COURSE DESCRIPTION CARD - SYLLABUS

Course name

Data Warehouses and Analytical Processing [S2Inf1-TPD>HURT]

Course

Field of study Computing	Year/Semester 1/1
Area of study (specialization) Data Processing Technologies	Profile of study general academic
Level of study second-cycle	Course offered in Polish
Form of study full-time	Requirements compulsory

Number of hours

Lecture 30	Laboratory classes 20	Other 0
Tutorials 0	Projects/seminars 45	

Number of credit points

6,00

Coordinators

dr hab. inż. Robert Wrembel prof. PP
robert.wrembel@put.poznan.pl

Lecturers

Prerequisites

A student should have a basic command in: programming languages, operating systems, database systems (relational data model, SQL language, tree index, conceptual and logical database schemas, transaction management and concurrency control, query optimization).

Course objective

1. Pointing out practical problems related to designing, implementing, deploying, and maintaining data warehouse (DW) systems, for standard and big data. 2. Conveying knowledge on designing a DW w.r.t.: technical architectures, data modeling, desigining a data integration layer - ETL, physical data structures, optimizing analytical queries, and techniques for processing big data. 3. Presenting problems related to designing and implementing a DW and analytical applications, in particular: analytical extensions in SQL, using physical data structures (indexes, partitions, materialized views) in the process of query optimization. 4. Developing skills in solving practical problems related to: designing and implementing DW systems, assessing the applicability of the DW technology and data analytics to a given problem at hand, testing a designed solution w.r.t. efficiency and functionality. 5. Developing team work while doing projects. Developing skills on doing practical projects and solving practical problems in the area of DW and data analytics.

Course-related learning outcomes

Knowledge:
1. advanced knowledge on: (1) architectures of a dw system (standard and big data), (2) data modeling for analytcs, (3) data structures for a dw, (4) optimization techniques for analytical queries, (5) tools and ecosystems for building dws.
2. detailed knowledge on dw systems (architectures, techniques and tools for data integration, logical and physical data models, data structures, star query optimization, dw tuning, main memory dw appliances).
3. knowledge on current trends in dw architectures and technologies. awareness of the existing problems and limitations of current dw systems.
4. advanced knowledge on a dw system design cycle.
5. advanced knowledge on applying dw architectures, methods and analytical techniques, as well as phyiscal structures in solving real business problems (cf. project classes).

Skills:
1. a student is able to extract useful information from various technical and research sources (both in english and polish) on topics covered in this course. he/she is able to integrate and confront this information, interpret and assess it; is able to justify a proposed solution to a given problem (cf. project classes).
2. uses information-communication techniques to do projects.
3. is able to: (1) design and conduct experiments on software and architectures, (2) interpret obtained results and draw conclusions from them, (3) build and verify hypotheses in the scope of a development of dw systems, (4) contuct short research projects based on dw technologies (cf. project classes).
4. is able to apply analytical, simulation, and experimental methods to build and solve technical problems and simple research problems in the area of dw systems.
5. while solving technical and research problems, a student integrates knowledge from multiple areas of computer systems, among others: databases, data warehosues, operating systems, distributed systems, programing languages, the theory of computational complexity, internet technologies.
6. assesses the applicability of novel concepts, techniques, and software for dws (e.g., nosql systems, stream processing techniques, hadoop, spark).
7. is able to assess the applicability of methods and tools to solve a given technical/engineering problem, e.g., designing, implementing, or assessing components of a dw system.
8. by means of applying novel techniques, technologies, and software, solves complex tasks of designing, implementing, and deploying a solution.
9. based on requirement analysis a student is able to design a dw system or its fragment. to this end, he/she uses or adapts methods, techniques, and software to a given problem, or designs his/her own unique solution (por. project classes).
10. is able to work in a team, taking different roles (por. project classes). is able to conduct the requirement elicitation process and interact with a future user.

Social competences:
1. understands that in the area of data warehousing (as in other computer science domains) previously acquired knowledge and skills quickly become obsolete, which require life-long following a state-of-the-art solutions and learning.
2. understands the necessity of using an up-to-date knowledge and solutions in the area of dw (e.g., nosql systems, parallel processing architectures like hadoop and spark, stream processing architectures) in a process of solving technical and research problems.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes presented above are verified as follows:

Lectures: knowledge, skills, and competences assessed by means of a written test (a student is allowed to use any educational meterials). The test is composed of: 5-6 problem questions and 6-8 test questions (single or multiple choice). Maximum number of points is 40, out of this 6-8 for test questions. Points are granted only as integer values. Lectures are considered as passed if a student is graded at least 21 points. The following grading scale is used:
0-20: fail (ndst; 2.0), 21-24: C (dst; 3.0), 25-28: C+ (dst+; 3.5), 29-32: B (db; 4.0), 33-36: B+ (db+; 4.5), 37-

40: A (bdb; 5.0), where A is the highest grade.

Projet classes: knowledge, skills, and competences is assessed by means of: (1) evaluating project progress during weekly meetings, also in a form of video conferences, (2) evaluating project progress based on periodical student presentations, also in a form of video conferences, (3) periodical evaluation of a project technical documentation, (4) final studen presentation "defending" the project, (5) checking a final project product (whether the product conforms to the requirements), and (6) assessing the content of a project technical documentation.
Projects realized for companies are evaluated also by mentors from these companies; a project "defense" is attended by a company representatives.
Maximum number of points is 100 divided as follows: 50 pts - project product, 40 pts - project technical documentation, 10 pts - final presentation. Points are granted only as integer values. Project classes are considered as passed if a student is graded at least 51 points. The following grading scale is used:
0-50: fail (ndst; 2.0), 51-60: C (dst; 3.0), 61-70: C+ (dst+; 3.5), 71-80: B (db; 4.0), 81-90: B+ (db+; 4.5), 91-100: A (bdb; 5.0), where A is the highest grade.

In terms of laboratories, verification of the established learning outcomes is realized by:
- evaluation of the implementation of the tasks assigned in each class,
- evaluation of knowledge and skills related to the implementation of laboratory tasks by solving a test (with possible open questions) at the end of the semester.
- obtaining additional points for activity during classes, especially for:
- discussion of additional aspects of the issue,
- remarks related to the improvement of teaching materials.
In order to receive a passing grade, student must upload her/his project to ekursy website.
In terms of laboratory, the following grading scale is adopted depending on the number of points obtained:
<0;50%>: 2.0, (50%;60%>: 3.0, (60%;70%>: 3.5, (70%;80%>: 4.0, (80%;90%>: 4.5, (90%;100%>:5.0.

Programme content

The lecture program covers the following topics:
Common problems of data integration
Data integration architectures
Data warehouse architectures for a standard application
Data warehouse refreshing - ETL/ELT
Data warehouse modeling
Physical data structures for a data warehouse
Star query optimization techniques
Main memory data warehouse appliances
Architectures for big data processing

The laboratory program is divided into the following parts:
1. introduction to the exercise environment
- case study,
- data sources, data warehouse schema,
- basics of Agile BI methodology.
2. introduction to the operation of the Pentaho Data Integration tool
- basic concepts,
- repository,
- single data source based transformation,
- subtransformation.
3. support of multiple data sources
- expansion of existing transformations and subtransformation with an additional data source,
- data flow path control,
- methods of combining data.
4. additional transformations
- methods for eliminating duplicates,
- automatic generation of data for dimensions,
- loading fact table.
- fundamentals of Agile BI methodology.
5. advanced transformations

- data sources based on CSV files, detection of changes in data sources,
- operational data store, data warehouse refresh.
6. modern data sources
- XML documents, web services.
7. data profiling and cleaning, historical data
- incorrect data detection (reference data, data patterns),
- automatic error correction, fixing errors in data sources,
- storing historical data for changing dimensions.
8. improving the efficiency of the ETL process, thematic data warehouses
- bulk loading of data (Oracle, PostgreSQL, MySQL)
- calculating aggregates from data, example of thematic data warehouse.
9. data processing in data warehouses using SQL language and its extensions.

Classes are conducted in the form of exercise classes using computers, with each student working independently. Each task is preceded by a short presentation and then the discussed issues are practiced.

Course topics

none

Teaching methods

Lectures: standard lecturing (on-line lectures are avialable) with the support of slides. If lectures are physically delivered (with students present in a lecture hall), additionally a discussion on solving specific issues is initiated.

Project classes: take place either in a laboratory at the university or at the infrastructure provided by a company for which a project is developed. Studens solve practical problems given by a company. Each year, project topics differ - as they depend on current interests of companies. Generally, topics revolve around: designing a data integration architecture (e.g., ETL), assessing performance of data structures in DW systems (including Oracle, DB2, SQL Server), advanced applications for data analysis, designing and implementing analytical software, storing and analyzing data in NoSQL systems, optimization techniques for ETL processes, data lake architectures. Projects are conducted in groups comprised of 2-4 students (depending on a project scope). So far, project have been done for the following companies: Roche, IBM, Pearson/IOKI, Capgemini, Kogeneracja Zachód, PKO BP, Santander.

Laboratories: multimedia presentation, the presentation is supplemented with short examples presented in a traditional manner with the use of the blackboard, performing exercises in the data warehouse, discussing more difficult exercises at the blackboard, answering questions as they arise, solving problems as they arise.

Bibliography

Basic
Vaisman A., Zimanyi E.: Data Warehouse Systems - Design and Implementation. Springer Verlag, 2014
Jarke M., Lenzerini M., Vassiliou Y., Vassiliadis P.: Fundamentals of Data Warehouses. Springer, 2010, ISBN-13: 978-3642075643
Golfarelli M., Rizzi S.: Data Warehouse Design: Modern Principles and Methodologies. McGraw-Hill Osborne, 2009, ISBN-13: 978-0071610391
Additional
Jiang B.: Constructing Data Warehouses with Metadata-driven Generic Operators, and more: Architecture, Methodology, and Paradigm; Concepts, Algorithms, and Operators; Principles, Recommendations, and Exercises. DBJ Publishing, 2011, ISBN-13: 978-3033029200
Pentaho Data Integration documentation https://pentaho-public.atlassian.net/wiki/spaces/EAI/overview
Matt Casters, Roland Bouman, Jos Van Dongen: Pentaho Kettle Solutions, John Wiley & Sons 2010

Breakdown of average student's workload

	Hours	ECTS
Total workload	150	6,00
Classes requiring direct contact with the teacher	95	4,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	55	2,00