



Modulbeschreibung

M.Sc. Computer Engineering PO19 Interactive Systems and Visualization

Stand: November 2022

Modul- und Veranstaltungsverzeichnis

Kursname laut Prüfungsordnung

Advanced Image Synthesis

Course title English

Advanced Image Synthesis

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		

Prüfungsleistung

mündliche Prüfung, 45 Minuten

Beschreibung / Inhalt Deutsch

Die Vorlesung erläutert systematisch die grundlegenden Konzepte aktueller 3D-Umgebungen.

Inhalte im Einzelnen:

- Architektur von Graphikprozessoren
- Low level und high level Shadersprachen
- Graphik und Medienbibliotheken OpenGL und DirectX
- interaktive Reflektions- und Refraktionsberechnung
- Schattenberechnung
- Einführung in radiometrische Größen
- Radiosity
- Irradiance Volumes
- Precomputed Radiance Transfer
- Ambient Occlusion
- Terrain Rendering & Synthese
- Fur
- High Dynamic Range Imaging

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden lernen in dieser Veranstaltung fortgeschrittene Algorithmen moderner Grafiksysteme kennen. Aufbauend auf den Grundkenntnissen der 3D Computergraphik werden Kenntnisse über Algorithmen und Konzepte zur Generierung und Visualisierung von 3D-Welten erworben. Die Studierenden verstehen die Funktionsweise aktueller Grafikengines wie sie z.B. in der Filmindustrie, aktuellen Spielen und virtuellen- bzw. erweiterten- Realitätssystemen zum Einsatz kommen.

Description / Content English

In this lecture a number of state of the art concepts of 3D environments are explained in a systematic way.

In particular the lecture contains topics such as:

- the architecture of graphics hardware,
- low- and high-level APIs,
- OpenGL and DirectX,
- interactive reflection and refraction,
- methods to simulate shadow,
- an introduction to radiometric quantities,

radiosity,
irradiance volumes,
precomputed radiance transfer,
ambient occlusion,
terrain rendering and terrain synthesis,
hair and fur rendering, and
high dynamic range imaging.

Learning objectives / skills English

After attending this lecture, students will be able to understand and implement state of the art graphics algorithms. Students with a basic understand of 3D computer graphics will learn advanced concepts that build on that foundation, including algorithms for the generation and visualization of interactive 3D worlds. Students who take this course will understand the workings of current graphics engines used in both the movie and game industry for virtual and augmented reality scenarios.

Literatur

- Aktuelle Internetliteratur
- Eberly: 3D Game Engine Design, Morgan Kaufmann
- Fernando: GPU Gems Series, Addison-Wesley
- DeLoura et. al: Game Programming Gems Series, Charles River Media

Kursname laut Prüfungsordnung**Advanced Web Technologies****Course title English**

Advanced Web Technologies

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		

Prüfungsleistung

Die Prüfung besteht aus drei Teilleistungen: (1) Die abschließende mündliche Prüfung, in der die Kenntnisse der theoretischen Konzepte und aktuellen Webtechnologie, die in der Vorlesung, den studentischen Referaten sowie Projektpräsentationen vorgestellt werden, nachweist; (2) die Bewertung eines Referats zu einer aktuellen selbstausgewählten Webtechnologie sowie (3) die Bewertung der Projektergebnisse.

Die Gesamtnote ergibt sich entsprechend zu 33% aus der mündlichen Prüfung der theoretischen Konzepte, zu 17% aus der Bewertung der Ausarbeitung und Präsentation einer Webtechnologie, und zu 50% aus der Bewertung der praktischen Projektresultate und deren Präsentation.

Beschreibung / Inhalt Deutsch

Das World Wide Web und die ihm zugrundeliegenden Technologien gewinnen zunehmend an Bedeutung für die Entwicklung interaktiver Systeme. Diese Lehrveranstaltung greift im Kern eine Menge verschiedener Konzepte, Prinzipien und Methoden aktueller client- und serverseitigen Webtechnologien auf. Die grundlegenden Webtechnologien (u.a. HTML, HTTP, CSS, XML, JavaScript) werden in diesem Modul vorausgesetzt. Der Schwerpunkt in diesem Modul liegt auf weiterführenden Entwicklungen und aktuellen Trends (z.B. Erweiterungen und Weiterentwicklungen der Web-Standards, Bibliotheken, Werkzeuge und Entwicklungsumgebungen, Web-Entwicklungs-Frameworks) die durch Studierenden in Gruppen aufbereitet und im Plenum präsentiert und diskutiert werden. Darüber hinaus werden die vorgestellten Webtechnologien im Kontext von kursbegleitenden, von den Studierenden selbstgewählten Webprojekten erarbeitet.

Lernergebnisse / Kompetenzen Deutsch

Nach erfolgreichem Absolvieren des Moduls können Studierende Kenntnisse:

- Konzepte aktueller Webtechnologien erläutern und miteinander vergleichen
- aktuelle Webtechnologien und deren Kombination in Webanwendungen erläutern
- Probleme und mögliche Lösungsansätze mittels aktueller Webtechnologien mit selbstgewählten Beispielen aus Webprojekten beschreiben

Fertigkeiten:

- Anforderungen in Webprojekten analysieren und adäquat anzuwendende Webtechnologien begründet auswählen
- verschiedene aktuelle Webtechnologien für innovative Webanwendungen kombinieren
- Projekterfahrung einsetzen, um sich in neue Webtechnologien einzuarbeiten

Kompetenzen:

Basierend auf den im Modul erworbenen Kenntnissen und Fertigkeiten können Absolventen

- aktuelle Webtechnologien wissenschaftlich präsentieren und diskutieren
- aktuelle Webtechnologien effizient und effektiv in Webprojekten einsetzen

- kreative Lösungen in Webprojekten entwickeln
- verantwortlich und verlässlich in Entwicklerteams agieren

Description / Content English

The World Wide Web and its underlying technologies are increasingly gaining importance for the development of interactive Web applications. This course introduces concepts, principles and methods in current client and server-side Web technologies. Basic Web technologies such as HTML, HTTP, CSS, XML, JavaScript etc. are a prerequisite to take this course. The focus of this course is rather on advanced topics in emerging Web technologies. These include extensions of Web standards, combination of different Web technologies, Web toolkits and development environments, as well as Web libraries and frameworks. This course follows a student-centered, project-based learning approach. Web technologies will be proposed and presented in the lecture by student groups. The presented technologies are further investigated and applied in hands-on sessions as well as small student projects carried out throughout the course.

Learning objectives / skills English

After successfully completing the module, students can:

Knowledge:

- Explain and compare concepts of current web technologies
- Explain current web technologies and their combination in web applications
- Describe problems and possible solutions using current web technologies with self-selected examples from web projects

Skills:

- Analyze requirements in web projects and reasonably choose the appropriate web technologies to use
- Combine different current web technologies to create innovative web applications
- Harness existing knowledge and experience to learn new web technologies

Competencies:

Based on the knowledge and skills acquired in the module, students can:

- Present and discuss current web technologies scientifically
- Use current web technologies efficiently and effectively in web projects
- Develop creative solutions in web projects
- Act responsibly and reliably in project development teams

Literatur

Web

Kursname laut Prüfungsordnung**Cloud, Web & Mobile****Course title English**

Cloud, Web & Mobile

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

In dem Modul werden theoretische und praktische Aspekte des Cloud-Computing betrachtet. Die Vorlesung besteht aus zwei Themenblöcken. Im ersten Block des Moduls werden Algorithmen, Architekturen und Programmiermodelle für Cloud Systeme und Cloud-basierte Anwendungen besprochen. Hierbei werden zwei Perspektiven betrachtet: die des Cloud-Providers und die des Cloud-Anwendungsentwicklers. Im zweiten Block werden Front-End Technologien und deren Verzahnung mit Cloud-Anwendungen vorgestellt. Unter anderem werden hier Web Technologien und mobile Betriebssysteme (darunter Android und Windows Phone 7) vorgestellt. Die Vorlesung hat hierbei einen großen praktischen Bezug. In den übungen werden die vorgestellten Konzepte auch an kleinen Beispielen ausprobiert.

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden verstehen Architekturen und Algorithmen, die es einem Rechenzentrums-Betreiber erlauben hoch-skalierbare und verlässliche Anwendungen auf Rechner Clustern auszuführen. Sie können Anwendungen entwickeln, welche auf solchen Plattformen ausgeführt werden können. Die Studierenden wissen, wie Abrechnungsmodelle/Kostenmodelle für Cloud-Computing aussehen und welche Arten von Anwendung sich hierfür eignen. Sie besitzen Kenntnisse über Front-End Technologien, welche die Cloud-Anwendungen Endnutzern zugänglich machen, z.B. Web Technologien oder mobile Anwendungen.

Description / Content English

This lecture presents theoretical and practical aspects of cloud computing. The lecture is divided in two parts. The first part of the lecture introduces algorithms, architectures and programming models for cloud computing and cloud-based applications. Here both, the point of view of the cloud provider and the point of view of the cloud application developer, are considered.

In the second part of the lecture front-end technologies and their usage in combination with cloud applications are presented. This includes web technologies and mobile operating systems (including Android and Windows Phone 7). The concepts presented in the lecture will be used in the exercises.

Learning objectives / skills English

The students know and understand architectures and algorithms that enable datacenter operators to run highly scalable and reliable applications on computer clusters. They are able to develop applications that can be run on corresponding platforms. The students know common payment- and pricing-models for cloud computing as well as which model suits which application. They gained knowledge about front-end technologies that enable the usage of cloud applications for end-users, e.g. web technologies and mobile applications.

Literatur

- L. Lamport: Paxos made simple. <http://research.microsoft.com/en-us/um/people/lamport/pubs/paxos-simple.pdf>

- Google: Paxos made live – an engineering perspective. http://labs.google.com/papers/paxos_made_live.html
- Google: Bigtable: A Distributed Storage System for Structured Data. <http://labs.google.com/papers/bigtable.html>
- Google: The Google File System. <http://labs.google.com/papers/gfs.html>
- S. Gilbert, N. Lynch: Brewer's Conjecture and the Feasibility of Consistent, Available, Partition-Tolerant Web Services
- C. Petzold: Programming Windows Phone 7. http://download.microsoft.com/download/5/0/A/50A39509-D015-410F-A8F2-A5511E5A988D/Microsoft_Press_ebook_Programming_Windows_Phone_7_PDF.pdf

Kursname laut Prüfungsordnung**Cognitive Robot Systems****Course title English**

Cognitive Robot Systems

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3		1	
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Ein kognitives Robotersystem nimmt mit Sensoren die Umgebung und die eigene Körperlichkeit wahr, sammelt, strukturiert und verwendet selbstständig Wissen, trifft darauf basierend sinnvolle Verhaltensentscheidungen, und reagiert/agiert mit Aktuatoren flexibel in Echtzeit. In der Vorlesung werden moderne Architekturkonzepte, Verfahren der Raumrepräsentation und zur Selbstlokalisierung, Systeme für visuell basiertes Greifen von Objekten, einfache Regelungsverfahren, Wegplanung zur Roboter-Navigation, Online-Roboterlernen sowie Robotik-Simulation behandelt. Im Rahmen der Übung werden ausgewählte Themen anwendungsbezogen vertieft. Inhalte im Einzelnen:

- Anwendungen von kognitiven Robotersystemen
- Kognitive Wahrnehmungs-Handlungs-Systeme
- Bestandteile von Robotersystemen
- Sensorsysteme als Grundlage für die Autonomie
- Koordinatensysteme und Transformationen
- Visuell-basierte Regelung eines Roboterarms
- Arten der Umweltbeschreibung
- Wegplanung zur Roboter-Navigation
- Probabilistische Ansätze zur Roboterlokalisierung
- Online lernende Verfahren zur Roboter-Navigation
- Robotik Simulation
- Programmierung kognitiver Robotersysteme
- Robot Operating System

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden sollen Architekturen von kognitiven Robotersystemen kennen lernen. Sie sollen Verfahren zur Roboterregelung, zur Wegplanung und Roboternavigation, zur Eigenlokalisierung, sowie zum Roboter-Lernen verstehen und realisieren können, inklusive den zugrundeliegenden mathematischen und probabilistischen Methoden. Für bestimmte Problemstellungen sollen sie in der Lage sein, potentielle Konfigurationen vorzuschlagen und zu bewerten.

Description / Content English

Cognitive robot systems use sensors and cameras to perceive their environment, in order to acquire and process knowledge for goal directed behavioral decisions. Such systems can be robot vehicles (e.g. for map building), robot arms (e.g. for object grasping), or robot heads (e.g. for active vision). The main focus of the course is on methods to reach such intelligent robot behaviors. This includes architectures, space representation, self localisation, navigation, visual servoing, online robot learning, robotics simulation. Within the scope of the exercise, selected topics are deepened in an application-related manner. Contents at a glance:

- Applications of Cognitive Robot Systems
- Cognitive perception-action systems
- Components of robot systems
- Sensor components as basis for autonomy
- Coordinate systems and transformations
- Visual Servoing of a robot arm
- Representation of environment
- Robot motion planning
- Probabilistic robot localisation
- Online robot learning for navigation
- Robotics simulation
- Programming of cognitive robot systems
- Robot Operating System

Learning objectives / skills English

Students should get to know possible architectures of cognitive robot systems. They should understand selected methods to solve motion planning and robot navigation, self localisation and obstacle avoidance, and should be familiar with the basic mathematics. For selected problems, they should be able to propose and evaluate potential configurations for cognitive robot systems.

Literatur

- R. Arkin: Behavior-Based Robotics, The MIT Press, 1998.
- H. Choset, at al.: Principles of Robot Motion, MIT Press, 2005.
- J. Latombe: Robot Motion Planning, Kluwer Academic Publishers, 1991.
- S. NIKU: Introduction to Robotics, Prentice Hall, 2001.
- B. Siciliano, O. Khatib: Handbook of Robotics, Springer, 2008.
- Ausgewählte Zeitschriftenartikel.
- Aktuelle eigene Artikel sowie Bachelor-/Master-/Doktorarbeiten.

Kursname laut Prüfungsordnung

Computer / Robot Vision

Course title English

Computer / Robot Vision

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Die Veranstaltung behandelt Methoden zur Extraktion von geometrischen Strukturen aus Einzelbildern und bei dynamischen Szenen die Erfassung und Charakterisierung der Objektbewegungen aus Bildfolgen. Für Robotik-Anwendungen werden Methoden zur Kameramodellierung, und darauf basierend Methoden zur 3D Hindernislokalisierung und zur automatisierten 3D Szenenrekonstruktion behandelt. Inhalte im Einzelnen:

- Einführung (Anwendungen, Verarbeitungsablauf)
- Medium-Level Strukturextraktion (Geraden, Konturen, Aktive Konturen, Hough-Transformation)
- Kameramodellierung (Linsen, Kamermerkmale, Projektionsmodelle, Bildentstehung, Kamerakalibrierung)
- Bildfolgenanalyse (änderungsdetektion, Objektverfolgung, Optischer Fluss, Korrespondenzanalyse)
- Hindernisdetektion und Kartenerstellung (Objektlokalisierung, Kameralokalisierung, Dynamische Szenenrekonstruktion)

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden sollen die zu zugrunde liegenden mathematischen Ansätze verstehen und unter Verwendung einer Computer Vision Plattform entsprechende Verfahren implementieren, sowie über die Eignung ausgewählter Computer/Robot Vision Verfahren für bestimmte Aufgabenstellungen urteilen können.

Description / Content English

The course treats methods for extraction of geometric structures from single images and for dynamic scenes the extraction and characterisation of object movements from image sequences. For robot applications, methods for camera modelling, 3D obstacle localisation, and automatic 3D scene reconstruction are treated. Contens at a glance:

- Introduction (applications, processing steps)
- Medium-level processing (extraction of lines and contours, snakes, Hough transformation)
- Camera modelling (lenses, projections, calibration, image formation)
- Image sequence analysis (change detection, object tracking, optical flow, feature matching)
- Obstacle detection and map building (object and camera localisation, dynamic scene reconstruction)

Learning objectives / skills English

The students should understand the basic mathematics, be able to implement certain approaches on a Computer Vision platform, and judge the qualification of selected Computer/Robot Vision approaches for certain tasks.

Literatur

- D. Forsyth: Computer Vision - A Modern Approach; Prentice Hall, 2002.
- R. Hartley, et al.: Multiple View Geometry in Computer Vision, Cambridge University Press, 2004.
- N. Paragios, Y. Chen: Handbook of Mathematical Models in Computer Vision, Springer, 2006.
- S. Prince: Computer Vision - Models, Learning, Inference, Cambridge University Press, 2012.
- R. Szeliski: Computer Vision - Algorithms and Applications, Springer, 2011.
- E. Trucco, et al.: Introductory Techniques for 3D Computer Vision; Prentice Hall, 1998.
- Ausgewählte Zeitschriftenartikel.
- Aktuelle eigene Artikel sowie Bachelor-/Master-/Doktorarbeiten.

Kursname laut Prüfungsordnung

Computer Graphics

Course title English

Computer Graphics

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		

Prüfungsleistung

mündliche Prüfung, 45 Minuten

Beschreibung / Inhalt Deutsch

Die Vorlesung führt in die Grundlagen der Computergraphik ein. Sie stellt Begriffe und Algorithmen der Rastergraphik vor, führt in die wichtigsten Methoden der low level Bildverarbeitung ein und erarbeitet Modellierungs- und Beleuchtungsmodelle der 3D-Graphik.

Inhalte im Einzelnen:

- Geometrische Modelle, insbesondere Dreiecksmeshes und parametrische linien und Flächen
- Farbmodelle, Farbwahrnehmung
- Beleuchtungssimulation
- Reflexionen
- Strahlverfolgungsalgorithmen
- Beschleunigungsstrukturen
- Geometrische Transformationen
- Rastisierungsalgorithmen und APIs zur hardwarebeschleunigten Rasterisierung
- Abtasttheorie

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden lernen, die grundlegenden Komponenten der Computergrafik kennen. Sie beherrschen die Grundlagen der geometrischen Modellierung sowohl mit polygonalen Primitiven als auch mittels parametrischer Beschreibung. Sie sind insbesondere in der Lage für ein gegebenes Szenario die beste Repräsentation zu wählen. Weiterhin kennen Sie die wichtigsten Eigenschaften der Aussehensmodellierung. Sie wissen wie Licht und Farben Simuliert, Repräsentiert und wahrgenommen werden. Die Studierenden beherrschen, die beiden grundlegenden Verfahren zur dreidimensionalen Bildgebung: die Strahlverfolgung und die Rasterisierung. Auch hier sind Sie nicht nur in der Lage beide Ansätze effizient umzusetzen sondern auch das jeweils beste Verfahren für ein gegebenes Szenario auszuwählen.

Description / Content English

The Lecture introduces the basics of computer graphic. It presents terms and algorithms of raster graphic, introduces the most important methods of low level image processing and develops modeling –and lightening models of 3D graphics.

Content in particular:

- Geometric models, in particular triangular meshes and parametric lines and surfaces
- Color Models and color perception
- Shading and lighting
- Reflections
- Ray-Tracing algorithms
- Ray-Tracing accelerations
- Geometric transformations

- Rasterization and APIs for hardware-aided rasterization
- Sampling Theory

Learning objectives / skills English

The students learn the basic concepts of computer graphics. They will have insight into the generation of geometric models using both polygonal approximations as well as parametric lines and surfaces. They learn to make a difference between those two basic concepts and will be able to choose the most appropriate representation for any given task. Furthermore, they will know the most important appearance modelling methods. They will understand the basics of light- and color-simulation, representation, and perception. The students learn the two basic 3D rendering techniques, ray-tracing and rasterization. They understand the strengths and weaknesses of both approaches and are able to choose the most appropriate for any given scenario.

Literatur

- Aktuelle Internetliteratur
- Foley, Van Dam, Feiner, Hughes ... Computer Graphics: Principles and Practice Addison-Wesley
- Watt & Watt, Computer Graphics, Addison-Wesley
- Glassner, Principles of digital image synthesis, Morgan Kaufman

Kursname laut Prüfungsordnung**Digital Games Research****Course title English**

Digital Games Research

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS/SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Die Veranstaltung konzentriert sich auf interdisziplinäre Aspekte digitaler Spiele, wobei je nach Schwerpunkt der Veranstaltung einzelne Themen genauer beleuchtet werden.

Lernergebnisse / Kompetenzen Deutsch

Den Studierenden werden der Entwicklungsprozess sowie interdisziplinäre Aspekte digitaler Spiele als neuer Medientyp näher gebracht. Die Veranstaltung zielt darauf ab, den Studierenden aufzuzeigen, wie Spiele funktionieren, wie sie entwickelt werden, welche Konsequenzen ihre Nutzung mit sich bringt und wie sie als Instrument und Medium in verschiedensten Szenarien eingesetzt werden können. Es werden neueste Technologien und aktuelle Forschungsfragen berücksichtigt.

Description / Content English

The course is focused on interdisciplinary aspects of digital game design and development, concentrating on single current aspects of digital games in detail.

Learning objectives / skills English

Students will learn about development and interdisciplinary aspects of digital games as new media type. The course aims to teach students about how games work, how they are developed and programmed, what consequences their usage has and how they can be used as tools. It addresses advanced technologies and current research questions.

Literatur

Wird vorlesungsspezifisch ausgegeben / Lecture-specific

Kursname laut Prüfungsordnung**Distributed Systems****Course title English**

Distributed Systems

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Die Vorlesung befasst sich mit den grundlegenden Konzepten und Protokollen für verteilte Systeme.

Die Vorlesung beginnt mit Grundlagen zur verteilten Kommunikation:

- Serialisierung (ASN.1, CORBA XDR, SOAP)
- Remote Procedure Calls
- Verteilte Objekte

und widmet sich dann wichtigen Basisalgorithmen

- Physikalische Uhren
- Logische Uhren
- Transaktionen
- Synchronisation
- Replikation und Konsistenz
- Globaler Zustand

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden kennen die wesentlichen Grundlagen, Protokolle, Algorithmen und Architekturen Verteilter Systeme und können diese anwenden.

Description / Content English

The lecture presents important concepts and protocols for distributed systems.

The lecture starts with principles of distributed communication:

- Data serialization (ASN.1, CORBA XDR, SOAP)
- Remote procedure calls
- Distributed objects

The second part of the lecture presents important and often used distributed algorithms:

- Physical clocks
- Logical clocks
- Transactions
- Synchronisation
- Replication and consistency
- Global state

Learning objectives / skills English

The students know the principles, protocols, algorithms and architecture of distributed systems are able to apply these to real word problems.

Literatur

- 1 Coulouris/Dollimore/Kindberg: Distributed Systems - Concepts and Design, Addison-Wesley 2001 (3rd edition).
- 2 Tannenbaum/van Steen: Distributed Systems - Principles and Paradigms, Prentice Hall 2002.
- 3 Borghoff/Schlichter: Rechnergestützte Gruppenarbeit (in German), Springer 1998.

Kursname laut Prüfungsordnung**Electronic Communities and Social Networks****Course title English**

Electronic Communities and Social Networks

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Unterstützt durch technische Systeme, wie email-Systeme, Diskussionsforen, oder gemeinsame Arbeitsbereiche, wird die Bildung von virtuellen Gemeinschaften und Netzwerken ermöglicht. Diese Gemeinschaften besitzen u.a. gemeinsame Interessen, bilden virtuelle Unternehmen oder entwickeln gemeinsame Softwareprojekte (Open Source Prinzip). Aufgrund der inhärenten räumlichen und / oder zeitlichen Verteilung der Beteiligten ergeben sich spezielle Anforderungen zur Erhöhung der Wahrnehmung der Kooperationspartner untereinander bzw. bezüglich der erstellen / geteilten Artefakte und Dokumente.

Inhalte im Einzelnen:

- Modellierung von elektronischen Gemeinschaften
- Analyse und Visualisierung sozialer Netzwerke und von Themenstrukturen
- Vorschlagsgenerierung von Mitgliedern/Artefakten aufgrund von Interessensprofilen
- Mechanismen und Voraussetzungen zur Bildung elektronischer Gemeinschaften
- Evolution von und Trenderkennung in elektronischen Gemeinschaften

Lernergebnisse / Kompetenzen Deutsch

Durch diese Veranstaltung sollen die Studierenden in die Lage versetzt werden,

- gängige Verfahren zur Modellierung von elektronischen Gemeinschaften zu verstehen und hinsichtlich ihrer Einsatzmöglichkeiten/-grenzen zu beurteilen
- Methoden zur Vorschlagsgenerierung zu kennen und zu verstehen
- die Bedingungen für Entstehung und Wandel elektronischer Gemeinschaften zu verstehen und zu analysieren
- Verfahren zur Analyse und Visualisierung sozialer Netzwerke und von Themenstrukturen zu verstehen, selbst anwenden und entwickeln zu können und deren Eignung zu beurteilen

Description / Content English

Supported by technical systems, like email systems, discussion forums, or common work areas, the education is made possible by virtual communities and networks. These communities have common interests, build virtual enterprises or develop common software projects (Open Source principle). Due to the inherent spatial and/or temporal distribution of the involved arise special requirements for increasing of the perception of the cooperation partners and/or concerning provided/ divided artifacts and documents.

Contents in details:

- Modeling of electronic communities
- Analysis and visualization of social networks and topic structures
- Generation of proposal of members/artifacts due to interest profiles
- Mechanisms and conditions for the education of electronic communities
- Evolution of and trend recognition in electronic communities

Learning objectives / skills English

During this lecture, the students are into the position

- To understand methods for modeling electronic communities and evaluate their application type/borders regarding usual procedures
- To know and understand methods for generation of proposal
- To understand and analyze the conditions for development and change of electronic communities
- Procedures for analysis and visualization of social networks and understanding topic structures, to be able to use and develop their qualification

Literatur

- Borghoff / Schlichter: Rechnergestützte Gruppenarbeit, Springer Verlag, 2000
- Wassermann / Faust: Social Network Analysis, Cambridge University Press, 1994
- Wenger: Communities of practice: Learning, meaning, and identity, Cambridge University press, 1998

Kursname laut Prüfungsordnung**Entwicklung sicherer Software****Course title English**

Development of Safe and Secure Software

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Der Terminus "Sicherheit" wird in zwei Bedeutungen verwendet: Safety bedeutet, dass ein System funktioniert, ohne seine Umwelt zu gefährden. Security bedeutet, dass ein System vor Angriffen aus seiner Umwelt geschützt werden muss. Software muss so konstruiert werden, dass das System, innerhalb dessen die Software eingesetzt wird, geforderte Sicherheitseigenschaften erfüllt. Bisher wurden Safety und Security weitgehend getrennt betrachtet und behandelt. Jedoch sind immer mehr Systeme sowohl Safety- als auch Security-kritisch. Diese Veranstaltung zeigt Wege auf, Software so zu konstruieren, dass sie beiden Arten von Sicherheit gerecht wird.

Inhalt im Einzelnen:

- Konzept von Safety, Terminologie
- Konzept von Security, Terminologie
- Typische Safety-Anforderungen
- Typische Security-Anforderungen (Vertraulichkeit, Verfügbarkeit, Integrität und deren Derivate)
- Zusammenhang von Safety und Security, sich ergänzende und sich widersprechende Ziele
- Bedrohungsanalysen für Safety und Security (z.B. Hazard Analysis, Angreifermodellierung)
- Sicherheit von Systemen vs. Sicherheit von Software
- Bedrohungs- und Risikoanalyse
- Maßnahmen zur Etablierung von Safety- und Security-Eigenschaften (z.B. Sicherheitsarchitekturen, Sicherheitsinfrastrukturen, Protokolle)
- Standards für Safety und Security (IEC 61508, ISO 27001, Common Criteria)
- Prozess zur Entwicklung sicherer Software (Erhebung und Repräsentation von Sicherheitszielen, Abwägung konfliktierender Anforderungen, Auswahl von Sicherheitsmechanismen, Einfließen von gewählten Sicherheitsmechanismen in die Architektur der Software, Implementierungs- und Testaspekte)

Lernergebnisse / Kompetenzen Deutsch

- Safety und Security beschreiben und deren Zusammenhänge erklären können
- Sicherheit von Systemen mit Sicherheit von Software in Verbindung setzen können
- Techniken zur Spezifikation von Sicherheitseigenschaften kennen und anwenden können
- Rolle von Standards erklären können
- Techniken zur Etablierung von Safety und Security nennen und erklären können
- Erklären können, wie Software so konstruiert werden kann, dass Sicherheitsanforderungen von vorneherein berücksichtigt werden

Description / Content English

Safety means that a system works without endangering its environment. Security means to protect a system against attacks from its environment. Software must be developed in a way, that the system containing the software fulfills the safety and security requirements. Up to now, safety and security were mostly considered

separately. In the near future we will have an increasing number of systems that have to satisfy safety as well as security requirements. This course shows how to construct software that contributes to safety as well as security.

Content in detail:

- Concept of safety, terminology
- Concept of security, terminology
- Typical safety requirements (functional properties)
- Typical security requirements (confidentiality, availability, integrity, and their refined versions)
- Interrelationships of safety and security, supplementary and conflicting objectives.
- Safety and security of systems vs. safety and security of software
- Threat and risk analysis
- Measures to establish safety and security properties (e.g., architectures, infrastructure and protocols)
- Standards for safety and security (IEC 61508, ISO 27001 Common Criteria)
- Process for developing safe and secure software (Safety and security objectives elicitation, requirements engineering and analysis, selection of safety and security mechanisms, safety and security mechanisms in the software architecture, implementation and testing issues)

Learning objectives / skills English

The students are able to

- describe safety and security and explain their interrelationship,
- explain relationship between safety and security of systems and of software,
- know and use techniques for specification of security and safety properties,
- explain the role of standards
- know and explain techniques to establish safety and security properties,
- explain how to develop software in a way that security and safety requirements are considered from the beginning

Literatur

- Anderson, R. Security Engineering, Wiley 2001.
- Pfleeger, C. P. Security in Computing, Prentice Hall, 2003.
- Markus Schumacher, Eduardo Fernandez-Buglioni, Duane Hybertson, Frank Buschmann, and Peter Sommerlad. Security Patterns - Integrating Security and Systems Engineering. Wiley, March 2006.
- Nancy Leveson. Safeware: System Safety and Computers. Addison-Wesley, 1995.
- International Electrotechnical Commission. Functional safety of electrical/electronic/programmable electronic safety-relevant systems, 1998.
- Common Criteria for Information Technology Security Evaluation, 1999, siehe <http://www.commoncriteria.org>
- sowie weitere Literatur zu diesem Thema gemäß Mitteilung in der Veranstaltung

Kursname laut Prüfungsordnung**Fehlerdiagnose und Fehlertoleranz in technischen Systemen****Course title English**

Fault Diagnosis and Fault Tolerance in Technical Systems

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
4	SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	1		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Hohe Zuverlässigkeit und Verfügbarkeit spielen in der Automatisierungstechnik eine wichtige Rolle. Schlüsseltechnologien sind Fehlerdiagnose sowie fehlertolerante Systeme. Im Rahmen dieser Lehrveranstaltung werden statistische, daten-basierte und modellgestützte Methoden zur Fehlerdiagnose und zur fehlertoleranten Regelung sowie die erforderlichen Entwurfsalgorithmen und Tools vorgestellt.

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden sollen in der Lage, statistische, daten-basierte und modellgestützte Methoden zur Fehlerdiagnose und zur fehlertoleranten Regelung anzuwenden.

Description / Content English

A very critical and important issue concerning the design of automatic control systems with increasing complexity is to guarantee a high system performance over a wide operating range and meeting the requirements on system reliability and dependability. As one of the key technologies for the problem solution, advanced fault detection and identification (FDI) technology and fault tolerant systems (FTC) are receiving considerable attention. The objective of this course is to introduce basic model based FDI and fault tolerant schemes, advanced analysis and design algorithms and the needed tools.

The course consists of 6 parts.

Part I: Basic fault detection problems and the associated solutions.

The following two topics are addressed in this part:

- Basic statistical methods for change/fault detection
- Basic deterministic methods for change/fault detection

Part II: Basic data-driven methods

The following two topics are addressed:

- Basic data-driven methods for statistic processes
- A basic data-driven method for deterministic processes

Part III: model-based FDI methods

- Two essential problems
- Essentials: Modelling and residual generation
- Fault detection in stochastic systems
- Fault detection in deterministic systems

Part IV: Data-driven design of dynamic FDI systems

- Subspace identification technique (SIT) aided design of observer-based FDI systems

Part V: Fault isolation and identification schemes

- Basic isolation and identification methods
- Methods to a structural fault isolation (for dynamic processes)

Part VI: Fault-tolerant systems

Learning objectives / skills English

The students should be able to apply statistical, data-driven and model-based FDI and FTC methods to real cases.

Literatur

Steven X. Ding, Model-based fault diagnosis techniques, Springer-Verlag, 2008.

Selected publications in leading international journals.

Kursname laut Prüfungsordnung**Formale Spezifikation von Software-Systemen****Course title English**

Formal Specification of Software Systems

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Wie empirische Untersuchungen ergeben haben, scheitern Softwareprojekte oftmals wegen unklarer Anforderungen. Softwaresysteme sollten daher genau spezifiziert werden, bevor mit der Implementierung begonnen wird. Da die natürliche Sprache zu unpräzise für eine genaue Spezifikation ist, sollten formale Sprachen zur Softwarespezifikation verwendet werden. Diese ermöglichen es auch, die Eigenschaften des Softwaresystems zu analysieren, bevor es implementiert ist.

In der Veranstaltung wird die Spezifikationssprache Z gelernt.

Inhalte im Einzelnen:

- Motivation: wozu formale Spezifikation?
- verschiedene Arten formaler Spezifikationssprachen (daten- vs. verhaltensorientiert, algebraisch- vs. modellbasiert)
- Notation für Mengen und Prädikate
- Z-Methodik
- Object-Z
- Typchecker für Z

Lernergebnisse / Kompetenzen Deutsch

- Vor- und Nachteile formaler Spezifikationen nennen können
- Verschiedene Arten formaler Spezifikationssprachen aufzählen und deren Unterschiede erläutern können
- Formale Spezifikationen in Z und Object-Z lesen und interpretieren können
- Formale Spezifikationen in Z und Object-Z anfertigen können
- Den Begriff der Verfeinerung erläutern können
- Verfeinerungsbeweise führen können
- Typchecker anwenden können

Description / Content English

As empirical studies show, software projects often fail because of unclear requirements. Software systems should therefore be precisely specified, before the implementation begins. Since natural language is too imprecise for a detailed specification, formal specification languages should be used. Using such languages enables a developer to analyze the properties of a software system before it is implemented.

In this course, students learn the specification language Z.

Contents in detail:

- Motivation: why formal specification?
- Different types of formal specification languages

(data- vs. behavior-oriented, algebraic- vs. model-based)

- Notation for sets and predicates
- Z-Methodology
- Object-Z
- Type checker for Z

Learning objectives / skills English

- Be able to state state advantages and disadvantages of formal specifications
- Be able to name various types of formal specification languages and explain their differences
- Be able to read and interpret formal specifications in Z and Object-Z
- Be able to develop formal specifications in Z and Object-Z
- Be able to explain the notion of refinement
- Be able to perform refinement proofs
- Be able to use typecheckers

Literatur

- J. M. Spivey: The Z Notation - A Reference Manual, Prentice Hall, 1992, verfügbar unter <http://spivey.orient.ox.ac.uk/~mike/zrm/>
- J. B. Wordsworth: Software Development with Z, Addison-Wesley, 1992
- G. Smith: The Object-Z Specification Language, Kluwer Academic Publishers, 1999

Kursname laut Prüfungsordnung			
Game Architecture and Design			
Course title English			
Game Architecture and Design			
Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS/SS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Die Veranstaltung fokussiert sich auf das Design und die Entwicklung digitaler Spiele, wobei folgende Themen abgedeckt werden:

- Innovative Spieleideen
- Produktionsprozess digitaler Spiele
- Produktionswerkzeuge für Spiele
- Projekt Engineering
- Struktur von Spielen
- Gameplay & Game Balance
- Playtesting, Qualitätssicherung und Spielspaß
- Entertainment Interface Design
- Interactive Storytelling
- Character Development
- Game Business
- Zukunft der interaktiven Unterhaltung

Lernergebnisse / Kompetenzen Deutsch

Die Teilnehmer lernen ihr eigenes Computerspiel zu entwerfen und zu entwickeln, angefangen von der Idee bis zum voll spielbaren Prototyp. Die Lernziele sind:

- Design und Entwurfsprinzipien digitaler Spiele kennen lernen
- Arbeitsweise und Aufbau digitaler Spiele verstehen
- Den Prozess der Entwicklung digitaler Spiele verstehen
- Die Spieleindustrie kennen lernen
- Entwurf und Entwicklung eines eigenen Spiels im Team
- Intensive Anwendung von agilem Projektmanagement in einem Multimedia-Projekt

Description / Content English

The course focuses on the design and development of digital games covering the following topics:

- Game Innovation and Ideas
- The Production Process of Digital Games
- Game Production Tools
- Project Engineering
- Structure of Games
- Gameplay & Game Balance
- Playtesting, Quality Assurance and Fun
- Entertainment Interface Design
- Interactive Storytelling
- Character Development

- Game Business
- The Future of Interactive Entertainment

Learning objectives / skills English

Students will learn to design and to develop their own computer game, from the very idea to a complete playable game prototype. The course teaches students the following:

- Learn about the design and development of computer games
- Learn how digital games work and what they are made of
- Understand the process of game development
- Understand the games industry
- Design and produce your own game in a team
- Learn intensively about project management of multimedia projects with the latest agile software design methodologies

Literatur

- Fullerton, Tracy et al: Game Design Workshop, 2nd ed. Morgan Kaufmann, 2008.
- Ernest Adams: Fundamentals of Game Design, 2nd Edition, New Riders, 2009.
- Jesse Schell: The Art of Game Design. A book of Lenses, Morgan Kaufmann, 2008.

Kursname laut Prüfungsordnung**Gestaltung interaktiver Lehr-/Lern-Systeme****Course title English**

Design of Interactive Learning Systems

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Die Veranstaltung zielt auf die Bereitstellung von Wissen für den theoriegeleiteten Entwurf interaktiver und kooperativer Lehr-/Lernsysteme. Als Basiswissen gehören hierzu Medientheorien und mediendidaktische Ansätze sowie Gestaltungsprinzipien und Modellierungsmethoden für interaktive Systeme. Lernermodellierung wird als Spezialisierung der Benutzermodellierung eingeführt, CSCL-Umgebungen als Spezialisierung allg. kooperativer Systeme. Neue Entwicklungen im Bereich Metadaten und Lernprozessmodellierung (IMS-LD) werden aufgegriffen.

Inhalte im Einzelnen:

- Entwicklungslinien der Mensch-Computer-Interaktion
- Medientheorie und Mediendidaktik (überblick und Auswahl)
- Gestaltungsprinzipien und -methoden für interaktive und kooperative Systeme
- Entwicklungslinien des computergestützten Lehrens und Lernens
- Architektur und Implementierung intelligenter Lehr-/Lernsysteme
- Lernermodellierung
- verteilte gruppenorientierte Lernumgebungen / CSCL
- Metadaten und Lernprozessmodellierung

Lernergebnisse / Kompetenzen Deutsch

Die Veranstaltung soll den Erwerb folgender Fähigkeiten unterstützen:

- Befähigung zur fachkundig-kritischen Beurteilung des Designs von interaktiven und kooperativen Systemen, speziell im Bereich Lehr-/Lernsysteme;
- Kenntnis historischer Entwicklungslinien und wichtiger Beispielsysteme und Ansätze (Modellierungs- und Implementierungsmethoden);
- Befähigung zur Konzeption neuer Lehr-/Lernsysteme auf der Basis des aktuellen "State-of-Art".

Description / Content English

This lecture aims at providing knowledge about the theory guided design and development of interactive and cooperative teaching and learning systems. This comprises basic knowledge about media theories and approaches to media education ("didactics" in the European sense) as well as principles of designing and modelling interactive systems. Student modelling is introduced as a special variant of user modelling, as CSCL environments are introduced as variants general cooperative systems. New developments in the areas of learning metadata and learning process modelling (IMS-LD) are also taken up. The lecture is structured as follows:

- historical lines of development of interactive systems and HCI
- media theories and "media didactics" (survey with selected examples)
- design principles and methods for interactive and cooperative systems
- architecture and implementation of intelligent educational environments
- student modelling

- distributed environments for group learning / CSCL
- metadata and learning process modelling

Learning objectives / skills English

The lecture should support the acquisition of the following abilities:

- qualification for professional critical evaluation of the design of interactive and cooperative systems, especially in the area of didactical-/ interactive teaching and learning systems
- knowledge of development directions and important example systems and basic approach (modeling- and implementation methods)
- qualifications for conceiving a conception of new didactical -/ interactive teaching and learning systems on the basis of current „State-of-the-Art“

Literatur

- Dix/Finlay/Abowd/Beale: Human-Computer-Interaction (Prentice Hall, 1998 / Pearson, 2004)
- B. Shneiderman: Designing the User Interface (Addison-Wesley, 1998)
- Borghoff/Schlichter: Rechnergestützte Gruppenarbeit (Springer, 1998)
- E. Wenger: Artificial intelligence and tutoring systems (Morgan Kaufmann Press, 1987)
- P. Dillenbourg (Hrsg.): Collaborative Learning (Pergamon, 1999)
- M. Vogel: Medien der Vernunft (Suhrkamp, 2001)

Kursname laut Prüfungsordnung**Information Mining****Course title English**

Information Mining

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Information Mining beschäftigt sich mit dem Extrahieren von impliziten, noch unbekannten Informationen aus Rohdaten (Data Mining) bzw. Texten (Text Mining). Dazu sollen Computer in die Lage versetzt werden, Datenbasen automatisch nach Gesetzmäßigkeiten und Mustern zu durchsuchen und einen Abstraktionsprozess durchzuführen, der als Ergebnis aussagekräftige Informationen liefert. Das maschinelle Lernen stellt dafür die Werkzeuge und Techniken zur Verfügung.

Inhalte im Einzelnen:

- Ein- und Ausgabe
- Algorithmen: Klassifikation, numerische Vorhersage, Assoziationen, Clustering
- Evaluierung von Data-Mining-Methoden
- Implementierung: Maschinelles Lernen in der Praxis
- Aufbereitung der Ein- und Ausgabe
- Data Mining für zeitabhängige Daten
- Data Mining für soziale Netze
- Text-Clustering: flaches/hierarchisches Clustering, Evaluierung, Optimum Clustering Framework
- Text-Klassifikation

Lernergebnisse / Kompetenzen Deutsch

Studierende sollen die theoretischen Grundlagen von Information Mining-Methoden verstehen, diese Methoden beherrschen, entsprechende Evaluierungsverfahren anwenden können sowie Möglichkeiten und Grenzen solcher Methoden beurteilen können.

Description / Content English

Information Mining deals with the extraction on implicit information from raw data (Data Mining) or text (Text Mining). The goal is the development of methods for analyzing databases and discovering useful information by means of abstraction. For this purpose, machine learning methods are applied.

Contents:

- Input and Output
- Algorithms: Classification, Numeric Prediction, Associations, Clustering
- Evaluation of Data Mining Methods
- Implementations: Real machine learning schemes
- Transformations: Engineering the input and output
- Data Mining on Time-dependent Data
- Data Mining for Social Networks
- Text clustering: flat/hierarchic Clustering, Evaluation, Optimum Clustering Framework

- Text classification

Learning objectives / skills English

Students will understand the theoretic concepts underlying information mining methods, be able to apply these methods and evaluate the outcome. Furthermore, they will understand the possibilities and limitations of these methods.

Literatur

- Ian Witten, Eibe Frank, Mark Hall: Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufman, 2011.
- Gary Miner, John Elder IV, Thomas Hill, Robert Nisbet, Dursun Delen, Andrew Fast: Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications. Academic Press, 2012.
- Trevor Hastie , Robert Tibshirani, Jerome Friedman: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, 2009
- Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, Introduction to Information Retrieval, Cambridge University Press. 2008.

Kursname laut Prüfungsordnung**Information Retrieval****Course title English**

Information Retrieval

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Information Retrieval beschäftigt sich mit Vagheit und Unsicherheit in Informationssystemen. In dieser Lehrveranstaltung sollen weiterführende Konzepte aus diesem Bereich behandelt werden.

Inhalte im Einzelnen:

1. Modelle:

- Retrieval als unsichere Inferenz
- Aussagenlogische Modelle
- Prädikatenlogische Modelle
- spezielle probabilistische Modelle

2. Interaktives Retrieval

- Probabilistisches Ranking-Prinzip für interaktives IR
- Kognitive Modelle
- Gestaltung von User Interfaces für IR

3. Multimedia-Retrieval

- Syntax, Semantik und Pragmatik
- Retrieval von Bildern
- Audio Retrieval
- Video Retrieval

4. Retrieval von strukturierten und semi-strukturierten Daten

- IR und Datenbanken
- XML-Retrieval

5. Verteiltes Retrieval:

- Modelle: CORI, entscheidungstheoretisches Modell
- Architekturen: zentraler Broker, P2P, Grid
- Verteiltes Clustering

6. Implementierung von IR-Systemen:

- Zugriffspfade und Algorithmen
- Implementierung verteilter IR-Systeme

7. Evaluierung:

- nichtlinearen Rangordnungen
- interaktives Retrieval

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden sollen die weiterführenden IR-Modelle verstehen, sie sollen wissen, welche verschiedenen Methoden für Multimedia-Retrieval sowie Retrieval strukturierter und semi-strukturierter Daten es gibt, und diese hinsichtlich ihrer Eignung für konkrete Anwendungen beurteilen können. Sie sollen wissen, welche Modelle und Architekturen es für verteiltes Retrieval gibt. Ferner sollen sie die Funktionsweise verschiedener Implementierungen von IR-Systemen verstehen, und sie sollen weiterführende Evaluierungsmethoden anwenden können.

Description / Content English

Information retrieval deals with vagueness and imprecision in information systems. This course will teach advanced concepts of this field:

1. Models:

- Retrieval as uncertain inference
- Models based on propositional logic
- Models based on predicate logic
- specific probabilistic models

2. Interactive retrieval

- Probability ranking principle for interactive IR

- Cognitive models

- Design of user interfaces for IR

3. Multimedia retrieval

- Syntax, semantics and pragmatics

- Image retrieval

- Audio retrieval

- Video retrieval

4. Retrieval of structured and semi-structured data

- IR and databases

- XML retrieval

5. Distributed retrieval

- Models: Similarity-based, Decision theoretic framework

- Architecture: central brokers, P2P, Grid

- Distributed clustering

6. Implementation of IR Systems

- Access methods and algorithms

- Implementaiton of distributed IR systems

7. Evaluation

- nonlinear rankings

- interactive retrieval

Learning objectives / skills English

Students will learn about advanced IR models, they will know models for retrieving multimedia or (semi-)structured data and assess their suitability for a concrete application. They will know models and methods for interactive and distributed retrieval. They will learn about different methods for implementing an IR system, and they will be able to apply advanced evaluation methods.

Literatur

- P. Ingwersen, K. Järvelin: The TURN: Integration of Information Seeking and Retrieval in Context. Springer, 2005.
- R. Belew: Finding Out About. A Cognitive Perspective on Search Engine Technology and the WWW. Cambridge University Press, 2000.
- Reginald Ferber: Data Mining und Information Retrieval. dpunkt Verlag, 2003.
- Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, Introduction to Information Retrieval, Cambridge University Press. 2008.

Kursname laut Prüfungsordnung**Intelligent Learning Environments****Course title English**

Intelligent Learning Environments

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3		1	
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Computer und "maschinelle Intelligenz" werden häufig als Mittel zur Bewältigung der heutigen kritischen Bildungsherausforderungen diskutiert: Remote Learning, Lernen im eigenen Tempo, auf die eigenen Bedürfnisse und den eigenen Hintergrund abgestimmtes Lernen, Bereitstellung einer hochwertigen Bildung an und für alle. In diesem Kurs sind alle Masterstudenten mit technischem oder nichttechnischem Hintergrund willkommen. Im Laufe des Semesters werden wir Themen an der Schnittstelle von Künstlicher Intelligenz in der Bildung, Bildungstechnologien und Mensch-Computer-Interaktion behandeln und praktische Übungen durchführen, um unser Verständnis für intelligente Lerntechnologien zu vertiefen. Im Einzelnen werden wir Folgendes besprechen:

- Einführung in Bildungstechnologien
- Künstliche Intelligenz im Bildungswesen (AIED)
- Student Modeling
- Intelligente Tutorenmodelle (ITS)
- Kollaborative Lernumgebungen / MOOCs
- Lernmanagementsysteme / Offene Bildungsressourcen
- Fairness, Rechenschaftspflicht, Transparenz und Ethik in AIED.

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden lernen den aktuellen Stand der Forschung im Bereich der Bildungstechnologien mit einem Schwerpunkt auf Künstlicher Intelligenz im Bildungswesen kennen. Sie werden mit algorithmischen Techniken zur Modellierung von Kognition und Wissen vertraut gemacht und erkunden, wie diese Darstellungen in der Praxis eingesetzt werden. Die Studierenden erforschen verschiedene Lernumgebungen, die von "intelligenten" Algorithmen unterstützt werden, und lernen den Einsatz von Technologie als Werkzeug und Mittel zur Orchestrierung des Lernens kennen.

Description / Content English

Computers and 'machine-intelligence' are frequently discussed as the means for addressing today's critical educational challenges: learning remotely, learning at one's own pace, learning according to one's needs and background, providing quality education to all and for all.

In this course, we welcome all master-level students with technical or non-technical background. Through the semester, we will cover topics on the intersection of Artificial Intelligence in Education, Educational Technologies, and Human-Computer Interaction and we will carry out hands-on exercises to deepen our understanding regarding intelligent learning technologies. Specifically, we will go over the following:

Introduction to educational technologies

- Artificial intelligence in education (AIED)

- Student Modeling
- Intelligent Tutoring Systems (ITS)
- Collaborative learning environments / MOOCs
- Learning Management Systems / Open Educational Resources
- Fairness, Accountability, Transparency and Ethics in AIED.

Learning objectives / skills English

Students will learn about the state-of-the-art research in Educational Technologies with a focus on Artificial Intelligence in Education. They will familiarize with algorithmic techniques for modeling cognition and knowledge, and they will explore how these representations are used in practice. Students will explore various learning environments supported by "intelligent" algorithms and will learn about using technology as a tool and means for orchestrating learning.

Literatur

- How People Learn: Brain, Mind, Experience, and School: Expanded Edition (2000), National Research Council. Washington, DC: The National Academies Press.<https://doi.org/10.17226/9853>.
- Handbook of design in educational technology (2013), Rosemary Luckin, Sadhana Puntambekar, Peter Goodyear, Barbara Grabowski, Joshua Underwood, and Niall Winters (eds).
- International handbook of computer-supported collaborative learning (2021) Cress, U., Oshima, J., Rosé, C., & Wise, A. (2021).Computer-Supported Collaborative Learning Series,19.
- Ausgewählte Veröffentlichungen (Forschung/Nachrichtenartikel)

Kursname laut Prüfungsordnung**Interaktive Systeme****Course title English**

Interactive Systems

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Die Vorlesung behandelt fortgeschrittene Methoden und Techniken der Mensch-Computer-Interaktion. Insbesondere werden neue Formen der Interaktion wie zum Beispiel interaktive Informationsvisualisierungen, Sprach- und Gesteninteraktion und adaptive Nutzerschnittstellen diskutiert. Wo erforderlich, werden relevante Methoden des maschinellen Lernens eingeführt. Neben theoretischen Grundlagen werden aktuelle Systembeispiele und Forschungsergebnisse vorgestellt und diskutiert.

Inhalte im Einzelnen:

- Modelle und modell-basierte Entwicklung interaktiver Systeme
- Interaktive Informationsvisualisierung, Visual Analytics
- Natural Interaction, interaktive Oberflächen
- Tangible Interfaces, Interaktion bei Ubiquitous Computing
- Perzeptive Interfaces: natürlichsprachliche Schnittstellen, gestenbasierte Interaktion und Body Motion-Erkennung
- Multimodale Schnittstellen, Fusion und Fission von Modalitäten
- Nutzer- und kontext-adaptive Nutzerschnittstellen

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden kennen die wesentlichen Konzepte, Modelle und Techniken zur Konzeption und Realisierung fortgeschrittener interaktiver Systeme und können diese im Zusammenhang darstellen und erläutern. Sie sind in der Lage, Interaktionsformen gezielt auszuwählen, zu gestalten und zu realisieren. Sie kennen wesentliche Forschungsfragen im Bereich innovativer interaktiver Systeme und können neue Entwicklungen einordnen und bewerten.

Description / Content English

The lecture covers advanced methods and techniques of human-computer interaction. In particular, novel forms of interaction are discussed, such as interactive information visualizations, speech and gesture recognition, and adaptive user interfaces. Where appropriate, pertinent machine learning methods are introduced. In addition to theoretical foundations, current system examples and research results will be presented and discussed.

Content overview:

- Models and model-based development of interactive systems
- Interactive information visualization and visual analytics
- Natural interaction, interactive surfaces
- Tangible interfaces, interaction in ubiquitous computing
- Perzeptive interfaces: natural language user interfaces, gesture-based interaction, recognition of body movements
- Multimodal Interfaces, fusion and fission of modalities

- User-adaptive and context-adaptive user interfaces

Learning objectives / skills English

The students know the essential concepts, models and techniques for the conceptual design and realization of advanced interactive systems and are able to represent and explain them in their respective context. They are able to select suitable forms of interaction as well as designing and realizing the interfaces. They are aware of the essential research questions in the area of innovative interactive systems and are able to classify and rate new developments.

Literatur

- Preim, B., & Dachselt, R. (2015). Interaktive Systeme. Bd. 2 (3. Aufl.). Springer.
- Ware, C. (2013). Information Visualization – Perception for Design (3rd edition). Morgan Kaufmann
- Jacko, J. A., & Sears, A. (Eds.). (2012). The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications (3rd edition). CRC Press.
- Olsen, D. (1998). Developing User Interfaces: Morgan Kaufmann Publishers
- Card, S. K., MacInlay, J. D., & Schneiderman, B. (1999). Readings in Information Visualization: Using Vision to Think. San Francisco, Cal.: Academic Press/Morgan Kaufman.

Kursname laut Prüfungsordnung**Internet of Things: Protocols and System Software****Course title English**

Internet of Things: Protocols and System Software

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Die Vorlesung vermittelt eine Einführung in das Themengebiet des „Internet der Dinge“ (IoT), in dem Milliarden eingebetteter Systeme (Sensoren, Aktuatoren) in Echtzeit kontinuierlich Daten über die reale Welt im Internet verfügbar machen. Behandelte Themen sind insbesondere: Kommunikationsprotokolle (z.B. IEEE 802.15.4, NBLoT, 6LoWPAN, MQTT), Sensordatenmodellierung und -verwaltung (z.B. linked data, RDF, SSN), Datenzugriff und Plattform-APIs (z.B. web-basierte Systeme, SPARQL, kontinuierliche Anfragen, Complex Event Processing). Neben der Vermittlung theoretischen Wissens, wird in der Übung – im Rahmen von Gruppenprojekten – die praktische Programmierung von IoT-Systemen vermittelt, z.B. mit Arduino-Geräten und Raspberry Pies.

Lernergebnisse / Kompetenzen Deutsch

Die Veranstaltung vermittelt Studierenden ein Verständnis des zukünftigen Internets der Dinge (IoT), der neu auftretenden Anforderungen sowie der technischen Grundlagen, Konzepte, Architekturen und Protokolle. Die Studierenden sollen diese sowohl theoretisch bewerten als auch praktisch einsetzen können, weswegen die Vorlesung von einer praktischen Übung begleitet wird. Schwerpunkte sind insbesondere die IoT-Gerätevernetzung und IoT-Systemsoftware. Hierbei sollen die Studierenden vor allem lernen, welche Unterschiede zu klassischen Internettechnologien und Systemen / Plattformen existieren und woraus diese resultieren.

Description / Content English

The so-called Internet of Things (IoT) is the next step in the evolution of the Internet and is widely expected to change our world in the most fundamental way. Billions of small embedded electronics will make our physical world „smart“, continuously delivering real time information about the state of people, physical structures and the environment, like movements, heat levels, pollution levels and air pressure. In addition, the world becomes ‘programmable’ and physical environments can be changed automatically by software services running in the Cloud.

This course introduces students to the Internet of Things (IoT), its challenges and technologies. Topics of interest include: communication protocols (e.g. IEEE 802.15.4, NBLoT, 6LoWPAN, MQTT), data modelling and storage (e.g. ontologies, linked data, RDF, SSN), data access and platform APIs (e.g. web systems, SPARQL, continuous queries, complex event processing). Besides providing theoretical knowledge, the course also aims at teaching students how to use IoT technologies to realise real systems. To this end, students perform group projects to develop IoT software for current prototype hardware platforms like Arduino and Raspberry Pies.

Learning objectives / skills English

The course introduces students into challenges, concepts and technologies of the Internet of Things (IoT). Students will learn the theoretical backgrounds. They will be able to analyse and assess existing IoT systems and will practise to design and implement new ones. The focus of the course will be on IoT networking and system software. Which concepts and technologies are used in IoT? How and why do they differ from existing Internet technologies? These questions will be answered in the course.

Literatur

Aktuelle Forschungsveröffentlichungen
Details werden in der Vorlesung diskutiert

Kursname laut Prüfungsordnung

Learning Analytics

Course title English

Learning Analytics

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		

Prüfungsleistung

Die Prüfung besteht aus zwei Teilleistungen:

(1) Eine mündliche Prüfung zum Nachweis der in der Vorlesung erarbeiteten theoretischen Konzepte und Methoden sowie

(2) eine Projektarbeit für die konkrete Anwendung und technische Umsetzung der erarbeiteten Theorien.

Die Gesamtbewertung ergibt sich zu 50% aus der Note der Vorlesung, geprüft über eine mündliche Prüfung zum Semesterende und zu 50% aus der semesterbegleitenden Projektarbeit.

Beschreibung / Inhalt Deutsch

In den letzten Jahren hat Learning Analytics (LA) viel Aufmerksamkeit auf sich gezogen, da Anwender, Institutionen und Forscher zunehmend das Potenzial sehen, das LA hat um die Zukunft des technologie-basierten Lernens zu gestalten. LA ist ein aufstrebendes Data Science Forschungsgebiet, das sich mit der Entwicklung von Methoden beschäftigt, die Bildungsdaten nutzen um den Lernprozess zu unterstützen. Erforschung und Entwicklung von LA-Systemen ist ein interdisziplinäres Feld, das Kompetenzen der Informatik, Psychologie, Pädagogik und Didaktik umfasst. LA basiert auf fundierten informatischen Methoden (Statistik, Visualisierung, Social Network Analysis, Machine Learning, Web/Data Mining, Recommender Systems, Visual Analytics, Big Data etc.), die auf das Lernen angewandt werden.

Der erste Teil des Kurses bietet einen systematischen Überblick über dieses Gebiet und seine Schlüsselkonzepte durch ein Referenzmodell für LA, welches auf vier Dimensionen basiert, namentlich Daten, Umgebungen und Kontext (Was?), Akteure (Wer?) Ziele (Warum?) und Methoden (Wie?). Der zweite Teil des Kurses nimmt die vier Dimensionen des LA-Referenzmodells in den Fokus. Dafür werden aktuelle Methoden und Techniken zur Entwicklung innovativer LA-Systeme in Bezug auf jede dieser Dimensionen vorgestellt. Der letzte Teil des Kurses widmet sich aktuellen Trends und Themen der LA-Forschung, die im Rahmen eingeladener Vorträge vorgestellt und diskutiert werden.

Die begleitenden Übungen sind praktische, projektartige Aufgabenstellungen. Ziel ist die Entwicklung und Evaluierung prototypischer LA-Komponenten, bei der die im Kurs erarbeiteten theoretischen Grundlagen Anwendung finden. Die Projekte werden im Verlauf der Vorlesung vorgestellt und diskutiert.

Lernergebnisse / Kompetenzen Deutsch

Nach erfolgreichem Absolvieren des Moduls können Studierende

Kenntnisse:

- die theoretischen und informatischen Grundlagen von Learning Analytics erklären
- einen systematischen Entwicklungsprozess für Learning-Analytics-Systeme beschreiben
- Voraussetzungen und Parameter für die Anwendung verschiedener Learning-Analytics-Methoden diskutieren
- aktuelle Trends und Forschungsfragen in Learning Analytics benennen

Fertigkeiten:

- adäquate Werkzeuge für die Implementation von Learning-Analytics-Systemen auswählen, diese praktisch anwenden und die erreichten Ergebnisse eigenständig beurteilen

- kleinere Learning-Analytics-Entwicklungsprojekte planen und umsetzen

Kompetenzen:

Basierend auf den im Modul erworbenen Kenntnissen und Fertigkeiten können Absolventen

- kreative Lösungen für Learning Analytics vorschlagen
- die Vor- und Nachteile verschiedener Learning-Analytics-Technologien abwägen
- in interdisziplinären Teams innovative Learning-Analytics-Systeme entwerfen und aufbauen
- Verantwortung in Teams übernehmen

Description / Content English

Learning Analytics (LA) has attracted a great deal of attention as practitioners, institutions, and researchers are increasingly seeing the potential that LA has to shape the future technology-enhanced learning landscape. LA is an emerging data science field that represents the application of big data and analytics in education. It deals with the development of methods that harness educational data sets to support the learning process. LA is an interdisciplinary field involving competences from computer science, cognitive psychology, and pedagogy. It leverages various computer science methods. These include statistics, big data, machine learning, data/text mining, information visualization, visual analytics, and recommender systems. The first part of the course will provide a systematic overview on this emerging field and its key concepts through a reference model for LA based on four dimensions, namely data, environments, context (what?), stakeholders (who?), objectives (why?), and methods (how?). In the second part of this courses, we will discuss various methods and techniques required to develop innovative LA systems, in relation to each dimension of the LA reference model. In the last part of the course, current topics and trends in LA research will be presented and dicussed in invited taks. The presented methods and technologies will be further investigated and applied in small student projects carried out throughout the course.

Learning objectives / skills English

After successfully completing the module, students can:

Knowledge:

- Explain the theoretical and technical foundations of learning analytics
- Systematically describe the development process of learning analytics systems
- Discuss requirements and parameters for the use of different learning analytics methods
- Identify current trends and research questions in learning analytics

Skills:

- Select adequate tools for the implementation of learning analytics systems, apply them in practice and assess the achieved results
- Effectively plan and implement small learning analytics projects

Competencies:

Based on the knowledge and skills acquired in the module, students can:

- Propose creative solutions for learning analytics
- Discuss the pros and cons of different learning analytics technologies
- Design and implement innovative learning analytics systems in interdisciplinary teams
- Take responsibility in project development teams

Literatur

- J. Han, M. Kamber: Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers, Second Edition, 2006
- M. Ester, J. Sander: Knowledge Discovery in Databases. Techniken und Anwendungen. Springer Verlag, 2000
- C. Bishop: Pattern Recognition and Machine Learning. Springer, 2006
- T. Munzner: Visualization Analysis and Design. CRC Press, 2014
- M. Ward, G. Grinstein, D.A. Keim: Interactive Data Visualization: Foundations, Techniques, and Application. A.K. Peters, Ltd, 2010
- C. Ware: Information Visualization: Perception for Design. Morgan Kaufmann, 2nd edition, 2004

Kursname laut Prüfungsordnung**Master-Arbeit (einschließlich Kolloquium)****Course title English**

Master-Thesis (including colloquium)

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
30	WS/SS	Deutsch/Englisch	1
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar

Prüfungsleistung

Prüfungsleistung: Durchführung, Dokumentation und Präsentation der Arbeit. Die Bewertung erfolgt durch zwei Prüfer.

Beschreibung / Inhalt Deutsch

Die Master-Arbeit ist eine Prüfungsarbeit, in der die oder der Studierende zum Abschluss des Studiums zeigen soll, dass er innerhalb einer vorgegebenen Frist von 6 Monaten ein Problem selbstständig unter Anleitung nach wissenschaftlichen Methoden bearbeiten kann.

Die Arbeit soll wie ein Projekt in der Praxis unter Beachtung von Methoden des Projektmanagements betreut und durchgeführt werden. Dokumentation und Präsentation (Kolloquium, deutsch oder englisch) sollen zeigen, dass die oder der Studierende in der Lage ist, Zusammenhänge und Ergebnisse verständlich und präzise darzustellen.

Lernergebnisse / Kompetenzen Deutsch

Die Master-Abschlussarbeit stellt eine Prüfungsleistung dar. Neben der fachlichen Vertiefung an einem Beispiel dient sie auch dem Erwerb und der Vertiefung folgender Soft-Skills:

- Selbstlernfähigkeit,
- Teamfähigkeit (Zusammenarbeit mit den Betreuern),
- Anwendung von Methoden des Projektmanagements,
- Kommunikationsfähigkeit: technische Dokumentation und Präsentation, im Fall englischer Präsentation auch Übung von Sprachkenntnissen.

Description / Content English

The master-thesis is an examination paper, in which the student should show that he can solve a problem self-contained under guidance by using scientific methods, within 6 months at the end of his studies.

This thesis is supervised and conducted like a project in practice considering methods of project management. Documentation and presentation (colloquium, German or English) should show that the student is able to illustrate relations and results in a coherent and precise way.

Learning objectives / skills English

The master-thesis represents an examination. Besides the professional engrossing by using an example the acquisition of soft skills are also gained:

- self-learning ability
- capacity of teamwork (working together with the supervisor)
- application of methods of project management
- communications skills: technical documentation and presentation, in case of an English presentation also practice of language skills

Literatur

Spezifisch für das gewählte Thema

Kursname laut Prüfungsordnung**Masterprojekt (AI, ISE)****Course title English**

Master Project (AI, ISE)

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
15	WS/SS	Deutsch	1
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
		8	4

Prüfungsleistung

Die erfolgreiche Teilnahme wird von der bzw. dem Lehrenden bestätigt, wenn eigenverantwortliche Mitarbeit an einem sich kontinuierlich entwickelnden Projekt innerhalb eines Semesters nachgewiesen wird. Prüfungselemente sind neben der zu bewertenden individuellen Leistung inklusive Programmierarbeit, Vortrag, schriftliche Dokumentation, auch die Rolle bei der Projektgruppenleistung. Besonderer Wert wird auch auf Selbständigkeit gelegt sowie die Umsetzung von wissenschaftlich aktuellen Konzepten und Verfahren.

Beschreibung / Inhalt Deutsch

Das Master-Projekt ist eine Einheit bestehend aus einem praktischen Teil und einem theoretischen Teil. Im praktischen Teil wird ein Software-System bzw. ein Hardware-Software-System realisiert, der begleitende theoretische Teil hat die Form einer Spezialvorlesung oder eines Seminars oder eines Kolloquiums. Eine Projektgruppe besteht im Allgemeinen aus maximal 12 Studierenden und bearbeitet über die Dauer der Vorlesungszeit eines Semesters eine abgegrenzte Aufgabenstellung, betreut und begleitet von Lehrenden der Informatik. Themen werden aus den vier möglichen Anwendungsbereichen im Master-Studiengang angeboten und sind jeweils an das Forschungsgebiet des gewählten Lehrstuhls angelehnt. Es werden klassische und neue Forschungsergebnisse aus dem jeweiligen Forschungsgebiet unter wissenschaftlicher Anleitung anhand eines konkreten Szenarios realisiert/angewendet. Beispielhafte Themen kommen aus den Bereichen: "Verteilte, Verlässliche Systeme", "Intelligente Technische Systeme und Wissenschaftliches Rechnen", "Interaktive und Kooperative Systeme", "Information Engineering".

Lernergebnisse / Kompetenzen Deutsch

Basierend auf den Qualifikationen, welche durch das Bachelor-Studium sowie durch das bis dahin fortgeschrittene Master-Studium erworben wurden, trägt das MA-Projekt zur Befähigung zu eigenständigem wissenschaftlichen Arbeiten bei (zusammen mit dem MA-Seminar und der MA-Arbeit). Unter Anleitung durch wissenschaftliches Personal lernen die Studierenden zunächst, sich neue wissenschaftliche Ergebnisse zu einem bestimmten Forschungsgebiet der Informatik anzueignen. Danach lernen sie spezifisch im MA-Projekt, wie diese Ergebnisse, die oftmals als Konzepte, Spezifikationen, Prä-Algorithmen vorliegen, anwendungsbezogen in konkrete Systeme oder Sub-Systeme umgesetzt werden können. Die Gruppe von Studierenden wird dabei motiviert zu größtmöglicher Selbständigkeit sowohl bei der Analyse des Problems, der Aufteilung in Teilaufgaben, sowie auch bei der Einarbeitung der Studierenden in die jeweiligen Teilaufgaben, und der abschließenden Fusion der Ergebnisse. Großer Wert wird ebenfalls gelegt auf die zeitbeschränkte, verständliche Präsentation von Zwischen- und Endergebnissen. Durch Gruppenarbeit machen die Studierenden die Erfahrung, wie Literaturstudium und reale Umsetzung durch Koordination und Zusammenarbeit mit den Kollegen der Gruppe erfolgt. Damit wird die Befähigung zum späteren Arbeiten in einer Gruppe von angehenden Wissenschaftlern erworben.

Description / Content English

The Master project is a unit consisting of a practical and a theoretical part. In the practical part a software-system respectively a hardware-software-system is realized, the attendant theoretical part is a special kind of lecture or seminar or colloquium. A project group consists of maximal 12 students and deals with a well-defined

task, guided by a supervisor of computer science over the whole lecture period in one semester. The topics are offered from the four possible application areas of the master studies and abut to the field of research of the chosen professorship. Classical and new research results from the particular research field will be realized / applied under scientific guidance. Exemplary topic come from the fields: "distributed, reliable systems", "intelligent technical systems and scientific calculation", "interactive and cooperative systems", "information engineering".

Learning objectives / skills English

Based on the qualification, which were gained in the bachelor and the so far finished master studies, the master project accounts the self-contained scientific work. Under guidance of scientific staff the students learn how to acquire new scientific results from specific research fields of computer science. After that, they learn how to implement those results, which are often available as concepts, specifications, pre-algorithms, applied in concrete systems and sub-systems.

The group of students is motivated by the greatest possible self-confidence both the analysis of problems, distribution of subtasks and the incorporation with students of the particular subtasks and the final fusion of the results.

A great importance lies on the time-limited, comprehensible presentation of interim results and the final results. By teamwork, the students learn how literature review and real implementation through coordination and cooperation with colleges of the group are. Thereby the ability for later work in groups of soon-to-be scientists is gained.

Literatur

Wird individuell gewählt

Kursname laut Prüfungsordnung**Masterseminar Informatik****Course title English**

Master Seminar Informatics

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
4	WS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
			2

Prüfungsleistung

Seminarvortrag, Seminararbeit, Diskussion

Beschreibung / Inhalt Deutsch

Die Studierenden arbeiten sich in ein begrenztes Thema eines Forschungsgebietes ein, bereiten einen Vortrag dazu vor, führen diesen durch und beantworten dabei zugehörige Fragen. Hinzu kommt weiterhin eine schriftliche Ausarbeitung, die innerhalb einer vorgegebenen zeitlichen Frist zu erstellen ist. Das Seminarthema soll aus dem gewählten Anwendungsbereich stammen.

Lernergebnisse / Kompetenzen Deutsch

Durch die erfolgreiche Teilnahme an einem Seminar zeigen die Studierenden, dass sie ein begrenztes Thema eines Forschungsgebietes verstehen, aufarbeiten, einen Vortrag dazu vorbereiten, durchführen und Fragen beantworten, sowie eine Ausarbeitung dazu erstellen können, und zwar innerhalb einer vorgegebenen zeitlichen Frist. Im Gegensatz zum Seminar im Bachelor-Studiengang, werden im Seminar des Master-Studiengangs üblicherweise anspruchsvollere Themen höherer Aktualität behandelt, und eine höhere Selbständigkeit in der Bearbeitung durch die Studierenden erwartet. Damit trägt das Master-Seminar, zusammen mit dem Master-Projekt und der Master-Arbeit zur Befähigung zur selbständigem wissenschaftlichen Arbeiten bei.

Description / Content English**Learning objectives / skills English****Literatur**

Wird individuell mitgeteilt

Kursname laut Prüfungsordnung**Modellierung nebenläufiger Systeme****Course title English**

Modelling of Concurrent Systems

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		

Prüfungsleistung

Mündliche Prüfung (30 Minuten)

Beschreibung / Inhalt Deutsch

Nebenläufige Systeme – von denen verteilte Systeme ein Spezialfall sind – können von einem Benutzer oft schwer überschaut werden. Bereits kleine Beschreibungen oder Programme können unvorhersehbares Verhalten hervorrufen. Außerdem stößt man dabei auf Probleme (Deadlocks, wechselseitiger Ausschluss), die bei sequentiellen Systemen nicht auftreten können. Daher werden in dieser Vorlesung entsprechende Modellierungstechniken und Analysemethoden vermittelt, die zum besseren Verständnis solcher Systeme führen.

Inhalte im Einzelnen:

- Transitionssysteme
- Verhaltensäquivalenzen (Trace-äquivalenz, Bisimulation)
- Prozesskalküle (CCS, pi-Kalkül)
- Petri-Netze (mit Partialordnungstechniken)
- Graphtransformationssysteme

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden sollen Kenntnisse über verschiedene Modellierungstechniken für nebenläufige Systeme erwerben. Insbesondere sollen sie Prozesskalküle, Petri-Netze und Graphtransformationssysteme und ihre Einsatzgebiete kennenlernen. Neben den Modellen selbst sollen die Studierenden auch Analyse- und Spezifikationstechniken, wie beispielsweise Verhaltensäquivalenzen und Partialordnungstechniken anwenden können und ihre Eignung abschätzen lernen. Insbesondere soll in dieser Veranstaltung der Umgang mit formalen Beschreibungsmethoden geübt werden.

Description / Content English

Concurrent system - of which distributed systems are a special case - are often hard to understand from a user's perspective. Already small specifications or programs can lead to unforeseeable behaviours.

Furthermore problems (such as deadlocks, mutual exclusion) arise which do not occur in sequential systems. Therefore this course will teach modelling and analysis techniques which lead to a better understanding of such systems.

Table of contents:

- transition systems
- behavioural equivalences (trace equivalence, bisimulation)
- process calculi (CCS, pi-calculus)
- Petri nets (with partial order techniques)
- graph transformation systems

Learning objectives / skills English

The students will acquire knowledge about different modelling techniques for concurrent systems. Especially they will learn about process calculi, Petri nets and graph transformation systems and the areas where such models are applied. Apart from the models and modelling languages the students will apply analysis and specification techniques, such as behavioural equivalences and partial order techniques and learn to judge their adequacy and appropriateness. One of the main aims is to practice the use of formal modelling languages.

Literatur

- R. Milner: Communication and Concurrency. Prentice Hall, 1989.
- W.J. Fokkink (2000): Introduction to Process Algebra, Springer, 2000.
- Aceto, Ingolfsdottir, Larsen, Srba: Reactive Systems: Modelling, Specification and Verification, Cambridge University Press, 2007
- Reisig: Petrinetze: Modellierungstechnik, Analysemethoden, Fallstudien (Vieweg+Teubner, 2010).

Kursname laut Prüfungsordnung
Modellierung, Analyse, Verifikation
Course title English

Modelling, Analysis, Verification

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		

Prüfungsleistung

Mündliche Prüfung (30 Minuten)

Beschreibung / Inhalt Deutsch

Neben dem Einsatz auf dem Gebiet der Hardware-Verifikation halten Analyse- und Verifikationstechniken immer stärker Einzug in das Gebiet der Software-Verifikation.

Vor der Anwendung dieser Techniken ist es notwendig, das zu verifizierende System zu modellieren bzw. eine formale Semantik der zu behandelnden Programmiersprache anzugeben. Diese Vorlesung soll eine Einführung in die Gebiete Modellierung, Analyse und Verifikation geben.

Inhalte im Einzelnen:

1. Datenflussanalyse
- Fixpunkttheorie
- Monotone Frameworks
- Worklist-Algorithmus
- Anwendungsbeispiele: Compiler-Optimierung, Java Bytecode Verifier
2. Grundlagen der abstrakten Interpretation
- Galois-Verbindungen
- Sichere Approximation von Funktionen
- Abstraktionsverfeinerung

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden sollen Kenntnisse auf dem Gebiet der Verifikation und Analyse von Programmen erlangen. Dabei sollen sie Datenflussanalyse, deren Grundlagen (Fixpunkttheorie, monotone Frameworks) und ihre Anwendungen kennenlernen. Außerdem sollen sie Methoden aus dem Bereich der abstrakten Interpretation anwenden und deren Eignung für die Programmverifikation abschätzen können.

Description / Content English

Apart from being used in the area of hardware verification, analysis and verification techniques are being increasingly applied in software verification. Before using such techniques, the system to be verified has to be modelled or the considered programming language has to be equipped with a formal semantics. This course will give an introduction into modelling, analysis and verification.

Topics in detail:

1. Data flow analysis
- Fixed-point theory
- Monotonous frameworks
- Worklist algorithm
- Application examples: compiler optimization, java bytecode verifier
2. Foundations of abstract interpretation
- Galois connections
- Safe approximation of functions

- Abstraction refinement

Learning objectives / skills English

The students should gain knowledge about verification and analysis of programs. In particular they should get acquainted with data flow analysis, its foundations (fixed-point theory, monotonous frameworks) and its applications. Furthermore they should be able to apply methods from abstract interpretation and to evaluate their adequacy in program verification.

Literatur

- Flemming Nielson, Hanne Riis Nielson, Chris Hankin: Principles of Program Analysis. Springer-Verlag, 1999.
- Edmund M. Clarke, Orna Grumberg, Doron A. Peled: Model Checking. MIT Press, 2000.

Kursname laut Prüfungsordnung**Muster- und komponentenbasierte Software-Entwicklung****Course title English**

Pattern and Component based Software Development

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Die Veranstaltung behandelt musterbasierte und komponentenbasierte Softwareentwicklung. Diese beiden Ansätze legen

Wert auf die Wiederverwendung bereits vorhandenen Wissens bzw. bereits vorhandener Software.

Inhalt im Einzelnen:

- Definition von Mustern
- Verwendung von Mustern im Softwareentwicklungsprozess
- Muster für die verschiedenen Phasen (Analyse: Problem Frames, Analysemuster; Entwurf: Architekturmuster, Entwurfsmuster; Implementierung: Idiome; Test: Testmuster)
- Komponentenbegriff
- Verschiedene Komponentenmodelle (z.B. Enterprise Java Beans, Corba Component Model, OSGi)
- Spezifikation von Komponentenschnittstellen
- Nachweis der Interoperabilität von Komponenten
- Komponentenbasierter Entwicklungsprozess
- Kombination von Mustern und Komponenten in einem integrierten Entwicklungsprozess

Lernergebnisse / Kompetenzen Deutsch

- Grundideen von Mustern und Softwarekomponenten erläutern können
- Vorteile muster- und komponentenbasierter Softwareentwicklungsansätze gegenüber traditionellen Ansätzen beschreiben können
- Den verschiedenen Phasen des Softwarelebenszyklus Muster zuordnen können
- Für die besprochenen Musterarten Beispiele aufzählen und erläutern können
- Die verschiedenen Arten von Mustern anwenden können.
- Grundidee der komponentenbasierten Softwareentwicklung erläutern können
- Existierende Komponentenmodelle erläutern und beurteilen können.
- Komponenten spezifizieren können.
- Komponentenbasierte Software entwerfen können.
- Zusammenhänge zwischen Mustern und Komponenten erläutern können

Description / Content English

The course treats pattern- and component-based software development. Both approaches stress the reuse of existing knowledge or existing software.

Content in detail:

- Definition of Patterns
- Usage of patterns in the software development process

- Patterns for different phases (analysis phase: problem frames, analysis patterns; design phase: architectural patterns, design patterns; implementation phase: idioms; test phase: test patterns)
- Notion of component
- Different component models (e.g., Enterprise Java Beans, Corba Component Model, OSGi)
- Specification of component interfaces
- Demonstration of component interoperability
- Component-based development process
- Combination of patterns and components in an integrated development process

Learning objectives / skills English

- Be able to explain the basic ideas of patterns and software components
- Be able to describe advantages of pattern and component based software development approaches as compared to traditional approaches
- Be able to assign patterns to the different phases of the software development life-cycle
- Be able to give examples for the discussed pattern types and explain them
- Be able to apply the different kinds of patterns
- Be able to explain the basic ideas of component based software development
- Be able to explain and assess existing component models
- Be able to specify a component
- Be able to design component-based software
- Be able to explain the interrelationship between components and patterns

Literatur

- M. Jackson: Problem Frames. Analyzing and structuring software development problems, Addison-Wesley, 2001.
- Martin Fowler: Analysis Patterns: Reusable Object Models, Addison Wesley, 1997.
- F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad und M. Stal: Pattern-Oriented Software Architecture: A System of Patterns, John Wiley & Sons, 1996
- E. Gamma, R. Helm, R. Johnson und J. Vlissides: Design Patterns - Elements of Reusable Object-Oriented Software, Addison Wesley, 1995.
- Binder, R. V. (1999). Testing object-oriented systems: models, patterns, and tools. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- C. Szyperski: Component Software - Beyond object oriented programming, Addison Wesley, 2. Aufl. 2002.
- J. Cheesman und J. Daniels: UML Components - A Simple Process for Specifying Component-Based Software, Addison-Wesley, 2001.
- George T. Heineman and William T. Councill: Component-Based Software Engineering. Addison Wesley, 2001

Kursname laut Prüfungsordnung**Natürlichsprachliche Mensch-Computer-Interaktion****Course title English**

Natural-Language-based Human-Computer Interaction

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	WS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		

Prüfungsleistung

Schriftliche Klausurarbeit oder mündliche Prüfung

Beschreibung / Inhalt Deutsch

Inhalte im Einzelnen:

- Ebenen der Sprachanalyse
- Sprachliche Interaktion
- Probleme der Repräsentation im Computer
- Spracherkennung, Sprachgenerierung
- Systeme zur sprachlichen Interaktion von Mensch und Maschine
- Anwendungen von natürlichsprachlicher Mensch-Computer Interaktion

Zudem wird angestrebt, die Veranstaltung regelmäßig auf Englisch durchzuführen.

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden lernen, welche Probleme bei der natürlichsprachlichen Mensch-Computer Interaktion auftreten und wie diese mit Verfahren der Sprachtechnologie gelöst werden können.

Description / Content English

Contents in detail:

- Levels of language analysis
- Linguistic interaction
- Problems of representation in the computer
- speech recognition, speech generation
- Systems for the linguistic interaction of man and machine
- Applications of natural language human-computer interaction

It is also intended to hold the event regularly in English.

Learning objectives / skills English

Students learn which problems occur in natural-language human-computer interaction and how they can be solved using methods of language technology.

Literatur

- Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, Introduction to Information Retrieval, Cambridge University Press. 2008.

- Daniel Jurafsky and James H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing*, 2nd edition. Prentice-Hall, 2009.
- Computerlinguistik und Sprachtechnologie. Eine Einführung. Kai-Uwe Carstensen, Christian Ebert, Cornelia Endriss, Susanne Jekat, Ralf Klabunde. Heidelberg: Spektrum-Verlag, März 2004
- Ruslan Mitkov, *The Oxford handbook of computational linguistics*, Oxford University Press. 2003.

Kursname laut Prüfungsordnung**Neuroinformatik und Organic Computing****Course title English**

Neurocomputing and Organic Computing

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Die Veranstaltung behandelt wichtige Typen von vorwärts gerichteten Neuronalen Netzen, wie Mehrschicht-Perzeptron, Radiale Basisfunktionen Netze, Tiefe Faltungsnetze und Support Vektor Maschinen. übergreifend werden das grundlegende Problem des algorithmischen Lernens vorgestellt, nämlich das Bias-Varianz-Dilemma, sowie Lösungen diskutiert. Bezug nehmend auf Organic Computing werden self-X Fähigkeiten untersucht. Ein besonderer Wert wird darauf gelegt, einen Zusammenhang zu grundlegenden Techniken aus anderen Disziplinen herzustellen, wie Gradientenabstieg, lineare und quadratische Optimierung, statistische Entscheidungstheorie. Typische Anwendungen werden exemplarisch behandelt, wie Signalfilterung, Mustererkennung, Roboterkontrolle. Inhalte im Einzelnen:

- Einführung
- McCulloch-Pitts Zelle, Perzeptron, Adaline
- Statistische Entscheidungstheorie
- Mehrschichtnetze, Tiefe Faltungsnetze
- Netze radialer Basisfunktionen
- Bias-Varianz-Dilemma
- Netze von Support Vektoren
- Organic Computing

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden sollen für ausgewählte Typen von Neuronalen Netzen deren Struktur und Lernmethodik verstehen, die grundlegende mathematische Fundierung nachvollziehen können, die prinzipielle Wirkung und die mögliche Anwendbarkeit kennen. Sie sollen für ausgewählte Problemstellungen potentiell sinnvolle Netztypen und Lernverfahren vorschlagen können.

Description / Content English

The course treats important types of feed-forward neural networks, such as Multi-Layer Perceptron, Radial Basis Function networks, Deep Convolutional Networks, Support Vector Machines. In a comprehensive manner, the basic problem of algorithmic learning is treated, including Bias-Variance Dilemma, and solutions are presented. Related to Organic Computing, self-X competences are discussed. A special effort is put on relationships to basic techniques from other fields, e.g. gradient descent, linear and quadratic optimization, statistical decision theory. Typical applications include signal filtering, pattern recognition, robot control. Contents at a glance:

- Introduction
- McCulloch-Pitts Zelle, Perzeptron, Adaline
- Statistical decision theory
- Multi-Layer Perceptron, Deep Convolutional Networks

- Radial Basis Function Networks
- Bias-Variance-Dilemma
- Support Vector Machines
- Organic Computing

Learning objectives / skills English

The students should understand for certain types of neural networks their structure and learning method, as well as the mathematical foundation, and they should know possible applications. They have the competence to propose for certain types of problems, the potentially useful types of networks and learning procedures.

Literatur

- C. Bishop: Neural Networks for Pattern Recognition; Oxford Press, 1995.
- C. Bishop: Pattern Recognition and Machine Learning; Springer, 2006.
- I. Goodfellow, et al.: Deep Learning; MIT Press, 2016
- T. Hastie, et al.: The Elements of Statistical Learning, Springer, 2003.
- M. Mohri, et al.: Foundations of Machine Learning; MIT Press, 2012.
- R. Rojas: Neuronale Netze; Springer-Verlag, 1996.
- Z. Zell: Simulation neuronaler Netze; Addison-Wesley, 1994.
- Aktuelle eigene Artikel sowie Bachelor-/Master-/Doktorarbeiten.

Kursname laut Prüfungsordnung**Nicht-technischer Katalog MA****Course title English****Non-technical Catalog MA**

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
8	WS/SS	Deutsch/Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
			6

Prüfungsleistung

Die Art und Dauer der Prüfung wird vom Lehrenden vor Beginn des Semesters bestimmt.

Beschreibung / Inhalt Deutsch

Mit diesem Modul soll den Studierenden die Möglichkeit gegeben werden, im Rahmen des Studiums neben den rein technischen Veranstaltungen auch so genannte „nicht-technische Fächer“ nachweislich zu belegen. Die Veranstaltungen können aus dem gesamten Angebot der Universität Duisburg-Essen gewählt werden, wobei das „Institut für Optionale Studien“ (IOS) einen Katalog mit Veranstaltungen aus dem so genannten Ergänzungsbereich vorhält.

Lernergebnisse / Kompetenzen Deutsch

Ziel des Moduls ist Vertiefung der Allgemeinbildung der Studierenden und ggf. die Verstärkung der sprachlichen Kompetenz sowie eine Stärkung der Berufsbefähigung durch das Erlernen von Teamfähigkeit und Präsentationstechniken.

Description / Content English

This module offers the students the opportunity to, besides the pure technical courses they take, attend some so called „non-technical subjects“ and latter provide an attest for them. These courses can be chosen from the overall offers of the Duisburg-Essen university, whereby the „Institut für Optionale Studien“ (IOS) proposes a catalog containing courses which fall under the named supplementary area.

Learning objectives / skills English

The module aims at deepening the general knowledge of the students and resp. at improving their language skills as well as strengthening their professional qualifications through the learning of teamwork and expose techniques.

Literatur

Spezifisch für das gewählte Thema

Kursname laut Prüfungsordnung

Peer-to-Peer Systeme

Course title English

Peer-to-Peer Systems

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

Die Vorlesung vermittelt Kenntnisse über Historie, Struktur und Protokolle Peer-to-Peer-basierter Systeme. Hierfür werden wichtige P2P Systeme im Detail beschrieben, unter anderem:

- Napster
- Gnutella
- Cord, CAN, Pastry

Außerdem werden Anwendungsgebiete von P2P Systemen besprochen und der Unterschied zu klassischen Client/Server Systemen thematisiert. Besondere Fokus wird auf die folgenden Anwendungsgebiete gelegt:

- Filesharing
- Datenbanken
- Verzeichnisdienste
- Geographische Daten

Lernergebnisse / Kompetenzen Deutsch

Die Studenten verstehen wie P2P Systeme funktionieren, insbesondere wie sie strukturiert sind, wie ein Computer dem Netz beitreten oder es verlassen kann, und wie Daten in einem P2P Netzwerk gespeichert und wieder gefunden werden können. Des Weiteren wissen sie, welche Vor- und Nachteile P2P Systeme im Vergleich zu Client/Server Systemen haben.

Description / Content English

The lecture covers the history, structure and protocols of peer-to-peer based systems. Important P2P systems will be discussed in detail, for example:

- Napster
- Gnutella
- Cord, CAN, Pastry

Additionally, the lecture will discuss application scenarios of peer-to-peer based systems and the differences to classical client/server approaches. Es special focus will be on the following application domains:

- Filesharing
- Databases
- Naming services
- Geographical data

Learning objectives / skills English

The students understand how P2P systems work, i.e. how they are structured, how nodes can join or leave the network and how data can be stored and retrieved in P2P networks. Furthermore, students know in which

scenarios P2P systems can improve upon client/server systems and where to expect limitations of the P2P approach.

Literatur

Peer-to-Peer: Harnessing the Power of Disruptive Technologies (Andy Oram)

P2P Netzwerke: Algorithmen Und Methoden (Mahlmann & Schindlhauer)

Unter Vorbehalt: Peer-to-Peer Systems and Applications (Steinmetz)

Kursname laut Prüfungsordnung

Recommender Systeme

Course title English

Recommender Systems

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

In der Veranstaltung werden Konzepte, Methoden und Techniken für Systeme behandelt, die sich an unterschiedliche Kontexte adaptieren können. Dabei ist der Kontextbegriff weit zu verstehen und umfasst Aspekte wie Orts- und Zeitkontext, aktuelle Aufgaben des Anwenders oder die Eigenschaften des verwendeten Endgeräts. Grundlegende Formen der Adaption werden behandelt wie Filterung/Selektion von Inhalten und Diensten, adaptive Generierung von User Interfaces oder adaptive Präsentation. Ein wesentlicher Aspekt ist die Repräsentation von Adoptionswissen sowie geeignete Inferenz- und Lernverfahren. Unterschiedliche Techniken der Erfassung und Interpretation von Kontext werden dargestellt. Schließlich werden Anwendungen wie Ubiquitous Computing oder Recommender-Systeme vorgestellt, anhand von Darstellung, und auch durch Illustration mit Filmen.

Inhalte im Einzelnen:

- Begriffliche Grundlagen und Modelle kontext-adaptiver Systeme
- Kontextfaktoren und Modellierung von Kontext
- Ontologie-basierte Kontextrepräsentation
- Technische Grundlagen zur Erfassung des (externen) Kontextes
- Context-Sensing und Kontextinterpretation
- Regelbasierte und probabilistische Ansätze zur Repräsentation von Adoptionswissen
- Klassifikation unterschiedlicher Adaptionseffekte und zugehöriger Techniken
- Maschinelle Lernverfahren für adaptive Systeme
- Anwendungen im Ubiquitous Computing
- Spezifische Anwendungen wie Recommender-Systeme und persönliche Assistenten

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden kennen die wesentlichen Konzepte und Methoden für kontext-adaptive Systeme und können die unterschiedlichen Kontextaspekte analysieren und formalisieren. Sie sind mit unterschiedlichen Adoptionsmechanismen vertraut und können die zugrunde liegenden Inferenztechniken und Lernverfahren anwenden. Sie kennen die wesentlichen Eigenschaften unterschiedlicher Anwendungen wie Location-based Services oder nutzeradaptive Web-Anwendungen und können diese konzipieren, realisieren und beurteilen.

Description / Content English

The lecture covers concepts, methods and techniques of systems that can adapt to different contexts. In the process the term, context, is to be broadly understandable and deals with aspects such as location and time context, recent user's tasks or the characteristics of the terminal device.

Basic forms of adaption are covered such as filter/selection of contents and services, adaptive generation of user interfaces or adaptive presentation.

A central aspect is the representation of adaptive knowledge as well as inference- and learning methods. Different techniques of capturing and interpretation of context will be shown.

Finally programs like Ubiquitous Computing or Recommender Systems will be demonstrated by presentations and illustrative videos.

Full details of contents:

- Basic terms and models of context-adaptive systems
- Context factors and modelling of context
- Ontology-based context representation
- Technical basics for capturing (external) context
- Context sensing and context interpretation
- Rule-based and probability approach for representation of adaptive knowledge
- Classification of different adaptive effects and regarding techniques
- Automatic learning procedures for adaptive systems
- Applications in Ubiquitous Computing
- Specific applications such as Recommender Systems and personal assistants

Learning objectives / skills English

Students will know the essential concepts and methods for context-adaptive systems and will be able to formalize and analyze the different aspects of context. They will be familiar with the various forms of adaptive mechanisms and can apply the underlying inference techniques and learning procedures. Additionally, the students will get to know essential characteristics of diverse applications such as location-based services or user adaptive web-applications and will be able to design, implement and evaluate them.

Literatur

aktuelle Internet-Quellen

Kursname laut Prüfungsordnung

Scientific Visualization

Course title English

Scientific Visualization

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Englisch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
2	2		

Prüfungsleistung

mündliche Prüfung, 45 Minuten

Beschreibung / Inhalt Deutsch

Die Vorlesung führt in die Grundlagen der wissenschaftlichen Datenvisualisierung ein. Sie stellt Begriffe und Algorithmen zur effizienten Behandlung diskreter Datenstrukturen vor. Zur Motivation der Visualisierung solcher Daten dienen Beispiele aus der medizinischen Bildgebung und der numerischen Simulation. So wird der Aufbau eines CT-Scanners und die damit erzeugten Datensätze näher erläutert und grundlegende Verfahren aus der numerischen Simulation zur Lösung einfacher gewöhnlicher und partieller Differentialgleichungen beispielhaft dargestellt. Zur Visualisierung skalarer Daten werden direkte Ansätze, wie z.B. Schichtverfahren und Strahlverfolgung, aber auch indirekte Methoden, wie z.B. Marching Squares bzw. Marching Cubes, besprochen. Des Weiteren werden mehrere Verfahren zur Strömungsvisualisierung erläutert, z.B. glyphenbasierte Darstellung, linienbasierte Integrationsmethoden, dichte Strömungsvisualisierung und topologische Methoden.

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden lernen in dieser Veranstaltung die grundlegenden Algorithmen moderner Visualisierungssysteme kennen. Sie werden anhand einiger Beispiele aus der medizinischen Bildgebung und dem wissenschaftlichen Rechnen die Herkunft und die Eigenschaften üblicher Datensätze erklären können. Grundlegende Konzepte wie Interpolation, Triangulation und Filtermethoden werden bekannt sein. Sie werden verschiedenen Datentypen passende Visualisierungsansätze zuordnen können. Sie beherrschen die interaktive Darstellung und Analyse von großen skalaren Bild- und Volumendaten, Vektorfeldern, Terraindaten und Daten aus weiteren Informationsquellen.

Description / Content English

This lecture is an introductory course in scientific data visualization. Particular emphasis is put on the representation and processing of discrete data structures. The course starts with the source of scientific data, such as computed tomography and numerical approximations to partial differential equations. These methods serve as examples for typical data sources for scientific visualization. The main part of the lecture is concerned with methods for the visualization of scalar and vector valued volumetric datasets. For scalar datasets the methods covered include slicing, ray casting, marching squares and marching cubes. For flow fields concepts such as glyphing, line based-, dense-, and topological-visualization methods are discussed.

Learning objectives / skills English

Students who attend this lecture will be able to understand and implement state of the art scientific visualization algorithms. They will gain detailed knowledge about the source of typical datasets from both measurements as well as simulations. They will understand basic concepts such as interpolation, triangulation, and filtering. They will be able to choose the most appropriate visualization technique for a number of scenarios including analysis of large scalar and vector valued volumes, terrains, and information that lacks spatial components.

Literatur

- Aktuelle Internetliteratur
- Nielson, Hagen, Müller: Scientific Visualization , IEEE Computer Society Press
- Earnshaw, Wiseman: An Introductory Guide to Scientific Visualization, Springer Verlag
- Schumann, Müller: Visualisierung - Grundlagen und allgemeine Methoden, Springer Verlag

Kursname laut Prüfungsordnung**Wissensbasierte Systeme****Course title English**

Knowledge-Based Systems

Kreditpunkte	Turnus	Sprache	Pflicht/Wahl
6	SS	Deutsch	0
SWS Vorlesung	SWS Übung	SWS Praktikum/Projekt	SWS Seminar
3	1		
Prüfungsleistung			

Beschreibung / Inhalt Deutsch

In dieser Vorlesung diskutieren wir, wie Information in eine für den Computer nutzbare Repräsentation transformiert werden kann und wie diese Information in Wissen umgewandelt wird.

Wir starten mit der grundlegenden Frage, was Wissen überhaupt ist, und beschäftigen uns dann mit verschiedenen Arten, Wissen zu kodieren. Wir diskutieren, wie der Grad der Ähnlichkeit zwischen einzelnen Informationen bestimmt werden kann, um Fragen wie „Ist schwarz ähnlich zu weiß?“ beantworten zu können. Dazu nutzen wir eine Vielzahl von menschlich-kuratierten Ressourcen.

Im Teilgebiet des maschinellen Lernens, beschäftigen wir uns damit, wie der Computer lernen kann, Objekte in der realen Welt zu klassifizieren. Dies geschieht entweder mit Hilfe von vom Menschen aufbereiteten Trainingsdaten oder gänzlich unüberwacht. Wir diskutieren verschiedene, allgemein gebräuchliche Algorithmen aus dem Bereich des Maschinellen Lernens und beleuchten Eigenschaften dieser Algorithmen, die bei der Wahl des richtigen Algorithmus für ein konkretes Problem hilfreich sind.

Lernergebnisse / Kompetenzen Deutsch

Die Studierenden kennen verschiedene Arten von Wissen und Ebenen der Wissensrepräsentation. Sie kennen Algorithmen zur Ableitung und Verarbeitung von Wissensrepräsentationen für eine Reihe von praktischen Anwendungsproblemen und können sie erklären. Sie sind in der Lage, Vor- und Nachteile von verschiedenen Repräsentationen und Algorithmen für verschiedene Anwendungen zu diskutieren.

Description / Content English

In this lecture, we will discuss methods how to transform information into a representation suited for computers and how to turn this information into knowledge.

We will start with the foundational question of what knowledge is and advance towards various ways of encoding it. We discuss how the degree of similarity between pieces of information can be determined to answer questions such as „is the color black similar to white?“ by using a variety of human-curated resources. In the subfield of machine learning we explore how the computer learns to classify objects in the real world either from human-prepared data or on its own. We will discuss several machine learning algorithms which are commonly used today and highlight properties that one should consider when choosing between algorithms.

Learning objectives / skills English

Students know different kinds of knowledge and levels of knowledge representations. They know and can explain algorithms for deriving and processing knowledge representations for a number of real-world problems. The students can discuss advantages and disadvantages of different representations and algorithms for different applications.

Literatur

- George F. Luger: Künstliche Intelligenz . Strategien zur Lösung komplexer Probleme. Pearson Studium 2001

- Ronald Brachman, Hector Levesque: Knowledge Representation and Reasoning. Morgan Kaufmann Publishers 2004
- Stuart J. Russell, Peter Norvig: Künstliche Intelligenz. Ein moderner Ansatz. Pearson Studium 2004