Course code: **HADOOP/P**

Course title: **Big Data for developers**

Days: 3

Description:

Course intended for:

Training course focuses on developers who want to develop systems for storing and/or analysing big sets of data with the use of Apache Hadoop platform. Course is dedicated to both beginners and programmers who already have preliminary experience with the platform and want to expand or consolidate their knowledge.

Course objective

Participants will gain knowledge needed to work with Apache Hadoop system, including the implementation of effective algorithms on the basis of MapReduce as well as data storage and import into the system. Design patterns and best coding practices will be presented. In the course emphasis is put not only on theoretical aspects but mostly on the practical skills.

Course strengths

Course curriculum includes general introduction to the subject of Big Data along with a detailed presentation of Apache Hadoop tools on the level which enables the participants to start working in this environment. The training is unique since the issues presented during it are not sufficiently covered in the available literature. The curriculum is constantly updated due to the rapid development of these technologies. Presented knowledge is the result of several years of practice of trainers in building systems based on Apache Hadoop platform.

Requirements

Participants are expected to have basic programming knowledge in Java as well as knowledge of the basics of databases and SQL.

Course parameters

3*8 hours (3*7 net hours) of lectures and workshops (with more workshops than lectures). During the workshops, participants will solve data processing problems by implementing their own algorithms with the use of MapReduce paradigm.

Course curriculum:

sages

1. Introduction to Big Data
2. Hadoop
    I. Introduction and history
    II. Architecture and components
    III. Running mode
    IV. Introduction to the ecosystem
    V. Users and applications
3. HDFS
    I. Introduction to Distributed Files System
    II. Management with command line
    III. Access through www
    IV. API usage
    V. Import and export of data
4. Introduction to MapReduce
    I. Introduction to MapReduce paradigm
    II. Comparison of subsequent versions of MapReduce
5. Use ofJava API MapReduce
    I. Input and output formats, creating own formats
    II. Embedded and own types of data
    III. Partitioner and Combiner, when and how to use it
    IV. Data counters
    V. Data sorting
    VI. Configuration of tasks with the use of paradigms
    VII. Creation of own data comparators
    VIII. Realisation of data connections in w MapReduce
    IX. Tasks chains in MapReduce
    X. Use of compression to decrease the amount of data
    XI. Optimization of tasks in MapReduce
    XII. Use of DistributedCache
6. Examples of implementation of common algorithms in MapReduce paradigm
7. Other programming approaches
    I. Streaming – use of programs written in other programming languages
    II. Developing MapReduce algorithms with the Cascading library
8. Good programming practices in MapReduce paradigm
    I. Design patterns in MapReduce
    II. Unit tests in Testy Apache Hadoop environment
9. Starting and monitoring of tasks in a cluster
10. Creating task flow MapReduce
    I. Use of JobControl class
    II. Apache Oozie
11. HBase
    I. Introduction to HBase
    II. Use of HBase with API
    III. MapReduce in HBase
    IV. Unit tests in HBase
12. Use of Spring Framework library

Sages sp. z o.o., ul. Nowogrodzka 62c, 02-002 Warszawa
tel. +48 22 203 56 00 fax: +48 22 203 56 01
e-mail: office@sages.io web: www.sages.io

I. Project setup (Java + Maven)
II. Hadoop configuration in Spring
III. Handling the ecosystem
IV. Testing
V. Dependency Injection in MapReduce environment

13. Hive
    I. Introduction
    II. Creating and running queries
    III. Use of User-Defined Function
14. Pig
    I. Introduction
    II. Creating and using scripts
    III. Use of User-Defined Function
15. Overview of selected ecosystem elements
    I. YARN
    II. Flume
    III. Zookeeper