Towards smart file-based data stores

Reggie Cushing
CS3 2017

Motivation

- Data is huge.
- Data is collected mainly for processing.
- Steps in processing data:
  - Collecting and storing raw data.
  - Cleaning and restoring data.
  - Write/apply data analyses workflows to data.
  - Store intermediate and resultant data.
  - Goto step 2.
- Combine compute & data to streamline processing.

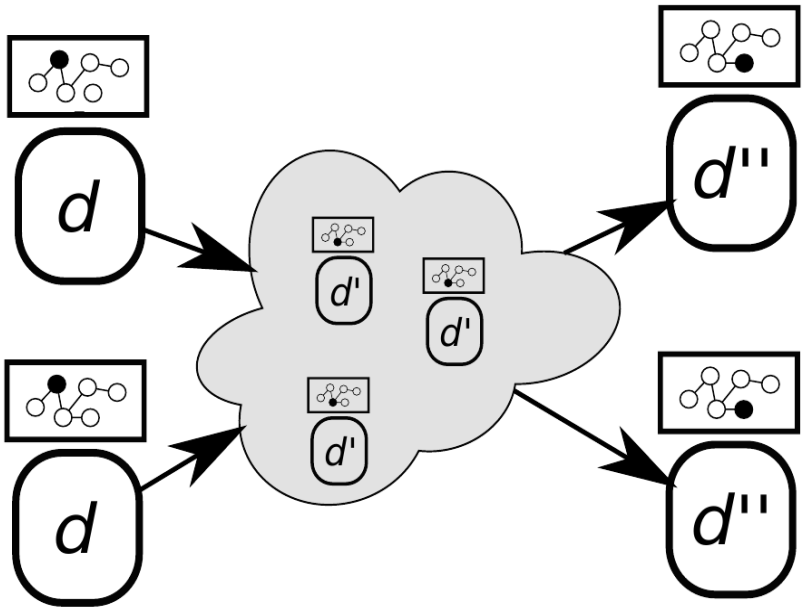
Combining file and compute

- How to combine data store with compute any backend?
  - A data store can manage *resources* instead of actual files.
  - A *resource* is an endpoint that either points to a data file or a *description* of how to generate the data file.
  - Through the *description,* the data store backend can generate the file on the fly.

Networking compute and data

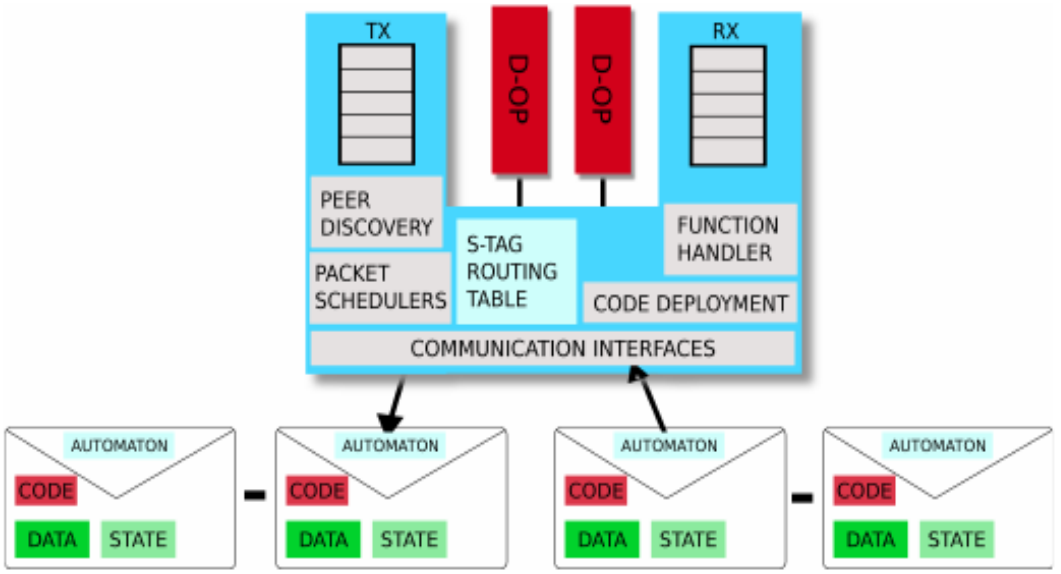
- An overlay network combines *data nodes* and *compute nodes.*
- Nodes discover each others' capabilites.
- If a data store can not satisfy a *resource* request, an announcement is done on the network.
- Capable compute nodes reply.

Generating data



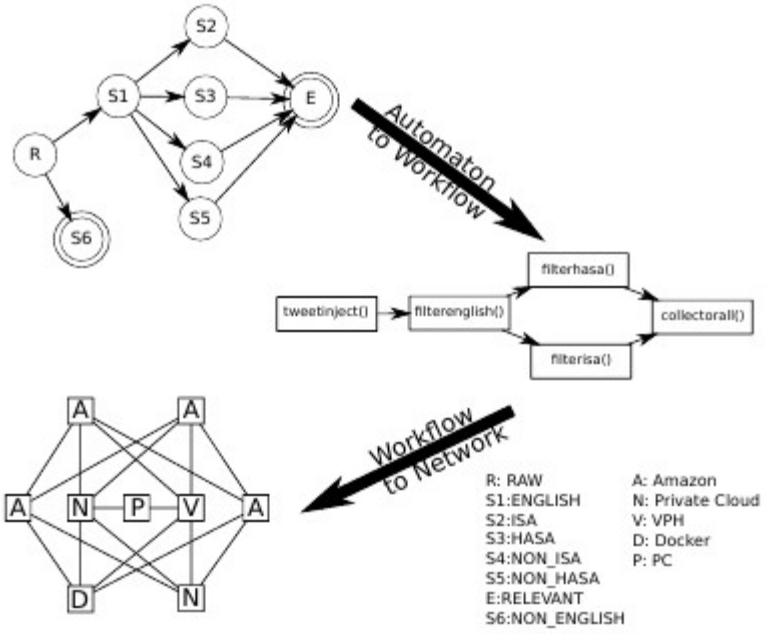
- A *resource* file is transformed from meta data to data in the overlay network

Nodes as data routers



- Compute nodes and data nodes can communicate between themselves to move and process data.

Workflow as a resource



- A *resource* can represent a workflow whereby multiple compute nodes are involved in producing the result.

Conclusion

- Combining data and compute into one framework will streamline data collection and analyses.
- Ability to regenerate data on the fly can reduce data footprint.
  - A data store can optimize to delete any *redundant* data.
- Real-time data (e.g. weather) can be represented as a *resource* also and generated just-in-time for processing.