Ludwig-Maximilians-Universität München
Institut für Informatik
Prof. Dr. Matthias Schubert
Evgeniy Faerman
Daniyal Kazempour

Munich, 02.11.2018

Big Data Management and Analytics

WS 2018/19

Tutorial 3: Hadoop, MapReduce and Scheduling

Assignment 3-1

Given a Hadoop setup with several DataNodes.

(a) One DataNode d is broken. How is this node failure being detected by Hadoop? How does the NameNode react to d with regards to IO requests? What can be stated regarding the availability of the data which was stored on the DataNode d? What can be the effects regarding the replication factor and what action is taken by the NameNode?

(b) The free space on one DataNode d falls below a certain threshold. What could be a potential scheme in context of rebalancing available space among DataNodes?

(c) Given the case of a highly frequent access on a specific file. What could be a scheme in order to rebalance the frequent file access?

(d) From a DataNode a data block is fetched. How is checked if the arrived data block is corrupted?

(e) By some failure the NameNode is no longer available. What are the implications regarding the functionality of the Hadoop setup? Is any manual intervention necessary? If yes, explain why.

(f) What are FsImage and EditLog? What would be the implications if any of these files are corrupted? What is done in order to be resilient against the case of corrupted FsImage and EditLog files?

(g) In a data center we have two racks r_1 and r_2. Each of the racks contains several DataNodes. The client is located on a DataNode d_1 on r_1. The DataNode with the client fails. Which of the replicas according to Hadoops replica placement strategy are available?

Assignment 3-2 *Matrix multiplication*

Given two matrices $A \in \mathbb{R}^{i \times j}$ and $B \in \mathbb{R}^{j \times k}$

(a) Describe the steps which are required to perform a matrix multiplication using MapReduce.

(b) Let now $A := \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$ and $B := \begin{pmatrix} 7 & 8 \\ 9 & 10 \\ 11 & 12 \end{pmatrix}$

Multiply A and B using MapReduce.

Assignment 3-3 *Resource distribution*

(a) Given $k \in \mathbb{N}$ number of tasks which require at least $b \in \mathbb{N}$ GB of RAM, and $n \in \mathbb{N}$ nodes which provide a_1,a_2,...,a_n GB of memory. How can the memory of n nodes be distributed among k tasks such that no task has less than ϵ_t GB of RAM? If a task does not get the required amount of memory, the task waits until sufficient memory is available. What is the runtime complexity?

(b) In which field(s) of computer science does the resource scheduling problem occur?

(c) What is the purpose of YARN and in how far does it relate to the mentioned resource distribution problem?