



Fakultät II – Informatik, Wirtschafts- und Rechtswissenschaften  
Department für Informatik

# Simulationsbasierte Analyse und Entwicklung von Peer-to-Peer-Systemen

Dissertation zur Erlangung des Grades eines  
Doktors der Ingenieurwissenschaften

von

**Dipl.-Inform. Ludger Bischofs**

Gutachter:

**Prof. Dr. Wilhelm Hasselbring**

**Prof. Dr.-Ing. Oliver Theel**

Tag der Disputation: 19. September 2008



Berichte aus der Informatik

**Ludger Bischofs**

**Simulationsbasierte Analyse und Entwicklung  
von Peer-to-Peer-Systemen**

Shaker Verlag  
Aachen 2008

**Bibliografische Information der Deutschen Nationalbibliothek**

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Zugl.: Oldenburg, Univ., Diss., 2008

Copyright Shaker Verlag 2008

Alle Rechte, auch das des auszugsweisen Nachdruckes, der auszugsweisen oder vollständigen Wiedergabe, der Speicherung in Datenverarbeitungsanlagen und der Übersetzung, vorbehalten.

Printed in Germany.

ISBN 978-3-8322-7747-5

ISSN 0945-0807

Shaker Verlag GmbH • Postfach 101818 • 52018 Aachen

Telefon: 02407 / 95 96 - 0 • Telefax: 02407 / 95 96 - 9

Internet: [www.shaker.de](http://www.shaker.de) • E-Mail: [info@shaker.de](mailto:info@shaker.de)

# Danksagung

Mein erster Dank gilt meinem Doktorvater Prof. Dr. Wilhelm Hasselbring für die zahlreichen konstruktiven Hinweise und die Ausübung eines gewissen Drucks bei gleichzeitiger Gewährung der nötigen akademischen Freiräume. Weiterhin danke ich Prof. Dr.-Ing. Oliver Theel für die Übernahme der Zweitbegutachtung.

Auch den von mir im Umfeld der Arbeit betreuten Studierenden Volha Angele, Farzad Farzamfar, Markus Fromme, Dieter Hildebrandt, Jens Krefeldt, Peter Schwenkenberg und Stefan Willer möchte ich danken für die investierte Arbeit bei der Erstellung ihrer Ausarbeitungen.

Ein weiterer Dank gebührt meinen aktuellen und ehemaligen Arbeitskolleginnen und -kollegen für die angenehme Arbeitsatmosphäre. Bei Stefan Gudenkauf und Guido Scherp bedanke ich mich darüber hinaus für das Korrekturlesen der Arbeit.

Oldenburg, im Oktober 2008

Ludger Bischofs



# Zusammenfassung

Peer-to-Peer-Systeme (P2P-Systeme) sind verteilte Systeme, die aus Millionen gleichgestellter Einheiten (Peers) bestehen können. Die Gleichstellung von Peers wird realisiert, indem ein Peer sowohl Client- als auch Server- Funktionalität übernimmt. Die Funktionalität eines Peers wird dabei üblicherweise durch eine P2P-Applikation implementiert, die das Verhalten des gesamten P2P-Systems vorgibt. Für die Kommunikation untereinander nutzen Peers in der Regel ein P2P-Overlay-Netzwerk, das auf einem physikalischen Netzwerk aufbaut.

Ein P2P-System mit einem vorhersagbaren Verhalten zu entwickeln ist eine schwierige Aufgabe, weil das Verhalten von vielen Einflussfaktoren wie dem verwendeten P2P-Suchverfahren und dem zugrundeliegenden physikalischen Netzwerk abhängt. Einige Einflussfaktoren können aufgrund ihrer Komplexität oder unvollständig bekannter bzw. verstandener Bestandteile nicht vollständig erfasst werden. Beispielsweise kann das voraussichtliche Nutzerverhalten möglicherweise nur auf der Grundlage empirischer Daten geschätzt werden.

Bei der Entwicklung komplexer, dynamischer Softwaresysteme wie P2P-Systemen wird häufig Simulation eingesetzt, um die Eigenschaften des zu entwickelnden Systems auf Basis eines Simulationsmodells frühzeitig analysieren zu können. Im Gegensatz zu Simulationen in Naturwissenschaften, wo die Trennung von Realität und Modell klar ist, verwischt diese Grenze bei der Simulation von Softwaresystemen: Das simulierte Modell ist selbst ein Softwaresystem. Bei P2P-Systemen wird beispielsweise ein „einfaches“ P2P-System modelliert und simuliert, um die Eigenschaften eines „realen“ P2P-Systems vorherzusagen.

Der Hauptbeitrag dieser Arbeit ist die als *PeerSE* (**Peer** Software **E**ngineering) bezeichnete Methode zur ingenieurmäßigen Entwicklung von P2P-Systemen, die eine kontrollierte Überführung eines Simulationsmodells in ein reales Softwaresystem ermöglicht. Die Methode beginnt mit einer vergleichenden Analyse von P2P-Systemmodellen und führt über die experimentelle Implementierung in einer Laborumgebung zu einem realen P2P-System, das in der Zielumgebung zum Einsatz gebracht wird. Ein Simulationsmodell für P2P-Systeme und ein Werkzeug für die Durchführung von Simulations- und Laborexperimenten gehören ebenfalls zur Methode. Simulation ist ein essenzieller Bestandteil der Methode zur Identifikation und zum Vergleich von Modellen, die vorgegebene Anforderungen erfüllen. Nachdem ein geeignetes Modell gefunden wurde, können Modellkomponenten wiederverwendet bzw. verfeinert werden, um ein Laborsystem zu implementieren. Die kontrollierte Überführung der Modellkomponenten zu Laborkomponenten beinhaltet einen Vergleich von Simulations- und Laborexperimenten unter Verwendung der gleichen Metriken. Die Anwendbarkeit der PeerSE-Methode wurde gezeigt, indem ein P2P-System für verteilte Softwareentwickler analysiert und realisiert wurde.



# Abstract

Peer-to-Peer (P2P) systems are distributed systems composed of up to millions of functionally equivalent entities (peers), which form P2P overlay networks on top of physical networks to communicate. The functionality of a peer is implemented by a P2P application which defines the behavior of the whole P2P system. The equivalence of peers is realized by providing client functionality as well as server functionality.

Implementing a P2P system with specified behavior is a difficult task because the behavior depends on many factors, such as the used P2P search methods and the underlying physical network. Some factors cannot be taken into account completely because of their complexity or unknown or not understood parts. For instance, the prospective user behavior may only be estimated based on observed data.

When engineering complex, dynamic software systems such as P2P systems, simulation is often used to analyze the properties of these systems based on simulation models in an early development phase. With simulations in natural sciences, the separation of reality and (simulation) model is clear: the reality exists in nature, while the model exists as software within some computer system. When simulating software systems, this separation is not so obvious: the simulated model is itself a software system. With P2P systems, for instance, a „simplified“ P2P system is modeled and simulated for predicting properties of „real“ P2P systems.

The new software engineering contribution of this work is the **Peer Software Engineering** (*PeerSE*) method, which allows a controlled transition from simulation models to real-world software systems. The method starts with a comparative analysis of simulation models for P2P systems and proceeds iteratively toward the experimental implementation in a laboratory setting and finally a real-world P2P system deployed in a target environment. The method includes a simulation model for P2P systems and a tool supporting the execution of simulation and laboratory experiments. Simulation is an essential part of the *PeerSE* method used to identify and to compare models fulfilling given requirements. When an appropriate model has been found, model components can be reused and further refined to implement a laboratory P2P system. To allow for a controlled transition of model components to laboratory components, the results of simulation and laboratory experiments are directly compared using the same metrics. The applicability of the *PeerSE* method has been successfully evaluated by analyzing and realizing a P2P system for distributed software development.



# Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>13</b>
1.1. Motivation	13
1.2. Eigener Ansatz	13
1.3. Wissenschaftlicher Beitrag	14
1.4. Gliederung der Arbeit	15
<b>I. Grundlagen</b>	<b>17</b>
<b>2. Peer-to-Peer-Architekturen</b>	<b>19</b>
2.1. Definitionen	19
2.2. Klassifikation von Peer-to-Peer-Architekturen	21
2.2.1. Reine Peer-to-Peer-Architekturen	22
2.2.2. Hybride Peer-to-Peer-Architekturen	23
2.3. Peer-to-Peer-Suchverfahren	25
2.3.1. Klassifikation von Peer-to-Peer-Suchverfahren	25
2.3.2. Blinde Peer-to-Peer-Suchverfahren	28
2.3.3. Heuristische Peer-to-Peer-Suchverfahren	30
2.3.4. Index-basierte Peer-to-Peer-Suchverfahren	30
2.3.5. DHT-basierte Peer-to-Peer-Suchverfahren	31
2.4. Effizienzeigenschaften von Peer-to-Peer-Systemen	34
2.4.1. Definition Effizienz	34
2.4.2. Ressourcenbedarf eines Peer-to-Peer-Systems	35
2.4.3. Effizienzbewertung	37
2.4.4. Effizienzmetriken	38
<b>3. Simulation von Peer-to-Peer-Systemen</b>	<b>41</b>
3.1. Definition Simulation	41
3.2. Klassifikation von Simulationsmodellen	41
3.3. Simulationsmodelle für Peer-to-Peer-Systeme	42
3.4. Vorgehensmodelle zur Durchführung von Simulationsstudien	43
3.4.1. Problembeschreibung	43
3.4.2. Zielfestlegung und Gesamtprojektplanung	43

3.4.3.	Modellbildung . . . . .	45
3.4.4.	Datenerfassung . . . . .	45
3.4.5.	Modellübersetzung . . . . .	45
3.4.6.	Experimententwurf . . . . .	46
3.4.7.	Ausführung und Analyse . . . . .	46
3.4.8.	Dokumentation und Auswertung . . . . .	46
3.4.9.	Implementierung . . . . .	47
3.5.	Simulatoren für Peer-to-Peer-Systeme . . . . .	47
3.5.1.	Domänenunabhängige ereignisdiskrete Simulationsframeworks . . . . .	47
3.5.2.	Netzwerk-Simulatoren . . . . .	48
3.5.3.	Peer-to-Peer-Simulatoren . . . . .	48
3.6.	Alternativen zur Simulation . . . . .	49
3.6.1.	Mathematische Analyse . . . . .	49
3.6.2.	Profiling . . . . .	50
3.6.3.	Monitoring . . . . .	50
3.6.4.	Experimente mit Labor- und Realsystemen . . . . .	50
3.6.5.	Vergleich der Analyseverfahren . . . . .	51

## II. Simulationsbasierte Analyse und Entwicklung von Peer-to-Peer-Systemen 53

<b>4.</b>	<b>Die <i>PeerSE</i>-Methode</b>	<b>57</b>
4.1.	Simulation als Teil der <i>PeerSE</i> -Methode . . . . .	57
4.2.	Modellkomponenten eines Peer-to-Peer-Systems . . . . .	59
4.3.	Die Phasen der <i>PeerSE</i> -Methode . . . . .	59
4.3.1.	Die Analysephase . . . . .	63
4.3.2.	Die Implementierungsphase . . . . .	68
4.3.3.	Die Validierungsphase . . . . .	70
4.3.4.	Die Fertigstellungsphase . . . . .	70
<b>5.</b>	<b>Das <i>PeerSE</i>-Simulationsmodell</b>	<b>73</b>
5.1.	Strukturmodellierung des <i>PeerSE</i> -Simulationsmodells . . . . .	73
5.2.	Verhaltensmodellierung des <i>PeerSE</i> -Simulationsmodells . . . . .	76
5.3.	Struktur- und Verhaltensmodellierung der Modellkomponenten . . . . .	77
5.4.	Initialisierung des <i>PeerSE</i> -Simulationsmodells . . . . .	80
5.4.1.	Initialisierung und Visualisierung der Modellkomponenten über <i>PeerSE</i> -Graphen . . . . .	80
5.4.2.	Initialisierung der Ereignisliste . . . . .	85
<b>6.</b>	<b>Anforderungen und Grobentwurf der <i>PeerSE</i>-Experimentumgebung</b>	<b>87</b>

---

6.1. Anforderungen der <i>PeerSE</i> -Experimentumgebung . . . . .	87
6.1.1. Funktionale Anforderungen . . . . .	87
6.1.2. Nichtfunktionale Anforderungen . . . . .	89
6.2. Grobentwurf der <i>PeerSE</i> -Experimentumgebung . . . . .	90
6.2.1. Hauptkomponenten der <i>PeerSE</i> -Experimentumgebung . . . . .	91
6.2.2. Externe Schnittstellen der <i>PeerSE</i> -Experimentumgebung . . . . .	94
6.2.3. Externe Komponenten der <i>PeerSE</i> -Experimentumgebung . . . . .	95

### **III. Evaluation 97**

#### **7. Realisierung der *PeerSE*-Experimentumgebung 101**

7.1. Vorarbeiten . . . . .	101
7.1.1. Der eaSim-Simulator . . . . .	101
7.1.2. Der OrgSim-Simulator . . . . .	103
7.1.3. Der PeerThing-Simulator . . . . .	103
7.1.4. Das RealPeer-Framework . . . . .	103
7.1.5. Verteilte Simulation . . . . .	104
7.2. Entwurfsentscheidungen . . . . .	104
7.2.1. Umsetzung funktionaler Anforderungen . . . . .	105
7.2.2. Umsetzung nichtfunktionaler Anforderungen . . . . .	107
7.2.3. Beispielhafter Hard- und Softwareeinsatz . . . . .	108
7.3. Feinentwurf und Implementierung . . . . .	110
7.3.1. Paketstruktur . . . . .	112
7.3.2. Das Paket <code>peerse</code> . . . . .	112
7.3.3. Das Paket <code>peerse.data</code> . . . . .	112
7.3.4. Das Paket <code>peerse.experiment</code> . . . . .	112
7.3.5. Das Paket <code>peerse.graph</code> . . . . .	115
7.3.6. Das Paket <code>peerse.gui</code> . . . . .	115
7.3.7. Das Paket <code>peerse.model</code> . . . . .	117

#### **8. Anwendung der *PeerSE*-Methode 119**

8.1. Anforderungen . . . . .	119
8.2. Analyse . . . . .	120
8.2.1. Problembeschreibung . . . . .	120
8.2.2. Zielfestlegung und Gesamtprojektplanung . . . . .	120
8.2.3. Modellbildung . . . . .	121
8.2.4. Datenerfassung . . . . .	131
8.2.5. Experimententwurf . . . . .	137
8.2.6. Ausführung und Analyse . . . . .	139
8.2.7. Auswertung . . . . .	152

8.3. Implementierung . . . . .	154
8.4. Validierung . . . . .	155
8.4.1. Experimentaufbau und -durchführung . . . . .	155
8.4.2. Experimentergebnisse . . . . .	156
8.4.3. Fazit . . . . .	165
8.5. Fertigstellung . . . . .	165
8.6. Zusammenfassung und Bewertung der <i>PeerSE</i> -Methode . . . . .	167
<b>9. Verwandte Arbeiten</b>	<b>171</b>
<b>IV. Zusammenfassung und Ausblick</b>	<b>173</b>
<b>10. Zusammenfassung</b>	<b>175</b>
<b>11. Ausblick</b>	<b>177</b>
<b>V. Anhang</b>	<b>179</b>
<b>A. Object-Z-Notation</b>	<b>181</b>
<b>B. Übungsblatt zum eaSim-Simulator</b>	<b>185</b>
<b>Abbildungsverzeichnis</b>	<b>187</b>
<b>Tabellenverzeichnis</b>	<b>191</b>
<b>Listings</b>	<b>193</b>
<b>Literaturverzeichnis</b>	<b>195</b>
<b>Index</b>	<b>211</b>