- How about using MapReduce on a Hybrid Cloud??
  - Designed for a single cloud
  - Unaware of data with multiple security levels
  - Manual splitting of processing required



- Need framework-level support to facilitate processing over hybrid clouds

Sedic – Objectives

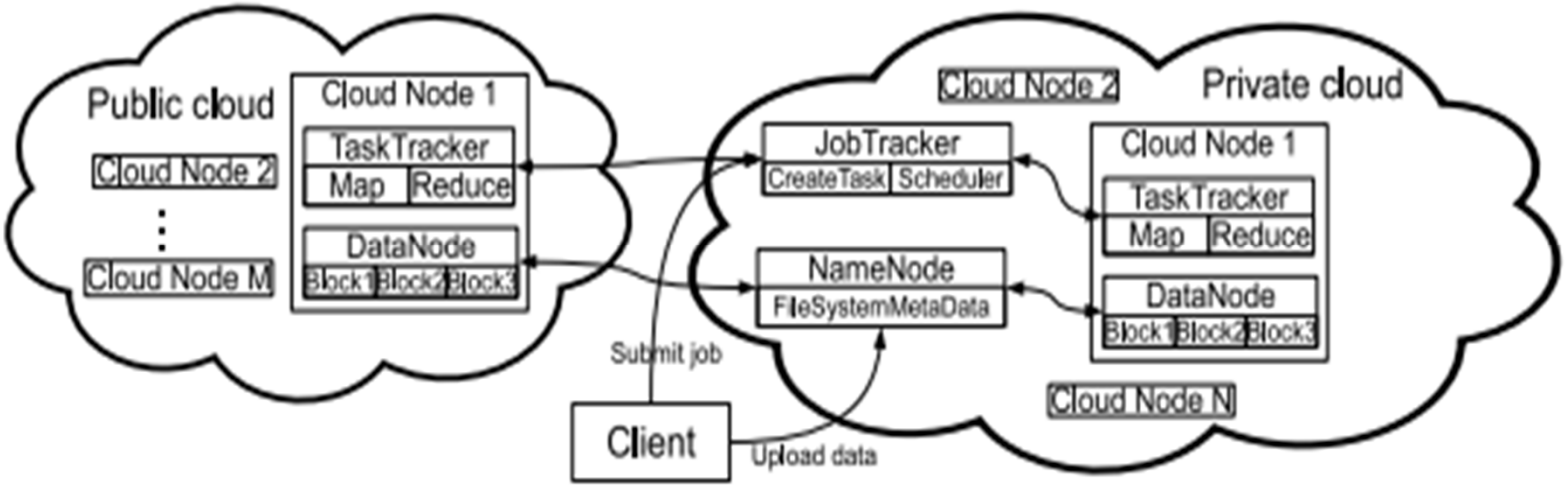
- High Privacy Assurance
  - Only public data is given to a commercial cloud
- Maximum public cloud utilization
  - Move as much computation to the public cloud as possible while respecting a user's privacy
- Scalability
  - Preserve MapReduce scalability while keeping a low privacy protection overhead
- Limited inter-cloud transfer
  - Since it is expensive
- Easy to use
  - Preserve end-user's MapReduce experience

Sedic – Design Overview

Table 1: Steps for a Privacy-Aware MapReduce

Users	• Label sensitive data, which can be done through a data-tagging tool (Section 3.1). • Submit to Sedic labeled data and a MapReduce job.
Sedic	• Analyze and transform the reduction structure of the job (Section 4). • Partition and replicate the data according to security labels (Section 3.1). • Create and schedule mappers across the public/private clouds (Section 3.2). • Combine the results on the public cloud and complete the reduction on the private cloud (Section 3.3).

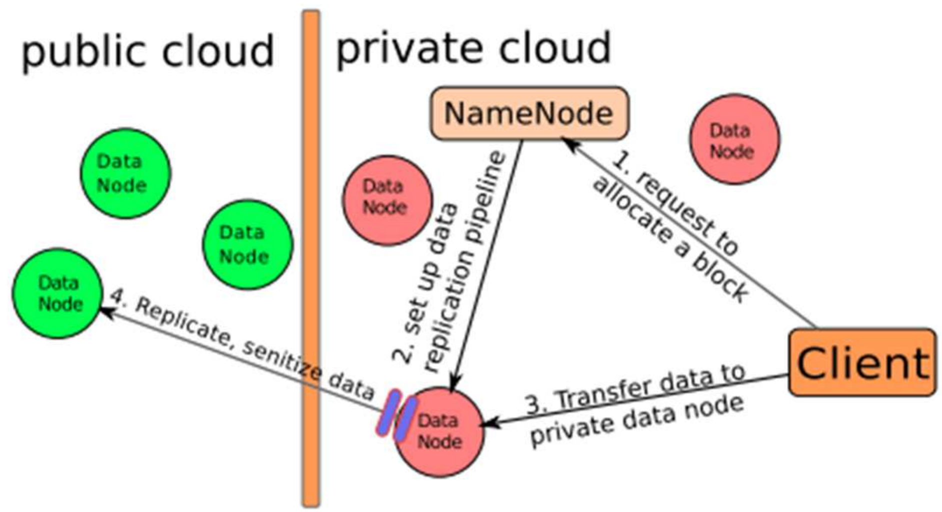
Sedic – Design



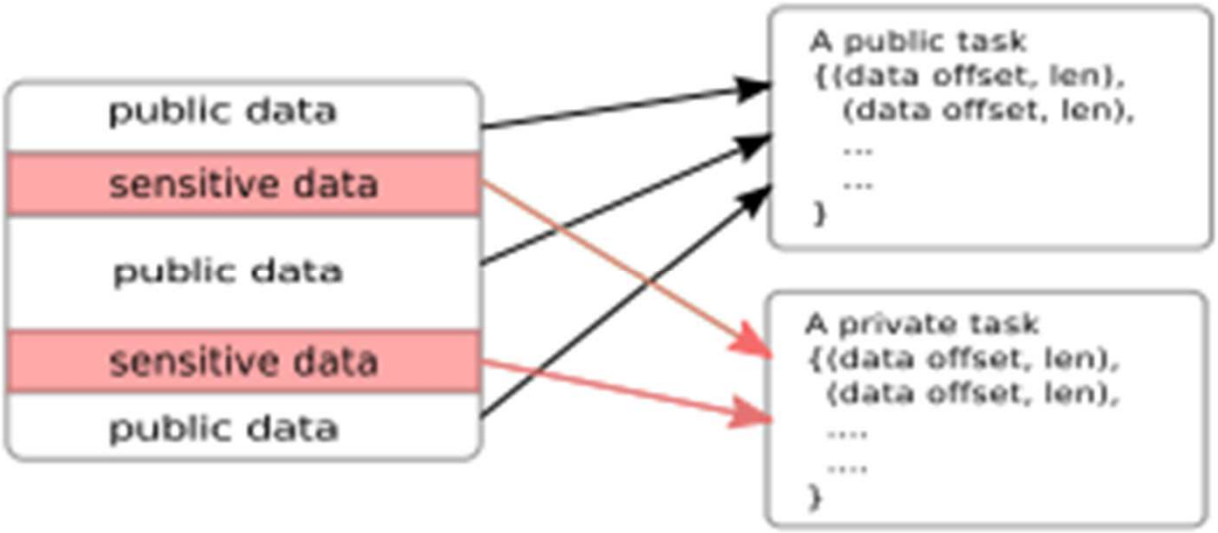
Sedic – Data Labeling and Replication

Data Labeling

Data Replication



Sedic – Map Task Management

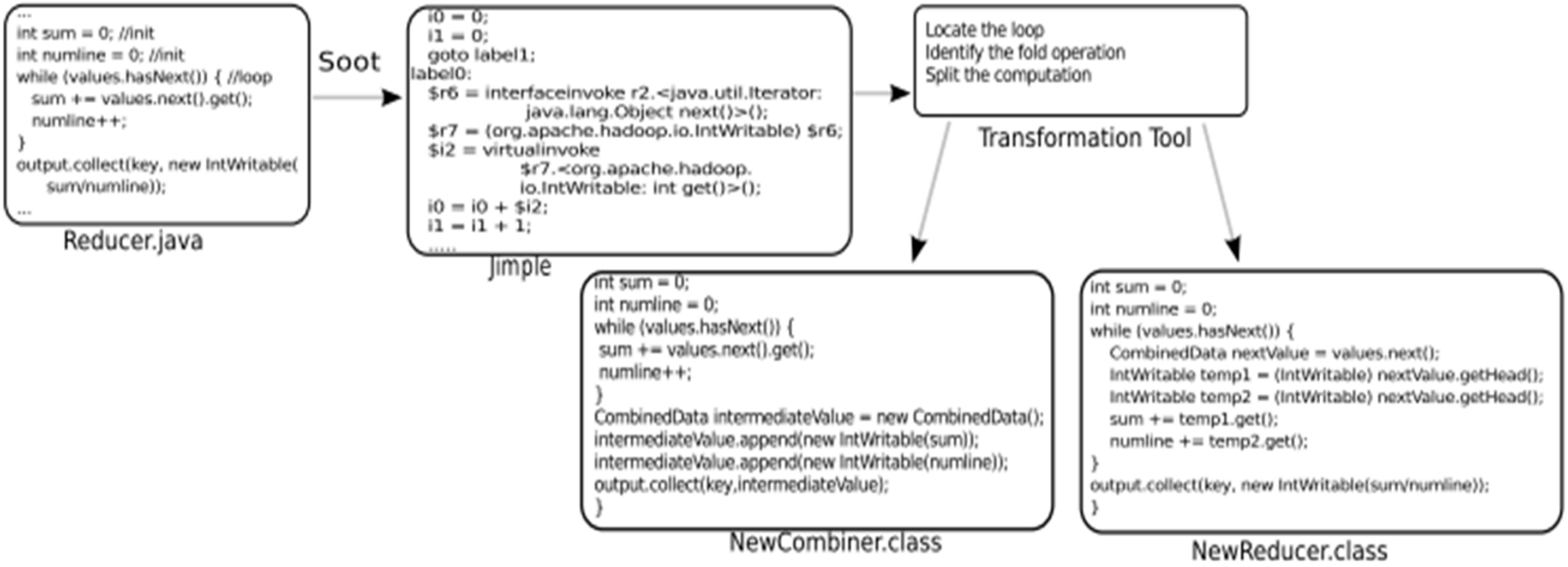


Sedic – Reduction Planning

- Move all public cloud Map outputs to private cloud
  - Very large inter-cloud communication
- User sets an upper limit for bandwidth and delay related with inter-cloud data transfer
  - Scheduler stops assigning Map's to public clouds once limit is reached
  - Constrains amount of public cloud computation

- Let public cloud perform Reduce too
  - Leverage associative and commutative properties of fold loop's in Reduce
- Extract loops to create Combiners that process data on public clouds

Sedic – Automatic Reducer Analysis and Transformation



Conclusions

- Sedic provides a privacy-aware hybrid computing paradigm
- Sedic schedules Map's such that tasks on private clouds operate on sensitive data while tasks on public clouds operate on non-sensitive data
- Sedic automatically extracts Combiner's from Reduce functions that allow public clouds to process data