Big Data

Tutorial #1

Isabelle Kuhlmann
2020-04-30

WeST
People and Knowledge Networks

Outline

1) Organization
2) Lecture Recap
3) Introduction to Apache Spark
4) First Assignment

Tutorials & Assignments/Exercises

Organization

Tutorials and Assignments/Exercises

- Tutorials will be uploaded by the end of the week
- Slides will be uploaded in OLAT as well
- If you have questions, please ask them!
  - Use the OLAT forum for this
- Exercises will be given at the end of each tutorial
  - No official submission/grading
  - You should do them anyway
- Exam Admission: group project
  - Further information within the next few weeks

What previously happened...

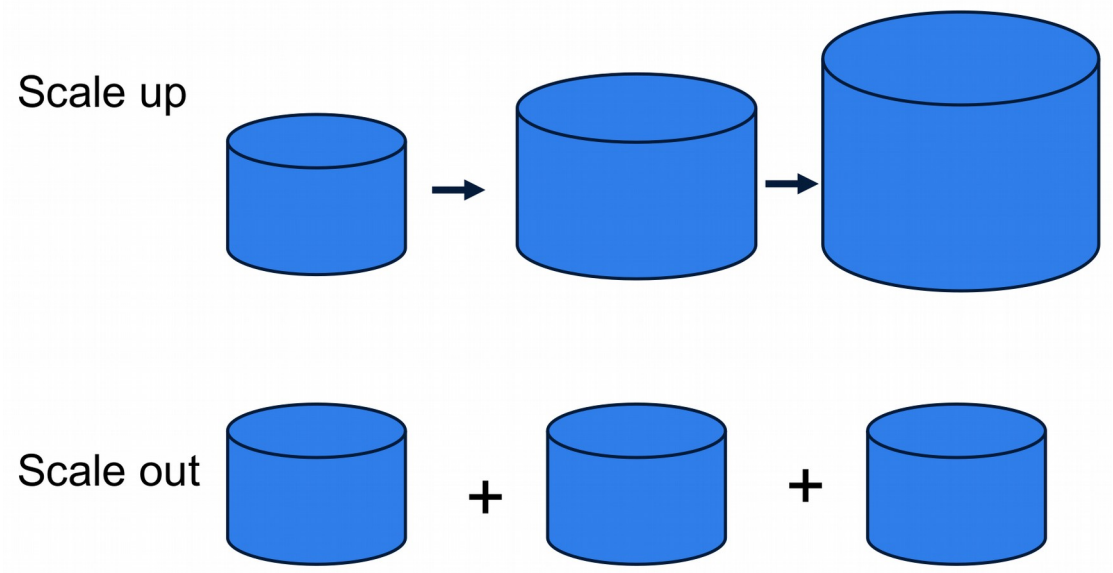
Lecture Recap

What is “Big Data”?

- *“Big data refers to data sets with sizes beyond the ability of commonly used software tools to capture, curate, manage, and process data within a tolerable elapsed time.”*
- Characteristics:
  - Volume
  - Variety
  - Velocity
  - Veracity
- “Big” is a relative term

Scale-out Instead of *Scale-up*

- Cloud computing
- Distributed computing
  - Split the work and perform it on several machines simultaneously



Tools

- Apache Hadoop
  - Hadoop Distributed File System (HDFS)
  - MapReduce
- Apache Hive
- **Apache Spark**
- Apache Storm
- Apache Mahout

Cloud Computing: 5 Characteristics

1. On-demand self-service
2. Broad network access
3. Resource pooling
4. Rapid elasticity
5. Measured service

Cloud Computing: 3 Service Models

1. Software as a Service (SaaS)
2. Platform as a Service (PaaS)
3. Infrastructure as a Service (IaaS)

Cloud Computing: 4 Deployment Models

1. Private cloud
2. Community cloud
3. Public cloud
4. Hybrid cloud

Cloud Computing – Concepts

- Virtualization
  - *Virtual* system on top of physical one
  - Multiple virtual systems share one physical system
    - When one virtual system is idle, others can use the physical resources
    - Hypervisor architecture
- Optimized for horizontal scaling
  - Parallel processing
- Pay-as-you-go
  - You only pay for the resources you actually use

Installation, overview, first applications

Introduction to Apache Spark

Installation

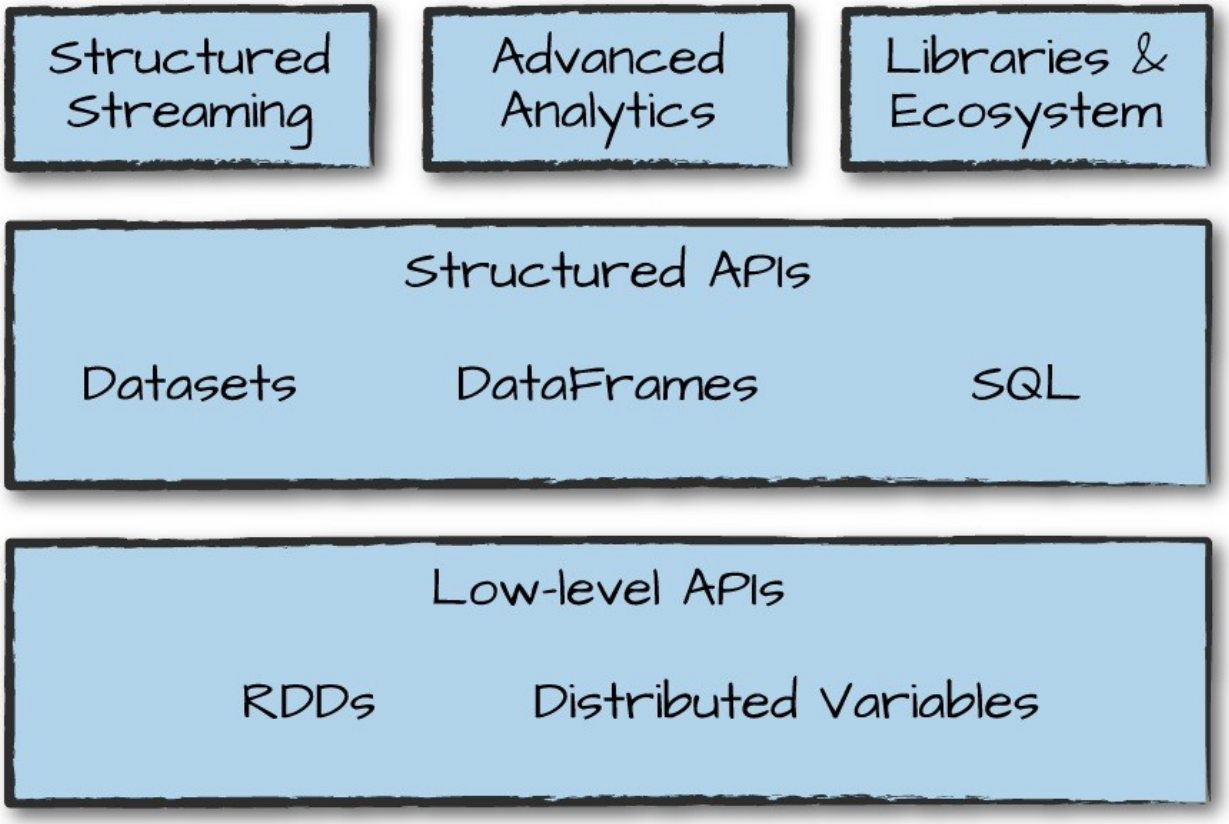
- Install Java (openjdk version 8)
  - Via `java -version` you can check whether you already have Java installed and which version it is
- Make sure you have Python 3.6 installed
- Download the current Apache Spark version (2.4.5 as of today)
- Unpack the download
- Now you can start a python shell using the `pyspark` command

How to Execute Python Code

- There are two different ways:
  - Launch the python console via the `pyspark` command
  - Use `spark-submit path/to/mySparkApp.py`
    - Here you have to set up a SparkContext manually
- Use the editor/IDE of your choice (e.g., PyCharm, VS Code, Atom)
- Demo – Example #1 (Hello World)

Overview

"Apache Spark is a unified computing engine and a set of libraries for parallel data processing on computer clusters."



Source: *Spark: The Definitive Guide – Data Processing Made Simple*, Bill Chambers and Matei Zaharia, O'REILLY, 2018

DataFrames – Concepts

- DataFrames are built on top of RDDs
  - Both RDDs and DataFrames are *immutable*
- Data is *partitioned*
  - Example:

Col 1	Col 2	Col 3	Col 4
Val 1.1	Val 1.2	Val 1.3	Val 1.4
Val 2.1	Val 2.2	Val 2.3	Val 2.4

Col 1	Col 2	Col 3	Col 4
Val 3.1	Val 3.2	Val 3.3	Val 3.4

Col 1	Col 2	Col 3	Col 4
Val 4.1	Val 4.2	Val 4.3	Val 4.4
Val 5.1	Val 5.2	Val 5.3	Val 5.4

DataFrames – Concepts

- Transformations
  - Narrow transformations (1 to 1) vs. wide transformations/*shuffles* (1 to N)
  - Transform a DataFrame (i.e., return a new DataFrame)
  - Lazy execution
    - A logical transformation plan is set up
    - Spark optimizes the order of transformations
  - Examples: `filter()`, `sort()`, etc.
- Actions
  - Trigger the execution
  - Eager execution
  - Examples: `show()`, `count()`, etc.
- Demo – Example #2

Example #3 (Using DataFrames)

- Check out this GitHub repository
  - Here you can find several datasets
- We will take a look at the **Flight Data** dataset
  - Can be found at `data/flight-data`
- We will complete the following tasks:
  1. Import data from a csv file
  2. Learn about the structure of our data
  3. Sort the data
  4. Filter the data (in Spark and in SQL)
- Documentation

Resilient Distributed Datasets (RDDs)

- Also incorporate the concept of transformations and actions
- Partitioned as well
- Collections of records, which can be Java/Scala/ Python objects
  - No known schema
- There are *generic RDDs* and *key-value RDDs*

Example #4 (Using RDDs)

- Let's take a look at `alice.txt`, which contains the text of Alice in Wonderland.
  - The file can be found under Tutorial Material/Tutorial01 in OLAT
- We will complete the following tasks:
  - Read the .txt file to create an RDD
  - Count the occurences of each word
  - Save the result
- Documentation

Now it’s your turn!

Exercise

Exercise

- Calculate the frequency of each word in *Alice in Wonderland* again, but this time, consider the following aspects:
  - Handle capitalized words.
  - Words cannot be empty.
  - Remove punctuation.
  - How do you handle terms like “don’t”, “she’s”, etc.? (Those are actually comprised of two words…)

Thank you for your Attention!

WeST
People and Knowledge Networks