

Verteilte Systeme und Erlang Programmierung <i>Distributed Systems and Erlang Programming</i>		Modulnummer:			
Bachelor Pflicht <input type="checkbox"/> Winf-Schwerpunkt-Pflicht <input type="checkbox"/> Winf-Schwerpunkt-Wahlpflicht <input type="checkbox"/> Winf-Wahl <input type="checkbox"/>		Schwerpunkt Computational Finance <input type="checkbox"/> E-Business <input type="checkbox"/> IT-Management <input type="checkbox"/> Logistik <input type="checkbox"/>			
Anzahl der SWS	The course will have a two-hour lecture Wednesdays every week. Most weeks will have an additional two-hour exercise session that is intended to flatten the programming learning-curve and introduce some useful tools. Details will be presented at the beginn	Kreditpunkte: 6	Turnus		
Formale Voraussetzungen: Keine					
Inhaltliche Voraussetzungen: -					
Vorgesehenes Semester: ab 1. Semester					
Sprache: Englisch					
Kommentar: This course will teach the foundation, implementation, applications of distributed systems. It will cover basic methods to handling concurrency, such as transactions, replication, and communication. Then, look at design of "real-life" scalable distributed computing systems, such as Map-Reduce (Google, Hadoop), Non-SQL storage (Amazon's Dynamo). In the course students will get hands-on experience with implementing the shown methods and systems. For this course we will introduce the programming language Erlang; a functional language that is designed to handle concurrency. Methods for making reliable implementations will be covered. The course will end with a larger assignment, where students pairwise will reliably implement a larger distributed system.					
Ziele: * Diskussion von Design-Alternative für ein verteiltes Computersystem * Analyse-Protokolle für Concurrency Control und Recovery, sowie für die Verteilung und Replikation * Implementierung von korrekter, robuster, effizienter und wartbarer Programme durch klare Aufgabentrennung (separation of concerns)					
Inhalte: Kenntnisse der * Grundlagen hinter Verteilte Systeme aus der theoretischen Entwürfe, praktische (Groß)-Systeme. * Message-Passing (Schauspieler-Modell) parallele Programmierung. * Wie parallele Programmierung für parallele Problemlösung eingesetzt. * Programmierung Strukturierungsprinzipien und Entwurfsmuster für den Umgang mit Softwarefehlern. Fähigkeiten in * Umsetzung Systeme, die Mechanismen für die Modularität, die Unteilbarkeit und Fehlertoleranz aufweisen. * Verwenden Sie verteilten und parallelen Algorithmus Skelette wie Karte-zu reduzieren, um Daten schreiben zu erkunden Programme. * Umsetzung einfach gleichzeitige / verteilte Server mit Message Passing, mit angemessenen Einsatz von synchronen und asynchronen Message-Passing. * Verwenden Programmierung Strukturierungsprinzipien und Entwurfsmuster für die Herstellung von zuverlässigen verteilten Systemen in Gegenwart von Softwarefehlern. * Lösen Sie anspruchsvolle, realistische Probleme mit den oben genannten Techniken.					
Unterlagen (Skripte, Literatur, Programme usw.): The course will be based on material that is freely available to the students * Distributed Systems: - Selected book chapters - Scientific papers * Erlang: - Learn you some Erlang for great good! http://learnyousomeerlang.com/ (If you prefer books, this can be bought.) - Armstrong; Making reliable distributed systems in the presence of software errors http://www.erlang.org/download/armstrong_thesis_2003.pdf					
Form der Prüfung: Larger written exercise followed by a shorter oral defence or longer oral exam.					
Arbeitsaufwand	<table style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: right;">0 h</td> </tr> <tr> <td style="text-align: right;">Summe 0 h</td> </tr> </table>			0 h	Summe 0 h
0 h					
Summe 0 h					
Lehrende: Michael Kirkedal Thomsen		Verantwortlich: Michael Kirkedal Thomsen			