Distributed Feature Calculation with Pydoop

Simone Leo

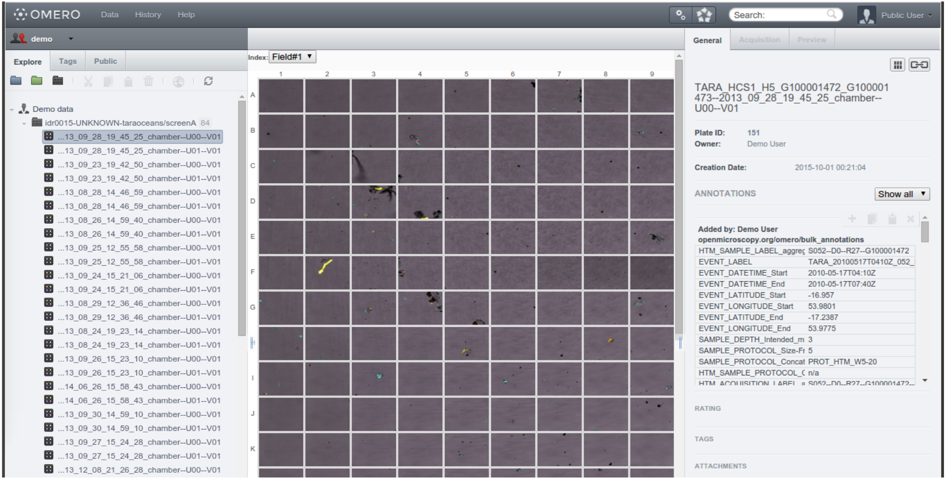
OME Tuesday Meeting

09/02/2016

The Problem

- Compute features on 37 TB of image data from the **IDR project**:
- http://idr-demo.openmicroscopy.org
  - 11 genetic/siRNA screens from published papers
- http://idr-demo.openmicroscopy.org/mito
  - Mitocheck screen (http://mitosys.org)
- http://idr-demo.openmicroscopy.org/tara
  - Tara Oceans study (http://oceans.taraexpeditions.org/en)

IDR on OMERO



- Demo data
  - idr0015-UNKNOWN-taraoceans/screenA 84
    - ...13_09_28_19_45_25_chamber--U00--V01
    - ...13_09_28_19_45_25_chamber--U01--V01
    - ...13_09_23_19_42_50_chamber--U00--V01
    - ...13_09_23_19_42_50_chamber--U01--V01
    - ...13_08_28_14_46_59_chamber--U00--V01
    - ...13_08_28_14_46_59_chamber--U01--V01
    - ...13_08_26_14_59_40_chamber--U00--V01
    - ...13_08_26_14_59_40_chamber--U01--V01
    - ...13_09_25_12_55_58_chamber--U00--V01
    - ...13_09_25_12_55_58_chamber--U01--V01
    - ...13_09_24_15_21_06_chamber--U00--V01
    - ...13_09_24_15_21_06_chamber--U01--V01
    - ...13_08_29_12_36_46_chamber--U00--V01
    - ...13_08_29_12_36_46_chamber--U01--V01
    - ...13_08_24_19_23_14_chamber--U00--V01
    - ...13_08_24_19_23_14_chamber--U01--V01
    - ...13_09_26_15_23_10_chamber--U00--V01
    - ...13_09_26_15_23_10_chamber--U01--V01
    - ...14_06_26_15_58_43_chamber--U00--V01
    - ...14_06_26_15_58_43_chamber--U01--V01
    - ...13_09_30_14_59_10_chamber--U00--V01
    - ...13_09_30_14_59_10_chamber--U01--V01
    - ...13_09_27_15_24_28_chamber--U00--V01
    - ...13_09_27_15_24_28_chamber--U01--V01
    - ...13_12_08_21_26_28_chamber--U00--V01

TARA_HCS1_H5_G100001472_G100001473--2013_09_28_19_45_25_chamber--U00--V01

Plate ID: 151
Owner: Demo User

Creation Date: 2015-10-01 00:21:04

ANNOTATIONS

Added by: Demo User
openmicroscopy.org/omero/bulk_annotations

HTM_SAMPLE_LABEL_aggreg	S052--D0--R27--G100001472
EVENT_LABEL	TARA_20100517T0410Z_052_
EVENT_DATETIME_Start	2010-05-17T04:10Z
EVENT_DATETIME_End	2010-05-17T07:40Z
EVENT_LATITUDE_Start	-16.957
EVENT_LONGITUDE_Start	53.9801
EVENT_LATITUDE_End	-17.2387
EVENT_LONGITUDE_End	53.9775
SAMPLE_DEPTH_Intended_m	3
SAMPLE_PROTOCOL_Size-Fr	5
SAMPLE_PROTOCOL_Concat	PROT_HTM_W5-20
HTM_SAMPLE_PROTOCOL_(	n/a
HTM_ACQUISITION_LABEL_a	S052--D0--R27--G100001472--

How to . . .

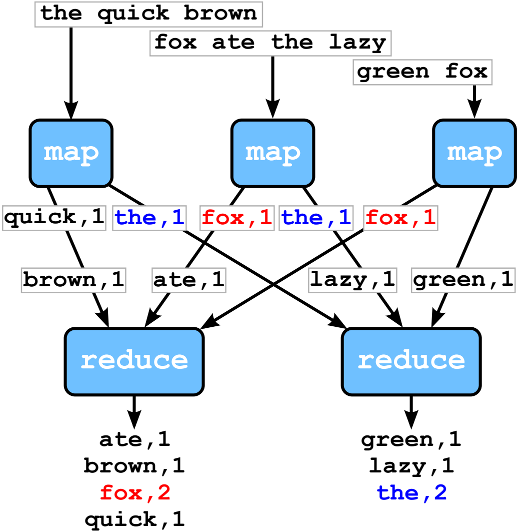
- Get data out of OMERO
  - OMERO script dumps individual planes to disk
    - As image (e.g., TIFF)
    - As .npy
  - OMERO script gets file paths, Bio-Formats reads images
- Convert data to a format that Python can read
  - Image & .npy already OK
  - Bio-Formats wrappers (python-bioformats, PIMS)
  - Avro
- Distribute the workload
  - Manually ☹
  - Multiprocessing
  - Hadoop

MapReduce and Hadoop

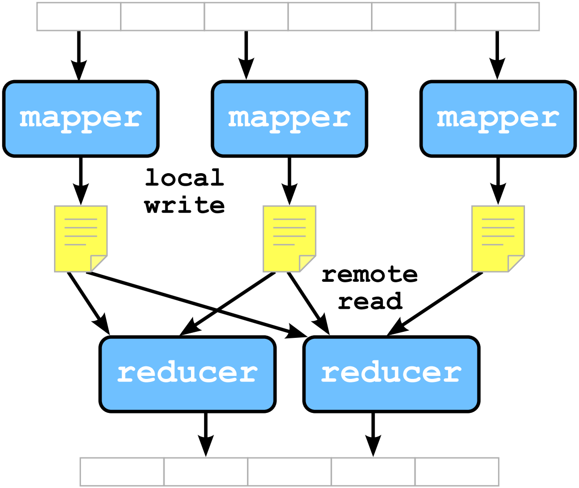
- Designed for data-driven applications
- Abstraction layer that hides parallelization details
- The developer writes two functions: `map` and `reduce`

```
function MAP($key, value$)
    for $word \leftarrow value$ do
        emit($word$, 1)
function REDUCE($key, values$)
    $count \leftarrow 0$
    for $v \leftarrow values$ do
        $count \leftarrow count + v$
    emit($key, count$)
```

MapReduce — Word Count



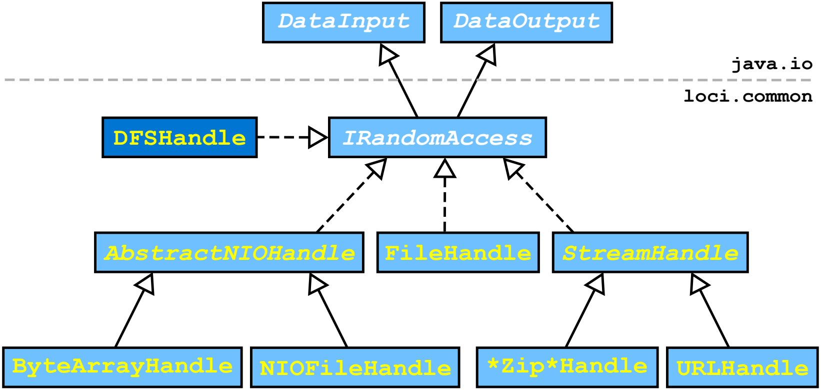
MapReduce — Execution Model



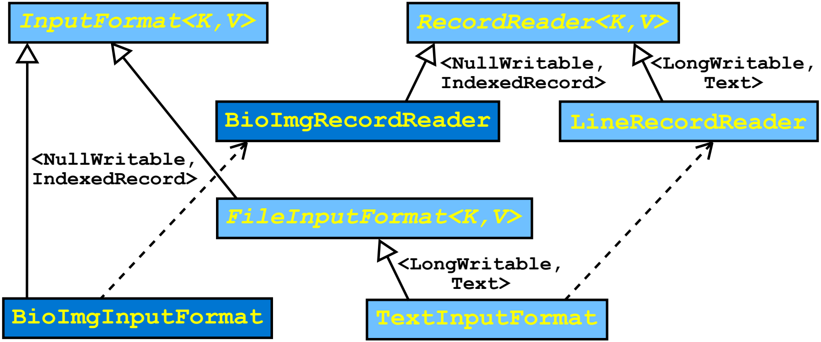
Pydoop-features

- Pydoop (http://crs4.github.io/pydoop)
  - Python Hadoop API
  - Brings Python's wealth of scientific libraries to Hadoop
  - ≈ 100 downloads / day
  - Support for transparent Avro serialization
- Pydoop-features (https://github.com/simleo/pydoop-features)
  - Bio-Formats-based Hadoop input format for bio images
    - Uses a custom HDFS-aware Bio-Formats handler
  - Pydoop-based MapReduce features computation
    - Map-only job
    - uses WND-CHARM
  - Avro-based serialization of images and output features

DFS File Handler for Bio-Formats



Hadoop Input Format for Bio Images



Preliminary Results

- Feature extraction on `dvs/kschleicher/140119`
  - 59 `.dv` files
  - Single-series, 180 image planes, 51 MB each
- 1. Input format assigns whole series to each map task
  - Analyzed all files in parallel on 59 CPU cores
  - Running time $\approx$ 33 min, 6% worse than ideal
- 2. Input format assigns plane range to each map task
  - 118 cores, each processing 90 out of 180 planes
  - Running time $\approx$ 19 min, 22% worse than ideal
  - optimal distribution level somewhere in between the whole series to single plane spectrum

Current Issues and Limitations

- Code assumes that all series have the same core metadata
- RGB support still WIP
- DFSHandle actually performs bad!
  - Hadoop is designed for streaming access to data
  - TIFF requires arbitrary seeks
  - Could be fixed by adding a caching layer to Bio-Formats
  - However, it does **not** have a significant impact on computationally intensive applications like WND-CHARM

References

- Jeffrey Dean and Sanjay Ghemawat. “MapReduce: simplified data processing on large clusters”. In: *Communications of the ACM* 51.1 (2008), pp. 107–113. DOI: 10.1145/1327452.1327492

- S. Leo and G. Zanetti. “Pydoop: a Python MapReduce and HDFS API for Hadoop”. In: *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing*. 2010, pp. 819–825

/over