Storm-on-YARN: Convergence of Low-Latency and Big-Data

Andrew Feng

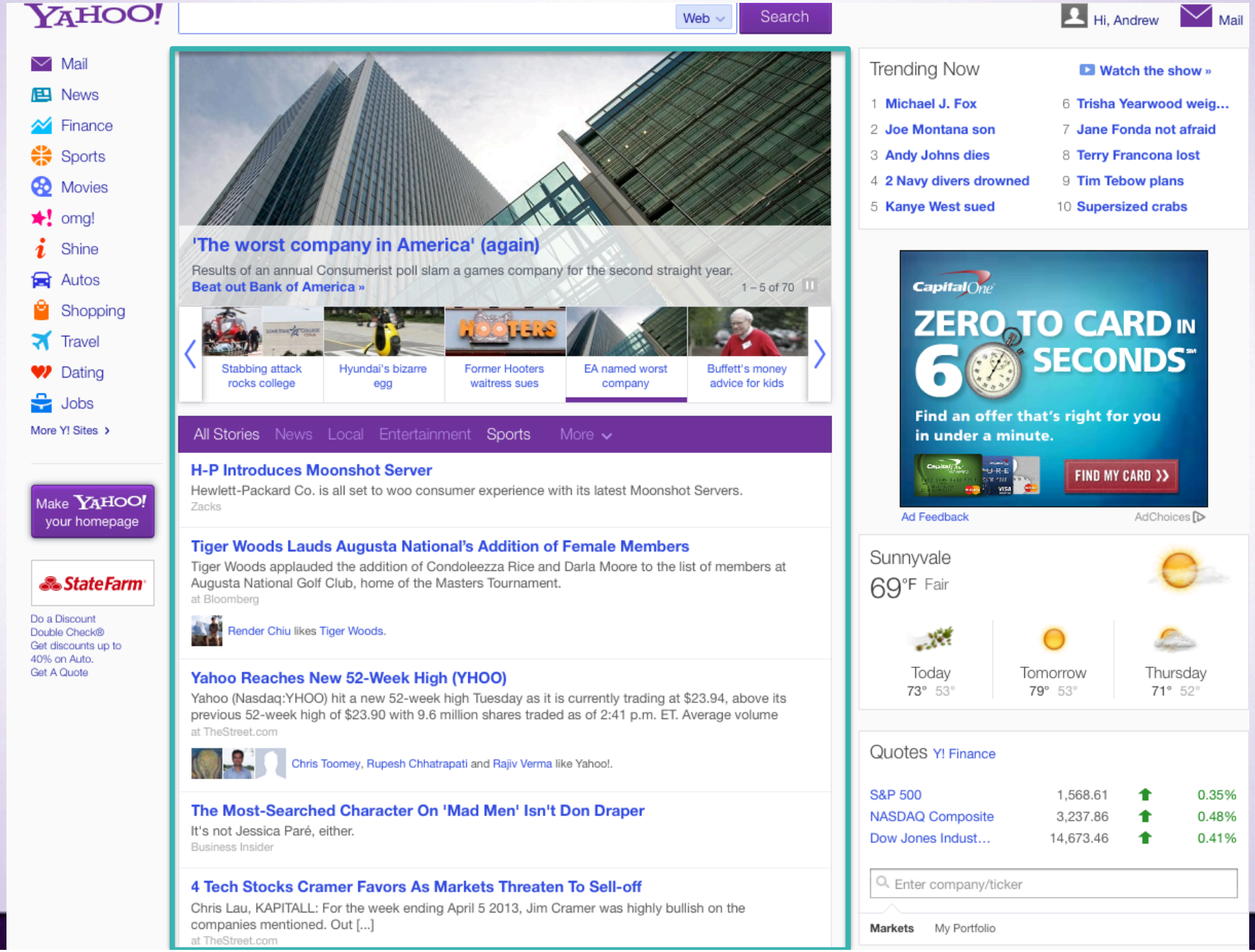
Self Introduction

- Current
  - Distinguished Architect, Yahoo! Hadoop Team
  - Core contributor at Storm project
- Past
  - Online advertisement
  - Personalization
  - Serving containers
  - Cloud services
  - NoSQL database
  - Application server

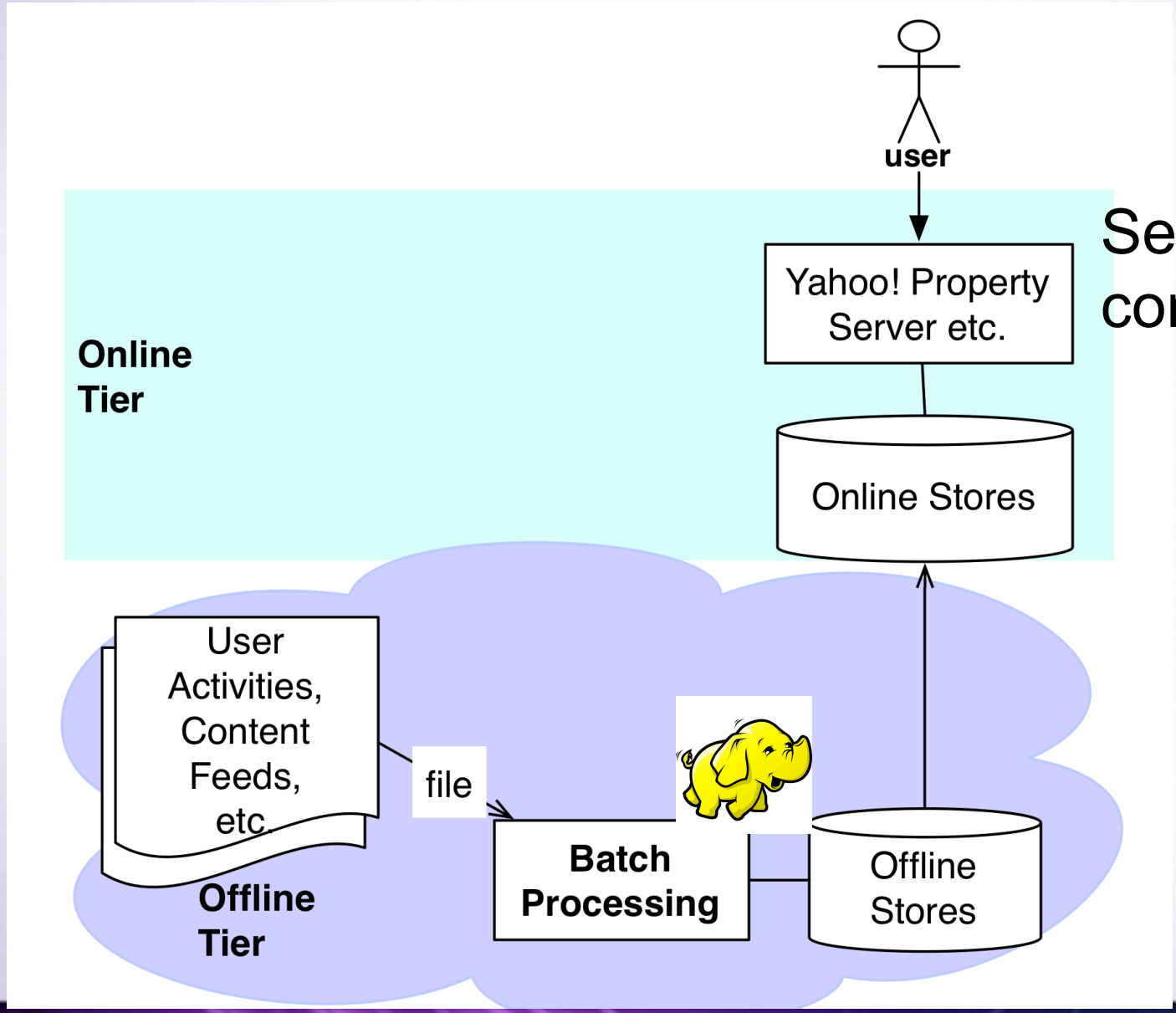
Agenda

- **Business motivation**
- Technical overview
- Open source

Yahoo!: Personalized Web



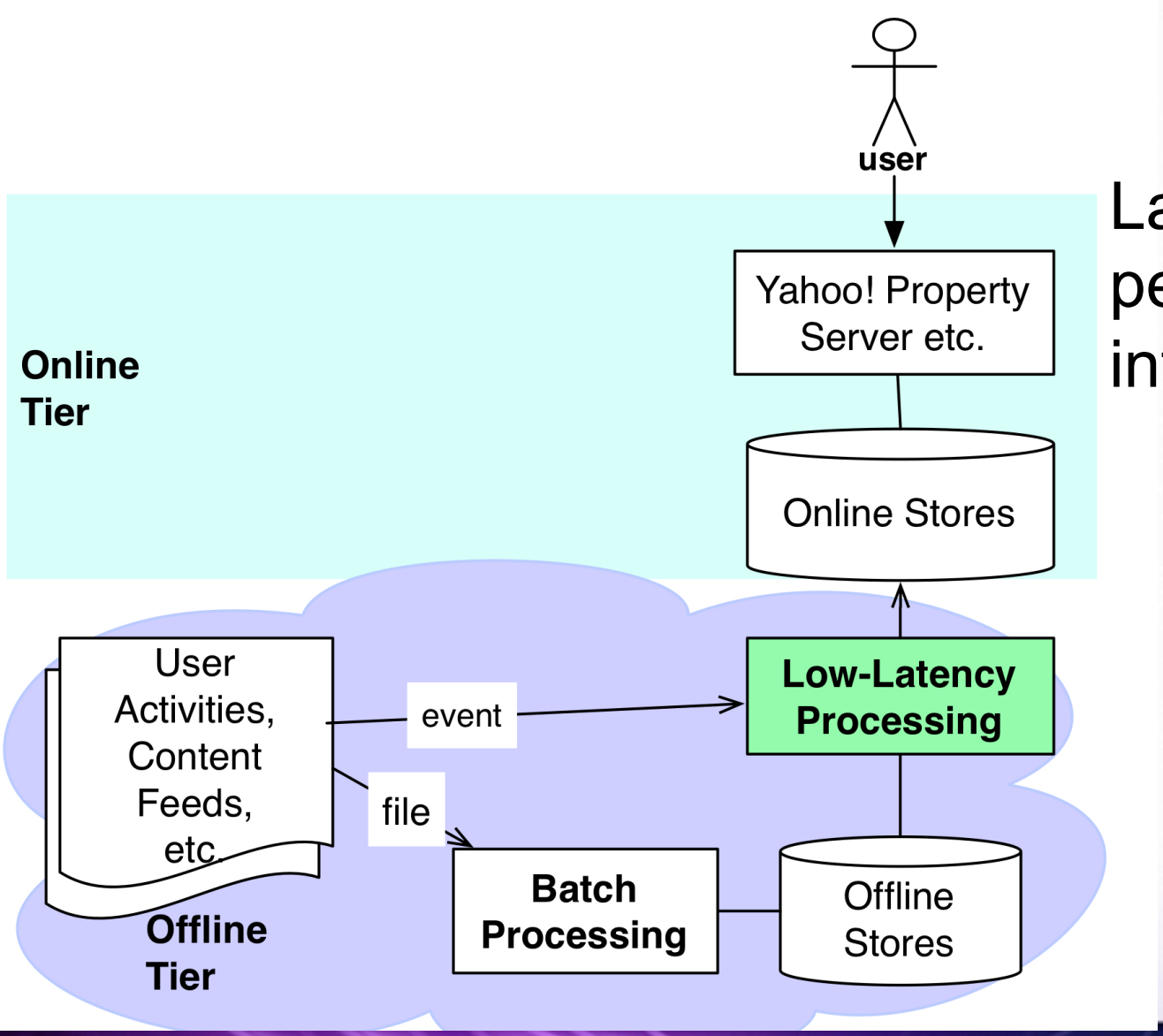
Personalization w/ Hadoop



Select relevant content & ads

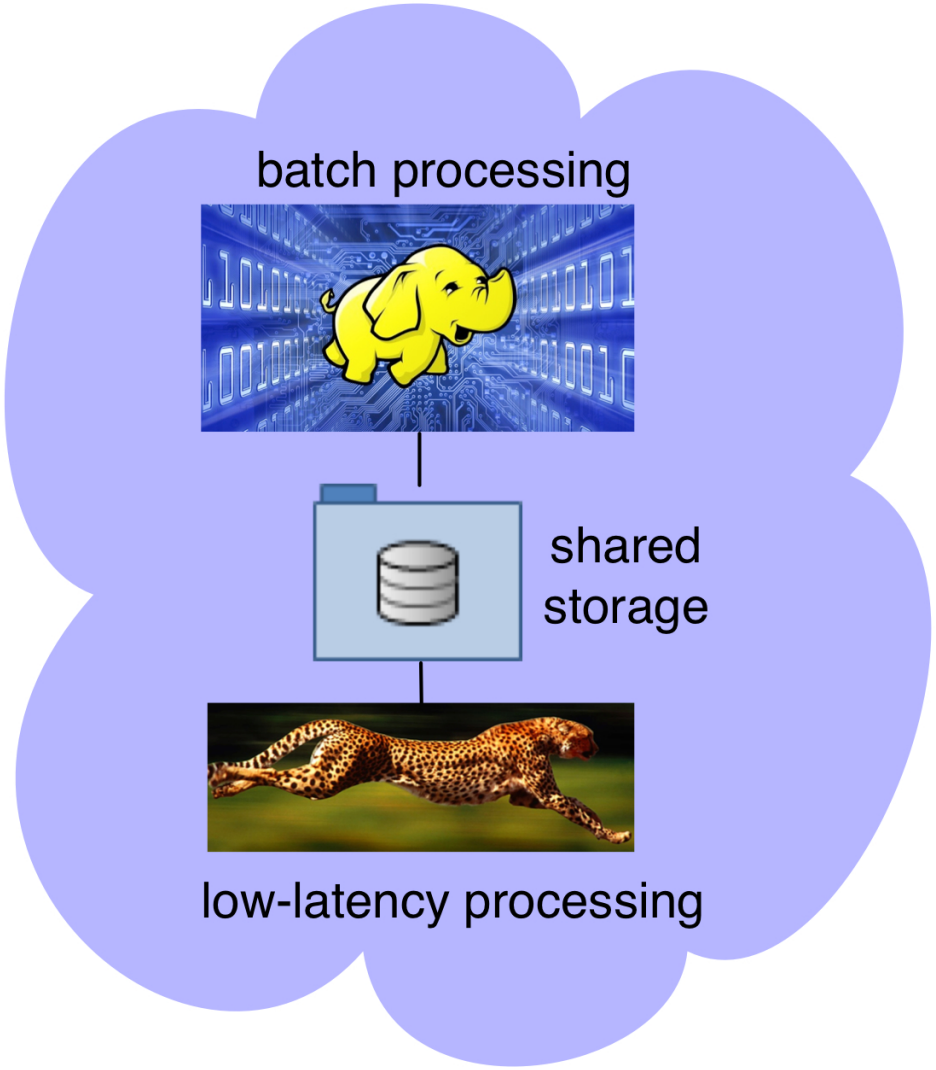
Understand user & content/ads

Personalization w/ Low-Latency



Latest content per current interests

Big Data + Low Latency: Design Pattern

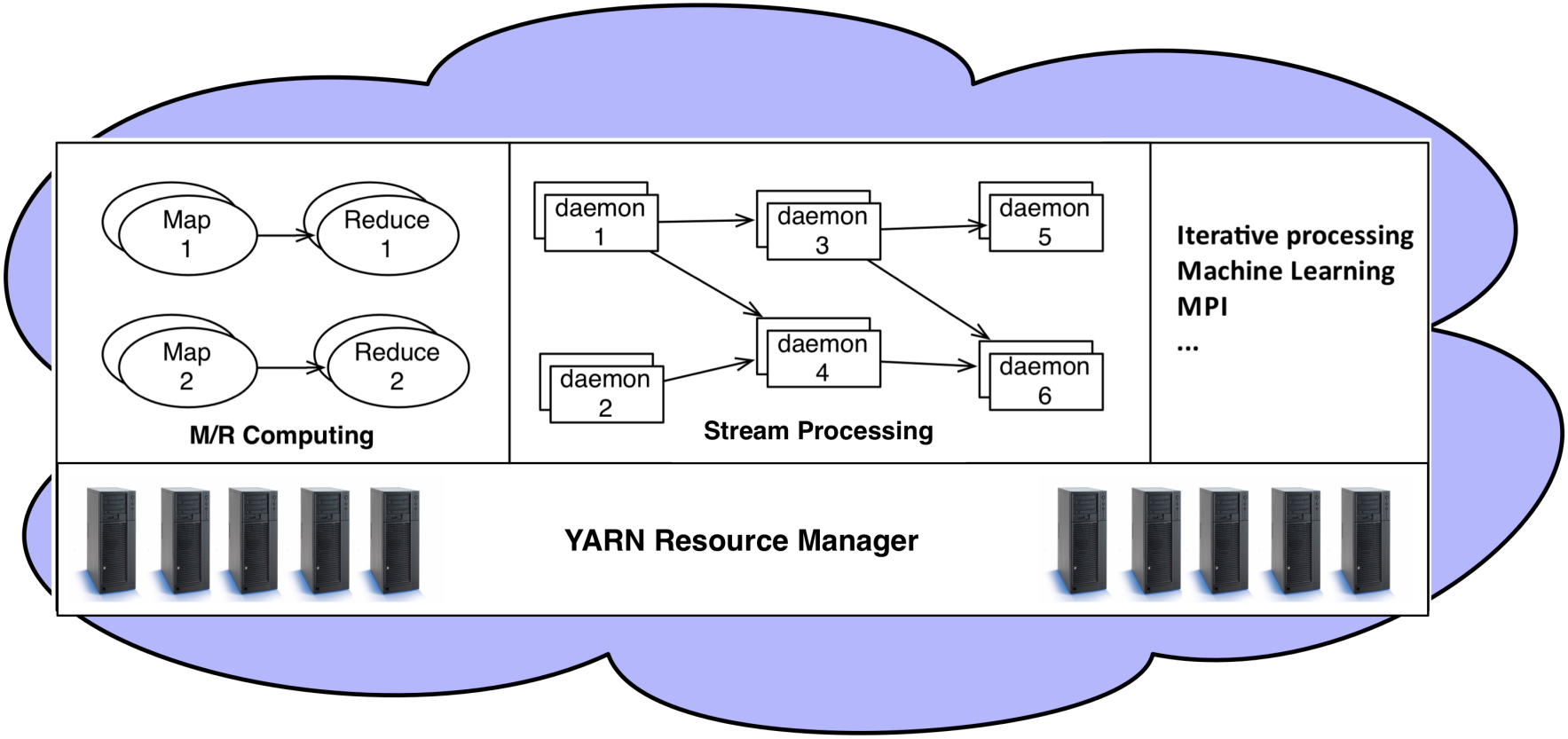


- Personalization
- Ad targeting
- Reporting
- Ad budgeting
- Fraud detection
- Trending topics

Agenda

- Business motivation
- **Technical overview**
- Open source

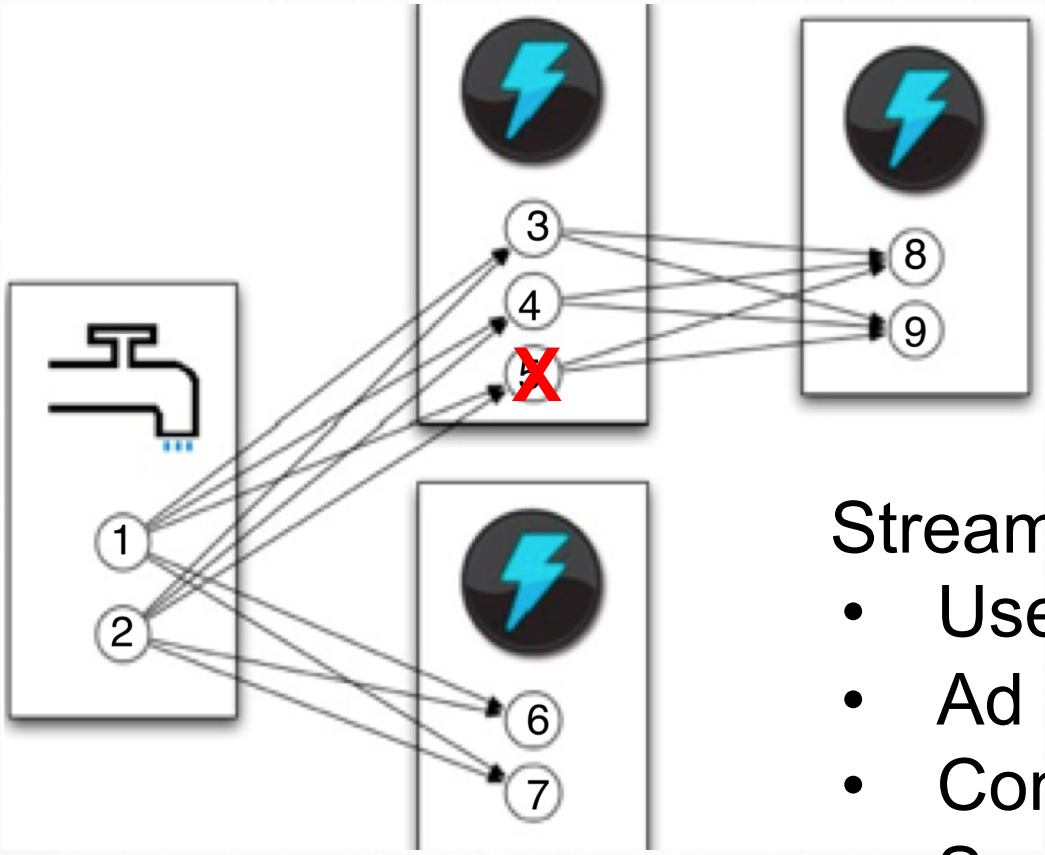
Hadoop YARN: MapReduce & Beyond



- Yahoo! deployed YARN into 30k+ nodes in production.
- YARN Apps … MapReduce, Storm, etc.

Storm: Distributed Stream Processing

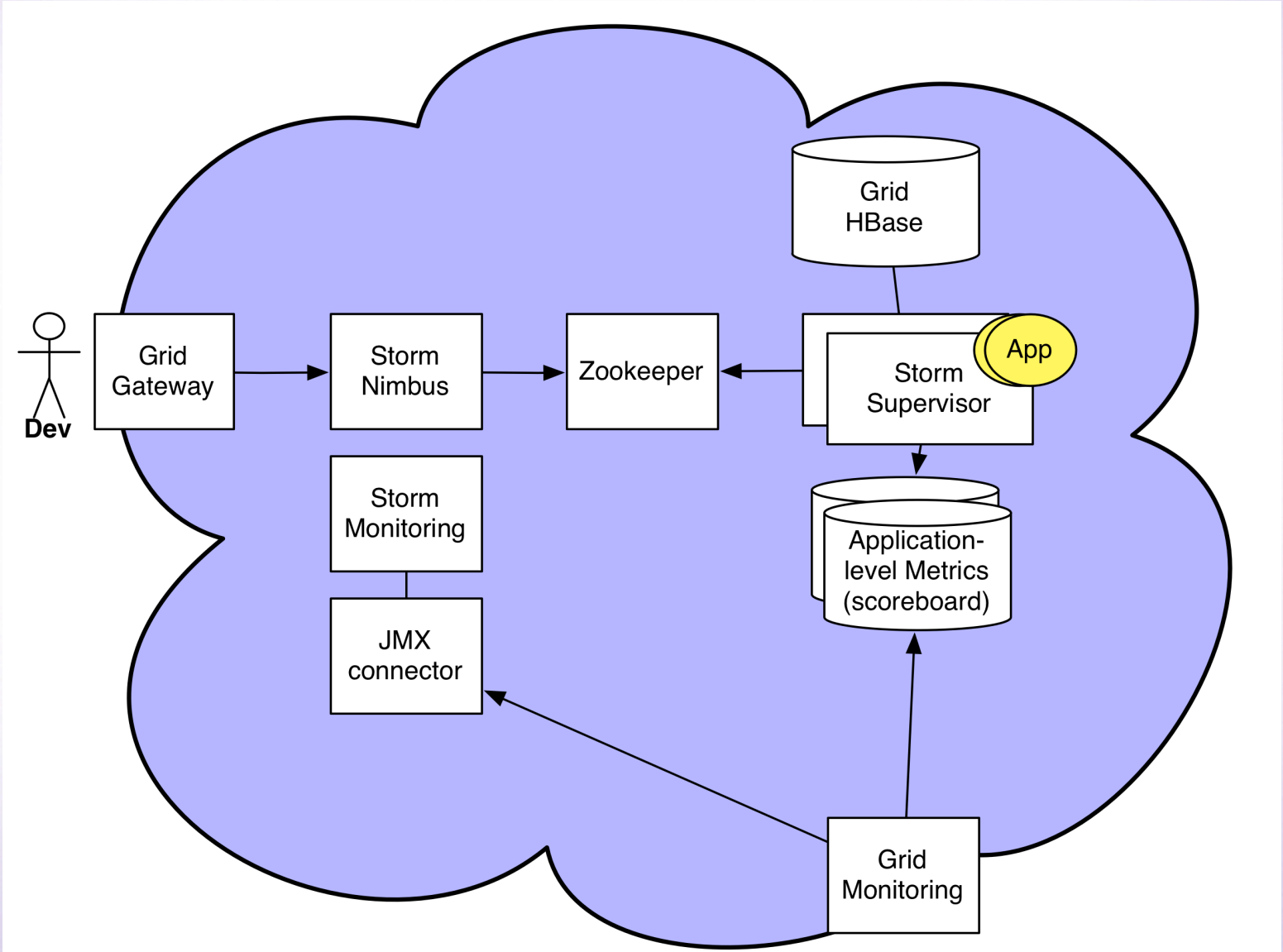
https://github.com/nathanmarz/storm



Streams

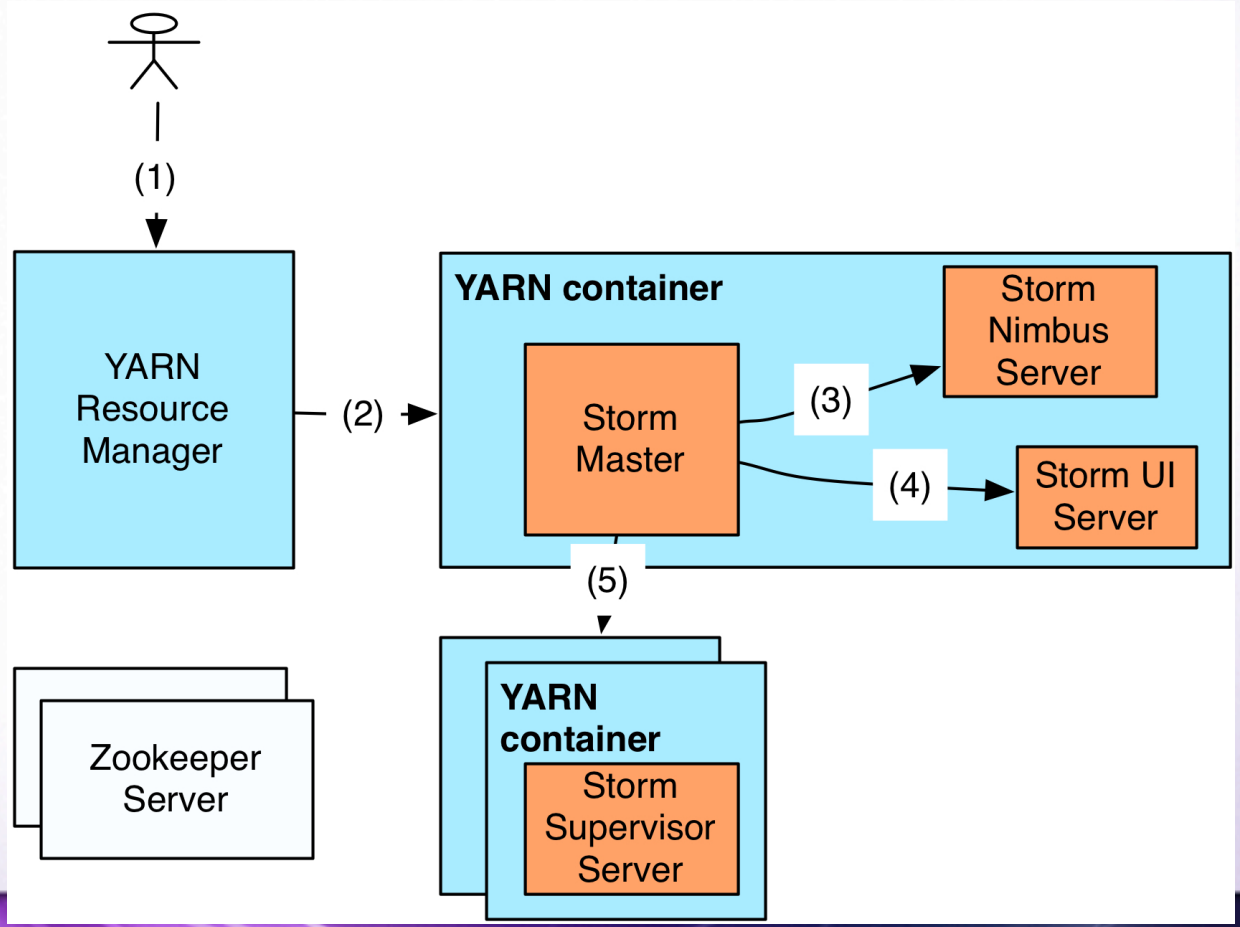
- User activities
- Ad beacons
- Content feeds
- Social feeds
- …

Storm Clusters on Hadoop Grid



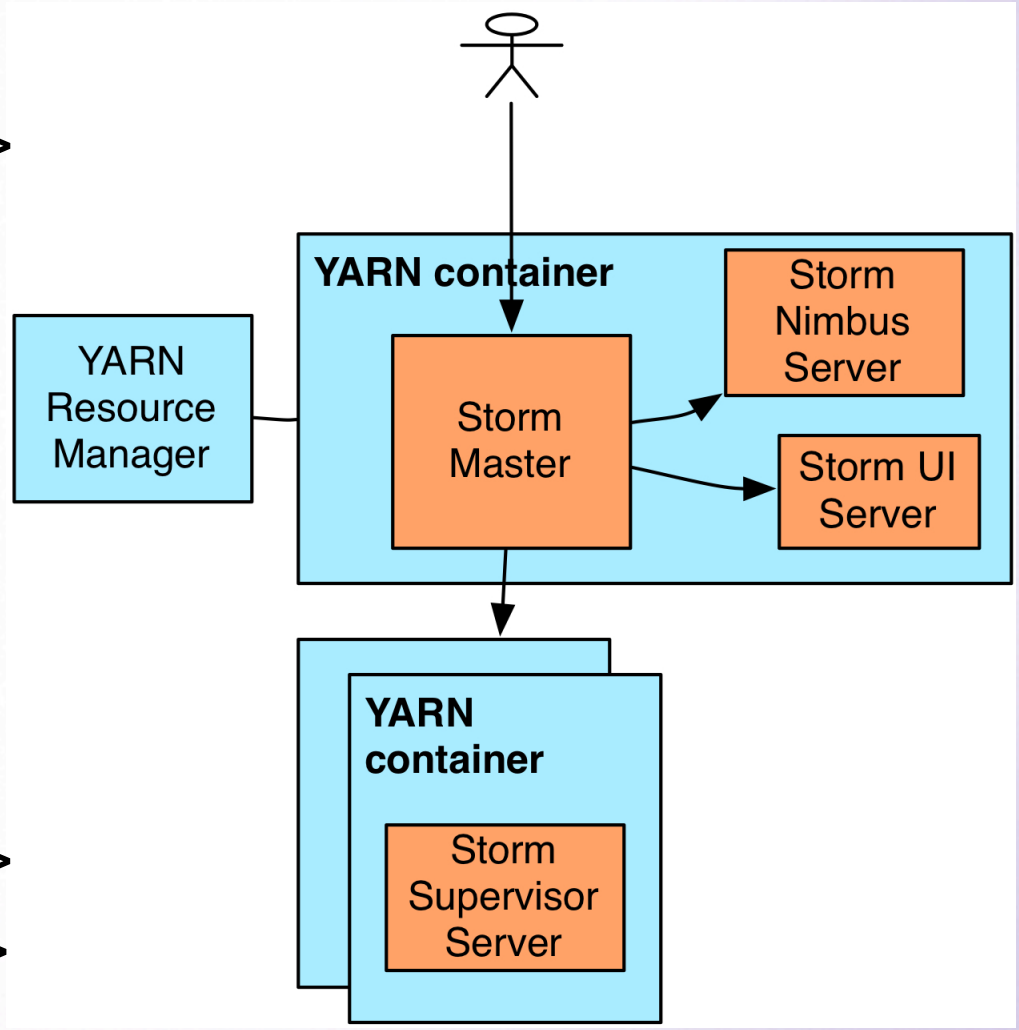
Storm-YARN: Launch Cluster

- storm-yarn launch <conf>
  - *Initial # of supervisors*
  - *memory size of allocated container*
- Result: <appID> of the newly launched Storm master



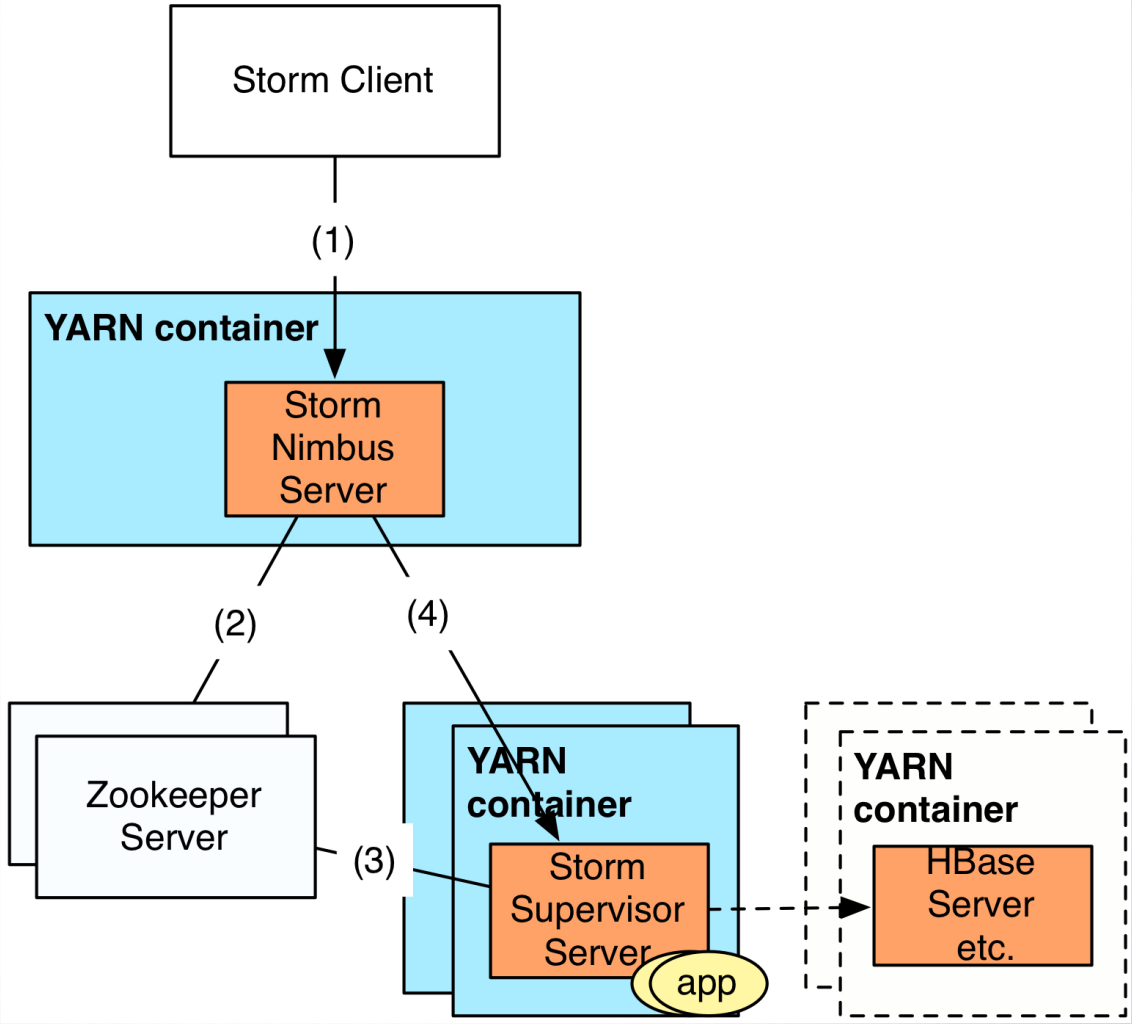
Storm-YARN: Manage Cluster

1. addSupervisors <appID> <count>
2. getStormConfig <appID>
3. setStormConfig <appID>
4. startNimbus <appID>
5. stopNimbus <appID>
6. startUI <appID>
7. stopUI <appID>
8. startSupervisors <appID>
9. stopSupervisors <appID>

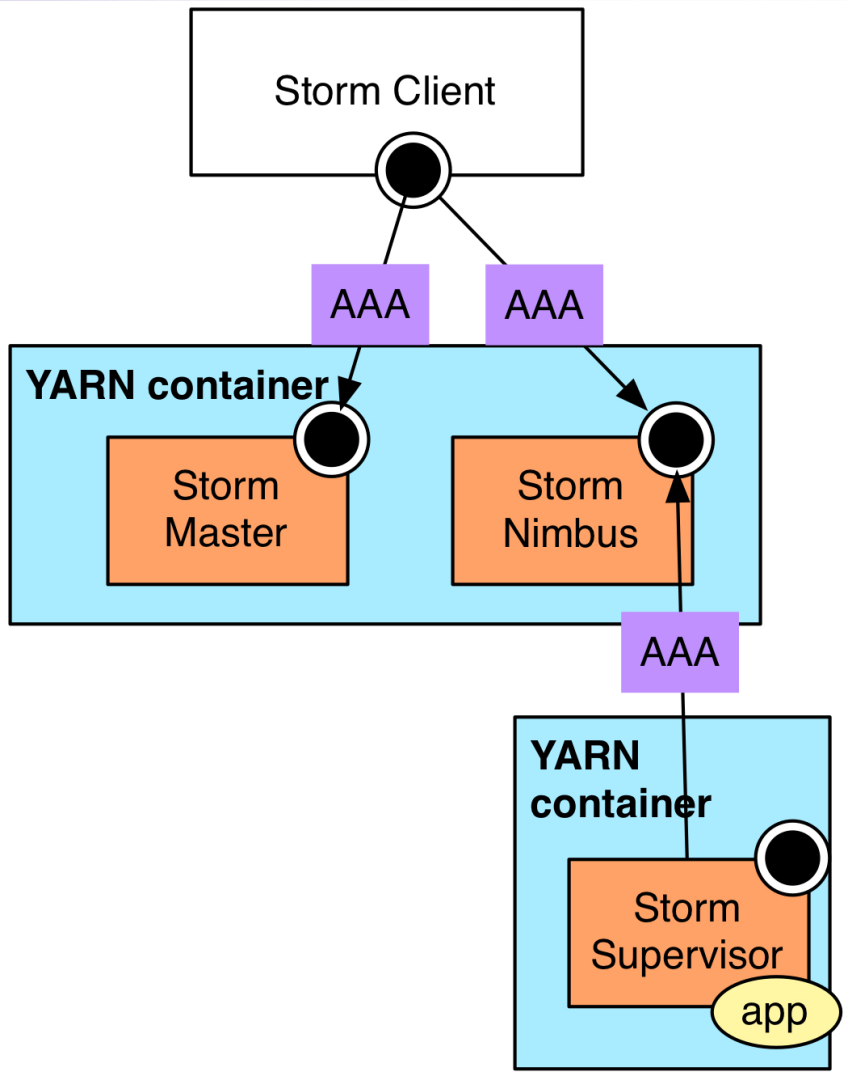


Storm-YARN: Deploy Apps

storm jar <appJar>



Authentication/Authorization/Audit



- Authentication plugins
  - Digest
  - Kerberos (soon)
  - None
  - *Bring your own*
- Authorization plugins
  - Accept all
  - Limited operations only
  - User whitelist
  - *Bring your own*
- Audit
  - Access log

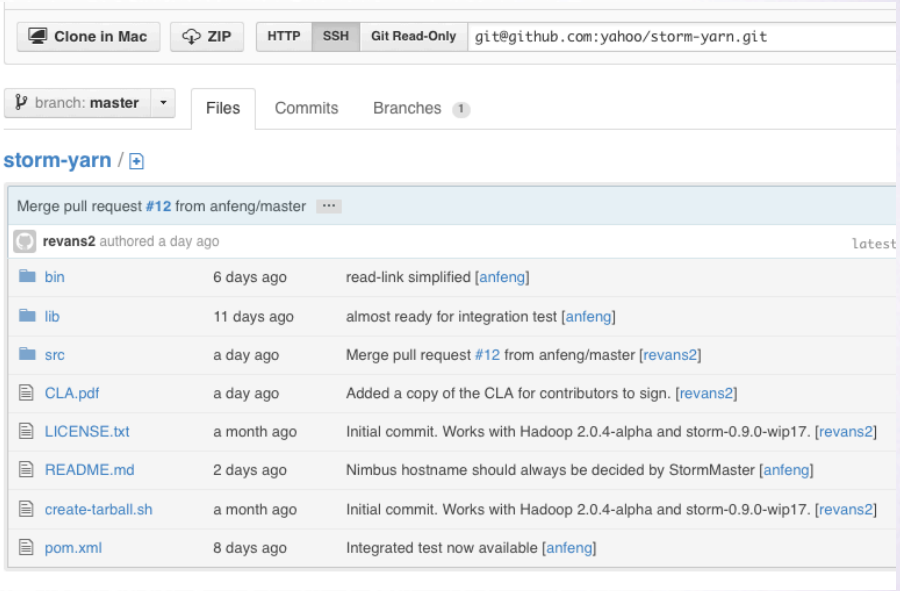
Agenda

- Business motivation
- Technical overview
- **Open source**

Storm-YARN: Open Source

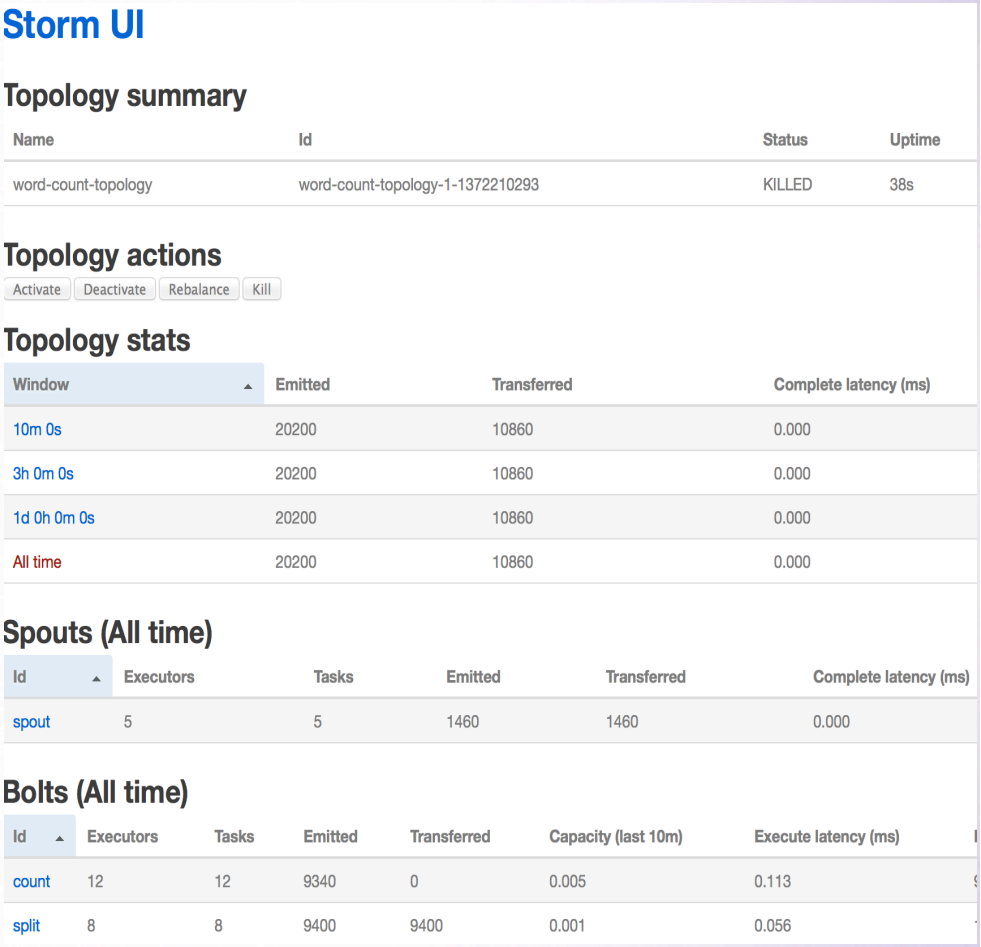
- Code released for early access
  - under the Apache 2.0 License
  - move to apache.org later
- Welcome contribution!
  - Submit proposals
  - Sign Apache style CLA
  - Submit git pull requests

https://github.com/yahoo/storm-yarn



Storm-YARN: **mvn test**

1. **storm-yarn launch**
   - ./conf/storm.yaml --stormZip lib/storm.zip --appname storm-on-yarn-test --output target/appId.txt
2. **storm-yarn getStormConfig**
   - ./conf/storm.yaml --appId application_1372121842369_0001 --output ./lib/storm/storm.yaml
3. **storm jar**
   - lib/storm-starter-0.0.1-SNAPSHOT.jar
   - storm.starter.WordCountTopology
   - word-count-topology
4. **storm kill**
   - word-count-topology
5. **storm-yarn shutdown**
   - ./conf/storm.yaml --appId application_1372121842369_0001



Storm UI

Topology summary

Name	Id	Status	Uptime
word-count-topology	word-count-topology-1-1372210293	KILLED	38s

Topology actions

Activate | Deactivate | Rebalance | Kill

Topology stats

Window	Emitted	Transferred	Complete latency (ms)
10m 0s	20200	10860	0.000
3h 0m 0s	20200	10860	0.000
1d 0h 0m 0s	20200	10860	0.000
All time	20200	10860	0.000

Spouts (All time)

Id	Executors	Tasks	Emitted	Transferred	Complete latency (ms)
spout	5	5	1460	1460	0.000

Bolts (All time)

Id	Executors	Tasks	Emitted	Transferred	Capacity (last 10m)	Execute latency (ms)
count	12	12	9340	0	0.005	0.113
split	8	8	9400	9400	0.001	0.056

Storm-YARN: Deployment

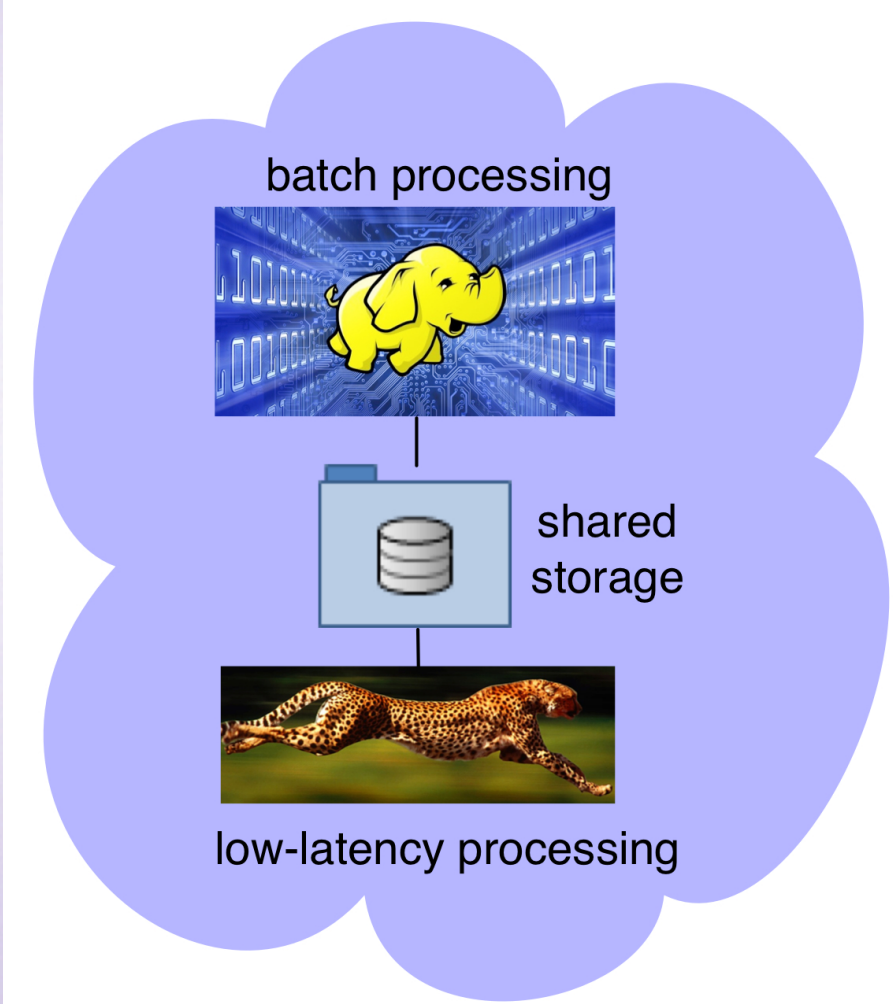
Install Storm S/W

1. hadoop fs –put storm.zip /lib/storm/<version>/storm.zip

Apply Storm-YARN

2. storm-yarn launch
   ➔ <appID>
3. storm-yarn getStormConfig <appID>
   ➔ <storm.yaml>
4. storm jar <appJar>

Conclusion



- YARN empowers the emergence of big-data & low-latency processing
- Yahoo! open source:
  - Storm-yarn @ github/yahoo
  - Spark-yarn @ spark-project.org


MORE THAN EVER BEFORE
?
Questions
Y!
Hadoop@Yahoo!