BIG DATA PROCESSING IN THE CLOUD: A HYDRA/SUFIA EXPERIENCE

Collin Brittle
Zhiwu Xie

Helsinki
June 2014
WHAT?
WHY?
SENSORS
SMART INFRASTRUCTURE

- Acceleration
- acoustics
- Temperature
- strain
- wind and flow
- load
DATA SHARING

• Encourage exploratory and multidisciplinary research

• Foster open and inclusive communities around
  • modeling of dynamic systems
  • structural health monitoring and damage detection
  • occupancy studies
  • sensor evaluation
  • data fusion
  • energy reduction
  • evacuation management
  • …
CHARACTERIZATION

- Compute intensive
- Storage intensive
- Communication intensive
- On-demand
- Scalability challenge
COMPUTE INTENSIVE

• About 6GB raw data per hour

• Must be continuously processed, ingested, and further processed

• User-generated computations

• Must not interfere with data retrieval
STORAGE INTENSIVE

• SEB will accumulate about 60TB of raw data per year
• To facilitate researchers, we must keep raw data for an extended period of time, e.g., \( \geq 5 \) years
• VT currently does not have an affordable storage facility to hold this much data
• Within XSEDE, only TACC’s Ranch can allocate this much storage
• What if hundreds of researchers around the world each tried to download hundreds of TB of our data?
ON DEMAND

- Explorative and multidisciplinary research cannot predict the data usage beforehand
SCALABILITY

• How to deal with these challenges in a scalable manner?
BIG DATA + CLOUD

• Affordable
• Elastic
• Scalable
FRAMEWORK REQUIREMENTS

• Mix local and remote content

• Support background processing

• Be distributable
FRAMEWORK REQUIREMENTS

• Mix local and remote content

• Support background processing

• Be distributable
OBJECTS AND DATASTREAMS

Local Object

Meta Meta File
OBJECTS AND DATASTREAMS

Local Object

Meta

Meta

File
REMOTE STORAGE

Amazon

EC2  S3  Glacier

Local Repository
FRAMEWORK
REQUIREMENTS

• Mix local and remote content

• Support background processing

• Be distributable
BACKGROUND PROCESSING

Redis

Public Server

Database

Worker

Worker

Worker

Clients
FROM QUEUES TO THE CLOUD

0100
0010

1100
0011

1010
0101

0101
0101
# FROM QUEUES TO THE CLOUD

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1010</td>
<td>1010</td>
<td>1100</td>
</tr>
<tr>
<td>0101</td>
<td>0101</td>
<td>0011</td>
</tr>
</tbody>
</table>
FROM QUEUES TO THE CLOUD

1010 1010 1100
0101 0101 0011
<table>
<thead>
<tr>
<th>1010</th>
<th>0101</th>
</tr>
</thead>
<tbody>
<tr>
<td>1010</td>
<td>0101</td>
</tr>
<tr>
<td>1100</td>
<td>0011</td>
</tr>
</tbody>
</table>

FROM QUEUES TO THE CLOUD
FROM QUEUES TO THE CLOUD

1010 1010 1010
0101 0101 0101

1100 0011
FROM QUEUES TO THE CLOUD

<table>
<thead>
<tr>
<th>1010</th>
<th>1010</th>
<th>1010</th>
</tr>
</thead>
<tbody>
<tr>
<td>0101</td>
<td>0101</td>
<td>0101</td>
</tr>
</tbody>
</table>

1100 0011

0011 1100
FROM QUEUES
TO THE CLOUD

1010  1010  1010
0101  0101  0101
<table>
<thead>
<tr>
<th>1010</th>
<th>1010</th>
<th>1010</th>
</tr>
</thead>
<tbody>
<tr>
<td>0101</td>
<td>0101</td>
<td>0101</td>
</tr>
</tbody>
</table>
FROM QUEUES TO THE CLOUD

1010 0101

1010 0101

1010 0101
FROM QUEUES TO THE CLOUD

<table>
<thead>
<tr>
<th></th>
<th>1010</th>
<th>1010</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0101</td>
<td>0101</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1010</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0101</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1111</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0000</td>
</tr>
</tbody>
</table>
FROM QUEUES TO THE CLOUD

<table>
<thead>
<tr>
<th></th>
<th>1010</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0101</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1010</td>
<td>0101</td>
</tr>
</tbody>
</table>
QUEUEING

<table>
<thead>
<tr>
<th>Name</th>
<th>Jobs</th>
</tr>
</thead>
<tbody>
<tr>
<td>audit</td>
<td>0</td>
</tr>
<tr>
<td>average</td>
<td>0</td>
</tr>
<tr>
<td>batch_update</td>
<td>0</td>
</tr>
<tr>
<td>characterize</td>
<td>0</td>
</tr>
<tr>
<td>data</td>
<td>0</td>
</tr>
<tr>
<td>event</td>
<td>0</td>
</tr>
<tr>
<td>high</td>
<td>0</td>
</tr>
<tr>
<td>low</td>
<td>0</td>
</tr>
<tr>
<td>failed</td>
<td>0</td>
</tr>
</tbody>
</table>

0 of 0 Workers Working
The list below contains all workers which are currently running a job.

<table>
<thead>
<tr>
<th>Where</th>
<th>Queue</th>
<th>Processing</th>
</tr>
</thead>
</table>

Nothing is happening right now...
QUEUING

### Queues

The list below shows all the registered queues with the number of jobs currently in the queue. Select a queue from above to view all jobs currently pending on the queue.

<table>
<thead>
<tr>
<th>Name</th>
<th>Jobs</th>
</tr>
</thead>
<tbody>
<tr>
<td>audit</td>
<td>0</td>
</tr>
<tr>
<td>average</td>
<td>1</td>
</tr>
<tr>
<td>batch_update</td>
<td>1</td>
</tr>
<tr>
<td>characterize</td>
<td>0</td>
</tr>
<tr>
<td>data</td>
<td>4</td>
</tr>
<tr>
<td>event</td>
<td>5</td>
</tr>
<tr>
<td>high</td>
<td>1</td>
</tr>
<tr>
<td>low</td>
<td>1</td>
</tr>
<tr>
<td>failed</td>
<td>0</td>
</tr>
</tbody>
</table>

### 1 of 1 Workers Working

The list below contains all workers which are currently running a job.

<table>
<thead>
<tr>
<th>Where</th>
<th>Queue</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AVERAGE</td>
<td>Sufia::Resque::MarshaledJob</td>
</tr>
</tbody>
</table>
FRAMEWORK REQUIREMENTS

• Mix local and remote content

• Support background processing

• Be distributable
FROM QUEUES TO THE CLOUD
DISTRIBUTED PROCESSING

- Database
- Public Server
- Clients
- Redis Master
- Redis Slave
- Private Server
- Private Server
- Private Server
SCALE UP

1 of 1 Workers Working

The list below contains all workers which are currently running a job.

<table>
<thead>
<tr>
<th>Where</th>
<th>Queue</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AVERAGE</td>
<td>Sufia::Resque::MarshaledJob</td>
</tr>
<tr>
<td></td>
<td></td>
<td>just</td>
</tr>
<tr>
<td></td>
<td></td>
<td>now</td>
</tr>
</tbody>
</table>
5 of 5 Workers Working

The list below contains all workers which are currently running a job.

<table>
<thead>
<tr>
<th>Where</th>
<th>Queue</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AVERAGE</td>
<td>Sufia::Resque::MarshaledJob just now</td>
</tr>
<tr>
<td></td>
<td>DATA</td>
<td>Sufia::Resque::MarshaledJob just now</td>
</tr>
<tr>
<td></td>
<td>DATA</td>
<td>Sufia::Resque::MarshaledJob just now</td>
</tr>
<tr>
<td></td>
<td>BATCH_UPDATE</td>
<td>Sufia::Resque::MarshaledJob just now</td>
</tr>
<tr>
<td></td>
<td>AVERAGE</td>
<td>Sufia::Resque::MarshaledJob just now</td>
</tr>
</tbody>
</table>
WE CHOSE SUFIA
WHAT IS SUFIA?

- Ruby on Rails framework…
- Based on Hydra…
- Using Fedora Commons…
- And Resque
FRAMEWORK REQUIREMENTS

• Mix local and remote content

• Support background processing

• Be distributable
QUESTIONS?

rotated8 (who works at) vt.edu