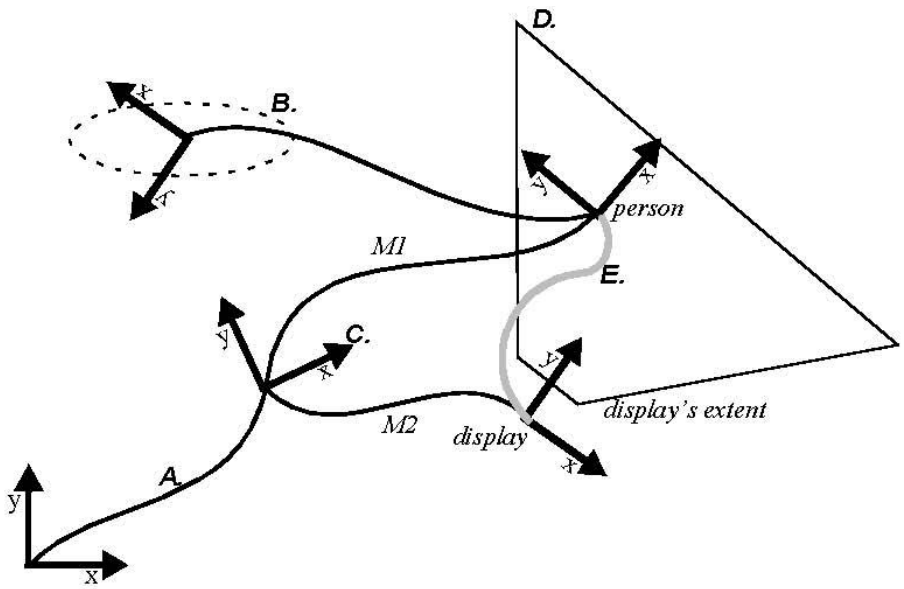


Abbildung 4.4: Das EasyLiving Geometric Model (Bild: Microsoft Research).

4.1.2 Philips HomeLab

Das HomeLab ist eine Initiative von Philips Research, welche als Testbett und Integrationsplattform für diverse Forschungs- und Entwicklungsaktivitäten dient. Ziel dieser Initiative ist es, die Auswirkungen von Ambient Intelligence auf das Alltagsleben des Menschen sowie das Interaktionsverhalten des Nutzers zu untersuchen [Res03]. Das HomeLab konzentriert sich hauptsächlich auf consumer electronics und Home-Entertainment-Anwendungen. HomeLab soll Philips erlauben, praktische, psychologische und soziale Implikationen von Ambient Intelligence zu erforschen „to better understand the moments people touch technology“ [Egg02, DYN04a].

HomeLab ist ein zweistöckiges Familienhaus, ausgestattet mit Ambient Intelligence Technologien, das von freiwilligen Drittpersonen bewohnt werden kann (siehe Abb. 4.5). Ein breites Spektrum von Technologien und Konzepten, die von Philips entwickelt wurden, können im HomeLab integriert werden, um dann im Rahmen eines Gesamtszenarios unter real-life conditions evaluiert zu werden. Eine Besonderheit von HomeLab ist, dass es „more a home that is also a Lab“ ist (siehe Abb. 4.6), wodurch dann eine angenähert echte Alltagsumgebung geschaffen wird. Das Lab umfasst ein Wohnzimmer, eine Küche, zwei Schlafzimmer, ein Bad, sowie einen „study“-Raum mit zwei observation rooms (siehe Abb. 4.5 oben rechts) und 20 Monitoren, von denen aus das ganze Haus durch one-way mirrors und durch Videokameras observiert werden kann. Über das Kontrollsystem für die Kameras und A/V-Signalverarbeitung können multidisziplinäre Forschungsgruppen, welche die Bewohner beobachten, die Videoausschnitte annotieren.

Die Geräte (Fernsehgerät, Lichtschalter, etc.) sind im Hintergrund verschwunden. Die Funktionalitäten (z. B. Film anschauen) sind aber in allen Räumen verfügbar. Unter realen Lebensbedingungen werden Interaktionsverhalten der (echten) Bewohner mit der (intelligenten) Umgebung, Machbarkeit der Technologien sowie das Vertrauen der Benutzer in Ambient-Intelligence-Anwendungen durch Beobachtung un-

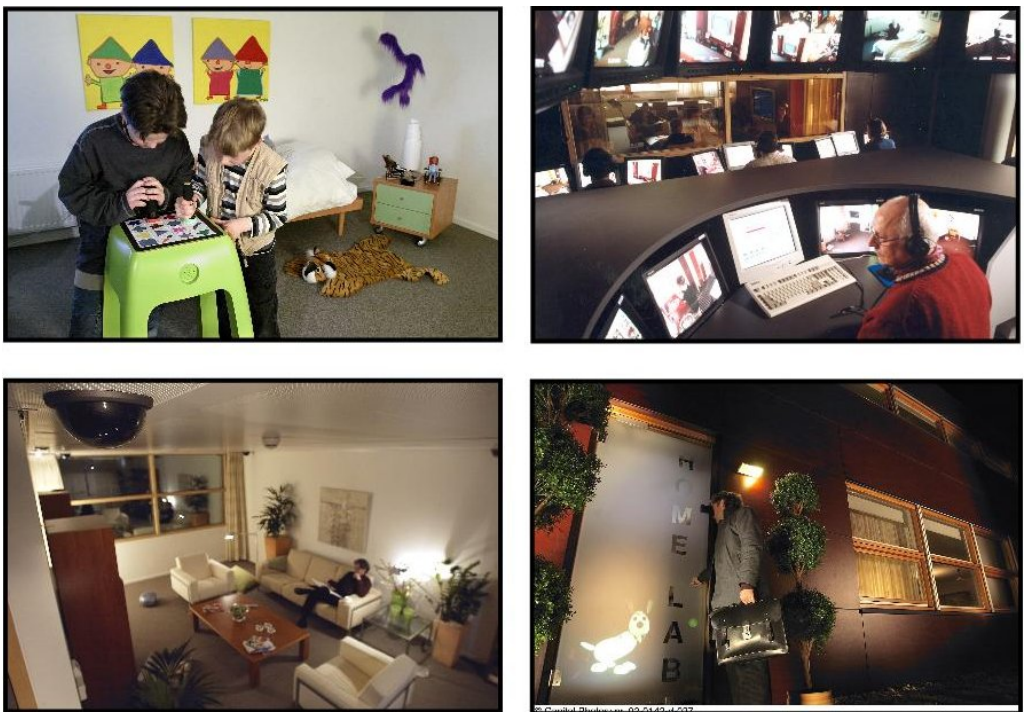


Abbildung 4.5: HomeLab bietet Real-life-Bedingungen für die Ambient-Intelligence-Forschung (Bild: Philips Research).

tersucht. Die Umgebung ist mehrbenutzerfähig und kann personalisiert werden. Sie kann die Menschen wieder erkennen (Gedächtnis!) und von deren Verhalten lernen.

Anwendungsdomäne

Seinen Bewohnern bietet HomeLab einige Anwendungen, die nicht in erster Linie für die Feasibility Studie selbst oder Haussteuerung (Positionierungssystem, Videoüberwachungssysteme, Multimodale Interaktion, Lichtsteuerung, Power Management, etc.) benötigt werden.

Folgendes Szenario soll die Anwendunsdomäne von HomeLab etwas näher aufzeigen. *„Bob and Linda have been invited to HomeLab to experience how a future living room might support the planning of their yearly summer holiday. After a tour around HomeLab, which surprisingly well matched the picture of what they would consider an ordinary home, they settle at the dinner table and start to browse the holiday brochures lying around. After a while, they begin to realize that the room seems to listen to their conversations, as pictures and movie clips started to appear on the wall that matches the actual holiday destinations they are discussing. Things get even more interesting when the colour of the lighting of the environment adjusts to the Tuscany pictures they are looking at. The experience is complete when they notice the soft Italian music that was playing in the background. After Bob and Linda selected their dream holiday, they join the research team to discuss the way they experienced the Ambient Intelligence room. They both tell the team they are impressed by the smartness of the room and the smooth way interesting media were displayed in the room only when it seemed appropriate and meaningful. At the end of their visit to HomeLab, Bob and Linda are given a tour backstage.“*[Egg02]

Abbildung 4.6: HomeLab Anwendungen (Bild: Philips Research)

Systemarchitektur

Das WWICE (Window on the World of Information, Communication and Entertainment) von Philips bildet die Basis für die verteilten Anwendungen. Ziel von WWICE ist es, Middleware und Infrastruktur für intuitiv zugreifbare und bedienbare A/V-Geräte (siehe Abb. 4.7) und Anwendungen bereitzustellen.

WWICE fokussiert auf In-Home Digital Networks (IHDN) Multimedia Anwendungen. Es erlaubt, Geräte verschiedener Hersteller miteinander zu kombinieren, sowie geräte-, orts- und netzwerkunabhängige Anwendungen zu definieren. WWICE benötigt keinen zentralen Server und erlaubt dem Programmierer sehr leicht seine IHDN-Anwendungen um neue Geräte und Funktionalitäten zu erweitern. Hierzu muss er lediglich die Beschreibung der Activity erweitern, welcher nur den benötigten Graphen und die im Falle von FollowMe-Operationen zu verlagernden Teile (Subunits) definiert.

Folgende Eigenschaften besitzt WWICE [BBE+00, DYN04a]:

Dynamik Durch die offenen Schnittstellen können neue Geräte und Funktionalitäten *graduell* und dynamisch im Gesamtsystem integriert werden.

Ubiquitärer Datenzugriff Der Zugriff auf Multimedia-Inhalte ist unabhängig vom physischen Speicherort, benutzten Gerät und dem Ort des Benutzers möglich.

Netzwerktransparenz Jegliches Detail über Netzwerktopology, device allocation, Verfügbarkeit von Ressourcen sowie die Umwandlung von Medienformaten ist für den Nutzer transparent. Dies wird von [BBE+00] in folgender Art angedeu-

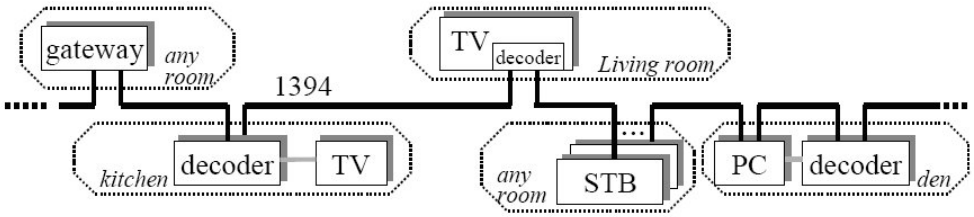


Abbildung 4.7: Geräte in einem In-Home Digital Networks (IHDN) (Bild: Philips Research)

tet: *As long as suitable devices and the required bandwidth suitable for a task are available somewhere, they will be found and allocated by the system.*

Ortsunabhängigkeit Die Dienste sind auf jedem Gerät und an jedem Ort verfügbar: *The user can move around the house and take with him anything he is currently using the system for (as long as the technical preconditions are fulfilled).*

Einheitliche Interaktion Der Nutzer findet für alle Umgebungen und Dienste ein einheitliches Bedienkonzept vor. *The user is presented a coherent user interface on all environments. While different capabilities of these environments might imply a different look, the UI concepts have to remain the same to flatten the learning curve for the user.*

Eine Schlüsselkomponente für die Realisierung von ortsunabhängigen Multimedia-Anwendungen (z. B. Follow-Me) ist der *Graph Mapper*, welcher die Elemente einer als Graph beschriebenen *Activity* in physische Units und Subunits (z. B. Fernsehgerät, Tuner) und Netzwerkpfade unterteilt. Hierzu muss der GraphMapper Geräte finden, die zu der in der Activity beschriebenen Subunit passen und verfügbar sind. Der Graph Mapper erlaubt einige Auswahlkriterien, um für eine Subunit ein Gerät zu finden. Dazu gehören Ort, Content, Quality-of-Service, sowie unterstütztes Streaming-Format. Um beispielsweise im Falle einer *Follow-Me* eine Activity Relocation durchzuführen, muss lediglich der Graph erweitert werden. Möchte man z. B. einen Film in einem anderen Raum weiterverfolgen, so muss der Graph lediglich um neue Subunits, also decoder und screen erweitert werden.

Context-Model und Sensorik

Das Schlüsselkonzept von WWICE ist die Einführung von *Activity*-, *Content*- und *Place*-Konzepten (siehe Abb. 4.8). Hierbei geht es darum, eine Aufgabe oder Aktivität von den für deren Erfüllung notwendigen Content, Methoden und Geräten zu trennen. Dem Benutzer soll transparent sein, wie eine Aktivität und mittels welcher Ressourcen sie durchgeführt wird. Er soll sich lediglich überlegen, was er wo machen möchte [BBE+00].

Zur Kontexterfassung und Beobachtung des Nutzers werden über 34 im ganzen Haus verteilte Kameras und Mikrofone verwendet, wobei diese möglichst unsichtbar eingesetzt werden. Video- und HF-Anlagen werden zum Identifizieren und Verfolgen von Personen benutzt. Die Umgebung kann über Sprach- und Gestenerkennung kontrolliert werden. Power-Control-Systeme erlauben die Fernsteuerung von Licht.

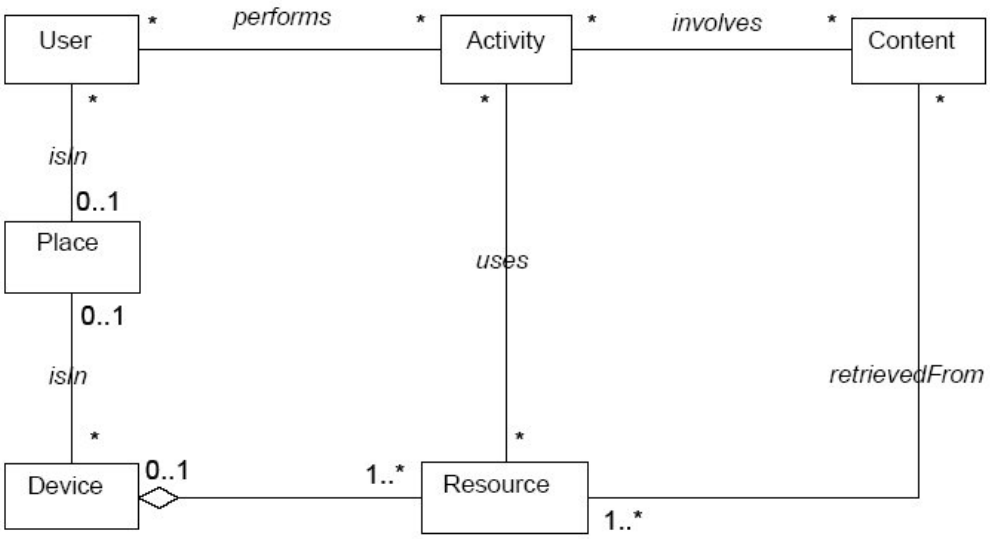


Abbildung 4.8: WWICE Domain Model (Bild: Philips Research)

4.1.3 FORSIP SIKOWO

SIKOWO [FOR04a] ist ein Projekt des Bayerischen Forschungsverbundes für Situierung, Individualisierung und Personalisierung in der Mensch-Maschine-Interaktion (FORSIP) und behandelt die Entwicklung einer situationsbezogenen und individuell abgestimmten Interaktion mit Wohnkomfortregelsystemen [FOR04b].

Zusammen mit den Teilprojekten SIPaDIM [FOR04b] und SIPBILD [FOR04c] stellt SIKOWO ein ganzheitliches System zur intelligenten Hausinstrumentierung dar. So kann das intelligente Haus den Wohnkomfort aktiv und interaktiv regeln und sich automatisch auf die Bedürfnisse des Bewohners einstellen.

Durch SIKOWO soll unter anderem die passive, anwesenheitsbezogene Erfassung der Situation in Wohnräumen ermöglicht werden. Die aktive, bedarfsgeführte Steuerung des Regelsystems (z. B. zur Temperaturregulierung) sowie die Steuerung der Jalousien und Beleuchtung der einzelnen Räume sollen den Wohnkomfort erhöhen und den Energieverbrauch minimieren.
So sind Helligkeit, Lüftung und Temperatur Indikatoren, die im Wohnraum für Behaglichkeit und das Gefühl des Zuhause-Seins sorgen. SIKOWO „sieht“ und „versteht“ durch Kontexterfassung, was im Raum vorgeht, passt Helligkeit und andere Parameter dementsprechend an und garantiert dadurch automatisch den höchsten Wohnkomfort, der für die entsprechende Situation und einen bestimmten Kontext möglich ist.

Anwendungsdomäne

Das SIKOWO-Projekt adressiert die Domäne des Wohnens. Folgendes Szenario soll die Anwendungsdomäne des Projektes näher erläutern.

Im Wohnzimmer eines mit einem SIKOWO-System eingerichteten Hauses ist es dunkel und kühl. Es befindet sich niemand in diesem Zimmer. Eine Person kommt nach Hause und geht ins Wohnzimmer. Kurz bevor sie das Zimmer betritt, geht automatisch das Deckenlicht an und die Heizung beginnt zu arbeiten. Die Person

tritt an den Schreibtisch und setzt sich an den Computer. Automatisch geht die Arbeitsleuchte über dem Schreibtisch an. Nachdem sie eine Weile am Schreibtisch gearbeitet hat, ohne sich wegzubewegen, wird das Deckenlicht gedimmt, die Arbeitsleuchte bleibt hell. Die Heizung stellt die Temperatur automatisch noch ein wenig höher, da die Person sich nicht bewegt und deshalb eine höhere Raumtemperatur zum Wohlfühlen braucht.

Systemarchitektur

SIKOWO beinhaltet sowohl implizite (situationsabhängige) als auch explizite Bedienparadigmen. Das Projekt möchte die Verwendung von zusätzlichen Bedienelementen zur Haussteuerung, wie TabletPC oder Handhelds reduzieren und nur auf implizite, situationsgesteuerte Interaktionsmöglichkeiten zurückgreifen.

Zu der wichtigsten impliziten Interaktionsmethode gehört die automatische Anpassung von Wohnkomfortparametern (Temperatur, Lüftung und Helligkeit, die mittels entsprechender Sensoren erfasst werden) mit Hilfe einer anwesenheitsbezogenen Erfassung der Situation in den ensprechenden Wohnräumen durch ein automatisches Mehrkamera-Bildverarbeitungssystem.

In der ersten Phase der Kontexterfassung wird versucht, bewegte Objekte wahrzunehmen und deren Aufenthaltsort zu bestimmen, wodurch auch alle für den Wohnkomfort entscheidenden Faktoren, wie z. B. den Zustand der Türen, Fenster oder Jalousien, registriert werden.

Im nächsten Schritt werden die Objekte klassifiziert, d. h. es soll unterschieden werden, ob sich beispielsweise ein Haustier, ein Kind oder ein Erwachsener im Raum aufhält. Auch Gegenstände müssen erkannt werden, so dass Verdeckungen von bewegten Objekten durch diese berücksichtigt werden können.

Die Aktivitätserkennung stellt die dritte Stufe der Kontexterfassung dar. Das System sollte unterscheiden können, ob eine Person zum Beispiel Sport treibt, liest oder schläft.

In der letzten Phase werden die Hausbewohner identifiziert, so dass ihre individuellen Bedürfnisse berücksichtigt werden können.

All diese Daten werden dem intelligenten Wohnkomfortregelsystem übermittelt, das dann die Teilsysteme entsprechend automatisch aktiviert.

Die aktiven Interaktionsmethoden werden durch zwei Teilsysteme von SIKOWO abgedeckt: Das Dialogmanagementsystem SIPaDIM [FOR04b] vermittelt dem Computer die Wünsche des Benutzers durch Moderation des natürlichsprachigen Dialogs zwischen Mensch und Computer. Dieses Teilprojekt entwickelt Verfahren, mit denen vom Benutzer gestellte Aufgaben in Kommandos an die Ausführungskomponente übersetzt werden. Ein Spracherkenner, ein Dialogmanager, eine Komponente zur Benutzermodellierung und eine Textgenerierungskomponente bilden die Hauptbestandteile. Auf diese Weise werden Benutzeräußerungen verarbeitet und an die angebundene Applikation weitergegeben. Der Zustand des gesamten Systems wird dabei stets kontrolliert, die Informationen werden in den Dialogverlauf und zur Formulierung von Mitteilungen an den Benutzer herangezogen [3].

Das zweite Teilprojekt von SIKOWO – das SIPBILD [FOR04c] – dient zur Erkennung von Mimik und Gestik in Videobildfolgen, wodurch die Bedienung von

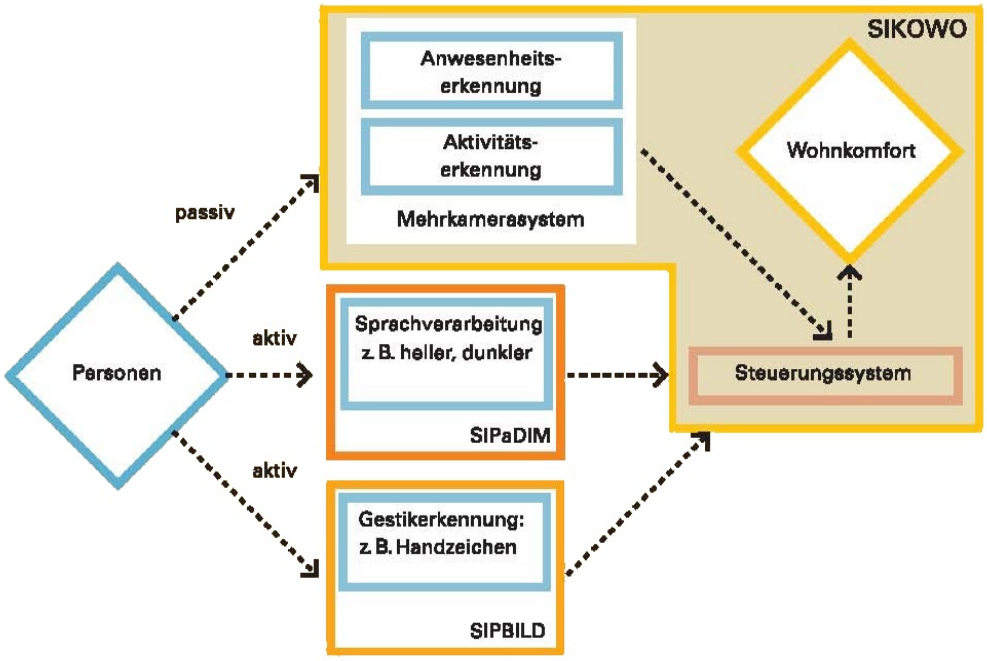


Abbildung 4.9: SIKOWO Architektur (Bild: FORSIP)

SIKOWO sehr intuitiv wird.

So kann der Benutzer den Computer durch Gesten dazu veranlassen, bestimmte Aufgaben zu erledigen, beispielsweise steuert er die Beleuchtung im Wohnraum, indem er Anweisungen in Form von Armbewegungen gibt: Das Bewegen eines Armes nach oben signalisiert dem Rechner z. B. *Helligkeit erhöhen*.

Das Wissen über den Gesamtzustand ist auf verschiedenen Komponenten verteilt und wird zwischen diesen in Form von Nachrichten ausgetauscht. Dazu kommunizieren die Module in einer Logik-Sprache, die in der Lage ist, alle Objekte und Vorgänge in der Domäne zu beschreiben und die leicht an neue Funktionalitäten angepasst werden kann.

Kontextmodell und Sensorik

Zur Personenortung und -identifikation benutzt SIKOWO ein Mehrkamerasystem. Die Gestik-, Mimik- und Helligkeitserkennung wird ebenso über das Kamerasystem realisiert, aus dem auch das systeminterne geometrische Modell des Raumes und der darin enthaltenen Objekte und Geräte erzeugt wird. Zur Spracheingabe und -ausgabe beinhaltet SIKOWO Mikrofone bzw. Lautsprecher, die je nach Kontext einzeln angesteuert werden können.

4.1.4 Fraunhofer inHaus Duisburg

Das Innovationszentrum Intelligentes Haus Duisburg (inHaus) 4.10 strebt eine Vernetzung der auf unterschiedlichen Standards beruhenden Komponenten in einem Haus an. In einer Entwicklungsumgebung werden geräteübergreifende Anwendungen erstellt. Der Benutzer soll eine praktische Unterstützung im Alltag erhalten und kann die Hausfunktionen ortsunabhängig steuern.

Die inHaus-Anlage (siehe Abb. 4.10) besteht aus einem Wohn- und einem Werkstatthaus, welches in einem Doppelhaus in Duisburg untergebracht ist. Das Wohnhaus setzt sich aus einem Wohnlabor für Probewohnphasen, einem Heimbüro für die Telearbeit zu Hause, einem vernetzten Fahrzeug und einem vernetzten Garten zusammen. Die Probewohnphasen werden jeweils sozial- und marktforscherisch begleitet. Das Werkstatthaus besteht aus einem inHaus-Büro, einer Entwicklungs-, Test-, und Wartungswerkstatt, einer Labor-Küche und einem Labor-Bad, einem Teleservice-Arbeitsplatz, der inHaus-Technikzentrale und einem Beratungsarbeitsplatz [Bau03].

Abbildung 4.10: Das inHaus Duisburg bestehend aus Wohn- und Werkstatt (Bild: inHaus).

Anwendungsdomäne

Zielsetzung der neuen Anwendungen ist die praktische Unterstützung der Bewohner im Alltag.

Integration und Vernetzung der einzelnen Geräte erlauben eine Steuerung von jedem Platz im Haus aus, ohne sich direkt beim Gerät befinden zu müssen. Damit ist beispielsweise eine Steuerung aus der Stadt oder vom Urlaubsort aus möglich.

So kann noch von der Autobahn aus der Backofen ausgemacht werden oder kontrolliert werden, ob das Licht im Badezimmer auch wirklich ausgeschaltet worden ist.

Im Mittelpunkt stehen dabei die Möglichkeiten, die sich aus der Verknüpfung verschiedener Gerätefunktionen ergeben. So können Ablaufszenarien aufgestellt werden, die zum Beispiel für die Gebäudesicherheit eine Anwesenheit der sich im Urlaub befindenden Bewohner simulieren. Es können die Jalousien geöffnet und abends unterschiedliche Lampen an- und wieder ausgestellt werden. Ebenfalls kann sich das Haus auf die Rückkehr der Urlauber einrichten, indem beispielsweise im Winter vorher die Heizung wieder hoch geschaltet wird.

Ein zentrales Einsatzfeld ist das Energiemanagement bzw. die Energieeinsparung. Die Temperatur wird z. B. beim Verlassen des Hauses über eine Funktion heruntergestellt. Über die Verbrauchsdaten und die Raumtemperatur kann ermittelt werden, wie effektiv die Heizung arbeitet. Das Sicherheitsmanagement schaltet Elektrogeräte mit Gefahrenpotenzial automatisch beim Verlassen des Hauses ab.

Die nötige Technik verschwindet dabei in den Hintergrund. So wird das Fernsehbild auf das matte Glas der Kleiderschrank-Front projiziert. Gleichzeitig sind von

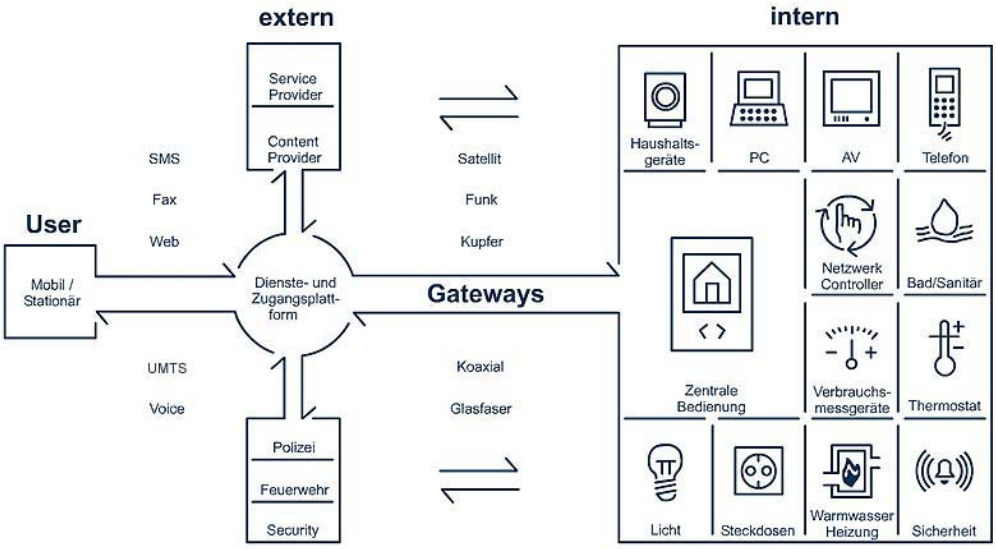


Abbildung 4.11: Interne und externe Vernetzung im inHaus (Bild: inHaus).

dort auch alle anderen Anwendungen wie Internet oder die Steuerung der Waschmaschine abrufbar.

Ein Bewohner kann bei einem Raumwechsel ein angefangenes Video weiterschauen, indem er nur den Ort der Ausgabe umstellt, ohne die Videokassette/DVD mit in den anderen Raum nehmen zu müssen. Gleiches gilt auch für CDs und Musik.

Im Badezimmer kann die Badewanne nach individuell eingestellten Vorgaben gefüllt und die Badewassertemperatur angezeigt werden. Sobald die Wanne gefüllt ist, meldet diese sich. Als weitere Schutzmechanismen agieren Kindersicherung, Überlaufschutz und Verbrühschutz.

Systemarchitektur

Die wichtigste Komponente im inHaus ist das Gateway [Fra04]. Ein Gateway verbindet zwei unterschiedliche Kommunikationssysteme miteinander. Im inHaus übernimmt diese Aufgabe das ISDN-EIB-Gateway, welches aus einem MiniaturNetzwerkcomputer in Verbindung mit einem Echtzeitbetriebssystem besteht. Dieses ist mit der Telehome-Platform der Deutschen Telekom in Berlin verbunden (siehe Abb. 4.11). Die Daten können entweder über ein GSM/GPRS-Modul oder über eine Ethernet-Schnittstelle in bestehende Netzwerke eingespeist werden. Für die Fernsteuerung werden ausschließlich Web-Technologien angewandt.

Eine flexible Middleware – basierend auf Open Services Gateway Initiative (OSGi) – erlaubt eine geräteübergreifende Kommunikation, vor allem innerhalb der Gebäude. Diese Middleware (siehe Abb. 4.12) überbrückt die Schnittstelle zwischen den externen und internen Netzwerkstandards. Dabei steht auf der einen Seite die Anbindung über Kabel, DSL, Stromleitung oder Radioverbindungen. Auf der anderen Seite hingegen wird eine Vielzahl von Standards wie EIB (European Installation Bus), LON (Local Operating Network), Konnex (EIB, BatiBus, EHS (European Home System) zusammen), Firewire, Wireless LAN und Bluetooth der verschiedenen Geräte verwendet. OSGi erlaubt nun geräteübergreifende Applikationen, indem es

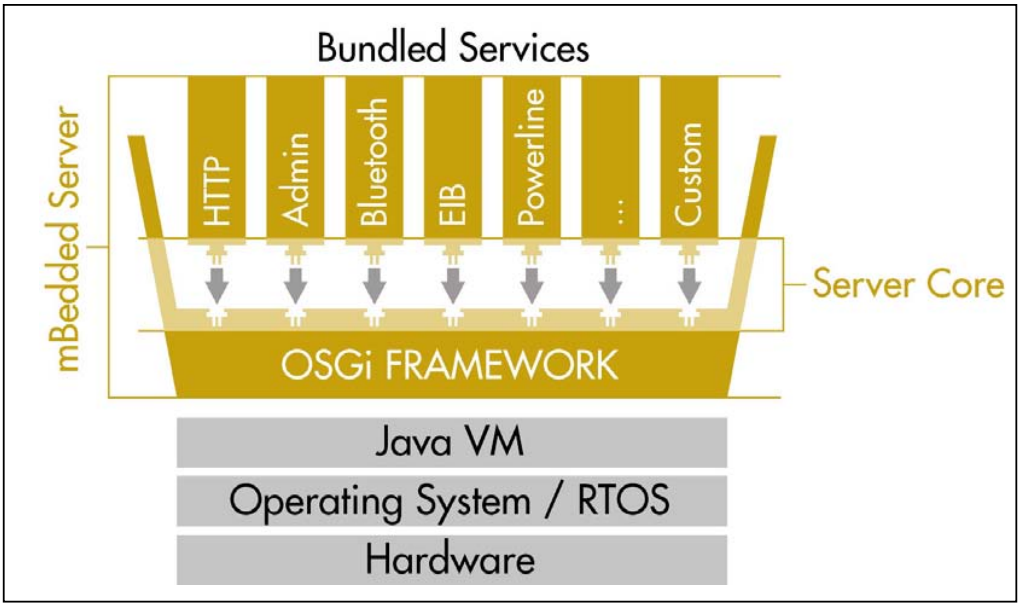


Abbildung 4.12: Die OSGi Middleware erlaubt transparenten Gerätezugriff (Bild: inHaus).

die Anbindung der unterschiedlichen Bussysteme unterstützt.

Im inHaus kann man mehrere übereinanderliegende Schichten unterscheiden (siehe Abb. 4.13). Zuunterst befinden die Sensor-, Aktor- und Kontrollerkomponenten. Darüber befinden sich die Hardware-Plattformen, auf denen sich die Kommunikationsinfrastrukturen, unter anderem EIB, LON, Ethernet/IP, wireless LAN, GSM und UMTS sowie die Betriebssysteme VxWorks und Embedded Linux anschließen. Nun folgt die Middleware mit OSGi, CORBA, Java und Jini. Zum Schluss kommen die Basisdienste und darüber die Systemanwendungen.

Die Unterhaltungs- und Multimediatechnik innerhalb des inHauses basiert auf Breitband-Vernetzungskonzepten wie IEEE 1394 (Fire Wire). Für weitere Datenübertragungen kommt das drahtlose Übertragungssystem Bluetooth sowie das Steuerdatenbussystem EIB (European Installation Bus) zum Einsatz. Ein weiteres System versorgt die einzelnen Räume mit Diensten wie TV oder LAN, indem eine Zentrale mit den Anschlussdosen in den einzelnen Räumen über Kabel inklusive EIB-Leitungen verbunden wird.

Die Interaktion mit dem Haussystem erfolgt integriert und über zentrale Systembedienung. Als Interaktionsgeräte kommen PC, Fernseher, Notepads, PDA und Smartphones mit einheitlicher Bedienlogik zum Einsatz. So können alle Geräte von verschiedenen Quellen aus bedient werden. Ein Beispiel wäre das Einschalten des Videorekorders mittels PDA, wenn man sich beispielsweise gerade in der Küche befindet. Die direkte Bedienung eines Gerätes soll damit aber nicht vollständig wegfallen, sondern lediglich ergänzt werden.

Die Bedienoberflächen der einzelnen Geräte bestehen aus HTML-Seiten und Java-Applets. Diese werden über einen zentralen, im Haus-Gateway eingebetetten Web Server abgerufen. Damit ist der Zugriff sowie die Steuerung des Residential Gateways neben dem Telefon auch über das Handy oder einen Internetbrowser möglich.

Eine situationsabhängige pro-aktive (implizite) Interaktion gehört nicht zum Konzept des inHauses Duisburg.

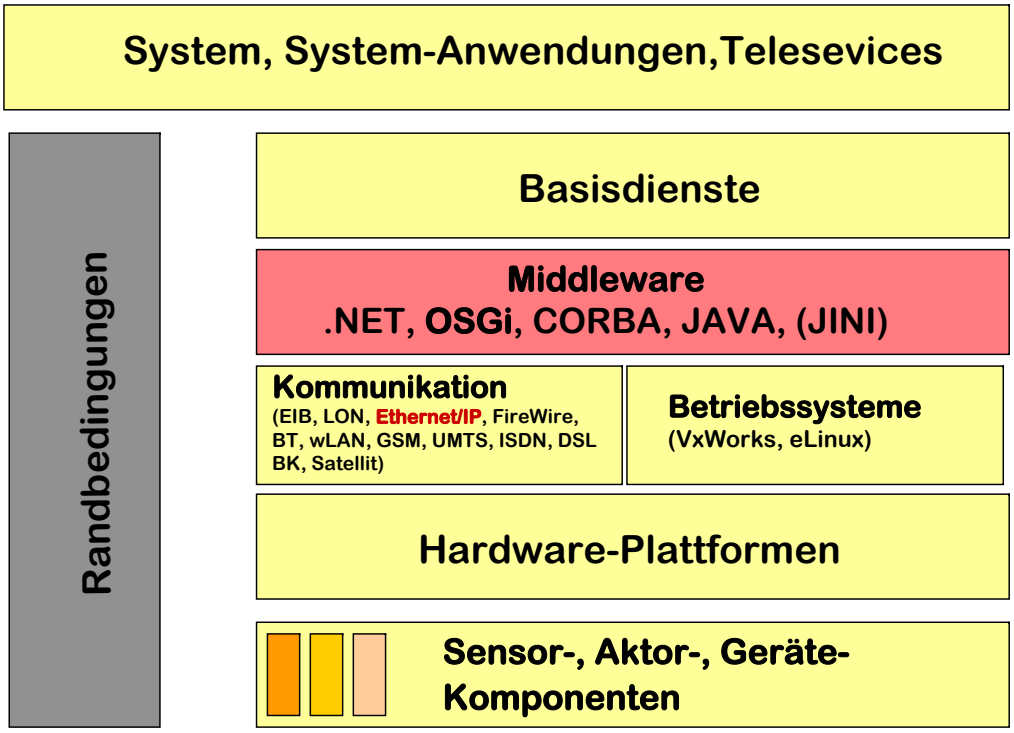


Abbildung 4.13: InHaus-Architektur (Bild: inHaus).

4.1.5 Funktionales Referenzmodell

Durch die Analyse der vorgestellten Projekte lassen sich AmI-Systembausteine und damit ein *Quasi-Referenzmodell* für AmI erkennen (siehe Abb. 4.14). Demnach lassen sich in den betrachteten Projekten (HomeLab, EasyLiving, inHaus Duisburg, SIKOWO) folgende Entitäten und funktionalen Systembausteine zusammenfassen.

Environment Control

Environment Control erlaubt dem Benutzer, mit seiner Umgebung sowie mit anderen Geräten, die ihn umgeben, zu interagieren. Um eine pro-aktive, situationsgesteuerte Unterstützung des Nutzers zu ermöglichen bzw. um eine Aufgabe in fremden Umgebungen durchzuführen (z. B. eine Multimedia-Präsentation abzuspielen), werden verfügbare Geräte aufgespürt (Device Discovery), organisiert und vernetzt. Hierzu wird für eine bestimmte Aufgabe das am besten geeignete Gerät ausgewählt. Kann eine Aufgabe nicht von einem einzigen Gerät durchgeführt werden, so müssen mehrere Geräte sich zu einer Geräte-Komposition zusammenfügen, miteinander kooperieren und eine Strategie entwickeln, um diese Aufgabe optimal durchzuführen.

Über Universal Environment Control wird beispielsweise in einem Konferenzraum das Licht eingedimmt, die Rollläden abgesenkt und eine PPT-Datei auf einen Projektor gesendet, um eine Präsentation optimal zu unterstützen. Über diese Komponente kann der Benutzer auch seinen Fernsehen oder ein komplettes Haus steuern.

Wichtige Merkmale von Universal Environment Control sind, dass

- in einem Raum die Position von Geräten und die angebotenen Dienste dynamisch erkannt werden, auch wenn die Geräte unsichtbar im Raum integriert

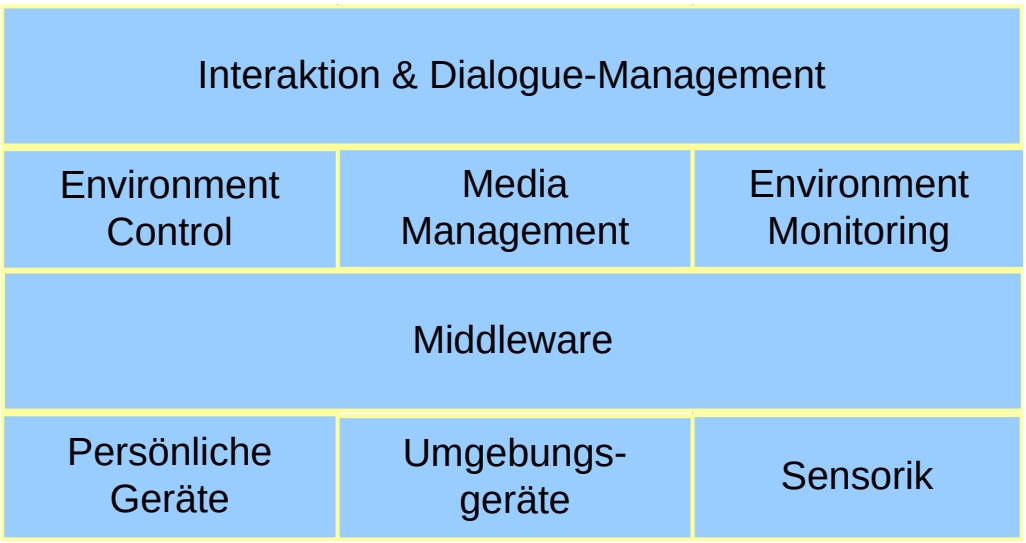


Abbildung 4.14: Funktionales Referenzmodell.

sind.

- Environment Control auch in *fremden Umgebungen* funktioniert. So können beispielsweise Licht und Fernsehgerät sowohl Zuhause, als auch im Hotelzimmer gesteuert werden. Dabei findet der Benutzer jedoch immer seine gewohnte, *personalisierte* User Interface vor.
- neue Geräte, die *ad-hoc* zu einer Umgebung dazu gekommen sind, *dynamisch* erkannt und unterstützt werden. Wenn z. B. ein DVD-Brenner verfügbar wird, erlaubt die Environment Control, Fernsehsendungen auf DVD aufzuzeichnen, ohne dass der Benutzer irgendetwas einstellen oder konfigurieren muss.
- für die Durchführung einer Aufgabe das beste Gerät dynamisch ausgewählt wird bzw. ein Team von (zielbasiert) kooperierenden Geräten zusammengestellt wird.
- der Benutzer bei Bedarf über eine intuitive User Interface (*explizite Interaktion*!) die physische Umgebung – Raum und die verfügbaren Geräte – direkt bedienen kann.
- Environment Control verteilt realisiert ist: Es läuft auf unterschiedlichen Geräten. Der Benutzer kann jedes verfügbare Gerät (i.e. PDA, Notebook, in einem Raum eingebetetes Gerät, Smartphone, Fernsehgerät, Projektor, Smartpen,...) für Kommunikation und Interaktion mit seiner Umgebung benutzen.

Media Management

Media Management sorgt für eine ubiquitäre Datenhaltung. Persönliche Daten eines Benutzers sind „anywhere and anytime“ und auf dem Gerät verfügbar, auf dem diese Daten benötigt werden. Media Manager bietet einen für den Anwender transparenten Zugriff auf Daten, die in unterschiedlichen Formaten und auf unterschiedlichen Geräten gespeichert sein können. Dabei bekommt der Benutzer die Untermenge an

verfügbaren Daten, die für seine aktuelle Situation relevant sind. Er kann verschiedene Sichten auf seine Daten haben (alle Dokumente von einer Person, alle für ein Thema relevante Daten, alle für eine Aufgabe relevanten Daten, alle an einem Datum erstellten Daten). Basierend auf (von externen Komponenten) gelieferten Prognosen für den Informations- und Unterstützungsbedarf des Anwenders werden Strategien entwickelt und Aktionen eingeleitet, um die ubiquitäre Verfügbarkeit von Daten zu garantieren. Wenn der Benutzer beispielsweise an einer Sitzung teilnehmen möchte, und es gibt dort keinen Netzwerkzugriff, so sorgt ein Media Manager dafür, dass die für das Meeting relevanten Daten (letzter E-Mail-Verkehr, Agenda, Präsentation) präventiv auf das Notebook des Benutzers übertragen werden.

Ein Media Manager kann als *a single point of information access* betrachtet werden, der unabhängig von Plattformen und Geräten arbeitet.

Interaktion und Dialogue-Management

Diese Komponenten nehmen eine besondere Rolle bei der Realisierung der Ubiquitous Computing Vision ein. Die *Multimodale Interaktion* – vor allem die Sprach- und Gestik-Interaktion – erlaubt es, menschenzentrierte, intuitive und besonders leicht zu bedienende Benutzerschnittstellen zu realisieren. Dadurch wird der Zugang zur AmI-Technologie erleichtert.

Persönliche Geräte und Umgebungsgeräte

Intelligente Umgebungen formen sich aus den in physischen Räumen integrierten Geräten und dort vorhandenen Alltagsgegenständen sowie aus den persönlichen Anwendungen der in den Räumen anwesenden Personen zu einer interaktiven und intelligenten Umgebung. Somit bilden die Geräte die Basis für die *Ausführung von Operationen*.

Environment Monitoring und Sensorik

Sensoren versorgen die *informatisierten* Alltagsgegenständen mit Daten und Informationen über ihre Umgebung. Damit können Umgebungen und Alltagsgegenstände sich und ihre Dienste einer bestimmten Situation (z. B. Helligkeit, Position des Benutzers, Position eines Objektes im Raum, Temperatur usw.) anpassen und *situationsabhängig* und pro-aktiv handeln. In den Kapiteln 4.4.1 und 4.3 wird auf das Thema *Context-Awarenes* ausführlich eingegangen.

Middleware

Um in dynamischen intelligenten Umgebungen den Benutzer bei der Durchführung seiner Ziele zu unterstützen, müssen die vorhandenen Anwendungen und die Infrastruktur in der Lage sein, miteinander zielgerichtet zu kooperieren. Um solch eine Kooperation zwischen den vorhandenen Informations-Anwendungen und den Alltagsgegenständen zu erlauben, muss eine Middleware verfügbar sein, welche

- *Benutzerziel* sowie seinen *Bedarf* an Informationen und IT-Unterstützung *identifiziert* (*Situation-Recognition*, z. B. der Nutzer will etwas präsentieren).

- aus den Nutzerzielen eine Reihe von Tasks ableitet (*Pro-Activity*; Licht einstellen, Projektor einschalten, ...).
- ad-hoc eine intelligente Umgebung formt und eine Strategie entwickelt, um die durchzuführenden Tasks auf die verfügbaren Geräte zu verteilen.
- den Geräten ermöglicht, miteinander zu kommunizieren. Dabei konkurrieren die verfügbaren Geräte, um einen Task zu bekommen. Wenn es z. B. als output-device einen Projektor und ein Fernsehgerät gibt, dann entscheidet die Umgebung, wer den Zuschlag bekommt
- Anwendungen aus der Umgebung abkoppelt, wenn z. B. ein Benutzer den Raum verlässt. Dabei muss es die Plattform erlauben, neue Anwendungen dynamisch zu integrieren, wenn beispielsweise eine Person mit einem neuen Anwendung einen Raum betritt.

Um eine komplexe Aufgabe durchzuführen, muss eine Komposition aus mehreren Geräten gebildet werden, die erst als Appliance Ensemble (also als Team) in der Lage ist, eine bestimmte Dienstleistung zu erbringen:

Ein Versammlungsraum, in dem mehrere Leute diskutieren, ist mit einem Drucker, einem Fernseher und einem PC ausgestattet. Die Tafel ist entsprechend gefüllt. Ein Teilnehmer hat mit seinem „camera integrated smart phone“ einige Fotos von der Tafel gemacht. Nun bietet das Handy dem Benutzer an, das Foto auszudrucken. Denn Smart Phone und Drucker bilden hier dynamisch ein Appliance Ensemble und sind als Team in der Lage, den Inhalt der „normalen“ Tafel auszudrucken. Ohne solch eine Komposition könnte man entweder Fotos machen oder etwas drucken.

Verlässt der Benutzer diesen Raum, bietet sein Handy diese Funktionalität nicht mehr an, auch wenn er wieder Fotos macht. Denn das Handy hat sich aus der Gerätekomposition „ausgeloggt“, als der Benutzer den Raum verlassen hat.

Diese Art von dynamischem, dezentral organisiertem, zielgerichtetem Zusammenspiel von Anwendungen ordnet diese Arbeit unter den synonymen Begriffen „Device Federation“, „Device Composition“, „Appliance Ensemble“ oder auch einfach **Middleware** ein.

4.2 Interaktionssysteme

Dieses Kapitel vergleicht einige wichtige Arbeiten aus Sicht der Mensch-Umgebungs-Interaktion. Im Vordergrund stehen hier die Aspekte der Interaktion insbesondere in adaptiven Medienräumen. Auf technische Aspekte wird hier bewusst nicht eingegangen. Ebenfalls verzichten wir auf eine Beschreibung der jeweiligen Systemarchitektur. Vielmehr wird eine kurze Übersicht des Bedienungskonzeptes des jeweiligen Systems dargestellt. Dann wird das analysierte System mittels des im Kapitel 2 entwickelten Klassifikationsmodells eingeordnet. Anschließend wird auf die Anforderungen von Kapitel 3 Bezug genommen, wobei auf eine wiederholende Beschreibung der jeweiligen Anforderungen verzichtet wird. Vielmehr verweisen wir auf die in Kapitel 3.4 dargestellten Zusammenfassung dieser Anforderungen.

4.2.1 u-Photo und StateSnap

Ein interessanter Ansatz für die Steuerung von intelligenten Umgebungen wird von Kohtake et al. (2005) [SAI+05] präsentiert. Das u-Photo-System verfolgt einen Bild-basierten Ansatz für die Darstellung der zu steuernde Umgebung. Der Benutzer kann mit seinem Gerät Bilder der Umgebung aufnehmen. Diese Daten werden mit weiteren Informationen überlagert. Es handelt sich hierbei um Icons, welche auf den im Foto dargestellten Objekten überlagert angezeigt werden. Als Beispiel kann der Benutzer in seinem Hotelzimmer ein Bild der Umgebung mittels eines PDAs aufnehmen. So kann das Bild die im Raum vorhandenen Computer, Bildschirme, Drucker, Musikanlage und Ventilator enthalten. Es wird dann zusammen mit der Geräte-Identifikation via WLAN auf ein Subsystem – den u-Photo-Creator – gesendet. Dieser erzeugt ein u-Photo, welches das aufgenommene Bild mit den jeweiligen 2-D-basierten grafischen Elementen beinhaltet. Das so erzeugte u-Photo wird an das PDA gesendet, wo es permanent gespeichert wird. Durch das überlagerte Bild (u-Photo) kann dann von dem PDA aus auf die physikalischen Geräte der Umgebung zugegriffen werden. Das von u-Photo adressierte Hauptproblem ist es, den Nutzern eine intuitive Möglichkeit für die Geräteauswahl in vernetzten Umgebungen zu ermöglichen. Mit Hilfe des Bild-basierten Interaktionsmodells nutzt u-Photo die menschliche visuelle Eigenschaft, Objekte der realen Welt auf einem Foto einfacher und schneller wiedererkennen zu können. Die Abbildung 4.15 zeigt ein Drop-Down-Menü für die Steuerung einer Klimaanlage. Die Benutzer können mit dem u-Photo System auch Medien von dem PDA auf einem Computer oder zu einem Drucker übertragen.

Abbildung 4.15: Das u-Photo-Benutzungsschnittstelle erlaubt einen funktionsbasierten Zugriff auf Geräte und Medien. Benutzer können durch das Foto die Geräte manuell auswählen und über das normale 2-D-Menü bedienen (Quelle [SAI+05], 2005).

Einordnung des u-Photo-Systems Bezogen auf die Klassifikation von Kapitel 2 ermöglicht das u-Photo-System eine explizite, funktionsbasierte Interaktion mit Medien und Geräten. Die angebotenen Modalitäten sind 2-D-Text und 2-D-Bild. Die Geräte werden geräteorientiert ausgewählt. Das u-Photo-System ermöglicht keine zielbasierte Interaktion. Entsprechend können auf der Dimension der Strategiebildung weder Makros noch dynamische Planung zur Anwendung kommen. Eine Ko-

existenz mit adaptiven Umgebungen wird nicht adressiert. Es ist an dieser Stelle zu erwähnen, dass wir das Konzept der räumlichen Adressierung auf der Basis von 3-D-Visualisierung bereits 2003 und 2004 konzipiert, umgesetzt und veröffentlicht haben (vgl. Nazari, 2004 [Naz04, NHK04]), während das u-Photo System erstmalig 2005 veröffentlicht wurde (vgl. Kohtake et al., 2005 [SAI+05]).

Bezug zu Anforderungen In Bezug auf die Anforderungen des Kapitels 3 adressiert das u-Photo System das Problem einer intuitiven manuellen Selektion von Geräten durch den u-Photo-Viewer. Das System bietet eine explizite Interaktion mittels persönlicher mobiler Assistenten an. Durch den Einsatz von selbsterstellten Bildern für die direkte Manipulation der Umgebung wird das wieder erkennen von fotografierten Geräten innerhalb des Systems erleichtert und die Bilder können durch anklicken im „u-Photo-Viewer“ ausgewählt werden. Somit erlaubt das u-Photo-System eine räumliche Adressierung von Geräten. Nutzer können die Geräte auswählen, ohne spezifische Geräteadressen zu kennen.

Jedoch wird dem Benutzer bei der Erkundung der Umgebung und insbesondere bei der Orientierung in unbekannten Räumen nicht weiter geholfen, denn er soll sich vielmehr selbst orientieren und die Bilder erzeugen. Im Gegensatz hierzu kann eine 3-D-basierte Darstellung dem Benutzer eine virtuelle Führung in der Umgebung ermöglichen.

Es gibt keine Möglichkeit für die Bedienung von unsichtbaren und kleinen Geräten. Das liegt daran, dass der Nutzer durch Fotografie die Menge der unterstützten Geräte einschränkt.

Der Einsatz von Augmented Reality oder 3-D-basierter Darstellung der Umgebung kann im Gegensatz zu solchen Foto-basierten Oberflächen hier eine Abhilfe bieten. Des Weiteren muss die Erkundung der Umgebung nach vorhandenen Geräten automatisch erfolgen, damit auch in die Umgebung unsichtbar eingebettete Geräte in die Interaktion einbezogen werden können.

Da das u-Photo-System eine explizite, funktionsbasierte Interaktion als Bedienungskonzept umsetzt, muss der Nutzer seine Bedienungsaufgaben einzeln an die jeweiligen Geräte senden. Eine Geräteauswahl wird zwar durch die räumliche Adressierung erleichtert, jedoch kann diese Art der Steuerung insbesondere in komplexen Umgebungen zu einer kognitiven Überlastung des Nutzers führen. Ein ganzheitliches Bedienungskonzept im Sinne einer zielbasierten Interaktion oder einer Hinzunahme von impliziter Interaktion könnte dies vermeiden. Eine Koexistenz des mobilen Assistenten mit adaptiven Umgebungen wird im u-Photo (vgl. Kohtake et al. (2005)) nicht behandelt.

Das Darstellungskonzept des u-Photos biete eine gute System-Repräsentation für die jeweilige Umgebung an. Der Nutzer kann auf Anhieb erkennen, in welchem Zustand sich die Geräte befinden und welche Funktionen die jeweiligen Geräte anbieten. Nachteilig ist es, dass die Zustände einer intelligenten Umgebung als eigenständige Entität nicht einbezogen werden. So kann der Nutzer beispielsweise nicht erfahren, welche Automatismen eine adaptive Umgebung aktuell durchführt oder ob eine reaktive Umgebung aktuell aktiv arbeitet. Ebenso kann die Benutzungsschnittstelle von u-Photo für eine adaptive Umgebung keine dynamische Visualisierung über

deren Situationserkennung geben. Der Benutzer kann so nicht erfahren, in welchem Zustand sich die Umgebung befindet oder was die Umgebung als nächstes implizit tätigen wird. Hier wird erneut darauf verwiesen, dass die Anforderung nach *Visibility & Affordances* eine für die Akzeptanz besonders wichtige Rolle einnimmt (vgl. Norman (1998)).

Ein besonderer Nachteil des bildbasierten Ansatzes besteht darin, die Benutzungsschnittstelle adaptiv zu gestalten. Obwohl Gerätezustände im u-Photo-System dargestellt werden können, erlaubt dieser Ansatz es nicht, auf die Entfernung von Geräten oder Positionsänderung zu reagieren. Somit sind die Forderungen nach Adaptivität und kontextabhängiger Darstellung nicht ausreichend erfüllt.

4.2.2 PHH-Assistent im „EMBASSI-Privathaushalt“

Das Projekt Elektronische Multimediale Bedien- und Service Assistenz (EMBASSI) ist ein BMBF Leit-Projekt, welches im Rahmen des Programms „Mensch-Technik-Interaktion“ durchgeführt wurde [HK02]. Es besteht aus drei voneinander lose gekoppelten Einzelszenarien. Dabei fokussiert ein Szenario darauf, dem Benutzer die typischen Aufgaben im Privathaushalt abzunehmen. Die anderen Szenarien behandeln das Fahrzeug und öffentliche Informationsterminals wie z. B. Geldautomaten. Die Vision von EMBASSI sieht es vor, dass der Nutzer seine Umgebung über Sprache und Gesten steuern kann. Ein wesentliches Element im EMBASSI-Projekt ist die Trennung von User Interface und Systemkomponenten in einem verteilten System.

Das Teilsystem für die Domäne Wohnzimmer bzw. „Privathaushalt“ ist aus mehreren vernetzten Komponenten realisiert. Diese ermöglichen insgesamt den Empfang, die Speicherung und Wiedergabe von digitalen Medien, insbesondere von Audio- und Video-Datenströmen. Um die Bedienung zu erleichtern, werden verschiedene Interaktionsformen bzw. Bedienassistenten angeboten. Dabei kommunizieren die verschiedenen Komponenten über die Agentensprache KQML [UMB97]. Der Assistent unterstützt auch bei der Auswahl einer TV - Sendung. Er stellt Informationen zum TV-Programm bereit, wie sie üblicherweise eine Programmzeitschrift zur Verfügung stellt. Die Interaktion mit dem Benutzer läuft über das multimodale Dialogmanagement. Der Assistent liefert nur Informationen zum TV-Programm zurück, er führt keine Bedienhandlungen von selbst aus.

Das multimodale Dialogsystem ist das wesentliche Teilsystem für die Interaktion im EMBASSI-System. Es lässt sich in drei Teilbereiche gliedern:

- Eingabekomponenten (Fernbedienung des GUI, Spracheingabe, Zeigestab oder Zeigegesten, evtl. Blickrichtungsanalyse und Mimik-Erkennung)
- Dialogsteuerung (amodale Dialogsteuerung, Verbindung zur Assistentenebene und zum Context-Manager)
- Polymodale Ausgabe (GUI, Sprachausgabe [Textgenerierung und Sprachsynthese], Avatare)

Das polymodale Eingabemodul (PMI) führt die verschiedenen multimodalen Eingaben zusammen. Der Dialogmanager verarbeitet den multimodalen Input. Dies

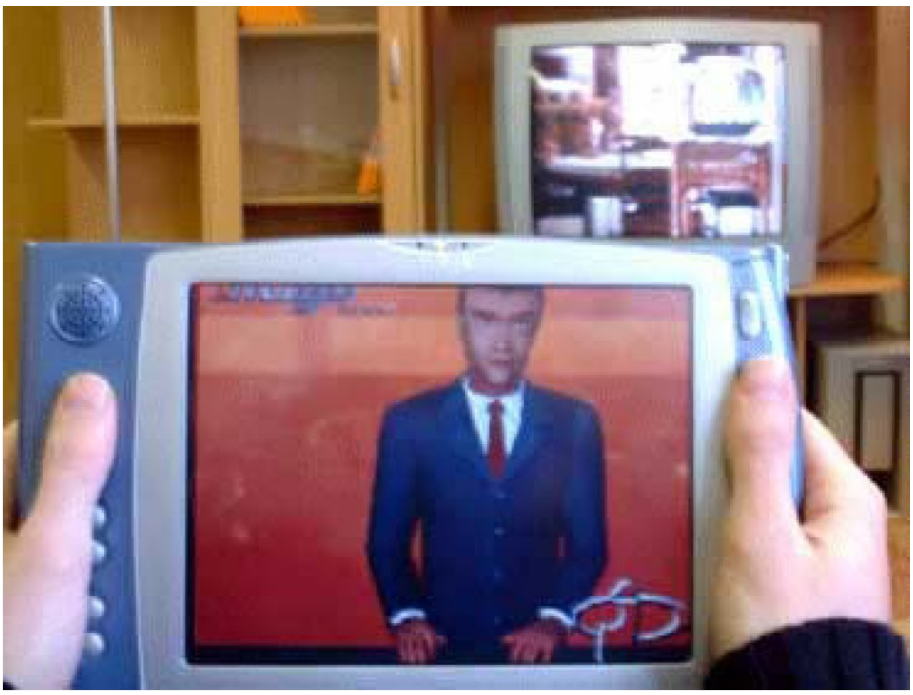


Abbildung 4.16: Avatar-basierte Benutzungsschnittstelle für EMBASSI-Privathaushalt.

umfasst die Analyse der Spracheingabe sowie die Einbeziehung von Gestik, Blickrichtung und manueller Eingabe über die Fernbedienung. Eine implizite Assistenz durch den multimodalen Dialog ist nicht gegeben. Die Dialogführung wird an Benutzer, Aufgabe und Situation angepasst und ist insofern adaptiv.

Die Aufgabe des polymodalen Ausgabemoduls (PMO) ist, die vom Dialog-Manager empfangene Ausgabe in amodaler Form auf die Ausgabe-Komponenten zu rendern. Als Ausgabe-Komponenten werden Avatar, GUI und Textgenerierung unterstützt. Im Rahmen der Bedienassistenz handelt das PMO nicht implizit, da jede Ausgabe durch den Dialog-Manager initiiert wird. Weiterhin werden durch das PMO auch keine Automatisierungen durchgeführt. Das gesamte Dialogsystem ist überwiegend passiv und reagiert auf explizite Eingaben des Benutzers. Die verschiedenen Eingabemodalitäten können zusammen und gleichzeitig oder alternativ benutzt werden. Die Ausgabemodalitäten werden vom Dialogsystem unter Berücksichtigung des Systemzustands ausgewählt, die Ausgabe erfolgt adaptiv. Die Komponente GUI ist für die Textgenerierung und für das Avatar für die Visualisierung zuständig. Sie besteht aus einer im TV-Bereich üblichen Kombination aus einer Infrarot-Fernbedienung (Einzeltasten und Cursorsteuerung) und der Anzeige auf dem Bildschirm (On Screen Display, OSD). Die Komponente PMI koordiniert die Eingabe des Benutzers.

Im Rahmen des Szenarios Privathaushalt kann der Nutzer über die bereitgestellten Schnittstellen Licht, Jalousien, und Ventilator steuern. Des Weiteren kann er auch mit dem Fernseher interagieren und beispielsweise über Sprache mitteilen: „Ich möchte heute Abend den Krimi auf Premiere aufnehmen“. Das EMBASSI-System erkennt, dass der Service „Premiere“ nicht empfangen werden kann, und informiert den Benutzer entsprechend. Darüber hinaus wird dem Benutzer eine Liste der empfangbaren Krimis angezeigt, aus der er einen zur Aufzeichnung auswählt [Töd03].

Einordnung des EMBASSI-Assistenten Im Bezug auf die Klassifikation von Kapitel 2 ermöglicht das EMBASSI-System eine explizite Interaktion mit Medien und Geräten. Es sind sowohl Assistenten zur zielbasierten Interaktion als auch zu funktionsorientierter Steuerung vorhanden. Jedoch handelt es sich hierbei um mehrere voneinander unabhängig entwickelte Systeme. Die angebotenen Modalitäten sind Animationen in 2-D-Grafik, Text und Video sowie Avatar (vgl. Abb. 4.16). Des Weiteren wird Sprachein- und Ausgabe verwendet. Derzeit unterstützt das EMBASSI System es nicht, die jeweils passende Interaktionsform für eine Situation automatisch auszuwählen. Eine derartige situationsabhängige Koordinierung von Ausgabemodalitäten wird im Folgeprojekt „DynAMITE“ entwickelt [Dyn04b]. Auf der Dimension der Strategiebildung können durch dynamische Planung aus Zielen einzelne Aktionen hergeleitet werden. Diese können dann in der jeweiligen Umgebung ausgeführt werden, um das Ziel zu erreichen [EK05]. Möchte der Nutzer funktionsbasiert vorgehen, so müssen die Geräte direkt ausgewählt werden.

Auf der Dimension der Initiative kann festgestellt werden, dass das Assistenz-System vom EMBASSI vom Benutzer hauptsächlich explizit benutzt wird. So kann der Anwender beispielsweise eine Liste mit Filmen anfordern, die ihn interessieren könnten. Er kann auch den Avatar um Hilfe bitten, einen Film aus dem Internet herunterzuladen. Weiter ist es möglich über den Assistenten den Zustand von einzelnen Geräten abzufragen. Neben der expliziten Interaktion erlaubt das Assistenz-System auch eine sehr eingeschränkte Art von impliziter Interaktion. Das EMBASSI-System informiert bspw. über aktuelle Zustände, Veränderungen und Handlungsmöglichkeiten. Es führt von sich aus Bedienhandlungen aus. Beispiele hierfür sind [Töd03]:

- Hinweise auf neue Film-Angebote
- automatisches Anpassen an veränderte Bedingungen wie Raumhelligkeit, Empfangsbedingungen ...

Ein weiteres Beispiel ist im Szenario „Ich möchte heute Abend den Krimi auf Premiere aufnehmen“ enthalten, da das System von sich aus Alternativen zum Krimi auf Premiere generiert. Der Benutzer muss nicht erst um einen Alternativvorschlag bitten. Weitere Beispiele für aktive Assistenz wären das Erscheinen des Avatars mit Hinweisen zur Bedienung in Folge von fehlerhaften Eingaben oder das Erinnern an vorgemerkte Filme. Eine implizite Assistenz könnte beispielsweise darin bestehen, dass diese Sendung selbsttätig aufgezeichnet wird, wenn der Benutzer entgegen seiner Gewohnheiten den Videorecorder nicht programmiert hat (vgl. [Töd03]).

Sowohl die explizite als auch die implizite Interaktion verläuft hier über demselben Assistenten, über den der Benutzer mit einem Fernseher interagiert. Dabei geschehen jedoch sowohl die Dateneingabe als auch die Ausgabe über dasselbe Fernsehgerät. Aus Sicht der Interaktion bestehen hier qualitative Unterschiede zu einem Fall, in dem es zwei voneinander unabhängige Assistenten gäbe, welche von außerhalb das Fernsehgerät jeweils implizit oder explizit steuern würden.

Im Falle des EMBASSI Assistenten handelt es sich eher um eine adaptive Benutzerführung als um eine Koexistenz voneinander unabhängiger Möglichkeiten für die Steuerung von dritten Geräten. Da die Dialogführung „vom Werk aus“ geord-

net verläuft, sind Konfliktfälle ausgeschlossen. Insofern besteht keine Parallelität der expliziten und impliziten Interaktion. Eine Koexistenz mit adaptiven Umgebungen wird also nicht adressiert.

Bezug zu Anforderungen Im Bezug auf die Anforderungen des Kapitels 3 adressiert das EMBASSI-Projekt das Problem einer intuitiven manuellen Selektion von Geräten durch den Einsatz von anthropomorphen Benutzungsschnittstellen. Insbesondere setzen die EMBASSI-Assistenten auf Gesten- und Sprachsteuerung. Somit kann der Nutzer Geräte auswählen, indem er auf sie zeigt oder diese nennt. Hauptsächlich verfolgt EMBASSI den Ansatz einer expliziten Interaktion mittels persönlicher, mobiler Assistenten. Mit einem der vielen EMBASSI-Assistenten können Benutzer die Umgebung zielbasiert steuern. Hierzu bietet EMBASSI ein planungsbasiertes Teilsystem an, welches aus Benutzerzielen dynamisch für die jeweilige Umgebung spezifische Strategien ableitet. Somit kann der Anwender auch in unbekannten Umgebungen interagieren, ohne Infrastrukturwissen zu besitzen. Auch wird er nicht mit der wachsenden Komplexität konfrontiert, da nur Ziele statt einzelne gerätebasierte Funktionen geäußert werden müssen. Da die Benutzer die physischen Geräte gar nicht direkt auszuwählen brauchen, könnten die Geräte auch im Hintergrund verschwinden und somit auch die in der Umgebung unsichtbaren in die Interaktion einbezogen werden. Möchte der Benutzer dennoch eine explizite, funktionsbasierte Interaktion mit einzelnen Geräten vornehmen, so stoßen die EMBASSI-Assistenten an ihre Grenzen. Als Möglichkeit gibt es zum einen eine sprachbasierte Benennung, wozu man jedoch Geräteadressen benötigt. Hiermit verbunden ist das bereits erwähnte Problem der kognitiven Überlastung bzw. der fehlenden Informationen. Zum anderen kann der Nutzer mittels Gesten auf ein Gerät zeigen. Dies ist problematisch, wenn die Geräte sehr weit entfernt liegen oder unsichtbar eingebettet sind.

Das Darstellungskonzept von EMBASSI biete keine grafische System-Repräsentation für die gesamte Umgebung an. Der Nutzer kann nicht auf Anhieb erkennen, in welchem Zustand sich die Geräte befinden. Lediglich bieten ein Avatar und ein GUI an, die Zustände von einzelnen Geräten zu erfahren. Welche Geräte überhaupt vorhanden sind, müssen die Benutzer manuell oder zielbasiert erkunden. So kann der Nutzer beispielsweise nicht erfahren, welcher Ziele in der aktuellen Umgebung unterstützt werden können. Dies gilt auch für den Avatar, welcher die unterstützenden Aktionen für jedes Gerät ausspricht. Hier wird erneut auf die Anforderung nach *Visibility & Affordances* hingewiesen. Eine Koexistenz der Assistenten mit adaptiven Umgebungen wird in EMBASSI nicht behandelt.

Nachteilig ist es auch, dass die verschiedenen Assistenten nicht ganzheitlich integriert sind. Eine integrierte Lösung könnte die Vorteile der verschiedenen Assistenten kombinieren. So bieten beispielsweise die GUI- und Avatar-basierten Assistenten zumindest eine eingeschränkte Systemrepräsentation, welche dem Benutzer bei der Erkundung der Umgebung weiterhelfen kann. Dies wird aber von den zielbasierten Assistenten nicht erfüllt, welche nur über Gesten und Sprache bedient werden können.

Auf dynamische Änderungen der Umgebung reagieren die Assistenten. Die 2-D-

Abbildung 4.17: Microsoft Surface: ein Computer für Multi-Touch Interaktion (Bild: Microsoft)

basierten Benutzungsschnittstellen passen sich an. Insofern sind Adaptivität und kontextabhängige Darstellung ausreichend erfüllt.

4.2.3 Greifbare Benutzungsschnittstelle und Multi-Touch

Einer der letzten Entwicklungen im Bereich von neuartigen Ein- und Ausgabeformen ist der Bereich von greifbarer Benutzungsschnittstelle sowie Multi-Touch-Systemen. Für letzteres kann man als Beispiel das iPod Touch und iPhone von Apple oder das Microsoft Surface™ sowie die Arbeit von Jeff Hann [Han05] nennen. Multi-Touch-Systeme vereinen Ein- und Ausgabegeräte und kommen ohne Maus, Tastatur und Extra-Display aus. Solche Systeme werden nur über Finger bedient und können mehrere Eingaben – also mehrere Finger – gleichzeitig verarbeiten. Dadurch stehen dem Benutzer eine Vielzahl von Bewegungs- und Eingabemustern ur Verfügung, welche vom Interaktionssystem für eine Manipulation von Objekten wie Fenstern oder Dokumenten benutzt werden können. Als Beispiel hierfür kann das Zoomen, Drehen, Verschieben, Löschen oder Browsen von Bildern genannt werden. Durch diese Art der Eingabe soll der Umgang mit Icons und Menüs reduziert bzw. ersetzt werden, um eine intuitivere Interaktion zu ermöglichen.

Eine weitere wichtige und neuartige Eingabetechnik ist die „Tangible Interaction“, welche durch eine „Tangible User Interface“ (TUI) ermöglicht wird. Ein TUI erlaubt den Benutzern „die Interaktion mit der Maschine durch physikalische Objekte“. Diese Technik wurde bereits 1997 von Hiroshi Ishii und Brygg Ullmer vom MIT Media Labs eingeführt. Kennzeichend für TUI ist die Verknüpfung digitaler Informationen mit physikalischen Objekten, die die digitalen Informationen direkt wahrnehm- und manipulierbar machen (vgl. Wikipedia [WIK06]). Die physikalische Objekte können dann von einem Menschen im realen Raum manipuliert (bewegt,

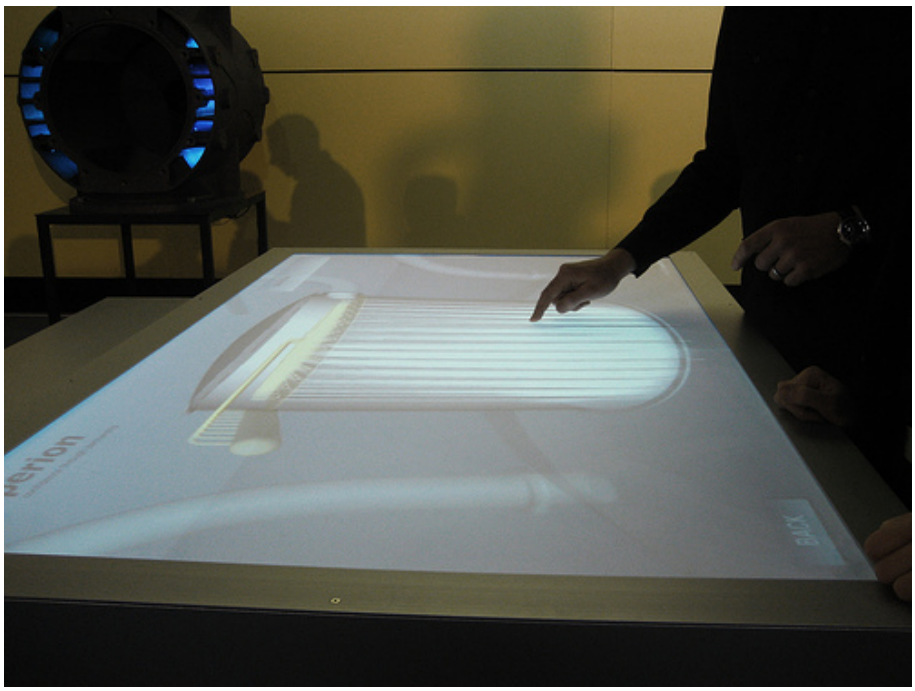


Abbildung 4.18: Multi-Touch-Table für Interaktion mit Virtuellen Umgebungen (Bild: Fraunhofer IGD)

gedreht, aufgehoben) werden, um den Zustand der korrespondierenden Objekten in der digitalen Welt zu beeinflußen.

Bislang wurden Anwendungen basierend auf TUI für die Interaktion in adaptiven Umgebungen wenig erforscht. Als Beispiel kann das Pendle-Projekt von Albrecht Schmidt et al (vgl. [VKVS05]) genannt werden. Das Pendle ist ein tragbares TUI, welche die Interaktion des Benutzers mit öffentlichen Bildschirmen (Public Display) koordiniert.

Wenn der Benutzer das Pendle um den Hals trägt, so wird das System in Abhängigkeit vom Benutzerprofil betrieben werden. Das Profil befindet sich auf dem Pendle und wird von dort aus auf dem System übertragen. Das Profil des Benutzers schreibt vor, welche Daten und Nachrichten auf dem öffentlichen Display angezeigt werden dürfen. Wenn der Benutzer das Pendel in den Händen trägt und damit vordefinierte Gesten erzeugt (vgl. Abb. 4.19), so wird das Public Display explizit vom Benutzer gesteuert. Damit ermöglicht das Pendle-System eine sehr einfache Art von Koexistenz zwischen Benutzer-initiierte, explizite Interaktion und System-initiierte, implizite Interaktion. Derzeit sind MusicPendle und ein NewsPendle realisiert.

Ferscha et al (vgl. [HRLF06, FVEW08]) stellen Konzepte für den Einsatz von TUI für die Steuerung von IPTV-Geräten vor. Um die wachsende Komplexität von Fernbedienungen gerecht zu werden, schlagen Ferscha et al (2008) den Einsatz von TUI vor.

Es sollen dem Benutzer physische Verknüpfungen angeboten werden, um die meistbenutzten Kommandos auszudrucken. Demnach werden orientierungsbewußte Artifakte entworfen, welche als greifbare Benutzungsschnittstelle dienen sollen. Die physikalischen Verknüpfungen sind mit Handgesten des Benutzers – also mit der Bewegung des greifbaren Artifaktes im realen Raum – zu aktivieren. Diese Gesten werden von einem in dem greifbaren Objekt integrierten Mikrokontroller erkannt

Abbildung 4.19: Das Pendle Projekt erlaubt eine personalisierte Interaktion mit öffentlichen Displays (Bild: HCILab)

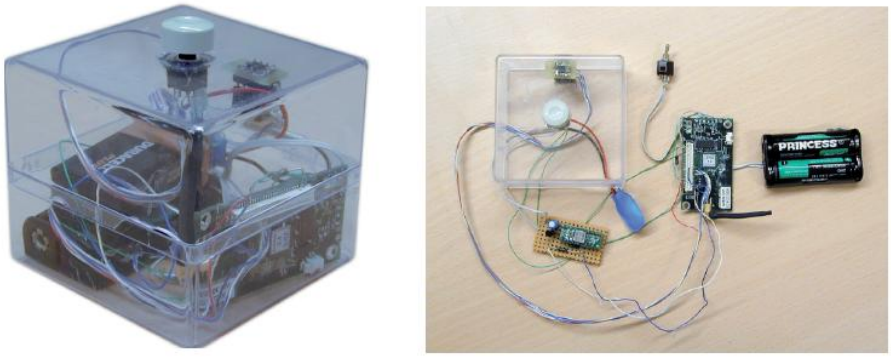


Abbildung 4.20: TA Cube ist ein TUI und stellt eine physische Verknüpfung für die Steuerung von Geräten dar (Bild: [FVEW08])

und als Infrarotsignal an das Zielgerät gesendet. *TA Cube* ist eine Realisierung des Konzeptes für eine greifbare Benutzungsschnittstelle für die Steuerung von IPTV-Settop-Boxen (vgl. Abbildung 4.20).

Bezug zu Anforderungen Im Bezug auf die Anforderungen des Kapitels 3 wird bei den oben vorgestellten Arbeiten das Problem einer intuitiven manuellen Selektion von Geräten nicht adressiert. Ebenso wird das Problem von fehlenden Mechanismen zur Interaktionsmanagement nicht adressiert. Die vorgestellten Arbeiten fokussieren hauptsächlich auf die Bereitstellung von intuitiven Ein- und Ausgabetechniken.

Jedoch besitzen diese Multi-Touch-Systeme ein großes Potential für die Entwicklung von Steuerungsassistenten für komplexe Umgebungen. Solche Systeme für die Interaktion in adaptiven Umgebungen müssen noch erforscht und entwickelt werden. Mit Hilfe von diesen Systemen könnten Benutzer Geräte sehr intuitiv auswählen, indem sie auf diese mit Fingern zeigen würden. Ebenso könnte das Browsen in großen Datenmengen sowie Navigation von 3-D-Szenen (vgl. Abbildung 4.18) vereinfacht werden.

4.2.4 Weitere Systeme

Zusätzlich zu den im letzten Abschnitt diskutierten Systemen werden an dieser Stelle die Projekte SIKOWO und HomeLAb aus der Interaktionsprespektive betrachtet. Auf die allgemeinen Ziele und Ansätze der beiden Projekte wurde bereits im Kapitel 4.1 eingegangen.

Sowohl SIKOWO als auch EasyLiving zeichnen sich dadurch aus, dass sie implizite und explizite Interaktionsmethoden anbieten, um die Nutzer-Interaktion zu unterstützen. Während beide Systeme bei der expliziten Interaktion auf intuitive Sprach- und Gestikerkennung setzen, bietet EasyLiving zusätzlich klassische Eingabemöglichkeiten wie Tastatur, Maus, Fernbedienungen oder Touchscreens, womit aber technikfremde Personen anfangs eher schlechter umgehen können. Bei der Eingabe von großen Datenmengen (z. B. beim Schreiben oder Ändern von Dokumenten) eignet sich die Eingabemöglichkeit über die Tastatur jedoch besser, weshalb sich EasyLiving sowohl für den Privathaushalt als auch für Meetingräume eignet, während SIKOWO ein reines Wohnkomfortregelsystem ist.

Im inHaus, wie auch im HomeLab, erhält der Benutzer einheitliche Schnittstellen für die Benutzung der Endgeräte sowie die Möglickeit, einzelne Aktivitäten unabhängig vom Ort auszuführen. Im HomeLab werden diese mittels einer Menubar, einer Housemap und einem Finder umgesetzt. Die Menubar listet die Aktivitäten auf, die gleichzeitig auf einen Anwender bezogen ablaufen. Die Housemap gibt eine Übersicht über die zurzeit laufenden Aktivitäten und über den Finder können Aktivitäten gestartet werden, indem der gewünschte Inhalt ausgesucht wird. Im inHaus liegt der Schwerpunkt auf der web-basierten Steuerung von Hausgeräten und daher werden auf den Bediengeräten beispielsweise Möglichkeiten für die richtigen Einstellungen der Waschmaschine aufgelistet.

4.3 Context Awareness

Es existiert eine Reihe von Modellen, die sich allgemein auf Umgebungs- und Geometriemodelle konzentrieren. Dazu gehören z. B. NEXUS Spatial World Model [BBR02a], Aura Location Model [JS02], EasyLiving Geometric Model [BMK+00a] oder LISTEN World Model [GS01].

Eine ausführliche Übersicht über Modelle für Ambient Intelligence liefert die Dissertation von Ulf Leonhardt [Leo98]. Martin Bauer und Christian Becker behandeln in [BBR02a] ebenfalls einen Vergleich und state-of-the-art für Geometriemodelle.

4.3.1 Was ist Kontext?

Dey und Abowd [IST04, ADB+99] definieren ein kontextabhängiges System als ein System, welches *Kontextwissen* benutzt, um dem Nutzer *relevante* Informationen und Dienste anzubieten, wobei die *Relevanz* von den aktuellen Aufgaben des Nutzers abhängt.

Definition von Kontext: jegliche Zusatzinformationen über den Nutzer, seine physikalische Umgebung, seine Aufgaben, welche es erlauben, *relevante* Informationen

und Dienste anzubieten, um den Nutzer bei der Durchführung seiner Aufgaben zu *unterstützen*.

4.3.2 Welche Komponenten hat Kontext?

Die aktuelle Forschung im Bereich von Context-Awareness behandelt folgende *Kategorien* von Kontextinformationen (oder einfach *Kontextkomponenten* bzw. *Kontextdimensionen*):

- Position und Orientierung von Personen, Objekten und Geräten. Hierzu bietet [BBR02a] eine guten Übersicht.
- Anwesenheit von Objekten und Geräten [BKMS00].
- Aktivitäten, benutzte Applikationen sowie Interaktionsmodalitäten einer Person in einem bestimmten Raum [BKL02, DK03, ZGB+04, NodAH04, Jov03, PHNP05].
- Identität von Personen in einem Raum [Hec01, BKMS00, HK03].
- Zeit (Datum, Uhrzeit, Kalender).
- Agenda einer Person, welche eine bestimmte Aufgabe betrifft [eGu99].
- Medien und Eignung von Geräten zur Wiedergabe bestimmter Medien [W3C04].
- Umgebungsdaten wie Licht, Lautstärke, Tempratur.

Eine Übersicht zu Kontextmodellen und Kontextrepräsentation geben Linnhoff-Poppien und Strang in [SLP04, CK00].

4.3.3 Wie wird Kontext dargestellt?

Es gibt verschiedene Wege zur Darstellung von Kontextdaten. Wegen mangelnder Standards und Referenzmodelle existiert keine einheitliche Darstellung von Kontextinformationen. Dies führt dazu, dass vorhandene Kontextinformationen einer Anwendung *nicht* einem anderen System zur Verfügung stehen.

Gängige Modellierungen sind [HIR02, IST04, SLP04]

Key-Value Paare In diesem einfachen Modell dient ein Kontextparameter (wie Licht, Anzahl von Personen, Identität oder Position einer Person) als Key und dazugehörige Werte werden als Values abgespeichert.

Tagged Encoding Basierend auf XML-Technologien werden Kontextinformationen innerhalb eines Tags dargestellt. Dabei werden Kontextparameter und Eigenschaften als *Tags* abgebildet. Beispiele hierzu liefern [MUM03, Gie00, HK03, MK00, WH03].

Objektorientierte Modellierung Angelehnt an OOM und UML-Technologien [UML06] werden komplette Domänen modelliert. Dabei können Kontextinformationen als Objektzustände abgebildet werden. Einige Arbeiten [HK03] fokussieren darauf, UML für den Einsatz in Ambient Intelligence und zur Kontextmodellierung anzupassen.

Logikbasierte Modellierung Mit am Ausdrucksstarksten für die Modellierung einer Domäne sind logikbasierte Sprachen. Hierbei werden gleich auch Inferrenz-Regeln zwecks Ableitung neuer Kontextinformationen (so genannte Context Prediction) in ein Modell eingebettet. Hierbei finden Methoden zur automatischen Klassifikation und Kategorisierung von Daten sowie Regelableitung aus anderen Gebieten (z. B. Data Mining und Knowledge Management) neuen Einsatz. Beispiele für logikbasierte Modelle sind [BKL02, Pin94, SAi02, B. 01].

4.3.4 Wie wird Kontext erfasst?

Je nach Art der zu erfassenden Kontextparameter können unterschiedliche Erfassungstechnologien zum Einsatz kommen. Die wichtigsten Kontextinformationen sind in den letzten Kapiteln (siehe Kapitel 4.3.2) als *Ort*, *Zeit*, *Umgebungsdaten* (Temperatur, Licht, ...), *vorhandene Geräte*, *Nutzeridentität*, aktuell ausgeführte *Aktivitäten*, *Agenda*, Zustand von *Geräten und Diensten*, sowie *Medien* identifiziert worden.

Neben kamerabasierten Systemen dienen hauptsächlich Sensoren zur Erfassung. Die am weitesten erforschten sensortechnischen Erfassungssysteme zielen auf die Bestimmung von Geometriedaten ab. Hierbei kommen diverse Sensoren wie Ultraschall [Nis00], Infrarot [Gie00], Wireless LAN [Soc03], Bluetooth [Loc05], RFID [WH03, Ahm04], Ultra-Wide-Band [Ubi05] zum Einsatz. Einen ausführlichen Überblick hierzu geben [Fle04, Sch99, Ahm04, Con04, Dur02].

Für Umgebungsdaten werden herkömmliche Sensoren benutzt. Einige Mini-Computer, welche typischerweise in Verbindung mit Sensoren zur Kontexterfassung und -verarbeitung dieser Informationen eingesetzt werden, werden bereits mit integrierten Licht-Sensoren, Mikrofon sowie Beschleunigungssensoren ausgestattet. Beispiele für solche Wireless-Sensors (auch Motes oder Sensor Nodes genannt) sind Smart-Its [The03], Intel Motes [Int05], Micaz [Cro05] und Smart Dust [Kri04].

Zur Ermittlung vorhandener Geräte werden üblicherweise existierende Protokolle zur Device Discovery eingesetzt. Vorreiter sind Jini, JXTA, UPnP [UPn04b] und Bluetooth.

Zur Erkennung von Objekten wie Mobiliar setzen [BMK+00a] und [FOR04a] kamerabasierte Systeme ein. Alternativ können Objekte auch über herkömliche Tag-basierte Systeme gefunden werden [Vog02].

Zur Ermittlung der Nutzeridentität kommen Video-basierte Systeme ([FOR04c] und [BKMS00]), biometrische Sensoren [Fra97, WHH+02], oder Tag-basierte Systeme wie RFID [Vog02] zum Einsatz. Andere Systeme wie ORL Active Floor [M.D97, VOR04] versuchen über die Analyse des Gangmusters auf einem Teppich auf die Identität des Nutzers zu schließen. Aktuell vom Nutzer ausgeführte Aktivitäten werden vom Forschungsgebiet *Activity-Recognition* adressiert. Für die Office-Domäne basieren die meisten Systeme auf der Analyse von Audio-Video-Daten [ZGB+04, NodAH04, Jov03]. Im Bereich des mobilen Nutzers (so genannte Out-Door-Umgebungen bzw. *Wearable Computing*) basieren die meisten Ansätze auf der Analyse von 3-D-Beschleunigungssensoren. Einen guten Überblick hierzu gibt [KASS04].

Informationen zu Agenda und Zeitplanung eines Nutzers können über diesbezüglich entwickelte Agenten geliefert werden. Hierzu existieren bereits Lösungen aus den Bereichen der Multi-Agenten-System (MAS) sowie Computer Supported Cooperative

Working (CSCW).

Gerätezustände werden meistens über die Bussysteme geliefert, welche zur Steuerung dieser Geräte eingesetzt werden. Hierzu zählen u. A. UPnP [UPn04b], HAVi [HAV02] und European Installation Bus [Eur04]. Informationen über Zustände von Software Services, Medien sowie verfügbaren Ressourcen liefern die jeweiligen Dienste bzw. Diensteplattformen. Zu solchen Service-Plattformen gehören Open Agent Architecture [SRI00], DynAMITE [Dyn04b], UPnP [UPn04b] und Jini [Jin00].

4.3.5 Wie wird Kontext in Anwendungen überschrieben?

Kontextinformationen werden von Context-Management-Systemen (CoMS) verwaltet. Grundsätzlich sind zwei Arten der Bereitstellung von Kontextinformationen zu unterscheiden.
Zum ersten kann eine Anwendung, welche solche Informationen verwenden möchte, *aktiv* danach fragen. Hierfür kommen XML/RDF Lösungen oder Datenbank-basierte *CoMS* in Frage. Beispiele für solche CoMS sind Microsoft EasyLiving [BMK+00a] und das MUMMY Projekt [MUM03].

Zum anderen können sich interessierte Systeme für relevante Kontextinformationen bei *Event-basierten* CoMS registrieren. Sobald *neue* Daten für die registrierten Kontextparameter verfügbar sind, werden die *subscribed* Anwendungen über Events benachrichtigt. Middleware zur Verwaltung solcher kontextabhängigen Anwendungen sind Solar [CK02], Open Agent Architecture [SRI00] sowie DynAMITE [SRI00].

4.4 Zusammenfassung und Herausforderungen

4.4.1 Zusammenfassung Ambient Intelligence

Die vier vorgestellten Projekte konkretisieren die Vision von AmI sowie Ubicomp und zeigen *exemplarisch* anhand von ausgewählten Szenarien in ausgewählten Domänen, wie die Zukunft der den Alltag durchdringenden, unsichtbaren, intelligenten, auf die Nutzerbedürfnisse eingehenden und intuitiv zu bedienenden Informationstechnologie aussehen könnte [Shi06].
Des Weiteren zeigen allein die hier betrachteten Projekte, wie unterschiedlich die Visionen von AmI und Ubicomp bei deren Verwirklichung ausfallen können. Getrieben von nicht technischen und meist strategischen Gründen werden jeweils einzelne Aspekte von AmI (Situation-Awareness, integrierte Kommunikationsinfrastruktur, intuitive Interaktionsparadigmen, Media Management) hervorgehoben.
Trotzdem lassen sich durch eine Analyse der Projekte (Visionen, Use Cases, Szenarien, Architektur, Interaktionstechniken) funktionale Systembausteine einer möglichen *AmI-Referenzarchitektur* identifizieren.

Vergleich von SIKOWO und EasyLiving

Bei den impliziten Interaktionsmethoden steht bei beiden Projekten das Mehrkamerasystem im Vordergrund. Es dient zur Personenortung, Mimikerkennung und Helligkeitsmessung, wobei SIKOWO zusätzlich die Personenidentifizierung mit Hilfe des

Kamerasystems durchführt, während EasyLiving dies momentan mittels eines Fingerabdruckscans realisiert. Im Gegensatz zu SIKOWO besitzt EasyLiving überdies Drucksensoren, welche die kamerabasierte Personenortung unterstützen.
Beide Projekte haben das Ziel, die Allgegenwart und das Kontextbewusstsein zu adressieren. Jedoch verschwinden bei SIKOWO die einzelnen Geräte und Komponenten fast komplett aus dem Blickfeld des Benutzers, während bei EasyLiving Komponenten wie Displays, Fernbedienungen oder Tastaturen immer zu sehen sind. Eine Vision ohne Desktop-Computing sowie die transparente Einbettung von Technikkomponenten in die Umwelt wird bei EasyLiving wohl aus unternehmensstrategischen Gründen nicht verfolgt.

Vergleich von HomeLab und inHaus Duisburg

Die Projekte inHaus und HomeLab zeigen die Möglichkeiten einer vernetzten Wohnumgebung auf, in der die Technik in den Hintergrund tritt. Durch Vernetzung eines Hauses werden die vielen Geräte und Dienste mit heterogenen Standards integriert und mittels einheitlicher Benutzerschnittstellen zugänglich gemacht.
Ein wesentlicher Unterschied zwischen den Projekten findet sich in deren Zielsetzung. Eine Gemeinsamkeit ist, dass alle Projekte komplette Häuser umfassen, in denen sich die Probanden längere Zeit aufhalten können.
Das HomeLab von Philips in den Niederlanden ist ein Testobjekt für Ambient Intelligence. Durch die intensive Beobachtung der Testbewohner in einer realistischen intelligenten Wohnumgebung sollen Systemverbesserungen erkannt und neue Anwendungen kreiert werden. Gleichzeitig gilt es, die Durchführbarkeit der Anwendungen zu testen. Philips erhebt dabei den Anspruch, durch seine Aktivitäten die Lebensqualität der Menschen verbessern zu wollen. Dabei wird untersucht, wie die Technik auf den Menschen wirkt und wie er mit ihr umgeht.
Die intelligenten Anwendungen werden im HomeLab um den Menschen herum aufgebaut. Diese Anwendungen sind den Menschen zum Beispiel beim Zugriff auf große Datenbanken behilflich, schaffen eine angenehme Wohnatmosphäre oder unterstützen in Form eines intelligenten Badezimmerspielgels. Dieser zeigt Zeichentrickfilme für die Kinder, teilt mit, wenn die Batterien (z.b: der el. Zahnbürste) fast alle sind und achtet gleichzeitig auf die Gesundheit.
Dagegen werden beim inHaus-Projekt die Anwendungen um das Haus herum mit dem Menschen als Bewohner aufgebaut. Anwendungen werden hauptsächlich entwickelt, die sich auf das Haus als funktionale Einheit beziehen und den Benutzer insofern in seinem Alltag unterstützen wollen, als dass sie die Steuerung und Kontrolle für die Bewohner vereinfachen beziehungsweise übernehmen. Dies liegt aber vor allem an den unterschiedlichen Projektinitiatoren. Während hinter HomeLab ein Elektronikkonzern steht, sind die Partner bei inHaus größtenteils Hausgeräte- und Hauszubehörhersteller, wie Hersteller von Sanitäranlagen oder Türschließsystemen. Akzeptanzuntersuchungen über die neue Technik werden auch im inHaus vorgenommen. Der Aspekt der Kommunikation zwischen Menschen fällt im inHaus weg. Hier steht vielmehr die Kommunikation zwischen Geräten im Vordergrund.

Die zentrale Rolle von Kontext

Eine *wesentliche* Komponente für AmI ist die Context-Awareness, welche in der Architektur von HomeLab, EasyLiving und SIKOWO berücksichtigt und in den Szenarien von inHaus zu erkennen – aber in der Umsetzung nicht vertreten – ist.
Die Wichtigkeit hängt direkt mit der Notwendigkeit von Context-Awareness für die Realisierung von „proaktiver“ Unterstützung des Nutzers, der „Adaptierung“ der Umgebung auf Nutzersituationen sowie der Bereitstellung von „impliziten“ Interaktionsmechanismen zusammen. Diese Eigenschaften sind Teile der Vision von Ambient Intelligence.
Verschiedene Aspekte von Context-Management (inklusive Kontextmodellierung, Kontextdarstellung, Kontexterfassung, Kontextbereitstellung, Sensorentwicklung, etc.) werden aktuell für unterschiedliche Domänen wie z. B. Heim, Büro, Meeting, Vorlesung, Anwendungen für Ältere, Touristik, *Wearable*, Verkehr, Logistik, Produktion von der bereits existierenden *scientific community* erforscht. Eine Übersicht hierzu gibt [SLP04].

4.4.2 Zusammenfassung Context-Awareness

Dieses Kapitel hat die unterschiedlichsten Aspekte von Context-Awareness dargestellt. Eine Betrachtung der vorgestellten Arbeiten macht deutlich, dass solche Modelle nur für klar definierte Anwendungsdomänen erstellt werden können. *Kontext* ist einfach zu komplex, um all seine Facetten (Nutzer, Umgebung, Aktivitäten, Geräte und Dienste, ...) in einem Modell aufnehmen zu können [Shi06].
Eine weitere ungeklärte Frage ist die *Tiefe* des Modells. Inwieweit sollen einzelne Kontextparameter abstrahiert werden?
Betrachten wir als Beispiel den Benutzer. Werden zu einem Benutzer lediglich persönliche Daten und Login-Daten modelliert? Oder sollen auch Präferenzen bezüglich Interaktion sowie konsumierte Unterhaltungsmedien modelliert werden?
Als ein weiteres Beispiel gibt es keine einheitliche Modellierung des Ortes. Während einige Kontextmodelle lediglich symbolische Beschreibung für Räume verwenden, bilden andere Ortsinformationen mathematisch ab, um *Geometric Reasoning* durchführen zu können. Einige, wie das Global Positioning System (GS-NASA), gehen von einem globalen Koordinatensystem aus, während andere, wie das AURA Location Model [JS02], für jeden Raum ein eigenes Koordinatensystem erlauben.

Schwächen existierender Kontextmodelle

Die existierenden Modelle scheinen in einem Entwickler-zentrierten, top-down Ansatz entstanden zu sein. Zumindest lassen sich keine Nutzer-zentrierten oder Aufgaben-zentrierten Design-Methodologien erkennen.

Sie sind meist entstanden, um eine bestimmte Anwendung zu entwickeln. Aufgrund des Top-Down-Ansatzes [1] und fehlender Spezifikationen lässt sich die Übertragbarkeit der Modelle für weitere Anwendungen nicht verifizieren.

[1] Ausgehend von konkreten Projektanforderungen entscheidet der Entwickler, aus welchen Entitäten und Beziehungen das Kontextmodell bestehen soll.

5 Domainanalyse

Ziel der Domainanalyse ist es, notwendige Daten für die Modellierung und Konzeption eines Interaktionssystems („Environment Controllers“) zu gewinnen. Dazu gehören insbesondere Daten über

- Hauptszenarien und Situationen der behandelten Domände (Meeting- und Büroumgebungen)
- Aktivitäten sowie Ressourcen, welche für die Ausführung dieser Aktivitäten benötigt werden
- das Kontextmodell als Grundlage zur Realisierung von AmI-Anwendungen und Interaktionslösungen
- Modell zur Abbildung der Relationen zwischen Geräten und Medien

Aus der Analyse soll ein Umgebungsmodell entstehen, welchem das explizite und implizite System gemeinsam zugrunde liegen soll. Durch die gemeinsame Nutzung soll eine Kommunikations- und Koordinierungsbasis geschaffen werden. Da detailierte Kenntnisse über zu unterstützende Situationen und Proaktivitäten nicht für beliebige Szenarien vorliegen, kann die Vollständigkeit des Modells nicht garantiert werden. Als Ausweg wird folgender Ansatz verfolgt: Zunächst wird ein Modell entworfen, welches für die Entwicklung eines expliziten Assistenzsystems mit adaptiven Schnittstellen vollständig ist. Darüber sollen die Anforderungen aus dem Kapitel 3.3 sowie die Anforderungen an kontextsensitive Anpassung von Benutzungsschnittstellen erfüllt werden können (vgl. Kapitel 3.3.3). Als nächstes wird das Modell derart erweitert, dass es gewisse Basisanforderungen erfüllt. Konkret wird das Model für die Unterstützung von impliziter Interaktion im Rahmen von typischen Präsentationsszenarien vorbereitet. Um die situationsabhängige Unterstützung von ausgewählten Szenarien zu verbessern, soll das Modell basierend auf einer Szenarienanalyse sowie den Erkenntnissen der Domainanalyse erweitert werden. Ein Anspruch auf Vollständigkeit wird hierbei nicht erhoben. Als letztes gilt die Anforderung an die Erweiterbarkeit des Modells. Diese bezieht sich sowohl auf die Spezifikation als auch auf die sensor-technische Realisierung des Kontextmodells.

Die Definition von Situationen oder Situationserkennung selbst liegt nicht im Fokus dieser Arbeit. Daher wird auch der Nachweis für die Vollständigkeit und Korrektheit der erkannten Situationen im Rahmen dieser Arbeit nicht behandelt.

Das Umgebungsmodell soll folgendes erlauben:

- Realisierung des Environment Controllers
- gemeinsame Kontextbasis für sowohl explizite als auch implizite Interaktion
- Koordinierung von expliziten und impliziten Interaktionen

5.1 Ansatz der Domainanalyse

Im Folgenden wird der Ansatz beschrieben, mit dem im Rahmen des beschriebenen Vorhabens die systematische und fundierte Gewinnung von Daten zur Entwicklung eines Umgebungsmodells vorgenommen werden soll.

Die Ziele der dazu aufgenommenen Szenarien- und Domainanalyse gliedern sich wie auch die Analyse selbst in die folgenden zeitlichen Abschnitte, in denen aufeinander aufbauend wichtige Erkenntnisse zur Ausformung eines Ambient- Intelligence Situations- und Umgebungmodells gewonnen werden. Die Ergebnisse stützen sich auf verschiedene Methoden der Datengewinnung und werden in einem benutzerorientierten Ansatz, das heißt durch Quantifizierung der Situations-, Aktivitäts- und Umgebungswahrnehmung, der Gewohnheiten und der Erwartungen von Nutzern erhoben.

Die Analyse – im Folgenden auch Studie genannt – wird in folgenden Schritten durchgeführt (siehe Abb. 5.1) :

- Situationsbestimmung durch Beobachtung und Interpretation gegenwärtiger Veranstaltungen und verfügbarer Technologien innerhalb eines iterativen observativen Prozesses. Dabei wird versucht, die menschliche Sicht und Interpretation der Vorgänge innerhalb bestimmter Szenarien (im ersten Schritt des Szenarios "Präsentation") zu erfassen und zu schematisieren. Ergebnisse sind Beobachtungsprotokolle und Schemata zur Erfassung und händischen Klassifikation von Vorgängen in ausgewählten Szenarien.
- Intentionsbestimmung von Benutzern durch Befragung mittels eines standardisierten Fragebogens, Verfeinerung des Modellentwurfs, Gewinnung einer repräsentativen Datenmenge von situationsabhängigen, intentionsbehafteten Aktivitätsmustern. Auf Grund der gewonnenen Erkenntnisse und durch fortgesetzte Beobachtungen weitere Ausformulierung und Erweiterung des Modellentwurfs.
- Bestimmung der Parameter für erfolgreiche Ansätze für situationsabhängige Pro-Aktivität auf Grundlage des so gewonnenen Erfassungsmusters und in den Grenzen der bekannten Modellparameter. Im Anschluss erfolgt die Ermittelung der charakteristischen Parametergruppen und Gewinnung kritischer Maße an aussagekräftigen Messdaten.

Die hier wiedergegebenen Ergebnisse beziehen sich auf die Abschnitte der Teile 1 und 2 und sind damit als Teilergebnisse der Studie zu verstehen.

5.2 Erste Iteration: Modellentwurf und Testschema in den Szenarien "Präsentation" und "Meeting"

In einem aufgabenorientierten Ansatz und aufbauend auf Annahmen aus dem eigenen Erfahrungsbereich wurde ein Testschema zur Erfassung der aus Nutzersicht

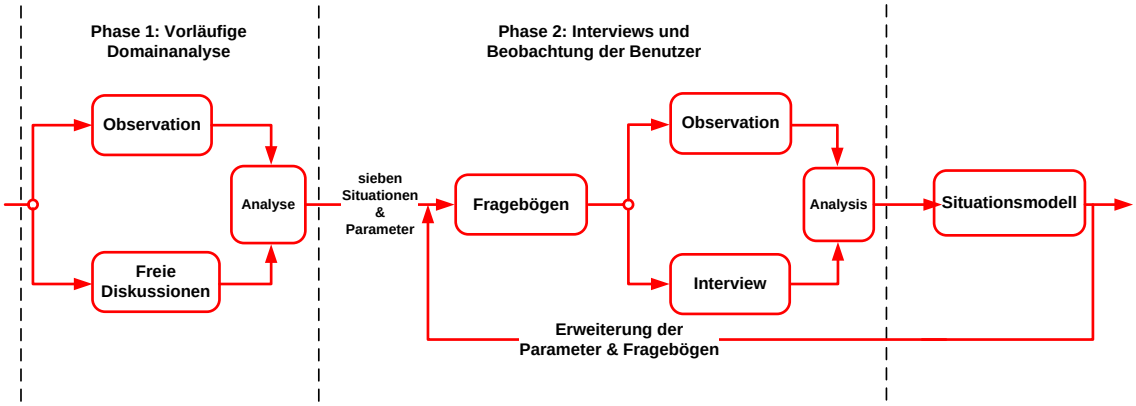


Abbildung 5.1: Iterativer, aktivitätsorientierter Ansatz der Nutzer-Studie.

wesentlichen Parameter von Situationen und Aktivitäten im Szenario "Präsentation" und "Meeting" erstellt. Wesentlich dabei war die Wahl der Vorgehensweise, d. h. die Orientierung an den für die menschliche Interaktion und Situationswahrnemung wesentlichen Parametern. Die Abstraktion der erfassten Modellgrößen auf technische Objekte und Vorgänge wurde bewusst soweit wie möglich aus dieser Phase der Modellgewinnung herausgehalten. Es wurden in der ersten Phase 11 Sitzungen in den Räumlichkeiten des Fraunhofer IGD analysiert.

5.2.1 Erfassungschema

Die Erfassung über dieses Testschema erfolgte durch händisches Erfassen in Form von Skizzen und Notizen. Das urspüngliche Schema wurde iterativ in mehreren Schritten um neu gewonnene und als zur Charakterisierung des Szenarios für wesentlich befundene Parameter erweitert.

Die Abbildung 5.2 zeigt ein Erfassungsschema für die Szenarion „Vortrag" im Raum 074 des Fraunhofer IGD. Hier wird anhand einer abstrahierten Abbildung des Vorlesungssaals und mithilfe verschiedener Symbole für Geräte, Services, Modalitäten, Medienobjekte oder perzeptive Erfassungsgrößen dem Beobachter ein Rahmen gegeben, die für das Verständis und den Ablauf der Vorgänge in einem Raum wichtigen Parameter festzuhalten.

Als eine wichtige Fragestellung bei der Domänenanalyse wurde beobachtet, welche Aktivitäten die Nutzer in Meetingsräumen und Hörsälen durchführen. Die Beobachtung soll helfen, die Aktivitäten aus den folgenden Gesichtspunkten heraus zu verstehen, um daraus Anforderungen zum Entwurf eines Environment Controllers zu gewinnen:

Was für Aktivitäten führen Nutzer in einer bestimmten Situation aus? Welche Situationen und Aktivitäten exisitieren? Welche Übergänge und Abfolgen zwischen Situationen oder Aktivitäten (kausal und zeitlich) gibt es? Welche Entitäten spielen bei der Durchführung einer Aktivität eine Rolle (Bewegung, Aufmerksamkeit, Lautstärkepegel, Positionen, Intentionen (als Schätzung aus Beobachtersicht))?

Wie führen die Nutzer diese Aktivitäten aus? Hierbei werden folgende Gesichtspunkte analysiert:

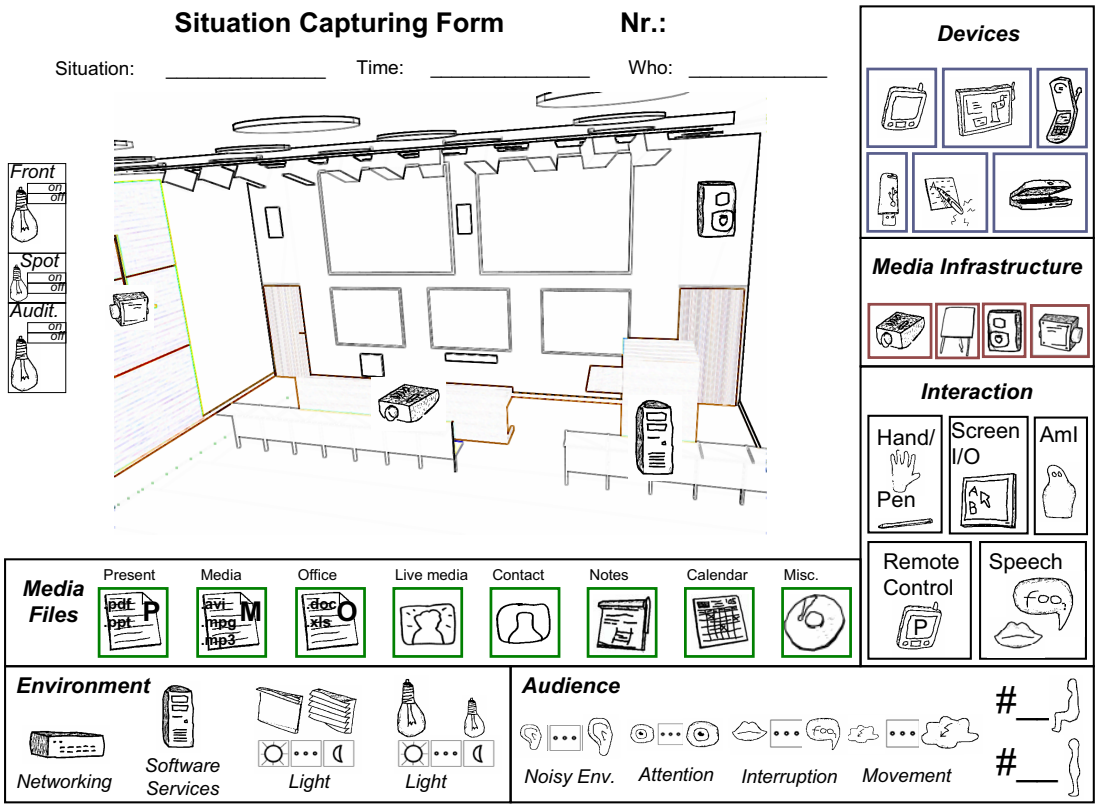


Abbildung 5.2: Erfassungsschema des Vorlesungssaals des Fraunhofer IGD.

- Welche Modalitäten *können* allgemein für eine Aktivität benutzt werden?
- Welche Interaktionssysteme bzw. Modalitäten werden vom Nutzer für die Durchführung einer bestimmten Task *bevorzugt*? Durch diese Analyse kann man die von einem Environment Controller zu unterstützenden Tasks ermitteln. Diese Tasks dienen dann als Grundlage zur Konzeption eines Environment Controllers (vgl. Kapitel 7.2.5) bzw. zum Entwurf des Interaktionsmodells (vgl. Kapitel 7.2).
- Welche Interaktionsmetaphern (implizit, explizit, zielbasiert, funktionsbasiert) sind für eine bestimmte Aktivität in einer Situation akzeptabel?
- Welche Geräte werden für die Durchführung einer Aktivität benutzt? Beispielsweise benötigt man für das „Starten einer Präsentation“ notwendigerweise Ausgabegeräte.
- Welche Medien(typen) sind für die Durchführung einer Aktivität notwendig?

Wo führen die Nutzer diese Aktivitäten aus? Welche Position(*en*) nimmt der Nutzer (Ausführender einer Aktivität) dabei ein?

Es gilt, im Allgemeinem die Beziehungen zwischen Aktivität, Geräten, Dokumenten sowie (funktionalen) Diensten zu analysieren, um daraus Anforderungen an einen Environment Controller zu gewinnen.

Die Typologien, welche innerhalb dieses Erfassungschemas zur Anwendung kommen, entstanden durch mehrfache Iteration und Verbesserungen im Rahmen zahl-

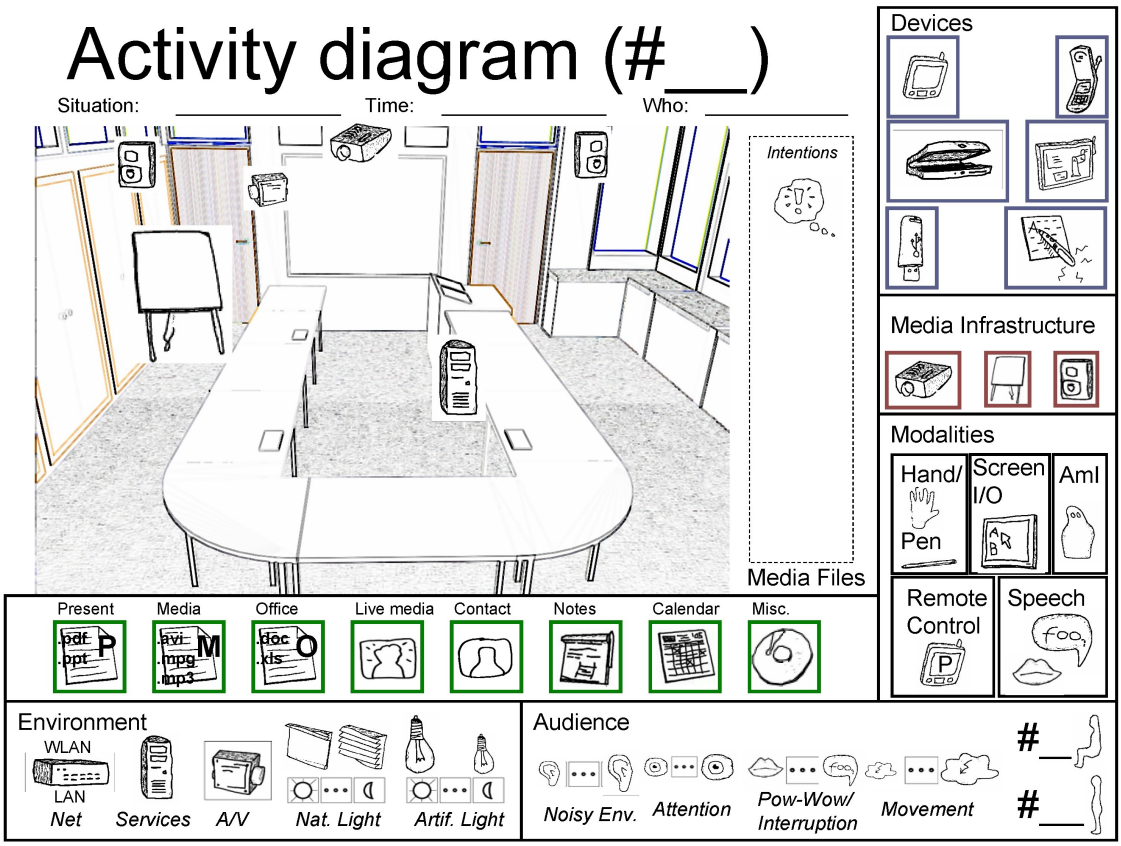


Abbildung 5.3: Testschema “Meeting“: Dieses Erfassungsschema bildet die o. g. Typologien auf Vorgänge in typischen Meetingräumen ab. Grundlage hierzu bildet ein abstrahiertes Modell des im Hause vorhandenen Telekonferenzraums “Raum 220“.

reicher Erfassungsvorgänge. Um parallel auch weitere Einflussgrößen weiter gefasster Szenarien aufzunehmen, wurde im gleichen Sinne auch ein Testschema zu dem übergeordneten Szenario “Meeting“ angefertigt und bei zahlreichen Gelegenheiten zur Erfassung eingesetzt.

5.2.2 Ergebnisse

Ziel und Teilergebnis dieser ersten Phase der Domainanalyse war die Klassifikationen der Akteure, Objekte, verwendeten Modalitäten und erkannten Situationen bzw. Aktivitätsgruppen, sowie die im Szenario “Präsentation“ auftretenden Situationen aus Nutzersicht. Soweit möglich, wurden offensichtliche Intentionen der Nutzer festgehalten oder auch auftretende Problemsituationen erfasst. Die hier beschriebenen Erfassungsschemata, s. Abb. 5.2 und 5.3, sind selbst Teil des Ergebnisses dieses Arbeitsschrittes.

Des Weiteren wurden *sieben* hauptsächliche Situationen für die Präsentationsszenarion identifiziert : *Setup*, *Welcome*, *Presentation*, *Explanation*, *Question & Answer*, *Discussion* und *Conclusion*. Zusätzlich wurden folgender Parameter (siehe Abb. 5.4) identifiziert, welche es erlauben, die o. g. Situationen eindeutig zu beschreiben. Dazu gehören: *Activity, Location, Interaction, Media, Device, Environment und Audience states (z. B. light setting und noise level, user movement, number of person,*

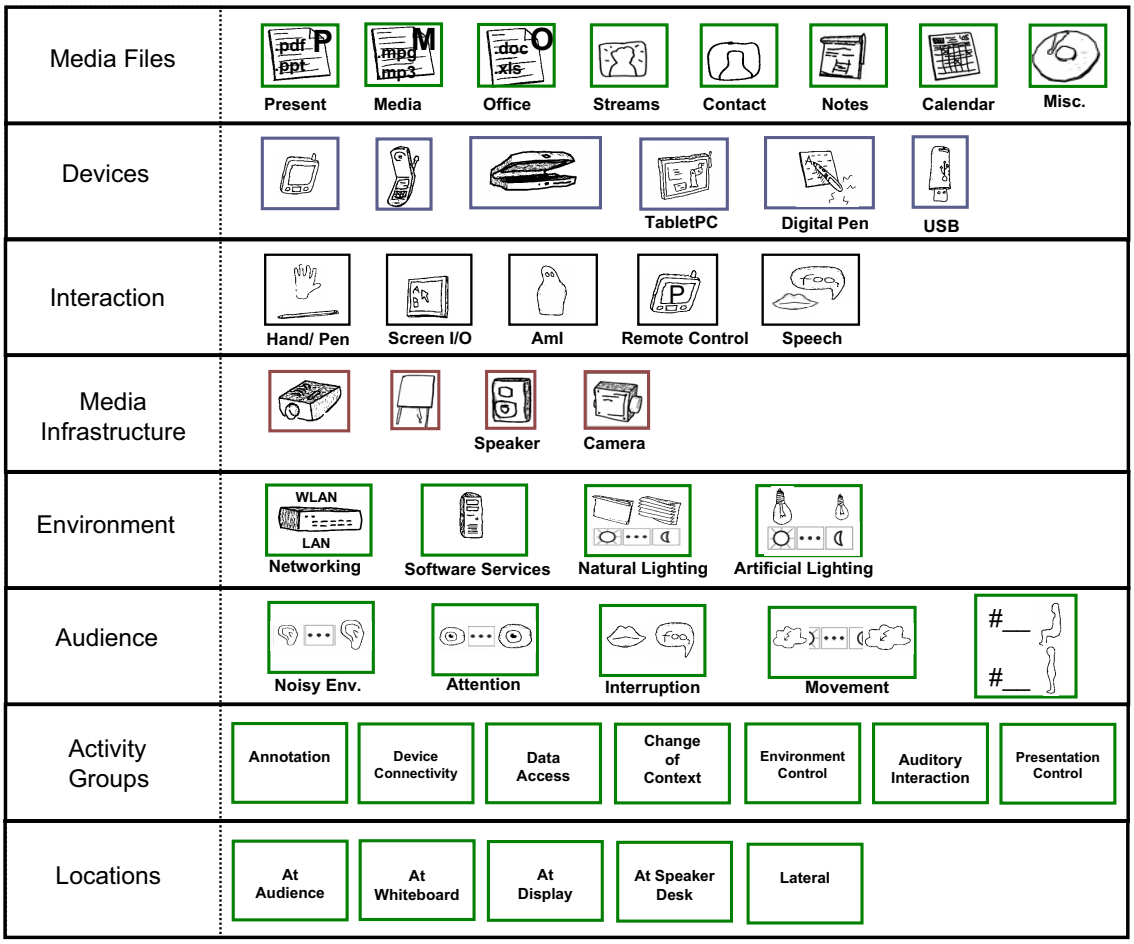


Abbildung 5.4: Im Rahmen dieser Analyse wurden für das weitere händische Erfassungsschema elementare “Systemparameter“ festgelegt.

level of attention, occurring interruptions).
Diese Parameter sind charakteristische Merkmale, mit denen Situationen und Aktivitäten von Präsentationsszenarien beschrieben werden können.

5.3 Zweite Iteration: Verfeinerter Modellentwurf, Erstellung situativ geführter Fragebögen

Aufbauend auf den Erkenntnissen der Testschemata erfolgte eine Analyse der Nutzungsgewohnheiten und Erwartungen der Anwender im Präsentationsszenario. Ziel der im Modus eines geführten Fragebogens durchgeführten Untersuchung war u. a. die Beantwortung folgender Kernfragen:

- Welche Geräte, Medientypen, Services und Interaktionen kommen im Szenario vor?
- Welche Objekte, Geräte und Services sieht der Nutzer als besonders wichtig an?
- Welche Aktivitäten führt der Nutzer in bestimmten Szenarien aus, welche sieht er als besonders kritisch an?
  - Welche Services, Medien, Geräte verwendet der Nutzer dabei?

  - Welche Positionen werden bevorzugt eingenommen?
  - Welche Steuerungsmöglichkeiten werden vom Nutzer wahrgenommen?
  - Welche Interaktionsmetaphern werden bevorzugt?
  - Welche pro-aktive Unterstützung erwartet der Nutzer in einer bestimmten Situation?

- Wie können die Antworten zu diesen Fragen bei der Erfassung und Auswertung der Parameter des Umgebungsmodells genutzt werden?

Zusätzlich wurden Hintergrundinformationen über Umfeld, Level und Anlass der jeweiligen typischen Präsentationstätigkeiten sowie beispielsweise der technischen Vorbildung der Nutzer aufgenommen.

a) b)

Abbildung 5.5: Durchführung der Domainanalyse. Die Bilder (a) and (b) zeigen die Situationen *Setup* and *Presentation*.

Die zweite Phase versucht, Daten für eine *semi-formale* Beschreibung von Situationen zu erheben. Die Datenerfassung erfolgte durch zwei Experimentatoren mittels manueller Datenerhebung mit Formularen und Notizen. Über eine Periode von 24 Monaten wurden Beobachtungen in *real-life* Präsentationen in mehreren Organisationen durchgeführt. In dieser Phase wurden mehr als 81 Sitzungen analysiert. Diese wurden in zwei verschiedenen Räumen des Fraunhofer IGD durchgeführt: in einem typischen Meetingraum und einem Hörsaal mit einer Kapazität von circa 100 Personen (vgl. Abb. 5.5). Des Weiteren wurden die Räume der TU Darmstadt sowie die Konferenzräume der INI-GraphicsNet Stiftung verwendet. Zwei Sitzungen wurden in den Räumen des Fraunhofer IPSI durchgeführt. Gegenstand der Beobachtungen waren nur Präsentationsszenarien. Andere Szenarien wie Brainstorming, informelle Meetings, oder Video-Konferenz wurden nicht behandelt. Die Präsentationen fanden in der Regel im Rahmen von internen Projekt-Meetings, Diplomarbeitsvorträgen, Vorlesungen sowie regulären Abteilungssitzungen statt. Die meisten Vortragenden waren Forscher oder Mitarbeiter der jeweiligen Organisation. Sie waren also mit der vorhandenen Geräte-Infrastruktur und den Bedienungssystemen vertraut. Ein kleiner Anteil von Beobachtungen bezog sich auf externe Vorträge, wo die Sprecher die vorhandene Umgebung nicht kannten.

Diese Beobachtung wurde zusätzlich durch eine Videoaufzeichnung für spätere Offline-Analysen unterstützt. Im Rahmen der Beobachtungen haben die Experimentatoren mit den Vortragenden Diskussionen geführt. Dabei ging es um die oben erwähnten Fragestellungen.

5.3.1 Geführte Fragebögen und Observationen

Zur Beantwortung dieser Fragen wurde zunächst ein Fragebogen erstellt, welcher sich primär am Aufbau und den Möglichkeiten des im Fraunhofer IGD vorhandenen "Raum 074" und an den dort vorhandenen medialen und infrastrukturellen Einrichtungen orientiert. Diese Fragebögen wurden dann für die anderen Umgebungen angepasst.

Als Form der Befragung diente ein geführter Fragebogen, um beispielsweise auch dem in der Thematik von Ambient-Intelligence-Unterstützung von Räumen unerfahrenen Nutzer eine Einführung in die Typologien, Begriffe und Ziele der Studie zu geben. Dazu wurden durch einführende Erläuterungen und einzelne Multimedia-Elemente Hilfestellungen gegeben, s. Abb. 11.1. Die Beantwortung erfolgte in Form von Multiple-Choice-Elementen, s. Abb. 11.2, oder durch Handskizzen, s. Abb. 11.3.

- Es sind hauptsächlich Präsentationen im Rahmen einer dienstlichen Veranstaltung.
- Eine Präsentation dauert ca. 10 bis 30 Minuten.
- Die befragten Personen halten ca. eine Präsentation im Monat.
- Zielgruppe dabei sind zumeist eigene Kollegen.
- Die Größe der Zuhörerschaft liegt zwischen 10 und 100.
- Inhalt und Ausrichtung der Präsentation ist meistens akademisch, Projektmeeting sowie *business* Präsentation.
- Die meisten Präsentationen finden lokal statt.

5.3.2 Ergebnisse

Die Hauptfragestellung dieser Untersuchung zielte neben grundlegenden Teilfragen nach Situationen, Aktivitäten, Geräten, Medien und Ausstattung innerhalb des Szenarios vor allem auf die Ermittelung eines Messraums mit stichhaltigen Parametern zur späteren automatischen Situationserkennung ab.

Um die Eignung des gewählten Erfassungsschemas für die weiterführende Klassifikation und v. a. der erfolgreichen Situationserkennung zu prüfen, wurde für die im Rahmen des Fragebogens erhobenen Aussagen eine statistische Auswertung durchgeführt, welche situationsweise die charakteristischen Aussagen nach relativen Häufig- keiten als Merkmalsvektor darstellt.

Hierzu wurden in einem ersten Ansatz die Aussagen zu

- Aktivitäten (in Aktionsgruppen zusammengefasst)
- Sprecherposition (Region)

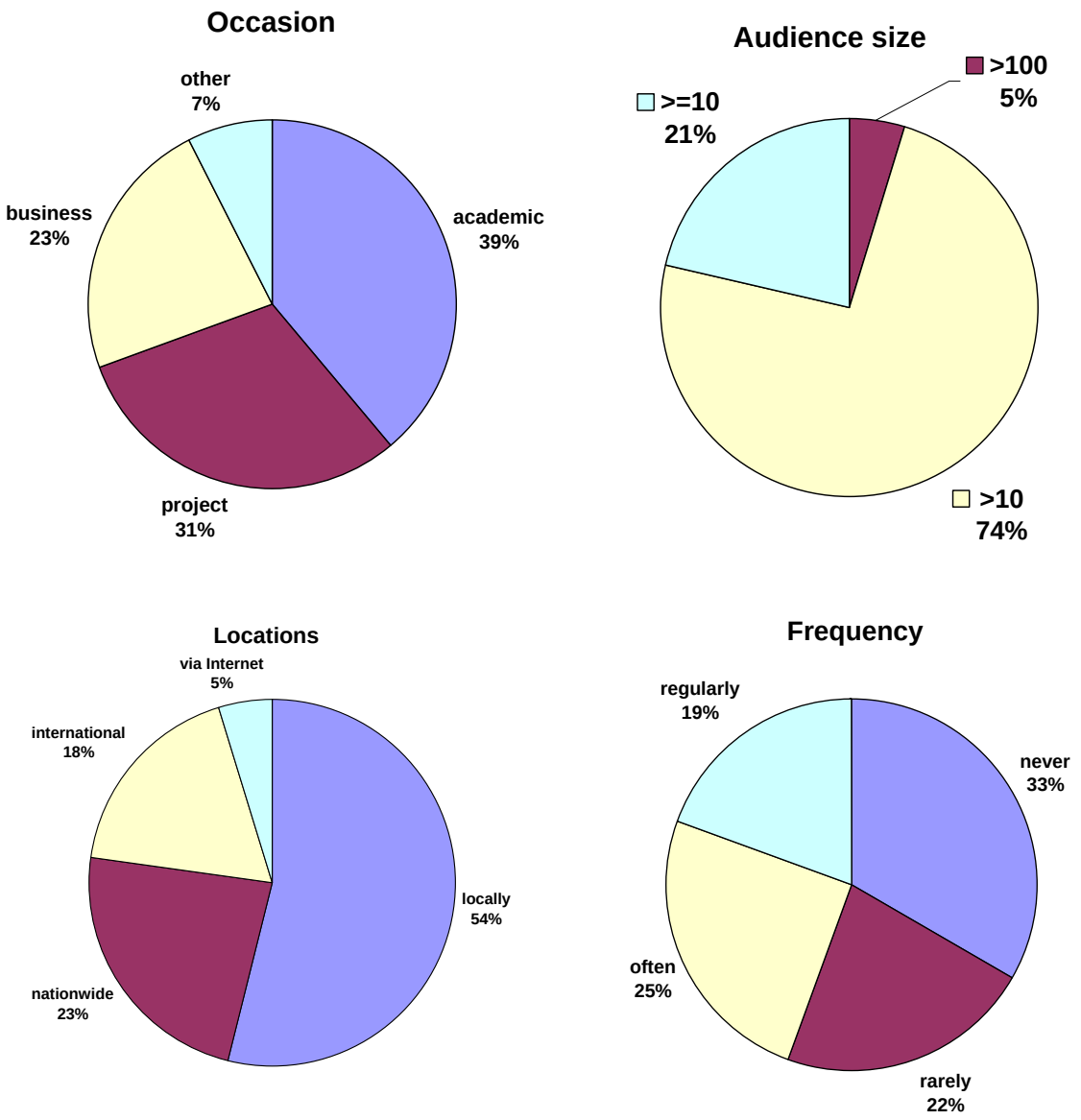


Abbildung 5.6: Hintergrund-Informationen über Testpersonen sowie beobachtete Vortragende.

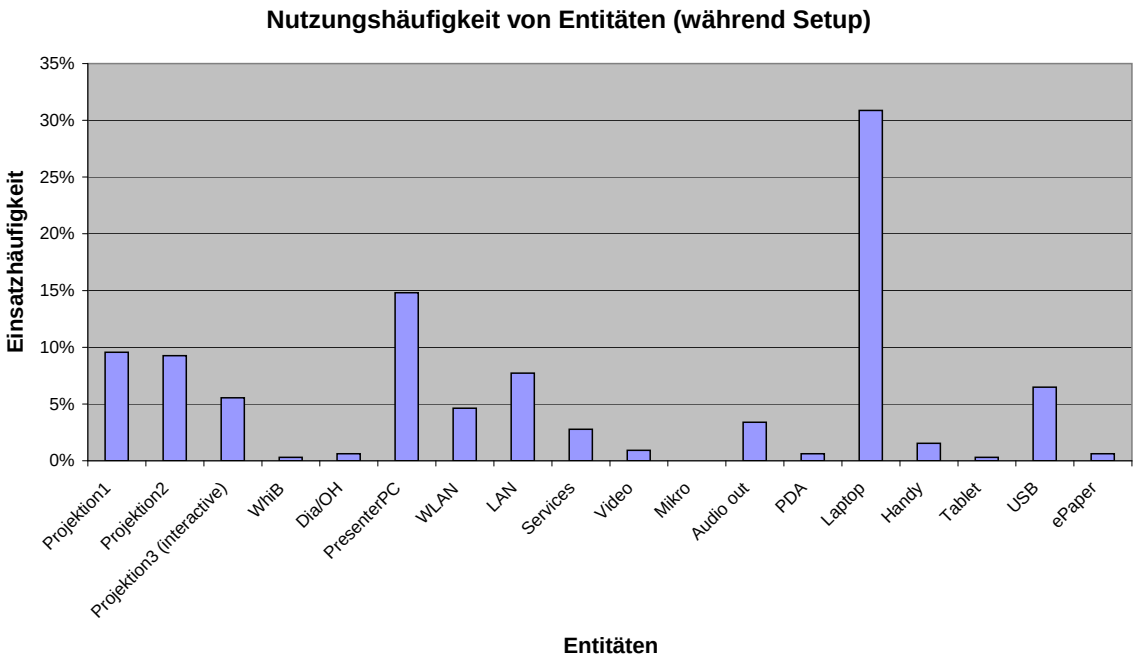


Abbildung 5.7: Häufigkeit, mit der Geräte im Rahmen eines Präsentationsszenarios in einem Hörsaal eingesetzt werden.

- und jeweils verwendeter Modalität

für die Charakterisierung der jeweiligen Situation herangezogen.

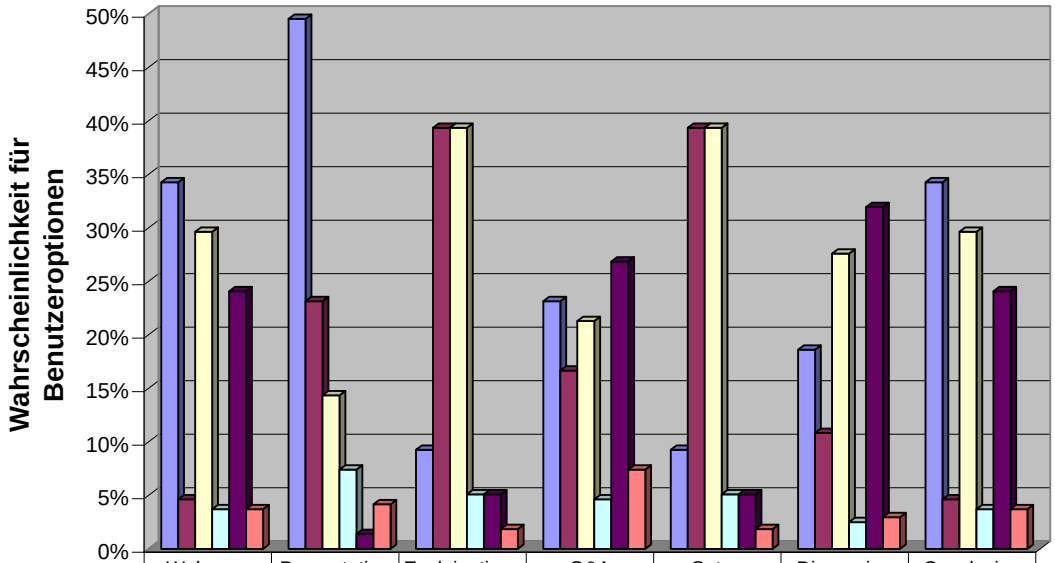
Die Auswertung der gewonnenen Merkmalsvektoren und ihre Zuordnung zu den vorgegebenen Situationen im Szenario erbrachte auf einer nichttechnischen, abstrakten Ebene ein erstes Bewertungsmaß für die Situationserkennung.

Ergebnis sind mehrdimensionale Merkmalsvektoren in Form von Häufigkeitsverteilungen über die erfassten und abgeleiteten Parameter. In Kombination mit den erweiterten Aufzeichnungen der Analyseschemata und folgender, erweiterte Erfassungsmethoden, wurde im weiteren Verlauf der zur Situationserkennung herangezogene Merkmalsraum noch erweitert.

Durch die Domainanalyse wurden folgende Ergebnisse und Erkentnisse gewonnen:

Situationen Im ersten Schritt der Analyse sind die Hauptsituationen untersucht und erkannt worden: *Setup*, *Welcome*, *Presentation*, *Explanation*, *Question & Answer*, *Discussion* und *Conclusion*.

Eindeutige Situationsbeschreibung Das sind mehrdimensionale Merkmalsvektoren, in Form von Häufigkeitsverteilungen über die erfassten und abgeleiteten Parameter. Diese Art der Klassifikation der Situationen mittels erläuterter Parameter erlaubt, die *sieben* Situationen eindeutig voneinander zu unterscheiden. Aus Tabelle 5.1 sowie den grafischen Darstellungen lässt sich leicht ablesen, dass bei den meisten Übergängen eine hohe Unterschiedlichkeit erkennbar ist, was in diesem Stadium der Untersuchung als eine prinzipielle Bestätigung des Analyseschemas angesehen werden kann. Damit ist die Grundlage für eine Situationserkennung und für die Realisierung von impliziten Interaktionslösungen gegeben. Niedrigere Werte, etwa bei den Übergängen zwischen "Presentation", "Question&Answer" oder "Explaination" korrelieren mit der faktischen Ähnlichkeit dieser Situationen.



	Welcome	Presentation	Explainations	Q&A	Setup	Discussion	Conclusion
At Podium	34,26%	49,54%	9,26%	23,15%	9,26%	18,61%	34,26%
At Interactive Display	4,63%	23,15%	39,35%	16,67%	39,35%	10,83%	4,63%
At White Board	29,63%	14,35%	39,35%	21,30%	39,35%	27,59%	29,63%
At Lateral	3,70%	7,41%	5,09%	4,63%	5,09%	2,50%	3,70%
At Frontal	24,07%	1,39%	5,09%	26,85%	5,09%	31,94%	24,07%
Other Locations	3,70%	4,17%	1,85%	7,41%	1,85%	2,96%	3,70%

Situationen

Abbildung 5.8: Die in einer bestimmten Situation von Nutzern eingenommenen Positionen.

Charakteristische Parameter Diese können mittels vorgestellter Parameter klassifiziert werden (siehe Abb. 5.4). Dazu gehören: *Activity, Location, Interaction, Media, Device, Environment und Audience states (z. B., light setting und noise level, user movement, number of person, level of attention, occurring interruptions).*

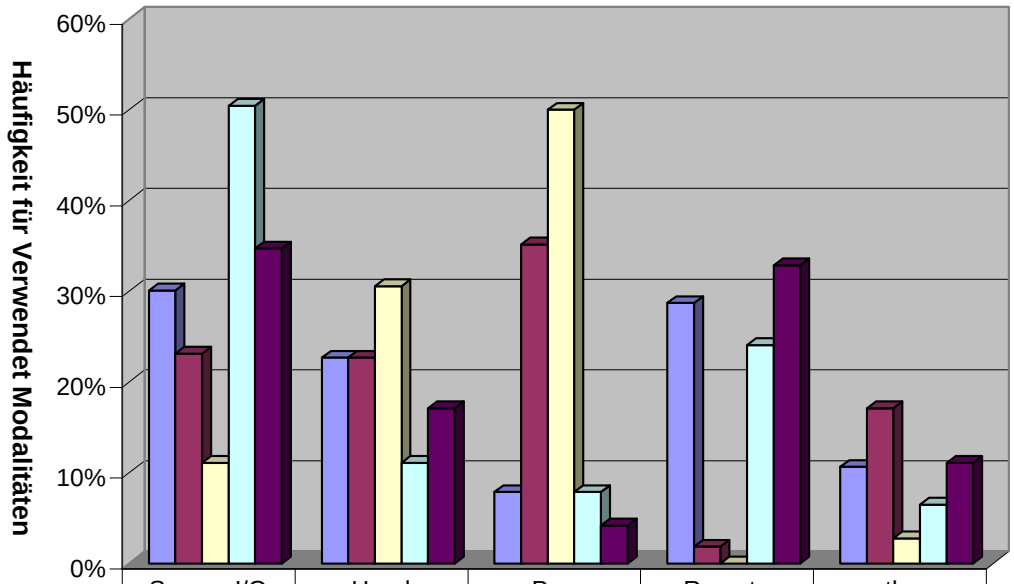
Diese Parameter sind charakteristische Merkmale für eine Situation, mit der Situationen und Aktivitäten von Präsentationsszenarien beschrieben werden können.

So zeigt zum Beispiel die Abbildung 5.8, welche Nutzer-Positionen für eine bestimmte Situation charakteristisch sind. Des Weiteren zeigt die Abbildung 5.9, welche Interaktionsmodalitäten für die Durchführung einer bestimmten Aktivität vom Nutzer bevorzugt werden.

Diese Parameter und die Erkenntnisse zu deren Zusammenhängen bilden eine Grundlage für den Entwurf eines Kontext- bzw. Umgebungsmodells (vgl. Kapitel 6). Dieses wird im weiteren Verlauf dieser Arbeit vorgestellt werden.

Aktivitäten Die Erkenntnisse über typische Aktivitäten und Aufgaben der Nutzer in der analysierten Domäne sowie die Beziehung der Geräte, Medien sowie Orte zu einer Aktivität dienen als Grundlage für ein funktionales Modell (vgl. Kapitel 7.1) bzw. eine hierarische Taskanalyse (HTA), welche im weiteren Verlauf dieser Arbeit vorgestellt wird. Die HTA bildet widerum die Grundlage für den Entwurf des Bedienungskonzepts und des Environment Controllers.

Geräte und Infrastruktur Die Abbildung 5.7 zeigt die in der analysierten Domäne



	Screen I/O	Hand	Pen	Remote	other
change-o-slide	30,09%	22,69%	7,87%	28,70%	10,65%
annotation	23,15%	22,69%	35,19%	1,85%	17,13%
free sketch	11,11%	30,56%	50,00%	0,00%	2,78%
data access	50,46%	11,11%	7,87%	24,07%	6,48%
change-o-context	34,72%	17,13%	4,17%	32,87%	11,11%

Modalitäten

Abbildung 5.9: Die für eine bestimmte Aktivität vom Nutzer bevorzugten Interaktionsmodalitäten.

eingesetzten Geräte sowie die Häufigkeit, mit der diese Geräte im Rahmen von Präsentationsszenarien eingesetzt werden. Diese Informationen werden benötigt, um eine Umgebung dynamisch nach vorhandenen Geräten zu durchsuchen (Device Discovery) und dabei die relevanten Geräte in der Bedienoberfläche aufzunehmen. Dieser Vorgang wird in Kapitel 7.2.3 beschrieben.
Die Abbildung 5.7 zeigt, wie häufig bestimmte Geräte im Rahmen einer Präsentation benutzt werden. Diese Information ist zum einen für den Entwurf eines Bedienungskonzepts relevant, da es Auskunft darüber gibt, welche Geräte für eine Aktivität vom Typ „Präsentieren“ in Frage kommen. Insbesondere hilft diese Information bei der Interpretation von *Makros*.

Dabei möchte der Nutzer eine bestimmte Datei auf ein Makro „Präsentieren“ ziehen und dem System überlassen, auf welchen Geräten dieser bestimmte Typ von Dokument präsentiert wird. Für diese Entscheidung ist auch die Relation zwischen Medientyp und Präsentationsgerät wichtig, was die Abbildung 7.3 verdeutlicht.
Zum anderen sind diese Informationen für die Automatisierung von Bedienoberflächen wichtig. Als Beispiel soll die Bedienoberfläche die in Frage kommenden Geräte für die Aktivität „Präsentieren“ automatisch anzeigen, sobald der Nutzer einen bestimmten Dokumententyp selektiert hat.

Medien Abbildung 5.10 zeigt die im Rahmen einer Präsentation verwendeten Medien-Typen. Dabei kommen am häufigsten Präsentationsmedien wie PDF sowie PowerPoint-Folien zum Einsatz. Als nächstes folgen Audio-Video-Medien wie

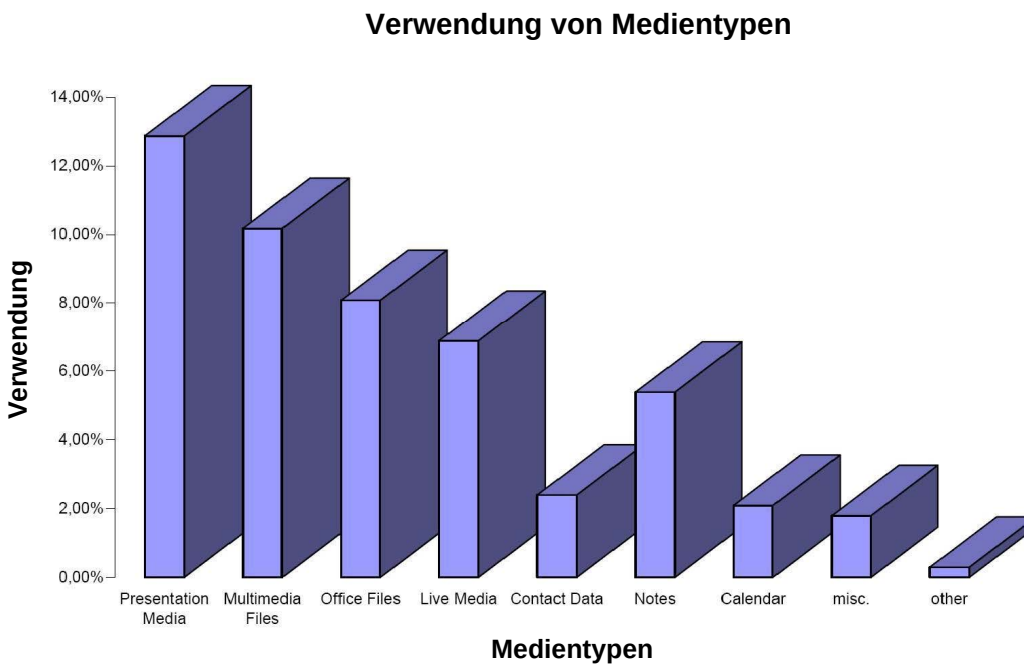


Abbildung 5.10: Welche Medientypen werden wie häufig (in %) in Präsentationsszenarien eingesetzt?

Filme und Grafiken. An dritter und vierter Stelle stehen Office-Dokumente wie Word und Excel sowie Live Media. Live Media kommen nur im Rahmen einer Video-Konferenz in Frage. Diese Informationen sind für eine automatische Vorauswahl der *wichtigsten* Medien für den Nutzer von Bedeutung, sodass eine Bedienoberfläche stets die für eine bestimmte Situation wichtigen Dokumententypen einer Medienquelle anzeigen und den Rest der Dokumente per Default ausblenden kann.

5.3.3 Bewertung der Ergebnisse

Teil der Auswertung war die Überprüfung der Stichhaltigkeit der Parameterauswahl im Sinne ihrer Aussagekraft für eine spätere Situationsanalyse. Dazu wurden die durch Auswertung der Befragungergebnisse erfassten Häufigkeitsverteilungen der Merkmalsvektoren auf ihre Abstände, d. h. ihr Maß an Unterschiedlichkeit, hin untersucht.

Zur Bewertung der Abstände der Häufigkeitsverteilungen der Merkmalsvektoren zweier Situationen wurde jeweils der Betrag der Differenzvektoren zwischen Messvektoren zweier unterschiedlicher Situtationen herangezogen. Dies wurde für alle Paare von Situationen nach dem Prinzip der Quadratischen Abstandsbewertung (RMS) durchgeführt:

$$Abstand(i,j) = \sqrt{\sum_{n=1}^{m} (V_n(i) - V_n(j))^2/m}$$

,

mit $V_n(i)$, dem n-ten Element des statistischen Merkmalsvektors für Situation i und Abstand(i,j), der Distanz der Merkmalsvektoren von Situation i und j. Die Er-

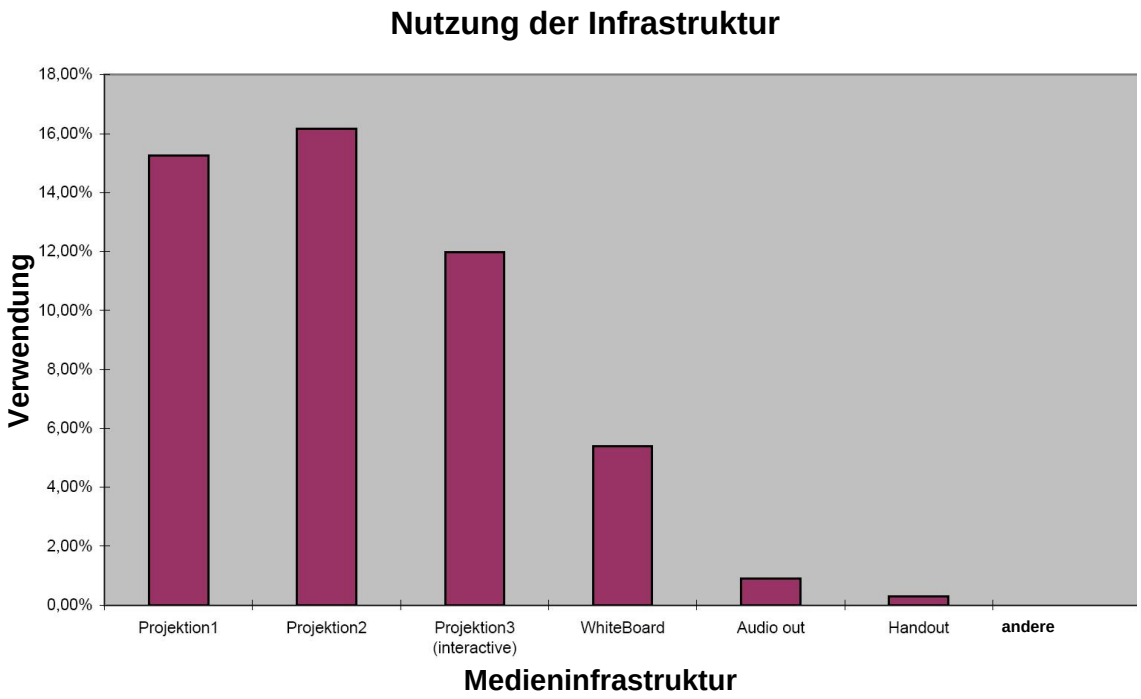


Abbildung 5.11: Wie häufig werden bestimmte Geräte für eine Präsentation benutzt?

gebnisse lassen sich in folgender Tabelle zusammenfassen: Aus Tabelle 5.1 sowie den grafischen Darstellungen lässt sich leicht ablesen, dass bei den meisten Übergängen eine hohe Unterschiedlichkeit erkennbar ist, was in diesem Stadium der Untersuchung als eine prinzipielle Bestätigung des Analyseschemas angesehen werden kann. Niedrigere Werte, etwa bei den Übergängen zwischen "Presentation", "Question & Answer" oder "Explaination" korrelieren mit der faktischen Ähnlichkeit dieser Situationen.

Abb. 5.12 zeigt die „Entscheidungssicherheit" der Situationserkennung bei einem typischen Durchgang durch alle berücksichtigten Situationen („Setup" bis „Conclusion"). Im Bereich der „ähnlichen" Situationen, z. B."Presentation" und „Explaination" (31,85%), brechen die Abstände drastisch ein, was bei einer späteren technischen Umsetzung in einem Schätzverfahren zu Fehlern führen kann. Hier müssen in den folgenden Analysephasen für eine zuverlässige automatisierte Bewertung und Schätzung der Situation noch weitere Informationsquellen für den Merkmalsraum bestimmt und für Schätzungsverfahren herangezogen werden.

5.4 Zusammenfassung und Abgrenzung

Im Rahmen der ersten Phase wurde aufbauend auf Annahmen aus dem eigenen Erfahrungsbereich ein Testschema zur Erfassung der aus Nutzersicht wesentlichen Parameter von Situationen und Aktivitäten im Szenario "Präsentation" und "Meeting" erstellt. Die Erfassung über dieses Testschema erfolgte durch händisches Erfassen in Form von Skizzen und Notizen. Es wurden in der ersten Phase 11 Sitzungen in den Räumlichkeiten der Fraunhofer IGD analysiert. Als eine wichtige Fragestellung bei der Domänenanalyse wurde beobachtet, welche Aktivitäten die Nutzer in Mee-

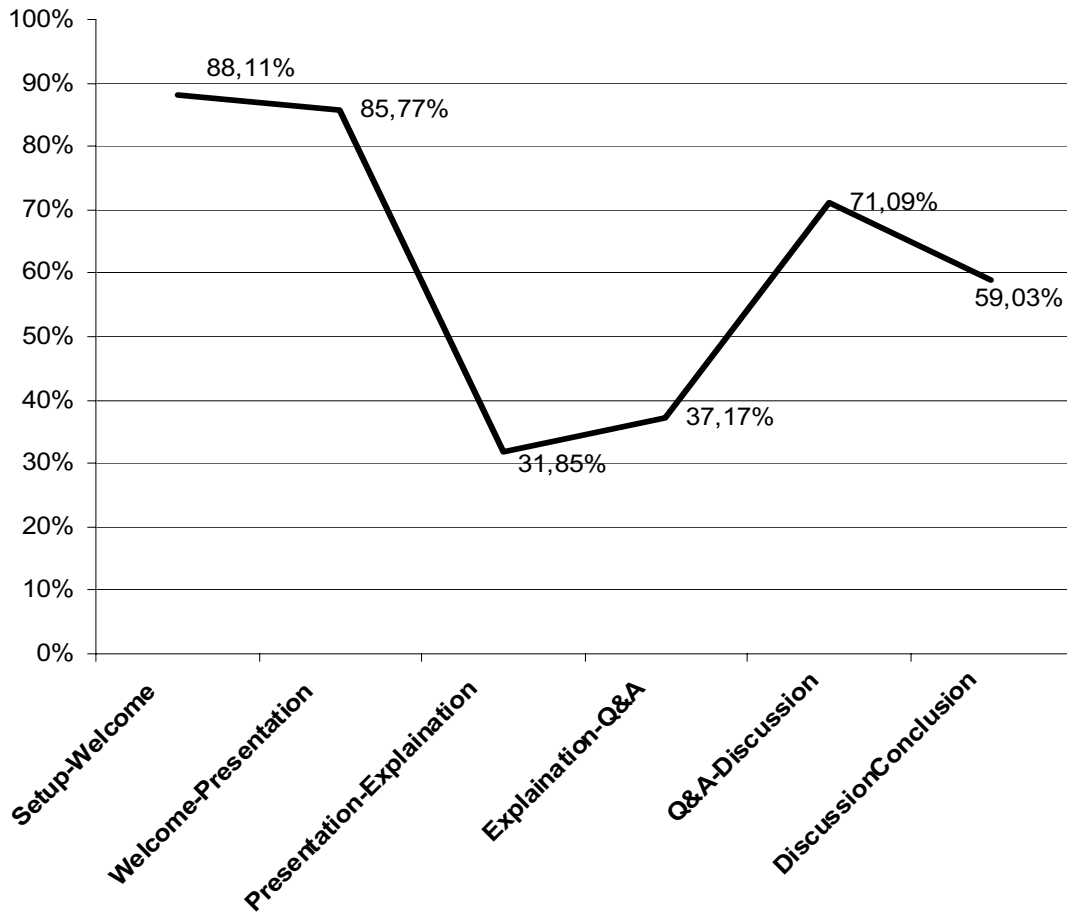


Abbildung 5.12: Abstände der Messvektoren für die Übergange innerhalb einer typischen Abfolge: Setup - Welcome - Presentation - Explaination - Q+A - Discussion - Conclusion

tingsräumen und Hörsälen durchführen. Die Beobachtung soll helfen, die Aktivitäten aus den folgenden Gesichtspunkten heraus zu verstehen. Teilergebnis dieser ersten Phase der Domainanalyse war die Klassifikationen der Akteure, Objekte, verwendeten Modalitäten und erkannten Situationen bzw. Aktivitätsgruppen sowie der im Szenario "Präsentation" auftretenden Situationen aus Nutzersicht.
Des Weiteren wurden *sieben* hauptsächliche Situationen für Präsentationsszenarien identifiziert. Zusätzlich wurden Parameter (siehe Abb. 5.4) ermittelt, welche es erlauben, die o. g. Situationen eindeutig zu beschreiben.

Die zweite Phase versucht, Daten für eine *semi-formale* Beschreibung von Situationen zu erheben. Die Hauptfragestellung dieser Untersuchung zielte neben grundlegenden Teilfragen nach Situationen, Aktivitäten, Geräten, Medien und Ausstattung innerhalb des Szenarios vor allem auf die Ermittelung eines Messraums mit stichhaltigen Parametern zur späteren automatischen Situationserkennung ab. Die Datenerfassung erfolgte durch zwei Experimentatoren. Über eine Periode von 24 Monaten wurden Beobachtungen in *real-life*-Präsentationen in mehreren Organisationen durchgeführt. In dieser Phase wurden mehr als 79 Sitzungen analysiert. Ziel der im Modus eines geführten Fragebogens getätigten Untersuchung war u. a. die Beantwortung folgender Kernfragen:

- Welche Geräte, Medientypen, Services und Interaktionen kommen im Szenario vor?

RMS	Setup	Welcome	Present.	Explan.	Q&A	Disc.	Concl.
Setup	0,00%	88,11%	68,29%	72,50%	80,62%	64,28%	44,43%
Welcome	88,11%	0,00%	85,77%	76,92%	66,00%	97,58%	74,10%
Presentation	68,29%	85,77%	0,00%	31,85%	56,15%	48,01%	58,01%
Explaination	72,50%	76,92%	31,85%	0,00%	37,17%	58,91%	69,23%
Q+A	80,62%	66,00%	56,15%	37,17%	0,00%	71,09%	77,12%
Discussion	64,28%	97,58%	48,01%	58,91%	71,09%	0,00%	59,03%
Conclusion	44,43%	74,10%	58,01%	69,23%	77,12%	59,03%	0,00%

Tabelle 5.1: Paarweise prozentuale Bewertung der Distanzen von Messvektoren bestimmter Situationen.

- Welche Aktivitäten führt der Nutzer in bestimmten Szenarien aus?
- Wie können die Antworten zu diesen Fragen bei der Erfassung und Auswertung der Parameter des Umgebungsmodells genutzt werden?

In Rahmen der vorliegende Arbeit konnte gezeigt werden, dass die verschiedenen Situationen des Präsentionsszenarien mit den Parametern *Ort* des Benutzers, ausgeführte *Aktivitäten* sowie die verwendete *Interaktionsform* eindeutig beschrieben werden können. Die Besonderheit dabei ist, dass eine eindeutige Beschreibung von Situationen auf einer semi-formalen Ebene und unabhängig von speziellen Sensoren und Erfassungstechnologien möglich wird.
Des Weiteren führt diese Arbeit ein verbessertes Situationsmodell für die Präsentationsszenarien ein. Die Aufteilung der Präsentationsszenarios in sieben Situationen scheint insofern richtig zu sein, dass diese Unterteilung mit existierenden Klassifikationen (vgl. Nijholt et al. (2004) [NodAH04]) große Überlappung hat. In der Art der Beschreibung und Ermittlung unterscheidet sich jedoch die vorliegende Arbeit, da die hier durchgeführte Untersuchung auf einer semi-formalen Ebene und unabhängig von speziellen Sensoren und Erfassungstechnologien erfolgte.

Die gewonnenen Erkenntnisse aus der Domainanalyse fließen in die nächsten Kapitel ein. Hauptsächlich sind die Kapitel zur Modellierung der Umgebung (Kapitel 6), Task-Analyse (Kapitel 7.1) und Konzeption eines Environment Controllers (Kapitel 7.2.5) betroffen. In letzterem werden insbesondere die Erkenntnisse und Anforderungen bezüglich Medien, Geräte und physikalische Speicherorte von Dokumenten verarbeitet.

6 Ontologie und Umgebungsmodell

Dieses Kapitel präsentiert Ontologien und Kontextmodelle. Ein detailliertes Verständnis über die Umgebung des Nutzers sowie über den Interaktionskontext ist eine Vorraussetzung für den Entwurf von intuitiven Interaktionslösungen. Die hierbei entstehenden Ontologien und Kontextmodelle sind eine Grundlage für die Entwicklung von Konzepten und Algorithmen im nächsten Kapitel dieser Arbeit.

Wichtiges Ziel dieser Dissertation ist die Entwicklung von intuitiven Assistenten zur Interaktion in adaptiven Umgebungen. Ein weiteres wichtiges Ziel ist es eine konfliktfreie Interaktion zur ermöglichen. Hierzu sind Mechanismen für die Koexistenz von Assistenzsystemen und adaptiven Umgebungen zu entwickeln.

Die Entwicklung von adaptiven Systemen – also die implizite Interaktion – ist zwar nicht das Ziel dieser Arbeit, jedoch ist eine leichte Integration von solchen Systemen zur ermöglichen. Denn eine Koexistenz und konfliktfreie Interaktion kann nur gewährleistet werden, wenn keine Zugriffe auf die Umgebung außerhalb des hier zu entwickelnden Systems erfolgen können.

Neben übersichtliche und einfache Schnittstellen zur Nutzung von Komponenten, welche einen zentralen Zugriff auf Geräten und Medien ermöglichen, ist auch die Entwicklung von Systemen zur impliziten Interaktion auf der Modellierungsebene zur unterstützen. Demnach soll das im Rahmen dieser Arbeit zu entwickelnde Kontextmodell sowohl für eine explizite als auch für implizite Interaktion einsetzbar sein. Deshalb soll das Modell nicht minimalistisch entworfen werden. Aus diesem Grund muss das Modell für möglichst ein breiteres Anwendungsfeld ausgelegt sein und eine einheitliche Sicht auf die Umgebung anbieten.

Eine Betrachtung der existierenden Modelle (vgl. Kapitel 4.4.2) macht aber deutlich, dass Kontextmodelle nur für klar definierte Anwendungsdomänen erstellt werden können. *Kontext* ist einfach zu komplex, um all seine Facetten (Nutzer, Umgebung, Aktivitäten, Geräte und Dienste, ...) in einem Modell aufnehmen zu können [Shi06].

Daher sollen folgende Anforderungen gelten, um sowohl die in dieser Arbeit zu entwickelndes Bedienungskonzept realisieren zu können als auch das Kontextmodell für nahe liegende Szenarien für eine situationsabhängige Benutzerunterstützung anbieten zu können. Dadurch soll das Kontextmodell eine gemeinsame Basis für explizite und implizite Interaktion in der ausgewählten Domäne bilden.

Zunächst soll mit diesem Modell die Entwicklung eines Bedienungskonzepts im Sinne dieser Arbeit unterstützt werden. Insbesondere soll eine kontextabhängige Darstellung unterstützt werden. Dabei soll die Darstellung in Abhängigkeit der Position von Objekten und Geräten sowie in Abhängigkeit von Interaktionskontext angepasst werden können. Zur Interaktionskontext gehören insbesondere die Entitäten Aktivität, Medien und Geräte.

Die Analyse der Hauptdomäne von Ambient-Intelligence Multimedia-Umgebungen aus dem letzten Kapitel liefert die notwendigen Erkenntnisse zum Interaktionskontext und

zeigt die Beziehungen zwischen diesen Entitäten auf. Diese Erkenntnisse sollen in einem Top-Down-Ansatz in der Kontextmodellierung einfließen.

Schließlich soll das Modell für Entwicklung von impliziten Interaktions-systemen erweitert werden. Hierzu werden die in [Shi06] analysierten Szenarien für situationsabhängige, implizite Interaktion in adaptiven Home-Office-Domäne herangezogen werden. Die aus dieser szenariobasierten Analyse resultierenden Beiträge für Kontextmodellierung werden verwendet, um das aus der Domainanalyse resultierende Kontextmodell zu ergänzen.

Die szenariobasierte Analyse wurde bereits in [Shi06] präsentiert. Daher wird im nächsten Abschnitt lediglich auf dessen Ergebnisse eingegangen, welche im Gesamt-Kontext-Modell einfließen sollen.

Die Szenarien nehmen das *Dokumenten-basierte Arbeiten* als das wichtigste Szenario für ein Büro angenommen werden. Die Hauptaktivitäten *Text-Schreiben, Diskutieren, Diagramm erstellen, nach Informationen suchen, Dateien verwalten* sowie ein Systemverhalten für die in der jeweiligen Situationen zu erbringende *pro-aktive Unterstützung* sind im Anhang beschrieben.

Die Szenarien selbst sowie die darauf basierende Herleitung von *Abfragen* und Entitäten zur Formung eines Kontextmodells sind im Anhang (vgl. Kapitel 11.3) zu finden.

6.1 Szenario-basierte Kontextmodellierung

Wie im Kapitel 4.4.1 schon aufgezeigt, wird für die Realisierung von Ambient Intelligence Context-Awareness benötigt.

In den Kapiteln 11.3.4 und 11.3.5 werden zwei Szenarien sowie die dazugehörigen Situationen und Pro-Activities analysiert [Shi06].
Durch die Analyse der Situationen und Pro-Activities der genannten Szenarien wurden High-Level Abfragen identifiziert, welche von einem Kontext-Management-System beantwortet werden müssen, wenn in den Situationen der genannten Szenarien die vorgestellten Pro-Activities erbracht werden sollen.

Anschließend wurden aus der Analyse der High-Level Abfragen[1] *konkretere* Elemente eines Kontextmodells gewonnen (vgl. auch [Naz05b, Shi06]):

Kontext-Dimensionen Dimensionen (oder Komponenten), welche das Modell beinhalten (z. B. *Person*, *Device*, *Location*, etc.) soll.

Eigenschaften Attribute und *Methoden*, welche eine bestimmte Kontextkomponente besitzt. Beispielsweise wird zu der Kontext-Komponente *Person* das Attribut *Name* modelliert. Methoden erlauben, dynamische Eigenschaften einer Kontext-Komponente (im Sinne von *Simple-Query*[2]) abzufragen. So kann man zu einem *Device* seinen *Energiestatus* abfragen.

Relationen Logische Verbindungen zwischen Kontext-Dimensionen, welche modelliert werden müssen, um eine High-Level Abfrage mittels Zusammensetzung von mehreren Simple-Query beantworten zu können. Beispielsweise ist ein *Device* mit einer Menge

[1] High-Level Abfragen liefern Informationen über mehrere Kontext-Dimensionen hinweg.
[2] Ein Simple-Query liefert Informationen, welche einer einzigen Kontext-Dimension zu entnehmen sind.

von spezifischen *Services* verknüpft, welche auf dem Gerät vorhanden sind.
Eine Relation zweier Kontext-Dimensionen erfordert eine *echte* Verknüpfung (z. B. Listen, Tabellen) der *Instanzen* dieser Kontext-Dimensionen. Beispielsweise müsste ein Kontext-Management-System *auflisten*, welche Instanzen vom *Service* tatsächlich auf einer Instanz vom *Device* existieren.

Basierend auf Analyseergebnissen dieser Szenarien werden in diesem Kapitel neue Elemente für das Kontextmodell eingeführt (vgl. auch [Shi06] sowie Kapitel 11.3.4 und 11.3.5).

Wir führen an dieser Stelle Teilräume (*Zonen*) ein, um mehrere Positionen eines Raumes zu einem Teilraum zusammenzufassen. Mehrere Positionen werden in Form von Punktemengen, welche einen Raum eingrenzen bzw. definieren (*Area*), dargestellt.

Sichtbarkeitsbereiche sind Zonen, in denen etwas sichtbar ist.
Damit gibt es für jedes Gerät eine Zone, in der es sichtbar ist. Dies bezeichnen wir als VisibilityZone eines Gerätes; diese ist relativ zum Gerät selbst definiert.

Die Eignung eines Gerätes für die Betrachtung eines bestimmten Dokumentes wird aber von einer Reihe anderer Faktoren mitbestimmt. Beispielsweise ist ein *sichtbares* Display lange noch nicht geeignet, um einen Film darauf anzuschauen, wenn die Sonne frontal auf den Bildschirm scheint. Ebenso ist die Eignung auch von dem darzustellenden Inhalt und Medienformat abhängig. Die Punkte wie Confidentiality, Schriftgrösse, Dokumententyp (Grafik, Foto, Diagramme, Tabellen, Text, Video, Anwendungen, ...), Umgebungszustand (Licht, ...), Entfernung des Betrachters sowie das mit dem Gerät und dem jeweils dargestellten Dokument beabsichtigte Vorhaben (Annotation-Activity, Presentation-Activity, ...) sind sehr eng mit der Sichtabrkeit verknüpft. Im Rahmen dieses Dokumentes wird darauf nicht eingegangen. Es existiert eine Reihe von Forschungsarbeiten zu diesem Thema („Multimodale und Multimediale Eingabe- und Ausgabekoordination"). Für weitergehende Informationen wird auf [Dyn04b, Naz06a] verwiesen.

Des Weiteren wird die Verdeckung eines Displays durch mobile Objekte (Geräte oder Personen) oder auch durch Wände abstrahiert. Im Rahmen des hier vorgestellten Modells wird auf die einfache Reichweite einer bestimmten Zone fokussiert. Dazu könnte auch z. B. die Hörbarkeitszone eines Telefons gezählt werden.

6.2 Domainanalyse-basierte Kontextmodellierung

Im Kapitel 5 wurde eine detaillierte Domainanalyse vorgestellt. Basierend auf dieser Analyse werden in diesem Kapitel die folgenden Kontext-Dimensionen eingeführt: Raum, Desktop, Zone, Person (*Room*), Aktivität (*Activity*), Gerät (*Device*), Dienst (*Service*) sowie Dokument (*Document*) und *URL* [Naz06a, NM06].

Das hier präsentierte Kontextmodell [Naz06a, NM06] besteht aus den o. g. Kontext-Dimensionen sowie Relationen, Attributen und Methoden dieser Kontext-Dimensionen.

Gemeinsame Eigenschaften von *Person*, *Device* und *Document* können zu *Object* generalisiert werden.

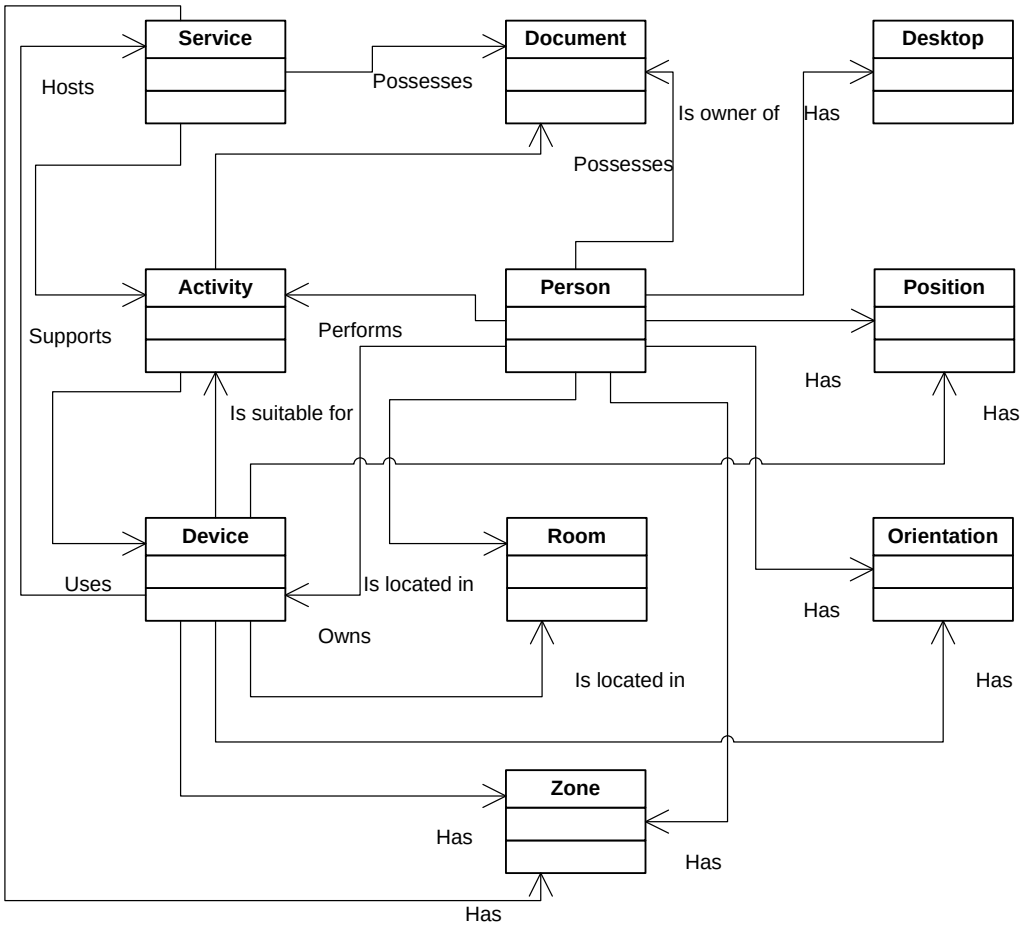


Abbildung 6.1: Existierende Entitäten und Relationen.

6.2.1 Basis-Entitäten

Die folgenden elementaren Entitäten werden zunächst eingeführt:

$$[Shape]$$
$$Vector == \mathbb{N} \times \mathbb{N} \times \mathbb{N}$$
$$Point == Vector$$
$$Location == Vector$$
$$Space == \mathbb{P}\, Point$$
$$[Action]$$
$$[Operation]$$
$$Activity == \mathbb{P}\, Action$$

DeviceType: definiert Geräteklassen, welche in Meetingäumen verfügbar sind. Verfügbare Gerätetypen sind MediaRenderer, MediaServer, Lighting, Cube, SmartBoard, WhiteBoard, BTDevice, WLANDevice, Sensing, EnvironmentController, Navigator, Agenda und Communicator.

ServiceType: definiert Klassen von Diensten (*software services*). Diese sind MediaManagement, EnvironmentControl, AudioInteraction, VideoInteraction, UPnPService, EnvironmentMonitoring.

MediaType: definiert Klassen von Dokumenten und Medien, welche in Meetingäumen verfügbar sind. Dies sind Presentation, Media (audio, video, pictures), Office, Live Streams, Contact, Notes, Calendar, Misc.

DocumentType: definiert Klassen von digitalen Mime Data. Diese sind VRML, HTML, JPG, PPT, MPEG, MP3, AVI, DOC, PDF, TXT.

ActivityType: definieren Aufgabentypen. Sie bezeichnen Aufgaben des Benutzers aus einer bestimmten Anwendungsdomäne. Diese Typen generalisieren logisch zusammengehörende Aufgaben zu Klassen, wie z. B. Presentation Control, Environment Control, Data Access oder Annotation. Aufgaben und Teilaufgaben dienen dazu, eine hierarchische Ordnung abzubilden. Die konkrete Auflistung kann der hierarchischen Aufgabenanalyse aus dem Kapitel 7.1 entnommen werden.

StateVariable: definiert einige Zustände für Geräte. Diese sind ON, OFF, STOP, UP, oder DOWN.

6.2.2 The Ambient: Kontext-Dimension-übergreifende Eigenschaften und Methoden

Position für Personen und Geräte Objekte haben eine eindeutige Position in dem Raum, in dem sie sich befinden. Diese Position wird durch einen Punkt (z. B. dem physikalischen Schwerpunkt des Objektes) mittels der Koordinaten (x, y, z) im lokalen Koordinatensystem des Raumes beschrieben (vgl. die Definition der Zonen). Diese Eigenschaft ist eine globale Gegebenheit. Jedoch können Geräte oder Personen über Positionierungsfähigkeiten verfügen, wodurch Geräte ihre eigene Position erfahren können. Diese Angaben können dann bei einer verteilten Implementierung anderen Entitäten (Room, Service, ...) mitgeteilt werden.

Räume haben ebenfalls Kenntnisse über Objektpositionen. Die Positionswahrnehmung eines Objektes (Geräte oder Personen) und die des Raumes müssen nicht zwangsläufig übereinstimmen. Dies ist beispielsweise dann der Fall, wenn das Objekt und der Raum unterschiedliche Genauigkeiten haben oder fehlerhafte Berechnungen bei der Positionierung durchführen.

Semantische Positionen Objekte können alternativ bzw. zusätzlich eine semantische Position (*Regions*) im Raum haben. Beispielsweise kann ein Objekt vor der Tür oder vor dem SmartBoard, ... liegen. Demnach ist eine *Region* ein bestimmter *Subroom* in einem Raum und besitzt einen der folgenden Namen als *Identifier*.

$$\begin{aligned} &Region ::= BehindTable \mid AtWhiteboard \mid AtDoor \mid AtWindow \\ &\mid AtWallCloset \mid BehindDoor \mid onTable \mid AtSpeakerDesk \end{aligned}$$

Umgebung eines Objekts Objekte haben eine besondere Zone, ihr *Surrounding*, die Umgebung des Objektes. Das ist, wenn man von *Nähe* eines Objekts spricht oder wenn eine Person sich an ihrem *Desktop* befindet. Diese Umgebung spannt ein eigenes lokales Koordinatensystem auf. Bei Polygonen, Kreisen und Kugeln liegt der Ursprung in der Mitte der Zone. So kann beispielsweise die Service-Zone einer Kamera relativ zu ihrer Umgebung adressiert werden.