



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Seminarausarbeitung AW2

Jens Ellenberg

Ein Wecker in einem ubicom Haus

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Der Wecker 2.0	1
2	Projekte	2
2.1	Projekt: ADDICT	2
2.1.1	Wo liegt der Fokus?	3
2.1.2	Wie wird der Kontext erstellt und verteilt?	3
2.1.3	Was wird mit dem Kontext gemacht?	4
2.1.4	Vergleich mit dem Projekt Wecker 2.0	4
2.2	Projekt: Search Application	5
2.2.1	Wo liegt der Fokus?	6
2.2.2	Wie wird der Kontext erstellt und verteilt?	6
2.2.3	Was wird mit dem Kontext gemacht?	6
2.2.4	Vergleich mit dem Projekt „Wecker 2.0“	7
2.3	Projekt: Assistive Meeting	7
2.3.1	Wo liegt der Fokus?	8
2.3.2	Wie wird der Kontext erstellt und verteilt?	8
2.3.3	Was wird mit dem Kontext gemacht?	8
2.3.4	Vergleich mit dem Projekt „Wecker 2.0“	9
3	Fazit	10
3.1	Zusammenfassung	10
3.2	Ausblick	10
	Literaturverzeichnis	11

1 Einleitung

1.1 Motivation

Die Vision von Mark Weiser von der Allgegenwärtigkeit von Computern hat viele Entwickler inspiriert [10]. Er hat in seiner Vision eine Zukunft gezeichnet, in der Computer sich mit ihrem Verhalten dem Menschen anpassen. Programme mit solchen Möglichkeiten nennt man im Allgemeinen „Context Aware“ beziehungsweise „Context Aware Computing“. Der „Wecker 2.0“ [5] ist eines dieser Programme aus einer intelligenten Wohnung, das in der Lage ist sich seiner Umgebung anzupassen. Der „Wecker 2.0“ erzeugt seine Intelligenz aus seinem Wissen über den Bewohner der Wohnung. Dieses Wissen wird für den Wecker als Kontext des Bewohners dargestellt. Der Kontext ist somit beim „Wecker 2.0“ der zentrale Mechanismus, durch den der Wecker seine besondere Flexibilität erhält. In dieser Ausarbeitung soll nun untersucht werden, welche weiteren Möglichkeiten es gibt Kontext einzusetzen. Dafür wurden drei Arbeiten zum Vergleich herangezogen, die Kontext in einer anderen Art und Weise einsetzen als der Wecker. Da der „Wecker 2.0“ in dieser Ausarbeitung als Ausgangspunkt dienen soll, wird hier noch einmal kurzer Überblick gegeben.

1.2 Der Wecker 2.0

In [5] wird ein Vorgehen skizziert, wie man einen Wecker der Zukunft gestalten könnte. Dabei kam heraus, dass es für ein kontextsensitives Programm notwendig ist, sich der Umgebung, in der es eingesetzt wird, bewusst zu sein. Die Umgebung, für die der Wecker genutzt wird, ist der Bewohner der Wohnung. Der Bewohner wird in den Ausprägungen Identität, Ort, Zeit und Aktion beschrieben, also in einem Kontext. Dieser Kontext wird dabei über verschiedene Sensoren aus der intelligenten Wohnung erfasst. Der Wecker nutzt diese Auswertung, um sich der Situation entsprechend zu verhalten und sein Verhalten anzupassen, sollte sich die Situation ändern. Dieses geschieht alles, ohne, dass der Bewohner der Wohnung sich darum kümmern müsste.

Der Kontext wird also beim „Wecker 2.0“ dazu genutzt, das Verhalten des Weckers (also der Software) anzupassen.

2 Projekte

Im Folgenden werden drei Projekte beschrieben. Bei der Beschreibung der Projekte wird ein besonderer Fokus auf den Einsatz des Kontextes innerhalb der Projekte gelegt.

2.1 Projekt: ADDICT

ADDICT (Context-Aware Data Design, Integration, Customization and Tailoring) ist ein Projekt der italienischen Universität „Politecnico di Milano“ des Departement „Elettronica e Informazione (Elektronik und Information)“. In diesem Projekt geht es darum, ein vollständiges Framework zu erstellen, bei dem Context Awareness eingesetzt wird, um Daten auf mobilen Endgeräten zur Verfügung zu stellen. Dieses Framework soll dem Nutzer bei der dynamischen Anbindung und Integration von neuen und zur Verfügung stehenden Informationsquellen unterstützen. Dabei soll nur ein angemessener, kontextbasierter Teil der Informationen auf das mobile Gerät geliefert werden. Die Idee dabei ist, dass nicht alle verfügbaren Informationen bzw. Daten notwendig sind. Je nach dem, welche Aufgabe gerade bearbeitet wird und in welchem Umfeld sich der Anwender befindet, braucht er nur einen bestimmten Teil der Informationen aus einer Datenbank. Die Situation und das Vorhaben des Anwenders ist der Kontext für das System. Wenn die Anwendung sich dem Kontext bewusst ist, kann sie die Menge der Informationen den Anforderungen anpassen, die sich aus dem Kontext ergeben. Daraus folgt der Vorteil, dass der Nutzer nicht erst selbst manuell die wichtigen Informationen suchen muss. Des Weiteren sind die Ressourcen bei mobilen Endgerät begrenzt. In einem W-LAN¹ steht nur ein Teil der Bandbreite zur Verfügung, wie in einem LAN². Dieses ist ein weiterer wichtiger Grund dafür, dass die Daten reduziert werden müssen.

Dieses Projekt hat sich zum Ziel gesetzt, das Problem in seiner gesamten Breite zu bearbeiten. Daher wird das Projekt durch verschiedene Teilprojekte realisiert. Die folgende Auflistung gibt eine Auswahl der verschiedenen Projekte wider.

Requirements for Context-Dependent Mobile Access to Information Services [9]: In diesem Paper von 2004 werden Voraussetzungen für das ADDICT Projekt vorgestellt.

¹Wireless Local Area Network

²Local Area Network

Es sind die Voraussetzungen um verschiedene Informationsquellen einem Nutzer von mobilen Geräten zugänglich zu machen. Dabei sollen durch Kontextinformationen die Daten aus den Informationsquellen reduziert werden.

Using Context for the Extraction of Relational Views [2]: In diesem Paper von 2007 wird gezeigt, wie man mit Kontextinformationen umgeht, die nur die Daten aus einer Datenbank bekommen, die zum Kontext passen. Diese Reduzierung der Daten wird erreicht, indem die Sichten auf die Datenbank mit dem aktuellen Kontext angepasst werden.

CADD: a tool for context modeling and data tailoring [1]: 2007 wurde auch ein Tool vorgestellt, mit dem sich verschiedene Kontexte entwerfen lassen. Für jeden Kontext kann mit diesem Werkzeug eine Sicht auf die Datenbank definiert werden.

Accessing and Documenting Relational Databases through OWL Ontologies [3]: 2009 wurden Methoden entwickelt, wie man durch die Sprachen OWL³ und SPARQL⁴ automatisiert Abfragen generieren kann, die der Datenbankabfragesprache SQL⁵ entsprechen.

A methodology for preference-based personalization of contextual data [7]: In diesem Paper von 2009 wird vorgestellt, welche Möglichkeiten es gibt, das bestehende Modell speziell auf die Bedürfnisse einzelner Personen anzupassen.

2.1.1 Wo liegt der Fokus?

Die Bearbeitung von Daten steht hier im Mittelpunkt des Projekts. Es geht darum eine Datenbank auch auf mobilen Geräten nutzbar zu machen. Da es im mobilen Umfeld nur reduzierte Ressourcen gibt, müssen die Daten auch entsprechend reduziert werden. Außer diesem allgemeinen Ziel, gibt es keinen speziellen Fokus in diesem Projekt. Vielmehr werden verschiedene Teilziele definiert, um das Gesamtziel des Projekts zu realisieren. Dabei behandeln alle Projekte die Idee, Kontext zu nutzen, um Zugriffe auf eine Datenbank anzupassen. Außerdem beschäftigen sich die Projekte mit den reduzierten Ressourcen von mobilen Anwendungen und passen existierende Sprachen und Werkzeuge dementsprechend an.

2.1.2 Wie wird der Kontext erstellt und verteilt?

Für die Erstellung des Kontextes wurde ein eigenes Tool entwickelt [1]. Auf der Abbildung 2.1 wird die Arbeitsweise des Tools dargestellt. Der Kontext wird zentral im Server aus ver-

³Web Ontologie Language

⁴SPARQL Protocol and RDF Query Language

⁵Structured Query Language

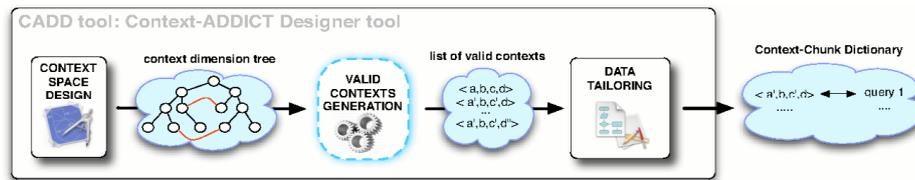


Abbildung 2.1: CADD tool

schiedenen Informationen zusammengestellt. Hierbei wird der Kontext durch einen Baum dargestellt. In diesem Baum bedeutet eine tiefere Ebene eine Spezialisierung des Zweigs. Der Kontext setzt sich aus den Dimensionen Rolle, Situation, Zeit, Ort und Interessensbereich zusammen. Für jede Ausprägung des Kontextes ist nun eine Sicht auf die Datenbank abgespeichert, die dann für die Abfragen genutzt wird.

2.1.3 Was wird mit dem Kontext gemacht?

Der Kontext wird hier dazu genutzt, Daten einzuschränken. Die Daten liegen in einer Datenbank vor und werden auf mobilen Endgeräten benötigt. Da bei mobilen Endgeräten nur reduzierte Ressourcen zur Verfügung stehen, können nicht alle Daten bereit gestellt werden. Durch den Kontext werden genau die Daten der Datenbank dem Nutzer präsentiert, die er in seiner Situation braucht. Dieses wird über spezielle Ansichten der Datenbank erreicht. Je nach Kontext ist eine andere Ansicht auf die Datenbank aktiv. Dadurch liefern die Abfragen an die Datenbank immer nur die Daten, die in der Situation des Anwenders notwendig sind.

2.1.4 Vergleich mit dem Projekt Wecker 2.0

Die Kontextdarstellung beim Wecker lehnt sich der Darstellung aus diesem Projekt an. In beiden Projekten ist der Kontext aktionsorientiert. Das bedeutet, dass die Aktion beziehungsweise das Vorhaben der Person, im Vordergrund stehen. Wie auch beim Wecker, wird der Kontext aus verschiedenen Informationen von einer eigenen Einheit erstellt und anschließend weitergegeben. Im Gegensatz zum Wecker werden aber hier nicht die Funktionen der Situation angepasst, sondern Daten werden so beschnitten, dass sie zur Situation passen. Dabei ist es auch bisher nicht angedacht, den Kontext für verschiedene Programme zu nutzen, sondern hier wird der Kontext nur für die Lösung des speziellen Datenproblems genutzt. Beim Projekt des Weckers ist der Wecker nur ein Programm von vielen in einem intelligenten Haus. Daher ist beim Wecker der Kontext auch nicht nur auf die Funktionen des

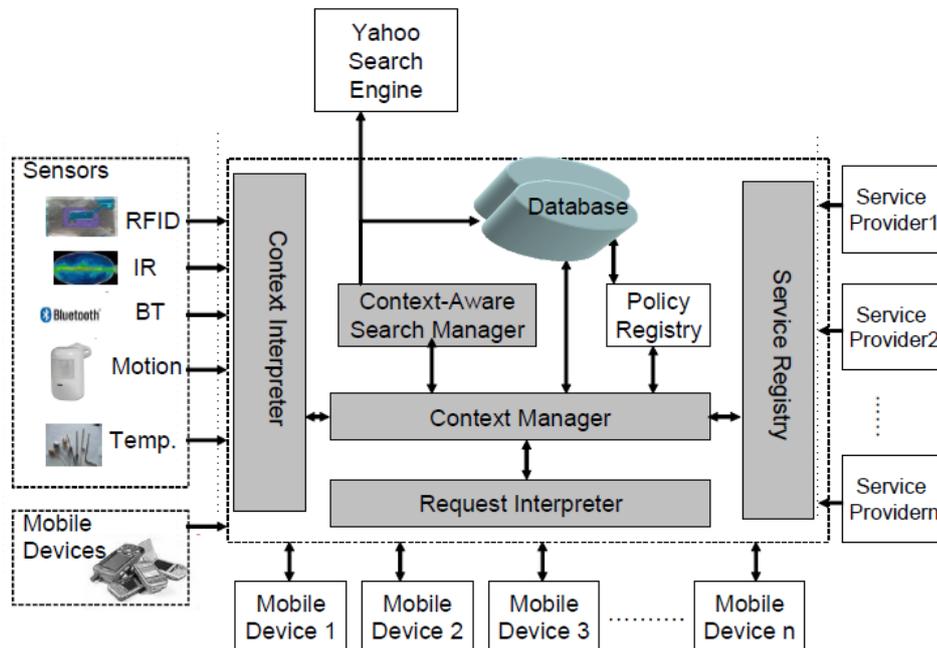


Abbildung 2.2: Context-Aware Server Architecture

Weckers zugeschnitten, sondern allgemein gehalten. Außerdem wird der Kontext im intelligenten Haus auch über eine Blackboard-Architektur [8] verteilt, die die Kopplung zwischen der Kontext-Erzeugung und der Kontext-Nutzung möglichst lose zu gestalten.

2.2 Projekt: Search Application

Dieses ist ein Projekt des „College of Engineering and Computing“ der „Florida International University“. In dem Projekt für eine „Client-Server Architecture for Context-Aware Search Application“ [6] sollen Suchanfragen von mobilen Endgeräten verbessert werden. Dabei ist die Idee, durch Kontextinformationen die Bedürfnisse des Anwenders zu erkennen und dadurch dann die Ergebnisse der Suchabfrage anzupassen. Bei mobilen Geräten sind die Eingaben für eine Suche laut dem Paper sehr reduziert. Beispielsweise würde man für eine Suche nach einem Geschäft, in dem man Äpfel kaufen kann, einfach nur den Begriff „Apfel“ eingeben. Bei der Architektur des Systems wird auch das mobile Endgerät in die Erfassung und Erstellung des Kontextes miteinbezogen werden. Da die Endgeräte in den letzten Jahren ihre Leistungsfähigkeit steigern konnten sind die mobilen Endgeräte in diesem Projekt ein entscheidender Faktor. Im mobilen Einsatz bezieht sich der Kontext auch auf das Umfeld des Anwenders und das, was er im Moment der Suchanfrage gerade vor hat. Daher

werden verschiedene Informationen auf den mobilen Geräten nicht nur gesammelt, sondern auch bearbeitet. Zusammen mit der Suchanfrage werden dann diese Informationen an den Server weitergegeben. Wie in Abbildung 2.2 zu sehen ist, werden mit diesen Informationen die Suchanfragen über den „Context-Aware Search Manager“ weiter bearbeitet. Die Datenbank, welche in der Abbildung zu sehen ist, wird zur Leistungssteigerung eingesetzt, um redundante Suchen schneller bearbeiten zu können.

2.2.1 Wo liegt der Fokus?

In diesem Projekt wird eine Client-Server Architektur vorgestellt. Der Fokus liegt in diesem Projekt bei der besonderen Bedeutung des Clients für die Erstellung des Kontextes. Neben den Verschiedenen Sensoren, die dem Server zur Verfügung stehen, um den Kontext zu erstellen, werden von dem mobilen Endgerät diverse Informationen gesammelt und vorverarbeitet. Dafür gibt es auf der Clientseite der Anwendung eine Software, die diese Vorverarbeitung übernimmt. Durch die Vorverarbeitung können die Suchanfragen nicht nur verbessert werden, sondern die Ergebnisse können auch an die Bedürfnisse des Anwenders angepasst werden. Je nach dem was gerade für den Anwender wichtig ist und was er gerade vorhat, kann so für den gleichen Suchbegriff ein anderes, dem Kontext angepasstes, Ergebnis geliefert werden.

2.2.2 Wie wird der Kontext erstellt und verteilt?

Der Kontext wird im Server erstellt, wie in Abbildung 2.2 zu sehen ist. Die Daten, aus denen der Kontext erstellt wird, stammen aber aus unterschiedlichen Quellen. Wie oben schon erwähnt, werden die Sensoren der Umgebung genutzt, die nichts mit der Suchanfrage oder dem Anwender zu tun haben. Zum anderen werden Informationen aus der Client Software genutzt um den Kontext zu erstellen. Durch diesen Mechanismus ist es dem Anwender auch möglich, seine Bedürfnisse in das System einfließen zu lassen da die Informationen aus dem mobilen Gerät personenbezogene Daten enthalten können. Genutzt wird der Kontext hier ausschließlich im Server, um die Ergebnisse der Suchanfrage anzupassen. Im Client kommt der Kontext nicht zum Einsatz.

2.2.3 Was wird mit dem Kontext gemacht?

Der Kontext wird eingesetzt, um die Suchbegriffe zu deuten und die Suchergebnisse einzuschränken. Dabei wird der Kontext wie eine Ontologie eingesetzt. In dem Paper [6] wird das

Beispiel „appel“ genannt. Hier kann sich der Suchbegriff auf das Obst oder auf die Firma beziehen. Welche Bedeutung gemeint ist, wird durch den Kontext festgestellt. Wird der Begriff von einer Hausfrau eingegeben, die gerade dabei ist, verschiedene Speisen einzukaufen, wird es sich bei dem Begriff wahrscheinlich um das Obst handeln. Ein Nutzer des iPhone hingegen wird wahrscheinlich an den Produkten der Firma Apple interessiert sein. In dem Paper wird leider der Mechanismus nicht weiter erläutert, mit dem von dem Kontext auf die Bedeutung geschlossen wird.

2.2.4 Vergleich mit dem Projekt „Wecker 2.0“

Beim „Wecker 2.0“ werden nicht nur die Daten für die Erstellung des Kontextes übertragen, sondern der Kontext selbst. Dadurch wird von den Daten abstrahiert. Dieses hat den Vorteil, dass die Systeme sich nicht mehr mit der Erstellung des Kontextes befassen müssen. Sie können sich darauf konzentrieren, was nach dem Auftreten des Kontextes zu tun ist. Andererseits muss der Wecker auch darauf vertrauen, dass der Kontext richtig erstellt wurde und dass mit dem Kontext das gleiche gemeint ist, was auch die erstellende Einheit mit dem Kontext meint. Beispielsweise könnte eine Person morgens gerade Brötchen und Honig auf den Küchentisch stellen. Dieses könnte der Context-Interpreter als „Frühstücken“ interpretieren. Das muss der Wecker dann genau so sehen, dass das Auf- und Abdecken des Geschirrs zum Frühstücken dazu gehört.

2.3 Projekt: Assistive Meeting

Dieses ist ein Projekt von der Tsinghua Universität von Beijing, China. Peng Dai und Guangyou Xu verfolgen hier das Ziel, Kontext in einer Konferenz zu nutzen, um den Konferenzraum mit Intelligenz auszustatten. In einer Konferenz interagieren verschiedene Menschen miteinander. Dadurch entstehen ständig neue Situationen, die auch eine andere Nutzung des Konferenzraumes mit sich bringen. Diese Situationen könnten beispielsweise eine Präsentation (eine Person spricht), eine Diskussion (alle Teilnehmer sprechen) oder vielleicht eine Pause sein (Unterhaltungen sind nicht relevant). In diesem Projekt „Context-Aware Computing for Assistive Meeting System“ [4] wurde ein Framework entwickelt, mit dem der Konferenzraum automatisch auf diese Änderung reagieren kann. Hierfür wurde ein hierarchisches Kontextschema erstellt, mit dem die Situation der Konferenz abgebildet wird. Je nachdem welche Situation durch das Modell beschrieben wird, werden die Geräte im Konferenzraum unterschiedlich gesteuert. Des Weiteren wurde ein neues dynamisches Kontextmodell entwickelt, durch welches eine bessere Kontexterfassung ermöglicht wird. Dynamisch bedeutet hier, dass die bei der Erstellung des Kontextes die Informationen des aktuellen Kontext genutzt werden.

2.3.1 Wo liegt der Fokus?

In diesem Projekt wird sich darauf konzentriert, den Kontext nicht einfach statisch zu generieren, sondern ihn mit einem dynamischen System zu erzeugen. In diesem System werden die Sensoren und die „Kontext Erzeugung“ durch den Kontext selbst beeinflusst. Durch dieses Vorgehen kann bei der Kontexterzeugung viel gezielter vorgegangen werden. Durch die Dynamik ist es möglich den Kontext auf verschiedenen Ebenen besser und genauer zu erfassen und mögliche Unsicherheiten besser auszugrenzen.

2.3.2 Wie wird der Kontext erstellt und verteilt?

Der Kontext wird durch Sensoren erfasst, die in dem Konferenzraum verteilt sind. Zu diesen Sensoren zählen unter anderem auch Kameras, die Bilder liefern. Durch eine Analyse der Kamerabilder entstehen so Informationen, die zum Teil einen höheren Level haben als rohe Sensordaten. Beispielsweise werden die Bilder dahingehend analysiert, ob auf ihnen eine Person zu erkennen ist. Der Kontext wird dann aus den Daten in einem Modul generiert. Bei der Erstellung des Kontextes gibt es verschiedene Hierarchie-Ebenen. Die Ebenen beschreiben die Situation in der Konferenz und gehen vom allgemeinen (Gruppenkonferenz) zum speziellen (A spricht zu B). Der Kontext wird in diesem Projekt direkt für den intelligenten Konferenzraum genutzt. Allerdings werden mit den Kontextinformationen neben dem Konferenzraum auch die Sensoren und die Kontext-Erkennung selbst gesteuert, so dass der Kontext an verschiedene Stellen übermittelt wird. Diese Module gehören aber alle zum selben System.

2.3.3 Was wird mit dem Kontext gemacht?

Durch den Kontext werden die Einheiten, beispielsweise Kamera oder Mikrofon, in dem Konferenzzimmer gesteuert. Diese richten ihre Funktionen nach den Anforderungen aus, die der aktuelle Kontext vorschreibt. Beispielsweise werden die Mikrofone auf den Teilnehmersitzen ausgeschaltet, wenn gerade ein Vortrag läuft. Des Weiteren wird der Kontext in einer dynamischen Schleife eingesetzt, wie in Abbildung: 2.3 zu sehen ist. Das bedeutet, dass der Kontext zum Erstellen des Kontextes herangezogen wird. Mit dem aktuellen Kontext werden die Sensoren zur Kontexterfassung (Multi-modal Signal processing) selektiert und fokussiert. Wenn beispielsweise gerade ein Vortrag läuft, dann ist es wichtig zu erkennen, wann der Vortrag unterbrochen wird und es eine Pause gibt. Hingegen muss nicht überwacht werden, welcher Teilnehmer gerade Spricht. Dieses wäre aber bei einer Gruppendiskussion notwendig. Außerdem fließt der aktuelle Kontext in die Kontexterfassung (Context Sensing Engine) mit ein. Wie in dem vorherigen Beispiel beschrieben, kann es in einem Vortrag nur bestimmte

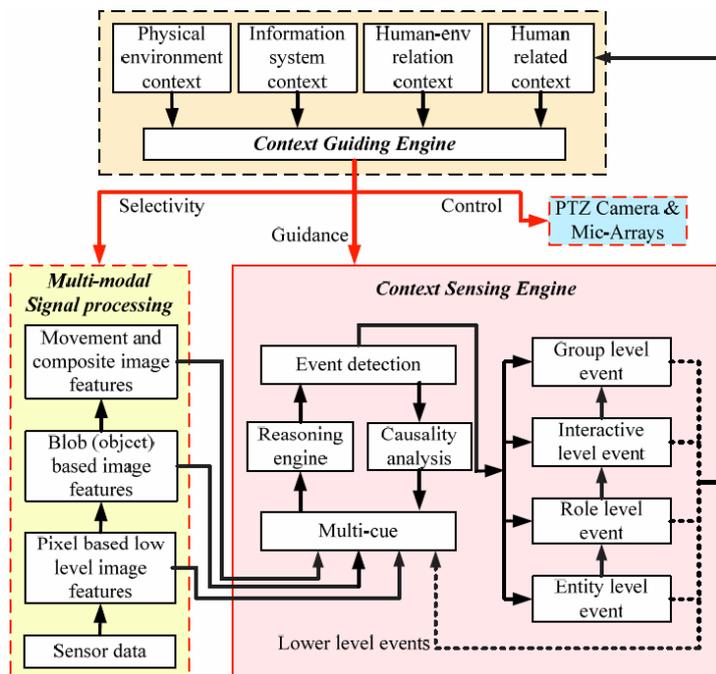


Abbildung 2.3: Dynamik Context Model

Kontextwechsel geben und die Kontexterfassung kann mit den zusätzlichen Informationen die Auswertung fokussieren.

2.3.4 Vergleich mit dem Projekt „Wecker 2.0“

Die Repräsentation wird hier mit dem Fokus auf die Aktionen zwischen den Personen dargestellt. Beim Wecker liegt der Fokus mehr auf dem Bewohner, daher unterscheiden sich die Darstellungsformen des Kontextes in diesen beiden Projekten. Wie auch beim Wecker werden hier die Funktionen durch den Kontext angepasst. Beim Wecker werden aber die Funktionen eines Programmes der Situation angepasst, wohingegen bei diesem Projekt Geräte in einem Raum gesteuert werden. In diesem Projekt wird der Kontext daneben für eine Rückkopplung genutzt. Beim „Wecker 2.0“ ist bisher keine Rückkopplung des Kontextes vorgesehen, durch den die Kontextererkennung verbessert werden könnte. Dieses ist aber eine sehr interessante Idee, mit der verschiedene Probleme des „Context Aware Programming“ reduziert oder abgemildert werden können. Daher sollte diese Dynamik vielleicht später beim „Wecker 2.0“ eingesetzt werden.

3 Fazit

3.1 Zusammenfassung

Kontext kann auf viele verschiedene Weisen eingesetzt werden. Kontext wird im Allgemeinen dazu genutzt um etwas anzupassen oder einzuschränken. Was hierbei aber angepasst werden soll, ist von diesem Konzept nicht abhängig. Bei dem Wecker 2.0 werden die Funktionen angepasst; bei der ADDICT wird die Suchanfrage der Situation angepasst und bei der Searchapplication ist es das Ergebnis der Suchanfrage. Der Begriff aus dem Projekt ADDICT „Tailoring“ (übersetzt: schneidern) trifft es recht genau. Man möchte aus einer Menge etwas heraus trennen und der Kontext gibt dafür das Muster vor. Bei der Menge kann es sich um Daten oder auch um Funktionen handeln. Der Kontext beschreibt, was gerade notwendig ist. Hierfür wird eine Beschreibung der aktuellen Situation genutzt, um Rückschlüsse auf „das Notwendige“ machen zu können.

3.2 Ausblick

Ein wesentlicher Teil des „Weckers 2.0“, ist die Kontext Erkennung und die Kontext Interpretation. Diese gilt es in den nächsten Schritten umzusetzen. Die Konzepte von Ontologien und Context Awareness sind außerdem sehr verwandt, wie im Projekt Search Application (Abschnitt: 2.2) zu sehen war. Dieses ist ein grundlegender Schritt und daher das erste, das für den Wecker geklärt werden sollte. Es sollte bei der Definition des Kontextes nicht darauf geachtet werden, wofür der Kontext eingesetzt werden soll, da es eine Vielzahl von Einsatzmöglichkeiten für den Kontext gibt. Eine Darstellung des Kontextes wie im Projekt ADDICT (Abschnitt: 2.1) könnte für den Wecker auch sinnvoll sein, da sie sehr flexibel ist. Um die Probleme der Kontextinterpretation zu begegnen, könnten die Ideen aus dem dritten Projekt einfließen (Abschnitt: 2.3). Dieses ist aber erst in den weiterführenden Schritten sinnvoll.

Literaturverzeichnis

- [1] BOLCHINI, C. ; CURINO, C. A. ; ORSI, G. ; QUINTARELLI, E. ; SCHREIBER, F. A. ; TANCA, L. : CADD: a tool for context modeling and data tailoring. In: *Proc. IEEE Intl. Conf. on Mobile Data Management (MDM)*, 2007, S. 221–223
- [2] BOLCHINI, C. ; QUINTARELLI, E. ; ROSSATO, R. ; TANCA, L. : Using Context for the Extraction of Relational Views. In: *Proc. 6th Intl and Interdisciplinary Conference, (CONTEXT), LNAI 4635*, 2007, S. 108–121
- [3] CURINO, C. ; ORSI, G. ; PANIGATI, E. ; TANCA, L. : Accessing and Documenting Relational Databases through OWL Ontologies. In: *Proc. Intl Conf. Flexible Query Answering Systems, LNCS 5822*, 2009, S. 431–442
- [4] DAI, P. ; XU, G. : Context-aware computing for assistive meeting system. In: *PETRA '08: Proceedings of the 1st international conference on Pervasive Technologies Related to Assistive Environments*. New York, NY, USA : ACM, 2008. – ISBN 978–1–60558–067–8, S. 1–7
- [5] ELLENBERG, J. : Ein Wecker in einem ubicom Haus / HAW Hamburg. Version:2010. <http://users.informatik.haw-hamburg.de/~ubicomp/projekte/master09-10-aw1/Ellenberg/bericht.pdf>. HAW Hamburg. – Forschungsbericht
- [6] GUI, F. ; GUILLEN, M. ; RISHE, N. ; BARRETO, A. ; ANDRIAN, J. ; ADJOUADI, M. : A Client-Server Architecture for Context-Aware Search Application. In: *Network-Based Information Systems, International Conference on 0* (2009), S. 539–546. <http://dx.doi.org/10.1109/NBiS.2009.75>. – DOI 10.1109/NBiS.2009.75. ISBN 978–0–7695–3767–2
- [7] MIELE, A. ; QUINTARELLI, E. ; TANCA, L. : A methodology for preference-based personalization of contextual data. In: *Proc. ACM 12th Int. Conf. on Extending Database Technology*, 2009. – ISBN 978–1–60558–422–5, S. 287–298
- [8] OTTO, K. ; VOSKUH, S. : Entwicklung einer Architektur für das Livingplace Hamburg / HAW Hamburg. Version:2010. <http://users.informatik.haw-hamburg.de/~ubicomp/projekte/master2010-proj/otto-voskuhl.pdf>. HAW Hamburg. – Forschungsbericht

- [9] SCHREIBER, F. A. ; TANCA, L. ; CELENTANO, A. ; CELENTANO, A. ; CA, U. ; VENEZIA, F. : Requirements for Context-Dependent Mobile Access to Information Services. In: *In Proc. 10th Int. Workshop on Multimedia Systems (MIS, 2004, S. 60–65*
- [10] WEISER, M. : The computer for the 21st century. In: *SIGMOBILE Mob. Comput. Commun. Rev.* 3 (1999), Nr. 3, S. 3–11. <http://dx.doi.org/10.1145/329124.329126>. – DOI 10.1145/329124.329126. – ISSN 1559–1662

Bildquellen

Die Quellen der Bilder sind:

- Abbildung 2.1 aus dem Paper [1]
- Abbildung 2.2 aus dem Paper [6]
- Abbildung 2.3 aus dem Paper [4]