

updated  
16.04.2010

Projektgruppe



Fabian Christ, Benjamin Nagel, Henning Wachsmuth

## Seminarthemen

8. April 2010

## Anforderungen im Seminar

Die Anforderungen an den **Seminarvortrag** und die **Seminararbeit** sind den Folien von Prof. Engels zu entnehmen.

Das Seminar findet am **1.6./2.6.2010** statt, die Ausarbeitung muss bis **18.6.2010** fertig sein.

# Überblick (I)



## Software Engineering

- Verfolgbarkeit von Anforderungen im Software-Entwicklungsprozess \*
- Klassische vs. agile Vorgehensmodelle im Software Engineering
- Vergleich von drei Open Source CMS-Systemen \*
- Komponentenbasiertes Software Engineering mit OSGi
- Automatische Modellgenerierung aus Systemspezifikationen
- Knowledge Representation/Persistence und Reasoning \*
- Reale Anwendungsgebiete des Semantic Web

*\* Auf jeden Fall zu bearbeiten!*

# Überblick (2)



## Information Driven

- Natürliche Sprachverarbeitung mit UIMA vs. GATE
- Kategorisierung von Textdokumenten mittels Klassifikation \*
- Clustering und Fingerprinting zur Erkennung von Ähnlichkeiten \*
- Semantische Suchverfahren im Detail
- Text Labeling mit Sequenzmodellen \*
- Die Entity-Recognition-Pipeline \*
- Semantic Role Labeling

*\* Auf jeden Fall zu bearbeiten!*

# Verfolgbarkeit von Anforderungen im Software-Entwicklungsprozess \*



- **Thema:** Konzepte zur Verfolgbarkeit von Anforderungen im Software-Entwicklungsprozess
- **Ziele:** Bestehende Ansätze für die Verfolgbarkeit von Anforderungen im Software-Entwicklungsprozess aufzeigen und kritisch betrachten.
- **Literatur:**
  - Chris Rupp (2007): Requirements-Engineering und Management , S. 409-442
  - Ramesh, Balasubramaniam und Matthias Jarke (2001):“Toward Reference Models for Requirements Traceability.”
  - Gotel, Orlena C. Z., und Anthony C.W. Finkelstein (1994):“An Analysis of the Requirements Traceability Problem”.

# Klassische vs. agile Vorgehensmodelle im Software Engineering



- **Thema:** Vergleich zwischen klassischen Vorgehensmodellen in der Softwareentwicklung mit neueren, agilen Vorgehensmodellen.
- **Beispiel:** Rational Unified Process (RUP) vs. Scrum
- **Ziele:** Unterschiedliche klassische und agile Modelle vorstellen, vergleichen und im Hinblick auf die Einsetzbarkeit in der PG bewerten.
- **Literatur:**
  - Kruchten, P.: The rational unified process : [an introduction], Addison Wesley, 1998
  - Alistair Cockburn: Agile software development. Addison-Wesley, 2002
  - Jutta Eckstein, Nicolai Josuttis: Agile Softwareentwicklung im Großen: Ein Eintauchen in die Untiefen erfolgreicher Projekte, dpunkt.verlag, 2004

# Vergleich von drei Open Source CMS \*



- **Thema:** Evaluation von 3 gegebenen Open Source CMS.
- **Ziele:** Erstellung eines Kriterienkatalogs für die Bewertung existierender CMS hinsichtlich der Verarbeitung von Software Engineering Artefakten. Es soll eine wissenschaftlich fundierte Empfehlung bezüglich der Auswahl eines CMS abgegeben werden können.
- **Literatur:**
  - Stahl, F., & Maass, W. (2003). Content Management Handbuch
  - Doyle, B. (2009, June). CMS Review, from <http://www.cmsreview.com>
  - BNP-Consulting (2007). CMS Benchmark, from <http://www.bnp.dk/I2I.0.html>

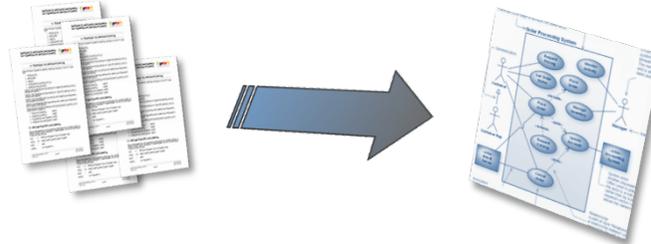
# Komponentenbasiertes Software Engineering mittels OSGi



- **Thema:** Komponentenbasierte Softwareentwicklung als Standardvorgehen im Software Engineering
- **Ziele:** Konzepte der komponentenbasierten Softwareentwicklung am Beispiel OSGi darstellen und möglichen Nutzen für die PG ableiten.
- **Literatur:**
  - (2009): OSGi Service Platform Core Specification
  - Heinemann and Council (2001): Component-Based Software Engineering: Putting the Pieces Together
  - Lau and Wang (2007): A Survey of Software Component Models

# Automatische Modellgenerierung aus Systemspezifikationen

- **Thema:** Automatische Umwandlung natürlichsprachiger Anforderungen eines Systems in UML-Modelle.



**Ziele:** Vorstellung bestehender Ansätze und verwendeter Techniken. Aufzeigen der Möglichkeiten und Grenzen automatischer Modellgenerierung.

- **Literatur:**
  - Ambriola and Gervasi (2006): On the Systematic Analysis of Natural Language Requirements with CIRCE
  - Liu et al. (2004): Natural Language Requirements Analysis and Class Model Generation Using UCDA
  - Gerber, A. & Raymond, K. (2003), MOF to EMF: there and back again.

# Knowledge Representation/Persistence und Reasoning \*



- **Thema:** Wissensrepräsentation und -speicherung als elementare Techniken für die Verarbeitung semantischer Informationen in Softwaresystemen
- **Ziele:** Techniken der Wissensrepräsentation und -speicherung verstehen und die Möglichkeiten des automatischen Folgerns darauf aufbauend vertiefen.
- **Literatur:**
  - Uren et. al. (2006): Semantic annotation for knowledge management: Requirements and a survey of the state of the art
  - Reeve and Han (2005): Survey of semantic annotation platforms
  - Bundy (1999): A Survey of Automated Deduction.
  - Yan et. al. (2008): Efficiently querying rdf data in triple stores
  - Vila (1994): A Survey on Temporal Reasoning in Artificial Intelligence

# Reale Anwendungsgebiete des Semantic Web



- **Thema:** Was ist das Semantic-Web und welche reale Anwendungen gibt es?
- **Ziele:** Erfassung der semantischen Technologien des Semantic-Web und kritische Betrachtung möglicher realer Anwendungen im Kontext der PG.
- **Literatur:**
  - Lausen et. al. (2003): Semantic Web Portals - State of the Art Survey
  - Fensel et. al. (2002): Semantic Web Application Areas

# Natürliche Sprachverarbeitung mit UIMA vs. GATE



- **Thema:** UIMA und GATE stellen Architektur und Framework für Technologien im Bereich der natürlichen Sprachverarbeitung (z.B. Information Extraction) zur Verfügung.



- **Ziele:** Vorstellen der beiden Frameworks und darstellen ihrer Unterschiede. Herausstellen, wann welches Framework nützlich ist.
- **Literatur:**
  - Ferrucci and Lally (2004): Building an example application with the Unstructured Information Management Architecture
  - Cunningham (2002): GATE, a General Architecture for Text Engineering
  - <http://gate.ac.uk/> und <http://incubator.apache.org/uima/>

# Kategorisierung von Textdokumenten mittels Klassifikation \*



- **Thema:** Maschinelle Lernverfahren eignen sich zur Klassifikation von Texten. etwa nach Topic, Genre, Intention oder Stimmung.

- **Beispiel:**

**iPhoneFan99**, 1.1.2010:  
Meine Meinung: Das iPhone  
ist toll! (...)



**Topic:** iPhone  
**Genre:** Forum / Blog  
**Stimmung:** positiv

- **Ziele:** Vorstellen, welche Ansätze zur Klassifikation von Texten existieren. Nutzen für unser Projekt prüfen.
- **Literatur:**
  - Joachims (1998): Text Categorization with Support Vector Machines: Learning with Many Relevant Features
  - Stein and Meyer zu Eissen (2008): Retrieval Models for Genre Classification

# Clustering und Fingerprinting zur Erkennung von Ähnlichkeiten \*



- **Thema:** Gleiche und ähnliche Dokumente können durch Clustering- oder Fingerprinting-Ansätze erkannt werden
- **Beispiel:** Information: Retrieval vs. Extraction  
**vs.**  
Retrieval in der Information Extraction  **Cluster:** ähnlich  
**Fingerprint:** verschieden
- **Ziele:** Vorstellen der Ideen und Ansätze der beiden Techniken und vergleichen ihrer Einsatzgebiete
- **Literatur:**
  - Hoard and Zobel (2003): Methods for Identifying Versioned and Plagiarised Documents
  - Broder et. al.(1997): Syntactic clustering of the Web
  - Fetterly (2003): On the Evolution of Clusters of Near-Duplicate Web Pages

## Semantische Suchverfahren im Detail

- **Thema:** Komparative Beschreibung von Konzepten für semantische Suchverfahren.
- **Ziele:** Das Ziel dieser Seminararbeit ist es, den state-of-the-art semantischer Suchverfahren zu beschreiben und die verschiedenen Ansätze miteinander zu vergleichen.
- **Literatur:**
  - Tim Finin, James Mayfield, Anupam Joshi, R. Scott Cost, Clay Fink (2005): "Information Retrieval and the Semantic Web"
  - Mangold, Christoph (2007): "A survey and classification of semantic search approaches."
  - Beispielanwendung „Wolfram Alpha“: <http://www.wolframalpha.com/>
  - Beispielanwendung „Zemanta“: <http://www.zemanta.com/demo/>

# Text Labeling mit Sequenzmodellen \*

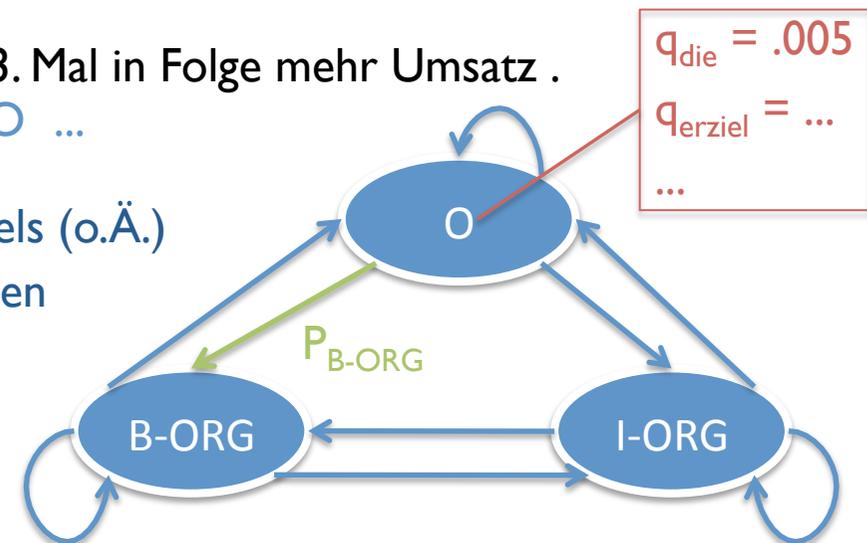
- **Thema:** Statistische Prozessmodelle dienen im Machine Learning dem Labeln von Sequenzen (z.B. Text). Die wahrscheinlichste Folge von Tags soll gefunden werden.

- **Beispiel:** Die Loewe AG erzielt zum 3. Mal in Folge mehr Umsatz .  
 ○ B-ORG I-ORG ○ ○ ○ ...

- **Ziele:** Vorstellen des Hidden Markov Models (o.Ä.) für das Labeln von Text und eines zugehörigen Sequence Labeling Algorithmus‘

- **Literatur:**

- Jurafsky and Martin (2009): Speech and Language Processing, Kapitel 6
- Manning and Schütze (1999): Foundations of Statistical Natural Language Processing, Kapitel 9
- Loper (2007): Sequential Model Structures Using Output Transformations



# Die Named-Entity-Recognition-Pipeline \*

- **Thema:** Die Erkennung von Entitäten ist eine klassische IE-Anwendung. Sie erfordert einige typische linguistische Vorverarbeitungsschritte.

- **Beispiel:**

Die	Loewe AG	erzielt	zum	3. Mal in Folge	mehr Umsatz	.
	DT NP0 NNI VBB	...				

- **Ziele:** Vorstellen der IE-Verarbeitungspipeline mit Fokus auf Named Entity Recognition.
- **Literatur:**
  - Manning and Schütze (1999): Foundations of Statistical Natural Language Processing, Kapitel 3+4
  - Jurafsky and Martin (2009): Speech and Language Processing, Kapitel 13+22
  - Tjong and De Meulder (2003): Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition

# Semantic Role Labeling

- **Thema:** Analyse der Aussagen in einem Satz: Erkennung der Komponenten, die eine semantische Rolle einnehmen.
- **Beispiel:**

acceptor	modal	neg.	<b>verb</b>	thing	accepted
He	would	n't	<b>accept</b>	anything of value	from
those he was writing about					.
accepted-from					
- **Ziele:** Vorstellen des Verfahrens und überprüfen, inwieweit nützlich und machbar in unserem Projekt
- **Literatur:**
  - Gildea and Jurafsky (2000): Automatic Labeling of Semantic Roles
  - Carreras and Màrquez (2005): Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling

## Pflicht-Themen

- Verfolgbarkeit von Anforderungen im Software-Entwicklungsprozess  
**Bearbeiter:** Christoph Fröhlich (Betreuer: BN)
- Vergleich von drei Open Source CMS-Systemen  
**Bearbeiter:** Denis Langlitz (BN)
- KnowledgeRepresentation/Persistence und Reasoning  
**Bearbeiter:** Stefan Middeke (FC)
- Kategorisierung von Textdokumenten mittels Klassifikation  
**Bearbeiter:** Steffen Beringer (HW)
- Clustering und Fingerprinting zur Erkennung von Ähnlichkeiten  
**Bearbeiter:** Jennifer Post (HW)
- Text Labeling mit Sequenzmodellen  
**Bearbeiter:** Enes Yigitbas (HW)
- Die Entity-Recognition-Pipeline  
**Bearbeiter:** Michael Meier (HW)

# Themenvergabe (2)



## Optionale Themen

- Klassische vs. agile Vorgehensmodelle im Software Engineering  
**Bearbeiter:** Razvan Hrestic (Silke Geisen)
- Komponentenbasiertes Software Engineering mit OSGi  
**Bearbeiter:** Thomas Kühne (FC)
- Automatische Modellgenerierung aus Systemspezifikationen  
**Bearbeiter:** Othmane Rhandor (FC)
- Reale Anwendungsgebiete des Semantic Web  
**Bearbeiter:** Mirko Rose (BN)
- ~~Natürliche Sprachverarbeitung mit UIMA vs. GATE~~
- ~~Semantische Suchverfahren im Detail~~
- ~~Semantic Role Labeling~~