

# Seminar: Intelligente Anwendungen im Internet

## Vortrag: Anpassung von Ubiquitous-Webanwendungen

Referent:

**Jochen Gerlach**

Dieser Vortrag basiert auf dem Text:

**Customisation for Ubiquitous Web Applications  
A Comparison of Approaches**

Autoren:

**Gerti Kappel, Birgit Pröll,  
Werner Retschitzegger, Wieland Schwinger**

05.11.2007



# Motivation: Anpassung von Ubiquitous-Webanwendungen

Das Internet entwickelt sich weiter:

- ▶ Anzahl der Endgeräte steigt.
- ▶ Vernetzung nimmt weiter zu.
- ▶ Computer dringen in immer unterschiedlichere Lebensbereiche vor.

⇒ Anwendungen müssen sich anpassen.

Wir wollen erarbeiten:

Was versteht man unter Anpassung?

Wie kann man Anpassung modellieren?

Wie sieht die Webanwendung von morgen aus?

Welche Anwendung gibt es schon heute, und wie sehen die Ansätze aus?

# Definition: Webanwendung

Eine *Webanwendung* kann allgemein definiert werden als ein Softwaresystem, das auf Technologien und Standards des Web beruht und web-spezifische Ressourcen bereitstellt[1].

Komponenten:

- ▶ *Web-Client (User-Agent)*
- ▶ *Web-Server*
- ▶ *Protokoll*

Vergleich zu Standardanwendungen:

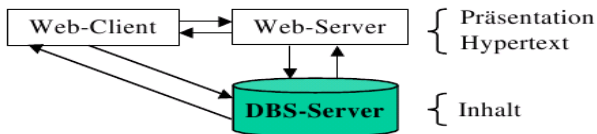
- ▶ besitzen i.d.R einen großen, oft anonymen Benutzerkreis
- ▶ unterliegen einen permanenten Veränderungsprozess
- ▶ immer verteilte Anwendungen
- ▶ oft sehr große Datenbestände
- ▶ ...

# 1. Generation: Statische-Webanwendung



- ▶ statische HTML-Dateien auf dem Webserver
- ▶ Anfragen durch das HTTP-Protokoll
- ▶ Vorteil: einfach und stabil
- ▶ **Nachteile:**
  - ▶ Änderungen sind kostenintensiv
  - ▶ neigt zu veralteten Informationen
  - ▶ oft Inkonsistenzen

## 2. Generation: Datenbankbasierte-Webanwendung

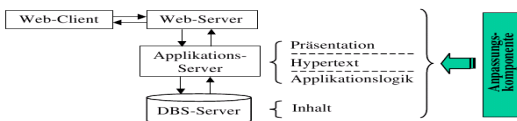


- ▶ generierung von Webseiten aus Datenbank-Daten
- ▶ mächtigere Programmiersprachen: Java-Servlets auf Serverseite und Java-Applets auf Client Seite  
⇒ zustandsbasierte Anwendung
- ▶ Möglichkeit: Einsatz eines Applikation-Servers für spezielle Webanwendungen

# Ubiquitous-Computing

- ▶ Vision von einer Zukunft der Allgegenwärtigkeit von Computern (Mark Weiser \* 23. Juli 1952 - † 27. April 1999)
- ▶ von einem Computer als Universalwerkzeug zu vielen spezialisierten Computern
- ▶ getestete Entwicklungen im Xerox Park:
  - ▶ Active Badge (ortsbasierte Personenerkennung)
  - ▶ XEROX PARC Tab (Vorläufer des PDA)
  - ▶ ...
- ▶ Problematiken des Ubiquitous-Computing
  - ▶ Vernetzung aller Haushaltsgegenstände mit dem Internet
  - ▶ Sicherheit und Datenschutz
- ▶ BIS JETZT: kein Ubiquitous-Computing wie geplant  
ABER: unterschiedliche Rechnerarchitekturen

### 3. Generation: Ubiquitäre-Webanwendung



#### Ubiquitäre-Webanwendung

Eine *Ubiquitäre-Webanwendung* stellt personalisierte Dienste zu jeder Zeit, an jedem Ort und für beliebige Endgeräte zur Verfügung, womit ein allgegenwärtiger Zugriff ermöglicht wird.

1. Anpassung einer Webanwendung an verschiedenen Klassen von Endgeräte.
2. Anpassung einer Webanwendung an Benutzerprofile.
3. Realisierung orts-/zeitabhängiger Dienste.
4. **Beispiel:** Anzeige des Mittagsmenüs auf mobilen Endgeräten.

⇒ Wir müssen also den Begriff der Anpassung genauer untersuchen

# Begriff: Anpassung 1/3

Was versteht man unter Anpassung?

- ▶ Ähnlicher Begriff: *Assimilation* beschreibt in der Lernpsychologie einen Prozess, über den das, was wahrgenommen wird, so verändert wird, dass es zu den gegenwärtigen vorhandenen Strukturen passt.[2] (passt nicht gut)
- ▶ Unter *Anpassung* versteht man in der Elektrotechnik die Abstimmung eines Messwertes auf einen anderen. Gewöhnlicherweise wird zwischen Spannungsanpassung, Leistungsanpassung, Impedanzanpassung und Stromanpassung unterschieden.[2] (passt auch nicht gut)
- ▶ Unter *evolutionärer Anpassung (Adaption)* versteht man Eigenheiten in Körperbau und Verhalten, die als evolutionäre Reaktion auf spezielle Umweltfaktoren gedeutet werden können.[2] (schon besser)

Interessante Begriffe sind sicherlich: Wahrnehmen, Verändern, Abstimmung, Verhalten

Wie aber sieht das genau bei Webanwendungen aus?



## Begriff: Anpassung 2/3

### Definition (Kontextsensitives System)

Ein System ist *kontextsensitiv*(*Context-Aware*), wenn es ohne explizite Nutzerinteraktion auf relevante Ereignisse in der natürlichen Umgebung reagieren kann, bzw. diese zur weiteren Bearbeitung nutzen kann.[2]

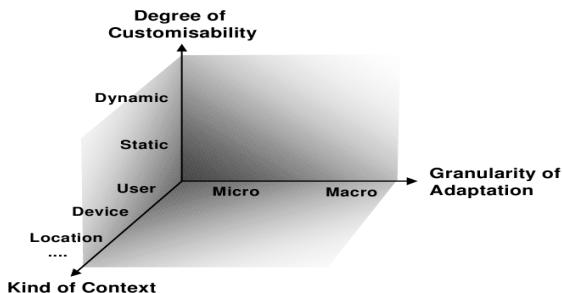
- ▶ Es verwendet Sensoren um Informationen über die Umgebung zu beziehen.
- ▶ Mit den Informationen werden Schlüsse über den Kontext gezogen.
- ▶ Kontext wird von der Anwendung verwendet, um ihr Verhalten anzupassen.

### Definition (Kontext)

Diejenigen Komponenten des Anwendungsbereichs, welche mit einem System interagieren, aber selbst nicht Bestandteil des Systems sind.[2]

# Begriff: Anpassung 3/3

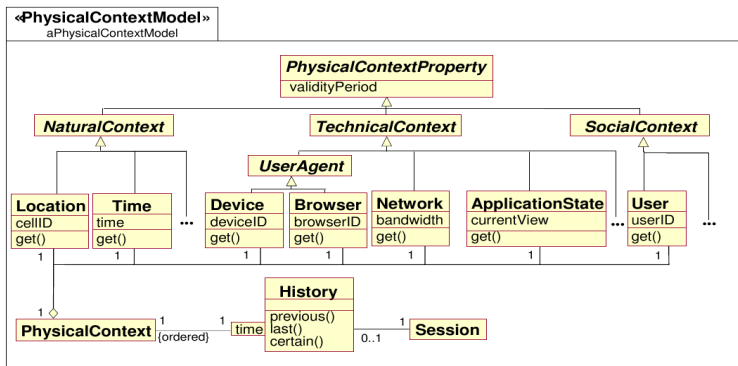
Anpassungsdimensionen:



## Abstraktionen der Anpassung:

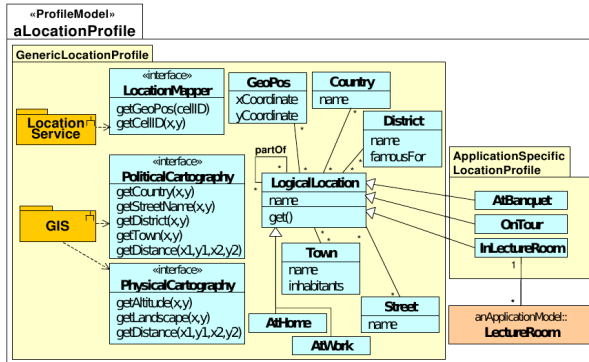
- ▶ Kontextabstraktion: Physikalischer-Kontext, Logischer-Kontext
- ▶ Anwendungsabstraktion: Inhalt-, Hyperbase-, Präsentationsebene

# Physikalischer-Kontext



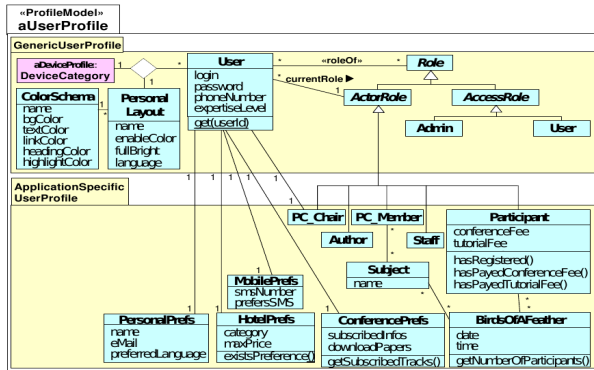
- ▶ **Pull-Based:** Beschaffung wenn Information benötigt wird
- ▶ **Push-Based:** Information löst ein Ereignis aus
- ▶ **Automationsgrad:** manuelle oder automatische Beschaffung
- ▶ **Dynamik:** Frequenz und Zeit der Beschaffung

# Logischer-Kontext: Beispiel Location-Profil 1/2



wird aus dem Physikalischen-Kontext gewonnen

# Logischer Kontext: Beispiel User-Profile 2/2



generischer Teil ist anwendungsunabhängig

# Anpassungsebenen

Anpassung findet auf 3 Anwendungsebenen statt:

## ▶ **Inhalts-Ebene:**

- ▶ Erzeugung, Integration, Aktualisierung, Bereitstellung der Daten, die eine Webanwendung zur Verfügung stellt.
- ▶ mögliche Elemente: Entität, Beziehung, Datenbank
- ▶ unterschiedliche Änderungshäufigkeiten und Qualitätsansprüche

## ▶ **Hyperbase-Ebene:**

- ▶ mögliche Elemente: Seiten, Links, Felder, Operationen
- ▶ beschäftigt sich mit Navigation
- ▶ Probleme der Desorientierung und kognitiver Belastung

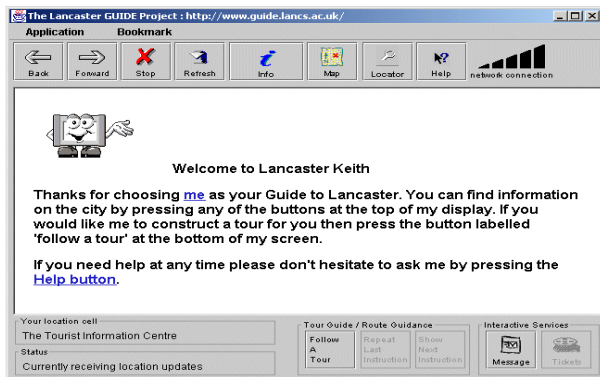
## ▶ **Präsentations-Ebene:**

- ▶ beschäftigt sich mit Layout
- ▶ Look and Feel der Benutzerschnittstelle
- ▶ permanente Notwendigkeit das Layout anzupassen

# Erste Umsetzungen von UC-Lösungen

<u>Name</u>	<u>Beschreibung(Autoren s. Referenz)</u>
Atalato et al.	auf UML basierter Modellierungsansatz [1]
ContextToolkit	Java und XML basierter Ansatz [1]
AVANTI	entwickelt um Benutzer mit Stadtinformationen zu versorgen [3]
Fox et al.	Framework mit starkem Ansatz zur Datenkompression [4]
Schmidt et al.	umfangreicher Toolkit um Kontextinformationen auszulesen [2]
Oracle 9i	datenbankbasierte Applikation-Server Lösung [4]

# The-GUIDE: Touristenführer für die Stadt Lancaster[2],[3]



Projektdauer: 1. April 1997 bis 31. Juli 1999

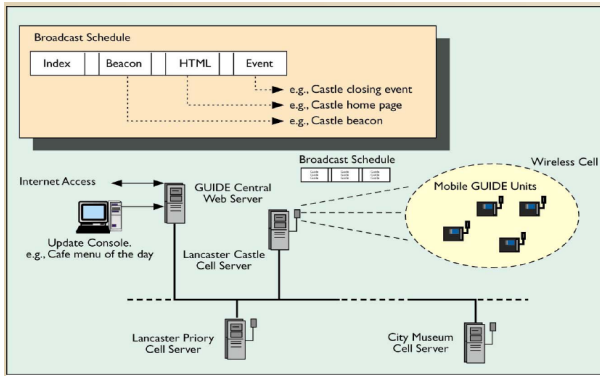


# The-GUIDE: Die Anforderungen



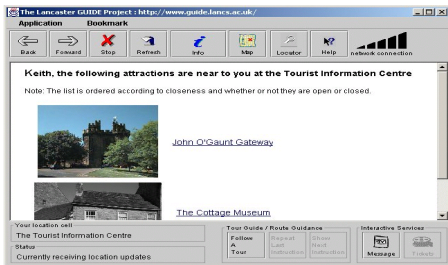
- ▶ freundliche touristengerechte Ausstrahlung
- ▶ *Flexibilität*: soll sich dem Rythmus der Benutzer anpassen
- ▶ *Dynamische Information*: Öffnungszeiten, Tagesgerichte
- ▶ *Context-Aware Information*: Informationen über den Benutzer, Informationen über die Umgebung
- ▶ *Interactive Services*: Inanspruchnahme von Services einiger Einrichtungen von beliebigen Orten  
**Beispiel**: Online Reisebuchung beim Tourismuscenter
- ▶ Senden und Empfangen von Nachrichten zwischen den Benutzern
- ▶ Planen von Touren durch die Stadt

# The-GUIDE: Technische Infrastruktur



- ▶ Linux-Server und Java-Web-Browser
- ▶ Endgeräte: Fujitsu Siemens TeamPad 7600 Tablet PC (Pentium 166 Mhz), PDA etc.

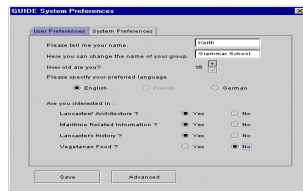
# The-GUIDE: Merkmale des Kontextes



- ▶ Profilbasiertes Kontextmodell:
  - ▶ Personal-Context: Name, Alter, Nationalität, Hobbies etc.
  - ▶ Environmental-Context: Context-Aware-Informationen, Geographische Informationen, Navigationspunkte, Location Objects
- ▶ kein Mechanismus vorgesehen um die vordefinierten Kontexteigenschaften zu erweitern
- ▶ pull-basierter Informationszugriff

# The-GUIDE: Bewertung der Anpassung

```
<HTML>
<P><B><FONT SIZE="5" FACE="Comic Sans MS">
The following attractions are near to you
now.</FONT><IB><IP>
<P><FONT FACE="Comic Sans MS">Note: The
list below is ordered according to closeness
and whether or not they are open or
closed.</FONT></P>
<P><FONT FACE="Comic Sans MS"><BR>
<GUIDETAG INSERT NEIGHBOURS>
</FONT></P>
</HTML>
```



- ▶ Anpassungselemente: Text- und Link-Anpassung
- ▶ Anfragen des Kontextes über HTML-Meta-Tags
- ▶ keine Separierung zwischen unterschiedlichen Aufgaben in der Anpassung
- ▶ Anpassung erfolgt i.d.R vollautomatisch
- ▶ dynamische Anpassung sobald der Benutzer die Context-Aware-Webseite betritt
- ▶ keine Erweiterbarkeit der vordefinierten HTML-Tag-Menge vorgesehen



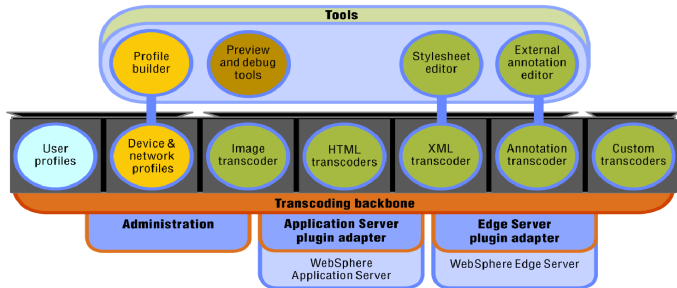
Teil von IBM-WebSphere[1].

Multi-Channel-Delivery

Erweiterbar um WebSphere-Personalisation- und WebSphere-Everyplace-Suite.

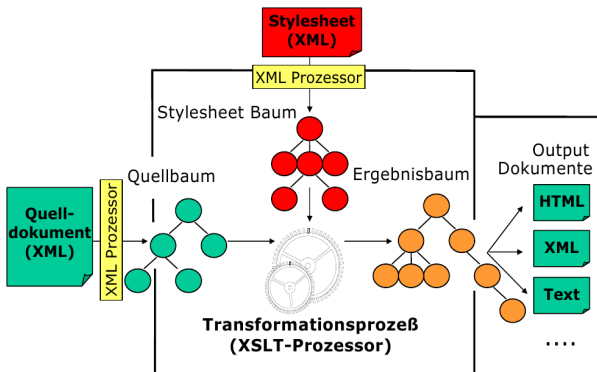
Anwendung: Informationsportale und E-Business Lösungen.

# Komponenten des Transcoding-Publishers



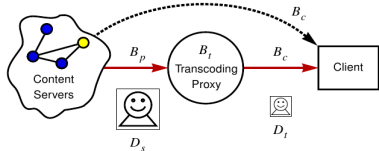
WebSphere-Transcoding-Publisher ist eine serverbasierte Software die auf dynamischen Wege, Webinhalte und Anwendungen, transformiert und für verschiedenartige Ausgabegeräte optimiert.

# Transcoding-Publisher: XML-Techniken[1]



- ▶ Sprachen: XML, XSL, XPath, XSLT
- ▶ **Beispiel:** Multi-Delivery, Datenaustausch, Datenintegration

# Transcoding-Publisher: Proxybasierte-Architektur[3]



Workstation	Color PC	TV browser	HHC	PDA	Smart Phone
					"bridge"
<b>size:</b> 256 x 256	192 x 192	128 x 128	96 x 96	64 x 64	-
<b>fidelity:</b> 38 KB	23 KB	8 KB	4 KB	0.6 KB	100 B
<b>color:</b> 24 bit RGB	24 bit RGB	256 colors	4 bit gray	B/W	-

## Beispiel Image-Transcoding:

1. Image-Content-Analysis: Bildtyp- und Verwendungszweck-Klassifikation
2. Image-Transcoding: Grösse, Wiedergabetreue, Farbe, Substitution nach Ausgabegerät bestimmen

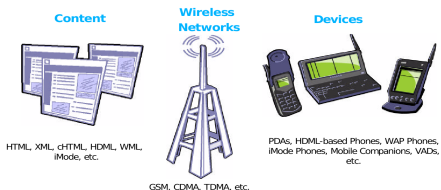


# Transcoding-Publisher: Kontextbewertung

- ▶ Umgebungseigenschaften: Location, Device, Network, User
- ▶ Physikalischer Kontext kann durch Request-Editors (logische Kontextinformationen) erweitert werden.
- ▶ Logischer-Kontext wird durch Profile festgelegt, die vom Entwickler bestimmt werden.
- ▶ Weiterer Logischer-Kontext durch "Page Annotations" möglich.
- ▶ bis jetzt keine Historie des Kontextes
- ▶ keine Möglichkeit abstrakteren Kontext zu erzeugen



# Transcoding-Publisher: Bewertung der Anpassung



- ▶ vordefinierte Anpassungsregeln (Document Editors)
- ▶ Textanpassung: Kürzen und Erweitern der Seite
- ▶ Transcoding: Bild- und Multimedia-Transformationsoperationen
- ▶ Stylesheet-Editor: Anpassungsoperationen über XSLT
- ▶ neue Anpassungen auch durch Servlets möglich
- ▶ komplexe Anpassungen möglich: Micro- und Macro-Ebene
- ▶ Anpassung wird nach der Anfrage des Benutzers vollautomatisch vorgenommen.
- ▶ Anpassung verändert Originaldokument und erfolgt dynamisch.

# Gesamteindruck und Ausblick

Es existiert noch kein einheitlicher Standard:

- ▶ nicht alle Umgebungsvariablen werden gleichzeitig berücksichtigt
- ▶ nur vergangenheitsbezogene Information wird berücksichtigt
- ▶ kein Standard für eine einheitliche Kontextrepräsentation
- ▶ neuer Ansatz: WUML basierend auf UML, Java, XML und ECA-Regelsystemen[3]

**Vielen Dank für Eure Aufmerksamkeit**

## Weitere ergänzende Literatur zum Ausgangstext 1/5:



[Websphere Transcoding Publisher.](#)

[http://www-306.ibm.com/software/pervasive/transcoding\\_publisher](http://www-306.ibm.com/software/pervasive/transcoding_publisher)



[Wikipedia - Die freie Encyclopedia.](#)

[www.wikipedia.org](http://www.wikipedia.org)



[Josef Fink, Alfred Kobsa, Andreas Nill.](#)

Adaptable and adaptive information provision for all users, including disabled and elderly people.

*The New Review of Hypermedia and Multimedia*, 4:163–188, 1998.  
[citeseer.ist.psu.edu/fink99adaptable.html](http://citeseer.ist.psu.edu/fink99adaptable.html)



[Armando Fox, Steven D. Gribble, Eric A. Brewer, Elan Amir.](#)

Adapting to Network and Client Variability via On-Demand Dynamic Distillation.

*Proc. Seventh Intl. Conf. on Arch. Support for Prog. Lang. and Oper. Sys. (ASPLOS-VII)*, Cambridge, MA, 1996.  
[citeseer.ist.psu.edu/fox96adapting.html](http://citeseer.ist.psu.edu/fox96adapting.html)

## Weitere ergänzende Literatur zum Ausgangstext 2/5:



Sigi Reich Werner Retschitzegger Gerti Kappel, Birgit Pröll.  
Web Engineering - Eine neue Disziplin zur systematischen  
Entwicklung von Webanwendungen.

[www.big.tuwien.ac.at/teaching/offer/ss03/we\\_vo/  
kapitel1.pdf](http://www.big.tuwien.ac.at/teaching/offer/ss03/we_vo/kapitel1.pdf)



Martin Glinz.

Spezifikation von Software - Kapitel 2 Systeme und ihr Kontext

[www.ifi.unizh.ch/req/ftp/ses/kapitel\\_02.pdf](http://www.ifi.unizh.ch/req/ftp/ses/kapitel_02.pdf)



G. Kappel, W. Retschitzegger, W. Schwinger.

Modeling Ubiquitous Web Applications: The WUML approach, 2001.

International Workshop on Data Semantics in Web Information  
Systems (DASWIS-2001), co-located with 20th International  
Conference on Conceptual Modeling Workshop (ER2001),  
Yokohama, Japan, November 27-30, 2001

[citeseer.ist.psu.edu/kappel01modeling.html](http://citeseer.ist.psu.edu/kappel01modeling.html)

## Weitere ergänzende Literatur zum Ausgangstext 3/5:



Gerti Kappel.

M7 XSLT.

[www.big.tuwien.ac.at/teaching/offer/ss05/we\\_vo/  
weM7XSLT.pdf](http://www.big.tuwien.ac.at/teaching/offer/ss05/we_vo/weM7XSLT.pdf)



Keith Mitchell Keith Cheverst, Nigel Davies and Adrian Friday.

Experiences of Developing and Deploying a Context - Aware Tourist Guide: The GUIDE Project.



Nigel Davies Keith Cheverst, Keith Mitchell.

The Role of Adaptive Hypermedia in a context-aware Tourist Guide.



P. J. Gill P. Washington.

Oracle9i Application, Server Wireless Edition in Action

Oracle Magazine, Jan.-Feb 2002

## Weitere ergänzende Literatur zum Ausgangstext 4/5:



Daniel Salber, Anind K. Dey, and Gregory D. Abowd.

The Context Toolkit: Aiding the Development of Context-Enabled Applications.

In *CHI*, pages 434–441, 1999.

[citeseer.ist.psu.edu/salber99context.html](http://citeseer.ist.psu.edu/salber99context.html)



A. Schmidt, K. A. Aidoo, A. Takaluoma, U. Tuomela, K. Van Laerhoven, and W. Van de Velde.

Advanced Interaction in Context.

*Lecture Notes in Computer Science*, 1707:89–??, 1999.

[citeseer.ist.psu.edu/schmidt99advanced.html](http://citeseer.ist.psu.edu/schmidt99advanced.html)



J. Smith, R. Mohan, and C. Li.

Content-based Transcoding of Images in the Internet.  
pages 7–11.

[citeseer.ist.psu.edu/smith98contentbased.html](http://citeseer.ist.psu.edu/smith98contentbased.html)



Weitere ergänzende Literatur zum Ausgangstext 5/5:



M. T. Siponen T. Alatalo.

Towards the OWLA methodology for development of Open.  
[citeseer.ist.psu.edu/520752.html](http://citeseer.ist.psu.edu/520752.html)