Debugging Spark Applications

Melike GEÇER

Supervised by
Dr. Haidar Osman
and
Prof. Oscar Nierstrasz

10.12.2019
Why is debugging hard in distributed systems?
Why is debugging hard?

Lack of tooling

Logs on the Resource Manager

Logs on the console

History Server
Why is debugging hard?  Data is always dirty

<table>
<thead>
<tr>
<th>employee_no</th>
<th>employee_name</th>
<th>designation</th>
<th>manager</th>
<th>hire_date</th>
<th>salary</th>
<th>Department_no</th>
</tr>
</thead>
<tbody>
<tr>
<td>7000</td>
<td>SMITH</td>
<td>CLERK</td>
<td>7902</td>
<td>12/17/1980</td>
<td>800</td>
<td>70</td>
</tr>
<tr>
<td>7499</td>
<td>ALLEN</td>
<td>SALESMAN</td>
<td>7698</td>
<td>2/20/1981</td>
<td>1600</td>
<td>30</td>
</tr>
<tr>
<td>7521</td>
<td>WARD</td>
<td>SALESMAN</td>
<td>7698</td>
<td>2/22/1981</td>
<td>1250</td>
<td>30</td>
</tr>
<tr>
<td>7566</td>
<td>TURNER</td>
<td>MANAGER</td>
<td>7839</td>
<td>4/2/1981</td>
<td>2975</td>
<td>20</td>
</tr>
<tr>
<td>7654</td>
<td>MARTIN</td>
<td>SALESMAN</td>
<td>7698</td>
<td>9/28/1981</td>
<td>1250</td>
<td>30</td>
</tr>
<tr>
<td>7698</td>
<td>MILLER</td>
<td>MANAGER</td>
<td>7839</td>
<td>5/1/1981</td>
<td>thousand</td>
<td>30</td>
</tr>
<tr>
<td>7782</td>
<td>CLARK</td>
<td>MANAGER</td>
<td>7839</td>
<td>6/9/1981</td>
<td>2450</td>
<td>10</td>
</tr>
<tr>
<td>7788</td>
<td>SCOTT</td>
<td>ANALYST</td>
<td>7566</td>
<td>12/9/1982</td>
<td>3000</td>
<td>20</td>
</tr>
<tr>
<td>7839</td>
<td>KING</td>
<td>PRESIDENT</td>
<td>2000</td>
<td>11/17/1981</td>
<td>5000</td>
<td>10</td>
</tr>
<tr>
<td>7844</td>
<td>TURNER</td>
<td>SALESMAN</td>
<td>7698</td>
<td>9/8/1981</td>
<td>1500</td>
<td>30</td>
</tr>
<tr>
<td>7876</td>
<td>ADAMS</td>
<td>CLERK</td>
<td>7788</td>
<td>1/12/1983</td>
<td>1100</td>
<td>20</td>
</tr>
<tr>
<td>7900</td>
<td>JAMES</td>
<td>CLERK</td>
<td>7698</td>
<td>12/3/1981</td>
<td>950</td>
<td>30</td>
</tr>
<tr>
<td>7902</td>
<td>FORD</td>
<td>ANALYST</td>
<td>7566</td>
<td>12/3/1981</td>
<td>3000</td>
<td>20</td>
</tr>
<tr>
<td>7934</td>
<td>MILLER</td>
<td>CLERK</td>
<td>7782</td>
<td>1/23/1982</td>
<td>1300</td>
<td>10</td>
</tr>
</tbody>
</table>
RQ1: How do developers debug distributed applications like Spark?

RQ2: What are the challenges that developers face when they debug such applications?

RQ3: How can we overcome these challenges?
Investigation Roadmap

- Setup Spark
- Extract Questions
- Reproduce Bugs
- Discover Strategies & Collect Requirements
- Developer Interviews
## Setup Spark

![Spark](https://images-na.ssl-images-amazon.com/images/I/71uVcJ3ARI._SX300_.jpg)

<table>
<thead>
<tr>
<th>ID</th>
<th>User</th>
<th>Name</th>
<th>Application Type</th>
<th>Queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>application_1561329314867_0002</td>
<td>melike</td>
<td>MainClass</td>
<td>SPARK</td>
<td>default</td>
</tr>
</tbody>
</table>

Java HotSpot(TM) 64-Bit Server VM warning: ignoring option MaxPermSize=256m; support was removed in 8.0

SLF4J: Class path contains multiple SLF4J bindings.

SLF4J: Found binding in [jar:file:/opt/hadoop-melike/m-m-local-dir/userscache/melike/filecache/10/_spark_libs_756452103123317951.zip/slf4j-log4j12-1.7.16.jar!/org/slf4j/

SLF4J: Found binding in [jar:file:/opt/hadoop-melike/m-m-local-dir/userscache/melike/filecache/10/_spark_libs_756452103123317951.zip/slf4j-log4j12-1.7.16.jar!/org/slf4j/

SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.

SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]

2019-06-24 00:40:02,758 INFO yarn.ApplicationMaster: Changing view acls to: melike
2019-06-24 00:40:03,981 INFO spark.SecurityManager: Changing view acls with: to: melike
Investigation Roadmap

- Setup Spark
- Reproduce Bugs
- Discover Strategies & Collect Requirements
- Extract Questions
- Developer Interviews
Extract Questions

Extract question on Stack Overflow

Talk to an expert and ask a recommendation

Have a pilot interview

Review the questions
## Extract Questions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Reason</th>
<th>Solution</th>
</tr>
</thead>
</table>
| Task Not Serializable      | The class is not serializable                  | - Make the class Serializable  
- Change the method to a function 
- Copy the value to a local variable |
| Stack Overflow             | Nested structures with many fields             | - Remove the nesting  
- Decrease the number of fields |
| Number Format Exception    | A String value found where an integer expected in the production environment | - Change the data  
- Discard the data |
| Stack Overflow             | Making an action after 1000 transformations    | - Put checkpoints  
- Use caching  
- Decrease the number of transformations  
- Increase heap size of Java |
Investigation Roadmap

- Setup Spark
- Extract Questions
- Reproduce Bugs
- Discover Strategies & Collect Requirements
- Developer Interviews
Interview Setup

Semi-structured interview

An interviewer, an interviewee and an observer

With their consent, we record the screen and audio.
Professional Information

1. How long is your professional experience in Industry?
2. How long is your experience in Scala?
3. How long is your experience in Spark?
4. How many «different» projects that use Spark did you work on in your experience?
5. When you encounter a Spark job failure, what are your first steps to investigate?
6. What are your tools for debugging failed Spark jobs?
Interviewees

Experiences of interviewees

- Industry: More than 3
- Scala: Less than 1
- Spark: Between 1 and 3

Number of «different» projects

- number of projects
  - 1
  - 2
  - 3
  - More than 3
Investigation Roadmap

Setup Spark

Extract Questions

Reproduce Bugs

Discover Strategies & Collect Requirements

Developer Interviews
<table>
<thead>
<tr>
<th></th>
<th>Action</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Checks the logs on the console</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>Sees the exception message</td>
<td>Log Interactions</td>
</tr>
<tr>
<td>C</td>
<td>Checks the logs on the Resource Manager</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>Inspects the class that threw the exception</td>
<td>Code Inspection</td>
</tr>
<tr>
<td>N</td>
<td>Inspects the script</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>Suspects something</td>
<td>Come Up w/ Hypothesis</td>
</tr>
<tr>
<td>F</td>
<td>Suggests to do something</td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>Makes changes in the code</td>
<td>Making Changes</td>
</tr>
<tr>
<td>P</td>
<td>Writes tests</td>
<td></td>
</tr>
<tr>
<td>R</td>
<td>Changes the script</td>
<td></td>
</tr>
</tbody>
</table>
## Coding Interview - Patterns

<table>
<thead>
<tr>
<th>Q</th>
<th>Runs their own tests</th>
<th>Test Execution</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Runs the test with the new changes</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>Runs the test without any changes in the code</td>
<td></td>
</tr>
</tbody>
</table>

| K    | Searches the exception on the internet | Seek Help |

| J    | Runs the code with the new changes on the cluster | Run On The Cluster |

| L    | Checks the size of the file on HDFS | Checks The File's Size |

<table>
<thead>
<tr>
<th>M</th>
<th>Checks the file on cluster</th>
<th>Checks The File</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>Checks the file on local</td>
<td></td>
</tr>
</tbody>
</table>
Debugging Flow

- Learning the job has failed
- Checking the logs
- Solving the problem
Learning the job has failed

START

- Orchestration Frameworks (e.g. Apache Airflow)
- Alerting System (e.g. Grafana, Prometheus)
- Manual Inspection
- Scripts (e.g. batch scripts to check if a task is still running)

...
Checking the logs

Look for the exception message ...

... on the console

Find the problem

The log is too long

... on the Resource Manager
Checking the logs … on the console

```console
2019-12-08 22:19:00,714 INFO yarn.Client: Application report for application_1575839673320_0002 (state: FINISHED) 
2019-12-08 22:19:00,717 INFO yarn.Client: 
    client token: N/A 
    diagnostics: User class threw exception: org.apache.spark.SparkException: Task not serializable 
    ApplicationMaster host: 130.92.65.82 
    ApplicationMaster RPC port: 0 
    queue: default 
    start time: 1575839003236 
    final status: FAILED 
    tracking URL: http://leela.unibe.ch:8088/proxy/application_1575839673320_0002/ 
    user: melike 

Exception in thread "main" org.apache.spark.SparkException: Application application_1575839673320_0002 finished with failed status 
    at org.apache.spark.deploy.yarn.Client.run(Client.scala:1132) 
    at org.apache.spark.deploy.yarn.Client$.main(Client.scala:1178) 
    at org.apache.spark.deploy.yarn.Client.main(Client.scala) 
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method) 
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57) 
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43) 
    at java.lang.reflect.Method.invoke(Method.java:606) 
    at org.apache.spark.deploy.SparkSubmit$.org$apache$spark$deploy$SparkSubmit$$runMain(SparkSubmit.scala:736) 
    at org.apache.spark.deploy.SparkSubmit$.doRunMain(SparkSubmit.scala:185) 
    at org.apache.spark.deploy.SparkSubmit$.submit(SparkSubmit.scala:210) 
    at org.apache.spark.deploy.SparkSubmit$.main(SparkSubmit.scala:124) 
    at org.apache.spark.deploy.SparkSubmit.main(SparkSubmit.scala) 

2019-12-08 22:19:00,742 INFO util.ShutdownHookManager: Shutdown hook called 
2019-12-08 22:19:00,745 INFO util.ShutdownHookManager: Deleting directory /tmp/spark-ba87ea2d-1919-486f-bff4-161c21b1be12 
melike@leela:~
```
Checking the logs ... on the Resource Manager
The problem is ...

... related to something you know

... related to something you don’t know

Ask colleagues

Look up on the internet
Solving the problem

…

… related to something you know

Come up with a hypothesis

… related to the code  … related to the data  … related to the parameters

…  …  …
Solving the problem

…

… related to the parameters

Check the pipeline

Check the job submitting script

Change the parameter

Check Spark metrics
Solving the problem - Check the pipeline
Solving the problem - Check the job submitting script

```
melike@leela:~$ spark-submit --num-executors 3 --master yarn --deploy-mode cluster --class ch.unibe.scg.MainClass ExampleOne-assembly-0.1.jar hdfs://leela.unibe.ch:9000//sales.txt.
```
Solving the problem - Check Spark metrics

e.g. Grafana dashboard
Solving the problem

... related to the data

Check the data

Check the size of the file

Reduce the data

Ignore the data

Re-create the data

Run it

...

Redraw the data

Ignore the data

Run it
Solving the problem

Use local debugger

Write tests

Check an operation at a time using a notebook

Make changes in the code

Run it

...
Epilogue

START

…

↓

Run it

↓

… on the local
(if possible)

if fails

… on the cluster

if succeeds

if fails

if succeeds

END
Postmortem Questionnaire

1. Have you seen these bugs before?
2. What do you do, after you see the exception message?
3. How do you know if there is a job failure?
4. What is the nastiest type of bugs you encountered in Spark applications?
5. Would you describe a tool that you miss when debugging Spark applications? What does it do? Why do you think it’s not there?
Have you seen these bugs before?

<table>
<thead>
<tr>
<th>Bug #1</th>
<th>Task Not Serializable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bug #2</td>
<td>Stack Overflow</td>
</tr>
<tr>
<td>Bug #3</td>
<td>Number Format</td>
</tr>
<tr>
<td>Bug #4</td>
<td>Stack Overflow</td>
</tr>
</tbody>
</table>
What is the nastiest type of bugs you encountered in Spark applications?

Spark is difficult for new learners.

Assigning executors to partitions.

A job fails with Memory Overhead.

«Other components are nastier.»
What are the challenges that developers face when they debug such applications?

- Logs don’t give much insight about the exception
  «Logs have messages that come directly from JVM»

Look for the exception message …

- … on the console
- The log is too long
- … on the Resource Manager

Find the problem

...
What are the challenges that developers face when they debug such applications?

Data format is changed by the data provider

… related to the data

Check the data
Check the size of the file

Reduce the data
Ignore the data
Re-create the data
Run it

… related to the data
What are the challenges that developers face when they debug such applications?

Debugging on local doesn’t always help. Some problems occur only on the cluster.

Use local debugger

... related to the code

Write tests

Check an operation at a time using a notebook

Make changes in the code

Run it

...
What are the challenges that developers face when they debug such applications?

Not being able to access the production environment

START

if fails

… on the local (if possible)

if succeeds

… on the cluster

if succeeds

if fails

Run it

END
How can we overcome these challenges?

3 categories of enhancements were proposed:

- Code
- Log
- Dashboard
Tool Recommendations to Enhance Coding

• Monitor data after each transformation
• Find the line in a data file, which causes a problem
• A distributed debugger
  • A unified debugger interface (Leske, Chiş and Nierstrasz, 2016)
  • A remote debugging model based on reflection (Papoulias et al., 2015)
Tool Recommendations to Enhance Logs

• A Log parser
  • A tool which separates important and useless parts of logs
  • Better exception messages, instead of taking the message from JVM
  • Using ML techniques to learn and advice solutions to certain problems

• See in which line the application dies like in Java
• See logs per stage on Spark UI
Tool Recommendation: Dashboard

• A tool which displays configuration of the cluster
  • Memory, GC, Network Bandwidth, and other parameters
• A tool which can compare different environments
  • when there's no access to the other environment
Investigation Roadmap

Setup Spark
Extract Questions
Reproduce Bugs
Developer Interviews
Build Tools
Requirement Collection & Discover Strategies
Duration vs. Number of Steps

B#1

B#2

B#3

B#4