

# End-User Installation Guide Generation Based on a Model Driven Approach

Diploma Thesis of

Name

At the Department of Informatics  
Institute for Program Structures  
and Data Organization (IPD)

Reviewer:	?
Second reviewer:	?
Advisor:	?
Second advisor:	?

Duration:: XX. Monat 20XX – XX. Monat 20XX

---

I declare that I have developed and written the enclosed thesis completely by myself, and have not used sources or means without declaration in the text.

**PLACE, DATE**

.....  
**(YOUR NAME)**

# Contents

<b>1. Motivation</b>	<b>1</b>
<b>2. Grundlagen</b>	<b>3</b>
2.1. Product Line Engineering . . . . .	3
2.1.1. Variability Modeling . . . . .	3
2.2. DocBook . . . . .	4
<b>3. Szenario</b>	<b>5</b>
<b>4. Verwandte Arbeiten</b>	<b>7</b>
4.1. Sensorinstallation für Ubicomp-Systeme in Heimumgebungen . . . . .	7
4.1.1. Eigenschaften einer Sensorinstallation . . . . .	7
4.1.2. Ablauf der Studie . . . . .	7
4.1.3. Aufbau des Kits . . . . .	8
4.1.4. Resultate der Studie . . . . .	9
4.1.5. Design Prinzipien . . . . .	10
4.2. Intelligente Augmented-Reality-Handbücher . . . . .	11
4.2.1. DFKI AR-Handbuch . . . . .	11
4.2.2. Second-sighted Glasses . . . . .	13
4.3. Dokumentengenerierung in industriellen Umgebungen . . . . .	13
4.3.1. Flexibler Ansatz zur Dokumentengenerierung . . . . .	13
4.3.2. Dopler Toolchain zur Unterstützung . . . . .	15
4.3.3. Implementierung der Dokumente . . . . .	16
4.3.4. Ergebnisse industrieller Studien . . . . .	17
4.4. Usability of User Interfaces Generated with a Model-Driven Architecture Tool	19
4.5. User comprehension performance for dataflow-based rules in smart Envi- ronments . . . . .	19
<b>5. Vergleich der Ansätze</b>	<b>21</b>
<b>6. Weiteres Vorgehen</b>	<b>23</b>
<b>7. Leichtgewichtige Generierung mit InstaGuide</b>	<b>25</b>
<b>Bibliography</b>	<b>27</b>
<b>Appendix</b>	<b>29</b>
A. First Appendix Section . . . . .	29



# 1. Motivation

Fortschritte in der Technologie haben Sensoren kostengünstiger und leichter zu benutzen gemacht. Diese Fortschritte reduzieren die Technologische Barriere für die Installation von Ubicomp-Systemen. Laut [BCL04] beschränkt sich der Installationsprozess auf die korrekte Platzierung und der semantischen Assoziation zwischen den Knoten der Anwendung und den Teil der physikalischen Welt den sie beobachten. Professionelle Installationen sind im Vergleich zu den Kosten der Anschaffung sehr groß und können die Ausbreitung solcher Systeme negativ beeinflussen. Die Endnutzer-Installation bringt mehrere Vorteile. Sie ist billiger und kann einem Ubicomp-System helfen schneller in der breiten Masse benutzt zu werden. Des weiteren lernen Benutzer das System besser kennen und können so mehr Kontrolle darüber haben und sie können das System an ihre speziellen Bedürfnisse anpassen.

Die Systeme die uns umgeben werden flexibler, enthalten mehr Intelligenz und sind größtenteils rekonfigurierbar. Dadurch entsteht eine hohe Komplexität bei der Installation, Konfiguration oder Wartung. Um den Nutzern eine Hilfestellung zu leisten könnte man individuell angepasste Handbücher erstellen. Dieser Ansatz kann jedoch nur effizient sein wenn es eine automatisierte Lösung gibt.

AR Systeme im Prototypstadium die sehr intuitiv sind.

Kann das aber auch mit einer Textuellen Benutzeranleitung funktionieren: Modellgetrieben Handbücher für intelligente, rekonfigurierbare Systeme erstellen um damit den Nutzern einen besseren Überblick über das System zu verleihen

Was ist der Vorteil ihres Lösungsansatzes? (auch ggü. anderen Ansätzen) Aktionen (Action): Welche Schritte wollen Sie gehen? Der Vorteil wäre eine automatisierte toolchain die Änderungen in dem System in dem Handbuch reflektieren und somit immer Aktuellen bleiben und dem Nutzer eine Sicht auf das System erlauben.



## 2. Grundlagen

### 2.1. Product Line Engineering

Um den Begriff des *Software-Product-Line-Engineering* (SPLE) zu erklären müssen zuerst ein paar andere Begriffe erläutert werden. Aus [PBvdL05] ist *Mass customisation* (kundenindividuelle Massenproduktion) die Großproduktion von Gütern die auf die individuellen Bedürfnisse der Kunden angepasst wurde. Die *Plattform* ist eine Technologiebasis auf der andere Technologien oder Prozesse aufbauen. SPLE ist ein Paradigma um Software-Anwendungen mit Hilfe von Plattformen und kundenindividueller Massenproduktion zu erstellen. Um die individualisierte Massenproduktion [massenanfertigung] zu ermöglichen werden Artefakte benötigt die an die verschiedenen Systeme der Produktlinie angepasst werden können. Diese Flexibilität der Artefakte wird *Variabilität* genannt. Die Aktivität, die als Ziel die Variabilität zu identifizieren und zu dokumentieren hat, heißt Variabilitätsdefinition. Im Prozess des Produktmanagements werden gemeinsame und variable Merkmale der Software-Entwicklungslinie identifiziert. Die Modellierung der Variabilität unterstützt die Entwicklung und die Wiederverwendung der Variablen Artefakte. Die Variabilität der Domänenanforderungen, -Architektur, -Komponenten und -Tests wird mit Hilfe des Variabilitätsmodells festgehalten.

...

#### 2.1.1. Variability Modeling

Definition 4-2: Variability Object A variability object is a particular instance of a variability subject.

Definition 4-2: Variability Object A variability object is a particular instance of a variability subject.

Definition 4-3: Variation Point A variation point is a representation of a variability subject within domain artefacts enriched by contextual information.

Definition 4-4: Variant A variant is a representation of a variability object within domain artefacts.

Variability Subject: Color Variability Object: Red, Green, Blue

Variation Point: Color of car Variants: red car, blue car

Definition 4-7: External Variability External variability is the variability of domain artefacts that is visible to customers.

Definition 4-8: Internal Variability Internal variability is the variability of domain artefacts that is hidden from customers.

Decision Model definieren?!

## 2.2. DocBook

DocBook ist eine Sammlung von Standards und Werkzeugen für Technische Publikationen. Der Standard wird von dem technischen Komitee OASIS (Organization for the Advancement of Structured Information Standards) [Doca] gepflegt und wurde von einem Konsortium von Software-Firmen erstellt um Computer-Dokumentation zu standardisieren.

Das Vokabular und der Inhalt der Elemente die ein Autor benutzen kann und die Verbindungen zwischen den Elementen wird durch das DockBook DTD (Document Type Definition) definiert. Das Format eignet sich besonders in folgenden Situationen:

- Sehr große Mengen an Inhalten
- Sehr strukturierte Inhalte
- Inhalte die zwischen inkompatiblen Systemen ausgetauscht werden müssen
- Inhalte die automatisierte Batch-Verarbeitung benötigen
- Inhalte die in verschiedenen Ausgabeversionen und -formen dargestellt werden müssen

Dadurch dass DocBook eine semantische Sprache ist, können Inhalte in einer präsentationsneutralen Form die ihre logische Struktur erfasst, dargestellt werden.

Das Markup ist ähnlich HTML mit Elementen die Bücher, Titel, Paragraphen oder Kapitel repräsentieren. Stylesheets die es erlauben die Inhalte in verschiedenen Formaten wiederzugeben sind auf [Docb] frei zugänglich.

The other major advantage of DocBook is the set of free stylesheets that are available for it. these stylesheets enable anyone to publish their DocBook content in print and HTML. The stylesheets are now developed and maintained as an open-source project on SourceForge.

Wie in [Sta07] erwähnt wird gibt es viele Textverarbeitungssysteme die DocBook unterstützen. Für kleine, einmalige Dokumente wird das System nicht empfohlen. Es ist auch für Dokumente die stark Layout-getrieben sind nicht geeignet. Da die Dateien in Klartext abgespeichert werden und nicht binär, können Versionskontrollsysteme benutzt werden.



### 3. Szenario

Das hier genannte Szenario wird an das Home Automation System Szenario aus [PBvdL05] angelehnt. Der Komfort der Nutzer sollte sich durch die Heimautomatisierung steigern. Daraus folgt, dass auch der Installationsaufwand gering gehalten werden sollte. Der Nutzer ist in diesem Szenario der einzige Stakeholder, es wird davon ausgegangen, dass er die Initiative für die Anschaffung des Systems hat. Die Heimautomatisierung deckt viele Bereiche des Lebens und des Hauses. Um das Anwendungsfeld zu begrenzen wird ein Ausschnitt aus einer Ubiquitären Anwendung für das eigene Heim betrachtet. Es werden nur die im Folgenden aufgeführten Komponenten in Betracht genommen:

- Regelgeräte für Heizkörper
- Sensoren und Aktuatoren für Fenster
- Ein LED-Strang
- Steckdosenaufsätze
- Server
- Installationsgerät (Smartphone, Tablet, Rechner)

[Graphik cu elementele care o sa le continua - fa-le dragutze, diagrame din carte ai putea refolosi] Des Weiteren könnten Infrarotsensoren eingesetzt werden um Benutzer im Raum aufzuspüren, die dem Konfigurationssystem erlauben Raumempfehlungen zu geben. Es werden keine anderen Ausgabegeräte, wie z.B. Bildschirme, benutzt, außer der Web-Oberfläche die von verschiedenen Geräten ansteuerbar ist. Eine weitere Randbedingung ist, dass keine Entertainment Funktionen, wie zum Beispiel Multi-Media Streaming, betrachtet werden. Es werden keine Sicherheitsaspekte wie das Nutzer-Management in Betracht gezogen. Das heißt alle Benutzer in diesem Szenario haben vollen Zugriff auf das System. Es werden keine Sicherheitskritischen Anwendungen wie Gesundheitsmonitoring oder Heimsicherung. Das System hat als Haupt Anwendungspunkt das Monitoring und Kontrollieren von Temperaturen und Stromfluss. Die Installation laut dem Generierten Handbuch ist die Forschungsfrage.

Das Heim-Gateway ist der Zentrale Server des Intelligenten Hauses [PBvdL05]. Anstatt einen Cloud-basierten Ansatz zu fahren wird der Gateway-Server als zentraler Server für die Kommunikation, Koordination und Datenhaltung dienen. Auf diesem Server kann das Modell der Anwendung, so wie sie auf den Benutzer zugeschnitten ist gespeichert und verwaltet werden. Des weiteren soll das Home Gateway den Code verteilen, die

Dokumentation auf dem neuesten Stand halten? Auf dem Home-Gateway Server werden folgende Daten über alle Komponenten des Systems (*Sensoren, Aktuatoren, Kontrolleinheiten*) abgespeichert:

- Informationen über die Art und Typ einer Komponenten
- Physische Position
- Semantische Assoziation
- Konfigurationsparameter
- Regeln(workflows) [bazat pe paperu care inca nu l-ai citit, sau poate gasesti ceva generic: k trebe sa existe reguli de control in absenta AI]

Variabilität in der Heimautomatisierung bedeutet, dass verschiedene Varianten des Produkts an die Endkunden kommen und diese auf unterschiedliche Weise installiert werden müssen. Die verschiedenen Möglichkeiten Sensoren und Aktuatoren zu kombinieren, kann die Assoziation und die verschiedenen Steuerungsalgorithmen ändern.

Die Erstellung der Assoziationen zwischen den Heimautomatisierungsgeräten und den von ihnen beobachteten Subjekten ist die Aufgabe des Anwenders.

Kontrolle des Systems durch Web-Oberflächen.

Es ist wichtig den Benutzern zu erklären was die Geräte machen, was diese gemacht haben und wie sie kontrolliert werden können um das "debuggen" (z.B. fehlerhafte Assoziationen) durch den Benutzer zu ermöglichen [EG01]. Dies gilt besonders im Fall in dem neue Geräte dazukommen, alte entfernt werden oder die Geräte von unterschiedlichen Herstellern kommen.

## 4. Verwandte Arbeiten

In diesem Kapitel werden ein paar Ansätze der verwandten Arbeiten näher beschrieben. Der Aufbau einer Studie zu Heimautomatisierung und deren Installation wird als erstes beschrieben. Danach kommen Ansätze die auf Augmented-Reality setzten und zum Schluß ein Überblick über eine Dokumentengenerierungstechnik wie sie in industriellen Umgebungen praktiziert wird.

### 4.1. Sensorinstallation für Ubicomp-Systeme in Heimumgebungen

#### 4.1.1. Eigenschaften einer Sensorinstallation

Die Installation von Sensoren wird in [BCL04] als eine Aufgabe mit zwei Dimensionen betrachtet: Platzierung und Assoziation. Die Platzierung ist wichtig um den Sensorknoten zu erlauben die korrekte physikalische Größe mit ausreichender Qualität zu messen. Die Positionierung wird weiterhin in zwei Unterfaktoren aufgeteilt: *Direktionalität* und *Proximität*. Unter dem Direktionalitätsfaktor versteht man wie sensibel der Sensor auf Fehlern in seiner Ausrichtung reagiert. Eine Kamera ist zum Beispiel sehr darauf angewiesen, dass sie die richtige Szene aufnimmt, ein Mikrofon hingegen ist nicht so sensibel auf seine Ausrichtung. Der Proximitätsfaktor gibt an wie nahe der Sensor an sein Messziel angebracht werden muss. Die Assoziation bezieht sich auf die semantische Verbindung zwischen dem Datenstrom eines Sensors und des gemessenen Objektes in der realen Welt. Eine Eins-zu-Eins Assoziation ist für Menschen die einfachste Variante und kann viel leichter nachvollzogen werden. Die Präzision einer Assoziation wird als Funktion der KnotenPlatzierung und dessen zu messenden Subjekts beschrieben. Es wird betont, dass die Assoziation eines proximitäts Sensors leichter durchzuführen ist als die eines direktionalen Sensors. Außerdem ist auch das Beobachten von Objekten leichter als das Monitoring von Räumen weil die Reichweite eines Sensors schwer einzugrenzen ist.

#### 4.1.2. Ablauf der Studie

Die Studie wurde in 15 Haushalten aus den Vereinigten Staaten durchgeführt. Die Teilnehmer wurden als repräsentative Gruppe von einer Marktforschungsfirma ausgesucht und es wurde darauf geachtet, dass sich keine Experten unter den Teilnehmern befinden sollen.

Als Entlohnung bekamen die Teilnehmer eine Geldsumme. Die Daten der Installationsdurchführung, die im Durchschnitt etwa 84 Minuten gedauert hat, wurden von zwei Mitarbeitern gesammelt. Diese haben mit Hilfe von Notizen und Fotos jeden Installationsversuch dokumentiert. Ein Fragebogen wurde von jedem Teilnehmer ausgefüllt.

Der Ablauf einer Installation hatte vier Phasen: *Einleitung, Erkundung, Installation der Sensoren und ein Interview*. In der Einleitung wurde eine Freigabeform und ein Fragebogen ausgeteilt (die Inhalte dieser Dokumente wurde nicht publiziert). Die Erkundungsphase diente um eine Übersicht des Konzepts hinter dem Ubicomp-System und dem Installationskit zu geben. In der Installationsphase wurden die Teilnehmer gebeten jeden der mitgelieferten Sensoren einzeln an ein Gerät oder in einem Raum aufzustellen. Die Evaluatoren haben während der Installationphase keine Fragen mehr über den Prozess beantwortet und haben versucht kein Feedback über die Handlungen der Teilnehmer zu geben um die Resultate der Studie nicht zu verfälschen.

Nach der Installationsphase wurde ein Interview durchgeführt. Die Fragen dienten dazu das Verstehen der Teilnehmer über die verschiedenen Installationsschritte und Komponenten des Systems zu erfassen und um die Probleme zu identifizieren würde das System einen Monat lang angebracht bleiben. Es wurde auch gefragt wie sie die Sensoren davon abbringen würden Daten zu sammeln.

#### 4.1.3. Aufbau des Kits

Das Kit wurde so konzipiert, dass die Installation keine komplexen Schritte, wie Änderungen an dem elektrischen Sicherungskasten, enthält. Der Inhalt des Kits ist wie folgender gewesen (siehe auch Abbildung 4.1):

- Mock-Up-Sensoren (Vibration, Strom, Bild, Geräusch)
- Ein Katalog mit Gegenständen
- Ein kompakter Computer
- PDA Gerät für Barcodes
- Anweisungen der Version A,B und C (vgl. Tabelle 4.1)

Ein Teil der mitgelieferten Sensoren wurden konzipiert um die Aktivität von Haushaltsgeräten zu messen (Vibration, Strom, Geräusche). Bewegungssensoren und Kameras wurden benutzt um die Aktivität der Menschen und das elektrische Licht zu überwachen. Die Sensoren wurden je nach Typ Farbkodiert und es wurden nur 2 Sensoren von jedem Typ, insgesamt 10, mitgeliefert. In Realität würde eine solche Ubicomp-Anwendung mehr als 50 Sensoren beinhalten. Nur die Platzierung und Assoziation der Sensoren wurde in betracht genommen. Zusammenarbeit oder Korrelation der Sensoren wurde nicht betrachtet, daher konnte die Installation jedes einzelnen Sensors unabhängig betrachtet werden.

Der Katalog mit Gegenständen enthielt Barcodes für alle möglichen Gerätetypen die in einem Haushalt auftreten können, wie z.B. Kühlschränke oder Waschmaschinen. Dieser Katalog in Zusammenhang mit den Barcodes auf den Sensoren und dem Barcodereader wurden benutzt um die Assoziation zwischen den Sensoren und ihren Subjekten durchzuführen. Der kompakte Computer sollte Daten von den Wireless-Sensoren auslesen. Die einzige Interaktion die der Nutzer mit dem Computer hatte war es ihn mit Strom zu versorgen. Ein Scanner und gedruckte Anweisungen wurden den Nutzern überreicht um sie durch den Installationsprozess zu führen. Es wurden verschiedene Anweisungen ausgeteilt um zu vermeiden, dass die Studie die Qualität der Nutzeranleitung bewertet. Das Ziel der Studie war es die Sensorinstallation zu betrachten. In Tabelle 4.1 werden die Unterschiede



Figure 4.1.: Bild des Kits

Table 4.1.: Die Unterschiede zwischen den Dokumentationen

Enthaltene Informationen	Dokumentation A	Dokumentation B	Dokumentation C
Installationsanleitung	✓	✓	✓
Gesammelte Daten	✓	✓	x
Funktion der Sensoren	✓	x	x

zwischen den Dokumentationstypen aufgezeigt. Die Namen der Sensoren in Dokumentation A und B waren gleich (*Strom*, *Vibration*, *Bewegung*, *Bild*, *Geräusch*). In version C der Dokumentation wurden nur Farben benutzt um zwischen den Sensoren zu unterscheiden. Alle Sensoren mussten mit Hilfe des PDAs assoziiert werden in dem der Sensor und das zu messende Objekt aus dem Katalog eingescannt wurden. Eine richtige Installation wurde nach den Positionierungs- und Assoziationsfaktoren bewertet. Es wird erwähnt, dass es keine statistisch signifikanten Unterschiede, zwischen dem Erfolg der Installation im Zusammenhang mit den verschiedenen Dokumentationen, gab. Von den 5 Nutzern mit Dokumentation C haben sich 3 solche Informationen aber gewünscht.

#### 4.1.4. Resultate der Studie

Die Resultate der Evaluierung werden im Folgenden zusammengefasst.

Von den 150 Sensorinstallationsaufgaben (15 Teilnehmer x 10 Sensoren) wurden 112 (75%) erfolgreich durchgeführt. Nur 5 Teilnehmer (30%) haben alle Aufgaben korrekt durchgeführt. Das Einlesen mit dem Barcode-Scanner wurde von 13 aus 15 (86%) Teilnehmern korrekt durchgeführt. Zwei der fünfzehn Teilnehmer konnten keine der Aufgaben erfüllen und haben auch die Assoziation mit Hilfe des Barcodes nicht verstanden. Dieselben konnten weder den Anweisungen auf dem PDA folgen, noch wussten sie wie die Sensoren zu deaktivieren sind. Nur 5 aus den 10 Teilnehmern, die Beschreibungen der von den Sensoren aufgenommenen Daten erhalten haben (Dokumentation A und B), konnten den konkreten Anwendungsfall der Sensoren nennen. Der am meisten fehlerhaft installierte Sensor war der

Geräuschsensor, mit 67% Erfolgsrate, gefolgt von dem Bildsensor, mit 70% Erfolgsrate. Im Vergleich dazu wurde der Stromsensor (ein Steckdosenaufsatz) in 87% der Fälle erfolgreich installiert (korrekte Assoziation 87% und korrekte Platzierung 93%). 8 von den 15 Teilnehmern (53%) haben wenigstens einen, aber nicht alle Sensoren korrekt installiert. Vier der neun fehlerhaften Bildsensorinstallationen folgten aus dem Übersehen oder nicht folgen der Ausrichtungsanweisungen. Mit Hilfe der Information über die Sensoren konnte ein Nutzer einen Sensor erfolgreich an seinen nicht standardkonformen Kühlschrank anbringen.

Obwohl vor der Studie alle Teilnehmer über die Datenerfassung unterrichtet wurden und der Aspekt, dass die Daten das Haus nicht verlassen mehrmals erwähnt wurde, gab es zwei Teilnehmer die sich geweigert haben die Bildsensoren zu installieren. Andere Teilnehmer hatten sich auch mit Bedenken über die Erfassung der Daten geäußert. Die Teilnehmer hatten auch Bedenken dass die Installation ihren Einrichtungen schaden würden und, dass Haustiere oder Kinder an die Sensoren kommen könnten.

#### 4.1.5. Design Prinzipien

Im folgenden werden die 5 in [BCL04] erhobenen Design-Prinzipien für die Endnutzerinstallation von Sensoren aufgeführt und erläutert. Die ersten zwei Prinzipien lehnen sich an traditionelle Prinzipien aus der Mensch-Maschine-Interaktion. Die letzten drei sind Erkenntnisse die aus der Studie gewonnen wurden.

1. **Das Konzeptuelle Modell des Benutzer bei bekannten Technologien angemessen nutzen**

Sensortypen deren Funktionalität ähnlich ist wie in anderen Anwendungsszenarien können die Benutzer beim Verständnis unterstützen, da die Konzeptuellen Modelle der Nutzer nicht mehr redefiniert werden müssen. Man muss aber darauf achten, dass vertraute Technologie nicht anders angewendet wird als im Normalfall. Falls dies dennoch auftritt, kann der Sensor sowohl im Namen als auch im Aussehen versteckt werden.

2. **Nutzbarkeit der Installation mit den häuslichen Anliegen im Gleichgewicht halten**

Designer und Anwender müssen Faktoren der Ästhetik und der Umgebung unter Betracht nehmen. Wenn Benutzer aus pragmatischen Gründen die Installation nicht durchführen wollen, können Technische Aspekte leicht in den Hintergrund treten. Ein System, dass die Privatsphäre oder das Eigentum des Besitzers verletzt kann genauso in der Box liegen gelassen werden wie ein System, das schwer zu installieren ist.

3. **Die Anwendung von Kameras, Mikrofonen und hochdirektionalen Sensoren vermeiden**

Nicht nur die Sensitiven Daten die solche Sensoren aufnehmen können sondern auch die verschiedenen Installationsmöglichkeiten in Verbindung mit unterschiedlichen Layouts der Wohnungen können diese Art von Sensoren problematisch machen.

4. **Inkorrekte Installationen sollten erkannt werden und auch Teilinstallationen sollten einen Mehrwert bieten**

Endnutzer wollen vielleicht einige Sensoren, die zum Beispiel sensitive Daten aufnehmen könnten, nicht installieren, obwohl sie einen Mehrwert in der ganzen Ubicomp-Anwendung sehen. Kann die Anwendung eine Teilinstallation entdecken und erlauben so kann trotzdem ein Mehrwert geboten werden.

5. **Den Nutzer über die Datensammlung, -speicherung und -übertragung unterrichten**

Benutzer die wissen was für Daten die Sensoren aufnehmen und wie diese Daten benutzt werden können mit Problemen bei der Installation besser umgehen. Für die Akzeptanz der Anwendung ist es kritisch, dem Benutzer zu erklären wie die Daten benutzt werden und ob diese das Haus verlassen.

Die Autoren erläutern auch weitere Aspekte die eine solche Anwendung in ihrer Akzeptanz und Installation unterstützen könnte. Für die Erkennung einer Fehlinstallation könnten Daten, die die Sensoren produzieren, analysiert werden um festzustellen ob der Sensor keine Daten oder stark korrelierte Daten mit einem anderen Sensor liefert. Dies kann geschehen wenn der Sensor falsch gerichtet ist oder die Abstände bei der physischen Platzierung eine ungewünschte Messung anderer Phänomene erlauben. Die Anwendung sollte daher ein Modell der sinnvollen Daten enthalten und entweder falsch angebrachte Sensoren ignorieren oder den Nutzer darauf aufmerksam machen und ihn bei der Fehlerbehebung unterstützen. Real-Time-Feedback der Sensoren kann das Konzeptuelle Modell des Nutzers verbessern und damit auch die Installation.

Die Einsatzgebiete die von einer Endnutzerinstallation profitieren könnten wären zum Beispiel Anwendung für die Unterstützung von älteren Personen. Das System könnte von dem Pfleger, der weiß was für spezielle Bedürfnisse seine Patienten haben, installiert werden. Auch Heim-Automatisierungssysteme würden durch die geringeren Anschaffungskosten erschwinglicher werden. Auch Feldstudien die normalerweise technisches Personal für die Inbetriebnahme des Systems benötigt werden könnten leichter und kostengünstiger aufgesetzt werden. Die Endnutzerinstallation gibt den Nutzern auch das Gefühl mehr Kontrolle über die Anwendung zu haben.

Die Autoren unterstreichen die Notwendigkeit einer sorgfältigen Dokumentation der Speicherung, Nutzung und Auswertung der gesammelten Daten. Auch wie dieser Prozess von den Nutzern kontrolliert werden kann sollte Teil der Dokumentation sein.

## 4.2. Intelligente Augmented-Reality-Handbücher

Ein Augmented-Reality-Handbuch ist ein digitales Handbuch, dass Schritt für Schritt Anleitungen auf einem Head-Mounted-Display (HUD) in das Blickfeld des Benutzers projiziert. Jeder Schritt wird dem Benutzer vorgezeigt bis dieser erfolgreich durchgeführt wurde.

### 4.2.1. DFKI AR-Handbuch

Der Augmented-Reality-Forschungsbereich des DFKI (Deutsches Forschungszentrum für Künstliche Intelligenz) arbeitet an der Entwicklung und Verbesserung von Augmented-Reality-Handbüchern, mit Hilfe von KI, um sie eines Tages in realen Szenarien einsetzen zu können [Nie13].

Die Erstellung von Material, das so genannte *Authoring*, für AR-Handbücher ist komplex und leidet unter hohen Zeit- und Geldaufwänden. Dieses Problem wird mit Hilfe des *programming by demonstration Ansatzes* gelöst. Das Authoring-Tool kann eine einmal gesehene Video-Sequenz in einzelne Handlungsabläufe zerlegen und kann diese mit einem stochastischen Übergangsmodell kombinieren (vgl. Abbildung 4.2). Mit Hilfe einer auf der Brille integrierten Kamera wird die Durchführung der Aufgabe analysiert und auf dem HUD werden die nächsten durchzuführenden Schritte eingeblendet. Dieses System ist leichtgewichtig in dem Sinne, dass keine speziellen Marker benötigt werden und das Erlernen der durchzuführenden Schritte automatisch passiert. Die transparenten Einblendungen der zu durchführenden Schritte werden auch automatisch von der Software erstellt.

In Abbildung 4.2 wird das Authoring gezeigt. Die einzelnen Handlungsabläufe werden aus der Video-Sequenz extrahiert und damit werden Klassifikatoren trainiert. Die Sequenz

wird in Teile aufgespalten die Eigenschaften wie *statisch*, *wiederholend*, *nicht wiederholend* haben. Die Einblendungen werden automatisch von der Software genriert und es besteht die Möglichkeit manuelle Annotierungen, wie Pfeile oder andere graphische Symbole, einzufügen. Die Automatischen Annotierungen werden synchron mit den Handlungen des Benutzers eingeblendet.

Mit diesem Ansatz können Techniker Referenzabläufe aufnehmen um sicherzustellen, dass alle zukünftigen Aktivitäten in derselben Art und Weise durchgeführt werden.

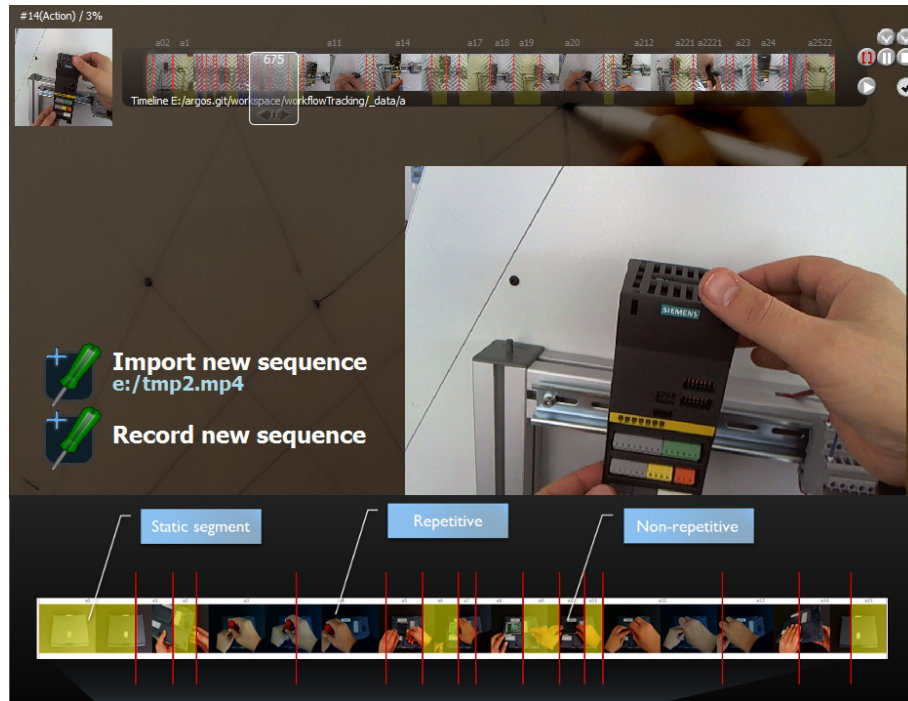


Figure 4.2.: Authoring für das AR-Handbuch, aus [Nie13]

Wenn der Nutzer eine Aufgabe mit Hilfe des Systems durchführt (Abbildung 4.3(a)) werden zuerst die notwendigen Schritte für die Teilaufgabe, wie in Abbildung 4.3(b), eingeblendet. Nachdem der Nutzer mit der Ausführung der Teilaufgabe anfängt wird auf das Sichtfeld der Status der Durchführung mit Hilfe einer Farbkodierung angezeigt. Dabei bedeutet in 4.3(c) die grüne Kodierung eine korrekte Durchführung.

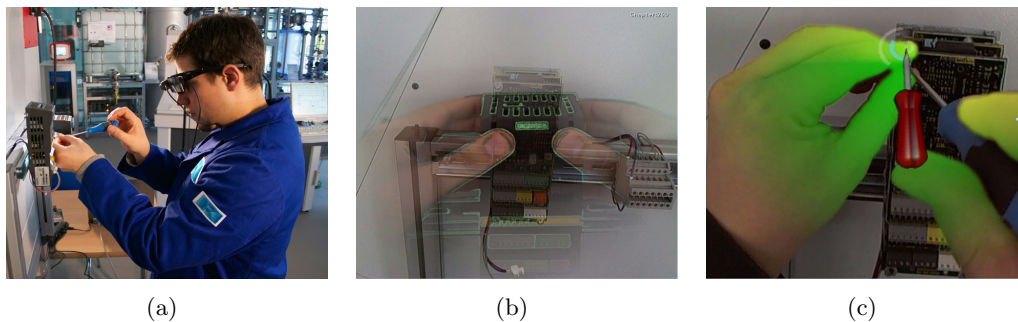


Figure 4.3.: Das AR-Handbuch System aus [Nie13]

Das AR System soll in Zukunft auch auf Android Geräte laufen. Dies bedeutet, dass auch private Anwender, z.B. bei der Installation von Haushaltsgeräten, davon Gebrauch machen könnten.



### 4.2.2. Second-sighted Glasses

Ein ähnlicher Ansatz, genannt Second-sighted Glasses, wird in [YNT11] vorgestellt. Hierbei handelt es sich um die Einblendung in das Sichtfeld des Nutzer von Varianten die er in der Interaktion mit verschiedenen Objekten haben kann. Das Ziel ist es dem Benutzer zu zeigen mit welchen Aktionen ein bestimmter Kontext erfüllt werden kann. Die Varianten, auch *Futures* genannt, werden abhängig vom Kontext in dem sich der Benutzer befindet angezeigt. Diese repräsentieren mögliche Abläufe die vom Kontextsensitiven Systemen definiert werden. Um den Objekten der Umgebung Intelligenz (Smartness) zu verleihen werden Sensoren oder Marker angebracht. Auch hier wird eine Assoziation zwischen den Sensoren der Realwelt und ihren Virtuellen Semantik erstellt. Diese Assoziationen können mit Hilfe einer Kamera erstellt werden. Da der Kontext eines Objektes schwer zu definieren ist wurde die Smart Object Event Modeling Language [YNK<sup>+</sup>09] entwickelt. Diese erlaubt es trotz ihrer Einfachheit die Lesbarkeit, Portabilität, Expressivität und die Unabhängigkeit zwischen Modell und Zielobjekt zu wahren. Temporale folgen können so definiert werden um komplexe Abläufe darzustellen. Auch eine graphische Oberfläche für die Modellierung von SOEML wird angeboten.

Die Autoren identifizieren bei diesem Ansatz die Notwendigkeit eine Benutzerstudie durchzuführen um zu klären ob die Benutzer die smarte Umgebung gut genug verstehen können und ob die Installation von den Entwicklern gut durchgeführt werden kann. Weitere Forschung muss noch betrieben werden um die Generierung von Animationen aus den Kontextdefinitionen zu erlauben und um die Zusammenarbeit mit anderen Kontextsensitiven Systemen zu erlauben.

AR-Technologien werden noch nicht in der breiten Masse angewendet. Es werden aber zunehmend vielversprechende Ansätze entwickelt. CastAR [EJ13] ist ein AR-System, dass für rund 200 US-Dollar demnächst auf den Markt kommen soll. Kostengünstige Hardware könnte es Ansätzen wie in [Nie13] oder [YNT11] erlauben sehr schnell in den Nutzermarkt einsteigen und breite Anwendung zu finden.

Die Vorteile dieser Ansätze sind die hohe Intuitivität und die Präzision der Aufgabenbeschreibung, da jeder Schritt dem Benutzer so Anzeigt wird wie er durchzuführen ist.

## 4.3. Dokumentengenerierung in industriellen Umgebungen

Der Ansatz basiert auf vorhergehenden Arbeiten [12,13][citeste] und erlaubt die Generierung von Dokumenten als auch von Softwaresystemen aus demselben Variabilitätsmodell. Der Ansatz basiert auf der Beobachtung, dass viele Entscheidungen die während der Ableitungsphase des Produktes gemacht werden, für beide Artefakte relevant sind. Der Ansatz wurde, mit Hilfe der DOPLERWerkzeuge [12,13], Erfolgreich auf zwei Produktlinien mit verschiedenen Reifegraden angewandt. Bei der ersten Produktlinie handelte es sich um eine Prozessautomatisierungsanwendung von Siemens VAI, dessen technische Variabilität schon modelliert war.[Citeste 15] [Pagina 2]

Der in [RHE<sup>+</sup>10] beschriebene Ansatz kann laut den Autoren unabhängig vom **Aufbaustadium/Reifegraden**, der Variabilitäts Modellierungstechnik oder der zu generierenden Dokumenttypen verwendet werden. Der Ansatz wurde durch die Tool-Suite DOPLERunterstützt.

### 4.3.1. Flexibler Ansatz zur Dokumentengenerierung

Der Ansatz beschreibt vier Schritte, wie in 4.4 dargestellt, welche iterativ zu durchlaufen sind:

### 1. Extrahieren und analysieren der Variabilität in Dokumenten

Im ersten Schritt werden existierende Dokumente herangezogen um die Variabilität zu extrahieren. Es wird empfohlen diesen Schritt durch einen Produktionslinienexperten mit Wissen über Variabilitäts-Modellierung durchführen zu lassen. Durch Workshops mit Domänenexperten der Produkt-Managements, des Vertriebs oder der Entwicklung können zusätzliche Informationen gewonnen werden. Die Dokumentanalyse bringt die Variabilitätspunkte und ihre dazugehörigen Varianten hervor.

### 2. Variabilitäts-Modelle erstellen oder anpassen

Der Produktionslinienexperte kann, mit dem gewonnenen Wissen, existierende Modelle anpassen oder neue erstellen. Falls existierende Modelle schon in der Automation benutzt werden könnten diese auch für Dokumentengenerierung benutzt werden. Die gewählte Technik sollte flexibel sein und fortgeschrittene Automatisierung während der Produktableitung zulassen.

### 3. Variabilitäts-Mechanismus und passenden Generator erstellen

Der Generator ist erforderlich um die produktspezifischen Dokumente zu generieren. Der Generator benötigt die Variations-Information die in den Dokumenten abgespeichert ist. Der Mechanismus mit dem man die Variations-Information in die Dokumente aufnimmt hängt vom Dokumentformat ab. Aus diesem Grund werden strukturierte Dokumentformate wie XML-basierte Formate benutzt. Auch das Format von Office Programmen wie Microsoft Word kann mit Hilfe von Markup-Informationen erweitert werden.

### 4. Dokumente um Variabilitäts-Information erweitern

Der letzte Schritt webt die Informationen in die Dokumente ein. Der Produktlinienexperte spezifiziert die Variationspunkte und Varianten die in Schritt 1 hervorgehoben wurden und stellt sie in Beziehung mit den Modellen aus Schritt 2. Die Beziehungen werden mit Hilfe des Mechanismus aus Schritt 3 realisiert. Während der Produktableitung kann der Generator die Informationen aus der Nutzereingabe und den Dokumenten heranziehen und produktspezifische Dokumente erstellen.

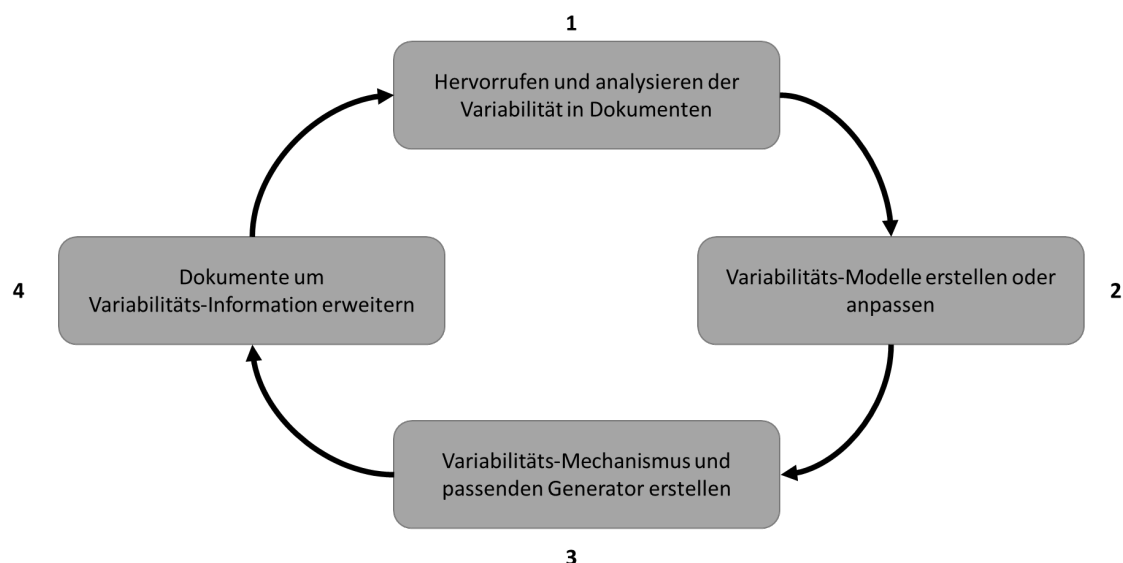


Figure 4.4.: Die vier iterativ zu durchlaufenden Phasen des Ansatzes zur Dokumentengenerierung aus [RHE<sup>+</sup>10]

### 4.3.2. Dopler Toolchain zur Unterstützung

Im [RHE<sup>+</sup>10] wurde der oben Beschriebene Ansatz implementiert.

Die Variabilitätsmodellierung wurde durch den Entscheidungsorientierten DOPLERAnsatz gemacht[12,15] und um die Variabilität in den Dokumenten zu definieren wurde das DocBook-System erweitert [16]. Als Generator wurde eine Erweiterung für DOPLER[13] erstellt.

Die Tool-Suite enthält ein Variabilitäts-Modellierungs-Tool, einen Konfigurations-Wizard um dem Nutzer die Variabilität während der Produktableitung zu zeigen.

[?? muss noch Erweitert werden um die DecisionKing & co. poate mai scoti si din celalalte papere]

Die DOPLERVariabilitätsmodelle enthalten Assets und Decisions. Die Assets sind die Artefakte der Produktlinie (Dokumente oder Technische Bestandteile). Die Abhängigkeiten zwischen den Assets können sein:

- Funktional
- Strukturell

DOPLERerlaubt es Assets mit arbitrary Granularität und mit domänenspezifischen Attributen und Abhängigkeiten, basierend auf einem set von Basistypen. Die Benutzer können Domänenspezifische Metamodelle erstellen um ihre eigenen Asset Typen, Attribute und Abhängigkeiten zu definieren.

Entscheidungspunkte werden als Decision(Entscheidung) modelliert. Eine Decision enthält eine eindeutige Identifikationsnummer, eine Frage und einen Typen(4.5. Die Frage wird dem Nutzer bei der Konfiguration gestellt und die Antwort auf die Frage bestimmt die Zusammensetzung des Dokumentes. Eine Entscheidung kann von anderen Entscheidungen in zwei Arten abhängig sein, siehe 4.6. Folgende sind die Abhängigkeitstypen der Decision:

- Hierarchisch
- Logisch

Eine hierarchische Abhängigkeit bedeutet, dass Entscheidungen in einer bestimmten Reihenfolge gemacht werden müssen. Wenn eine logische Abhängigkeit besteht dann ändert eine Entscheidung die Antworten der davon abhängigen.

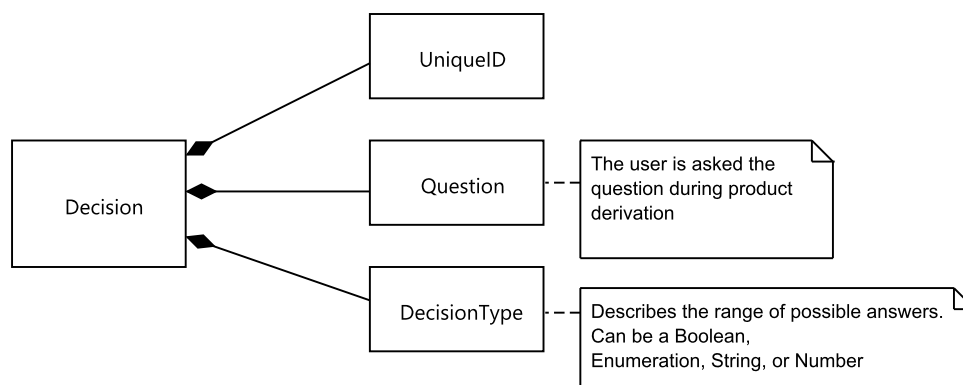


Figure 4.5.: Die Entscheidung und ihre Hauptbestandteile

Das Dopler-Meta-Model in 4.6 zeigt auch den Zusammenhang zwischen den Decisions und Assets. Die Inklusion eines Asset in das abgeleitete Produkt geschieht durch seine

Verbindung zu der Decision. Die Attribute eines Assets können von den Antworten auf die Decision Frage beeinflusst werden.

DOPLER erlaubt es domänenspezifische Assets zu definieren indem es ein meta-model für eine spezifische organisation oder Kontext erstellt. Um Dokumente zu generieren kann man so ganze oder nur Teile von Dokumenten als Assets in den Modellen repräsentieren. So wurden, wie in Abbildung 4.6, die Assets erweitert (inheritance) in dem das Document-Fragment eingeführt wurde. Dieses repräsentiert Teile eines Dokuments wie Kapitel oder Abschnitte. Um auch den Fall zu modellieren, dass bestimmte Fragmente Teil von anderen Fragmenten sind wurde noch eine Assoziation (verzi k aici e aggregation) zwischen den Fragmenten eingeführt.

Die explizite modellierung von Dokumentfragmenten erlaubt es, durch die höhere Abstraktion, die selbe Decision zu benutzen um mehrere Dokumentfragmente zu verwalten?!. Es wird dann auch möglich durch die selbe Decision sowohl Dokumentfragmente als auch Komponenten einzubeziehen.

Die Dokumentenfragmente als Assets werden benutzt um die grobkörnige Variabilität zu modellieren. Um die feinkörnigen Anpassungen durchzuführen wurde ein variabilitätsmechanismus basierend auf DocBook erstellt. Da DocBook standardisiert und sehr geeignet um elektronische Dokumente zu erzeugen (vergleiche Kapitel 2.2) wird es auch in diesem Ansatz benutzt.

### 4.3.3. Implementierung der Dokumente

Der Profiling-Mechanismus aus DocBook wird benutzt um Elemente und Attribute zu implementieren die die Variationspunkte in den Dokumenten definieren. Die zentrale Erweiterung zu DocBook ist das *doplerdoc* Attribut. Es koppelt das Markup-Element an ein Dokumentenfragment und erlaubt es so optionale oder alternative Texte zu definieren. Durch diesen Mechanismus sind auch Querverweise möglich. Platzhalter können in Verbindung mit dem *doplerdocplaceholder* Element erstellt werden.

Die am öftesten benutzten Variabilitätsmuster, laut [RHE<sup>+</sup>10], sind:

- Platzhalter
- Optionaler Text
- Alternativer Text
- Querverweise
- Einfache grammatikalische Variabilität (z.B. Pluralbildung)
- Multi-Media-Objekte (z.B. Bilder)
- Formattierung und Layout (durch XSL oder CSS Style-Sheets realisiert)

In Listing 4.1 wird ein Beispiel aus einem DocBook-Dokument mit der DOPLER Erweiterung aufgeführt. Das Kapitel *cooling* wird in die Dokumentation nur aufgenommen wenn das Dokumentfragment *cooling\_chapter*, aufgrund einer Decision aufgenommen wurde. Das Element *doplerdocplaceholder* mit dem *doplerdoc* Attribut dient als Platzhalter für den Kühlungsmechanismus *cooling\_mech*.

Listing 4.1: Beispiel der DocBook Erweiterungen für DOPLER aus [RHE<sup>+</sup>10]

```
<chapter id="cooling" doplerdoc="cooling_chapter">
  <para id="cooling_mechanism">
    The caster supports the
    <doplerdocplaceholder doplerdoc="cooling_mech"/>
  </para>
</chapter>
```

```

        mechanism . . .
    </para>
</chapter>

```

Mit dem Modellierungswerkzeug für Variabilität (z.B. DOPLER) wird ein Variabilitätsmodell erstellt. Die Antworten auf die Fragen der Decisions werden durch den Nutzer mit Hilfe eines Konfigurationsassistenten (z.B. DOPLERKonfigurationsassistent) in das Modell eingetragen.

Das variable Dokument, das zentrale Artefakt im Prozess der Dokumentengenerierung, entsteht durch Erweiterung mit Variabilitätsinformation und eine ständige Rückkopplung zu dem Variabilitätsmodell.

Der Dokumentengenerator kann mit Hilfe der schon aufgelösten Variabilität, durch Beantwortung der Fragen durch den Nutzer, die Anpassung durchführen. Der Dokumentengenerator ist auch dafür zuständig die DocBook XSLT Engine zu konfigurieren und auszuführen. Die variablen Dokumente werden mittels der DocBook Engine in ein Ausgabeformat (z.B. PDF) transformiert und mit Hilfe von XSL oder CSS formatiert, die auch von den Decisions abhängen könnten.

#### 4.3.4. Ergebnisse industrieller Studien

Die Erkenntnisse die aus den zwei industriellen Anwendungen des Ansatzes gewonnen werden im weiteren beschrieben. Im ersten Anwendungsfall waren Teile der Dokumentation für Benutzer schon als DocBook Quellen vorhanden. Des weiteren waren die Variabilitätsmodelle schon vorhanden und die Generierung von Produktkonfigurationen wurde schon unterstützt. In der ersten Phase der Methodik (siehe mai sus) wurden die Variationspunkte die, laut einer Befragung der Domänenexperten, am meisten auftreten, in Betracht genommen:

- Teile der Dokumentation sind optional abhängig von den Teilen des Systems
- Querverweise müssen konsistent sein
- Platzhalter kommen oft vor und müssen ersetzt werden (z.B Name des Kunden)
- Grammatikalische
- Spezifischere Dokumente müssen geliefert werden, wenn Kunden eigene Erweiterungen für das System entwickeln wollen.

In der zweiten Phase der Methodik konnten 70% der Decisions der Software-Konfiguration aus den Variabilitätsmodellen , benutzt werden. In den restlichen Fällen wurden Decisions spezifisch für die Dokumentengenerierung erstellt[nu zice daca le-o bagat dupa in varmodel original]. Ein Generator wurde danach erstellt der die Technische Benutzerdokumentation laut den Decisions aufbauen kann (wie oben beschrieben). Die Aufgabe der Erweiterung der Dokumente mit der Variabilitätsinformation (der 4. Schritt der Methode) wurde von den Domänenexperten durchgeführt, nachdem für jede Art von Variabilität die von den Domänenexperten erwähnt wurde (siehe auch oben) ein Beispiel gegeben wurde. Die so erhaltene Werkzeugkette unterstützt sowohl die Variabilitätsmodellierung der Software als auch der Dokumente. Ein einziges Entscheidungsmodell (Decision model) wird benutzt um die Produktkonfiguration als auch die Technische Benutzerdokumentation zu erstellen.

In der zweiten Industriellen Anwendung mussten Kundenspezifische Verkaufsdokumente erstellt werden. In diesem Fall wurde noch keine Variabilitätsmodellierung durchgeführt. Es wurden die Schritte der Methode durchgeführt. Die folgenden Variabilitätspunkte wurden von den Domäne:

- Zahlenindikatoren mussten abhängig vom Kunden und den ausgewählten Features berechnet werden[hier ist das besonders]
- Alternative Texte

Einige Variationspunkte hatten Einfluss auf mehrere Dokumente. Die Teile die hier die Rolle der Assets hatten, waren die verkaufbaren Einheiten. Decisions wurden modelliert um Kundendetails, Systemeigenschaften oder Werte für die Berechnung der numerischen Indikatoren(ROI) [Types of decisions mehr erläutern?]

Der Generator hatte die Aufgabe die Preise und ähnliches zu generieren.

Auch in DocBook sources gemacht, aus word dokumenten.

Man kann aus den Studien schließen, dass folgende Variabilitätspunkte relevant für die Dokumentengenerierung sind:

- 

[Related work] Was diese arbeit von den anderen hervorhebt ist dass es besonders für nicht technisches Personal geeignet ist. Dieser Ansatz ist auch besser weil beliebige Dokumente erstellt werden können. Laut [22 -SLR] gibt es viel andere Ansätze aber sie betrachten nur einen Typen von Dokumenten, nämlich die Anforderungsspezifikation. Most importantly, we use end- user customization tools based on decision models to resolve variability. DocLine

[Conclusion]

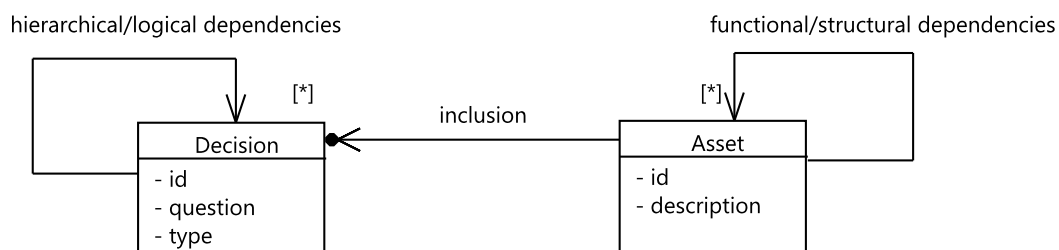


Figure 4.6.: DOPLER Meta-Model aus [RHE<sup>+</sup>10].

Das Diagramm 4.7 beschreiben.

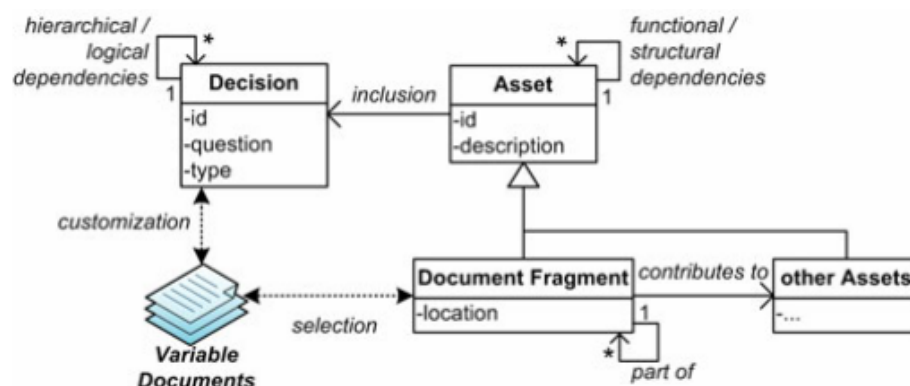


Figure 4.7.: DOPLER Meta-Model mit Dokumenten aus [RHE<sup>+</sup>10].

#### 4.4. Usability of User Interfaces Generated with a Model-Driven Architecture Tool

...??

#### 4.5. User comprehension performance for dataflow-based rules in smart Environments

???





## 5. Vergleich der Ansätze

Eine Übersicht der Ansätze findet man in den Tabellen 7.1 und 7.1.

Genauigkeit der Beschreibung: kann das Garantiert werden wenn menschen das System intepretieren?

Die Resultate der Studie die in [BCL04] durchgeführt wurde werden nicht detailliert behandelt und die Resultate die erwähnt werden dienen der Unterstützung bestimmter Folgerungen. Die Versionen der Handbücher werden kurz beschrieben und es wird nur eine in Abbildung 5.1 dargestellte Variante gezeigt. Es wird erwähnt, dass es keine statistisch signifikanten Unterschiede, zwischen dem Erfolg der Installation im Zusammenhang mit den verschiedenen Dokumentationen gibt. Die Rohdaten wurden nicht in das Paper integriert und es gab auch keine Sensoren die interagieren mussten. Es musste auch nichts konfiguriert werden.

Es kann passieren, dass Nutzer wichtige Anweisungen übersehen oder nicht befolgen, daher wäre es nützlich diese hervorzuheben.

Metamodell sollte Assoziation und direkionalität in betracht nehmen um erstens detailliertere Angaben über die Insallation zu geben wenn ein Sensor kompliziert zu installieren ist oder viele Faktoren dort einfließen. Aber auch um die nötigen Infomrationen zu enthalten um die Fehlerfindung und bewertung der Insallation durchführen zu können. Mass für komplexität: direkionalität.

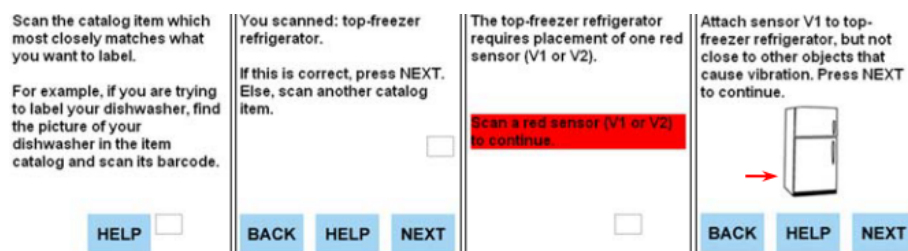


Figure 5.1.: Bild des Kits

Die Installation von Sensoren wird in [BCL04] als eine Aufgabe mit zwei Dimensionen betrachtet: Platzierung und Assoziation. Dies sind wichtige Anhaltspunkte um die Randbedingungen des Modells zu definieren.

Table 5.1.: Vergleich der Ansätze(unter dem Aspekt der Benutzbarkeit)

	DOPLER	AR-Handbook	Modellg
Ansatz	Entscheidungs orientiert		
Interaktionsmodus			
Quell-Artefakte			
Generierte Artefakte	Fertige Dokumente		
End-Nutzer	Vertrieb, Marketing, Produkt-Management		
Form der Unterstützung	Textuell	Graphisch, VR	G
Flexibilität			
Domänenexperten	JA	JA	
Wiederverwendung			
Aufwand für die Instandsetzung			
Aufwand für die Benutzung			
Genauigkeit der Beschreibung			
Synchronisierungsaufwand			
Aufwand für die Erstellung neuer Handbücher			
Anwendungsszenarien		physische Umgebung	

Expertenwissen wird bei jeder generierung von Dokumentation benötigt. Disadv: AR Handbook not linked to system model

Die Vorteile: Sehr intuitiv Automatisch erlernte Dokumentation der Schritte Unterstützung bei der Verifikation der Arbeitsschritte Passt die Videoausgabe an den Nutzer an

Probleme: Müssen wenigstens ein mal Aufgezeichnet werden Für Systeme mit vielen Varianten ein potenzielles Problem Änderung der Prozessschritte zieht ein neues Durchspielen nach sich Mit Software GUIs nicht ausprobiert Noch keine Nutzbarkeitsstudie Expertenwissen wird bei jeder generierung von Dokumentation benötigt. Disadv: AR Handbook not linked to system model

Textuelle Handbücher genau so gut oder besser?

## 6. Weiteres Vorgehen

Usability studies machen Am anfang um eine Hypothese zu erstellen und sehen was leute von so einem System erwarten

!System definieren!

Punkte finden die interessant sind/interessant zu beschreiben sind. Formative studie - ?  
Qualitativ, qunatitative?

Interviews erstellen -> Requirements sammeln -> Hypothese

Studien erstellen um zu sehen ob eine statistisch signifikanter Unterschied zwischen den statisch erstellten und den dynamisch generierten besteht.

Wizard of Oz studie => Referenzimplementierung machen

Idee: Seite erstellen die einen konfigurator darstellt. User konfigurieren lassen. Nach einiger zeit den usern das ding in die Hand drücken und arbeten lassen.

2x5/7 Leute sollte man haben für eine Usability studie Gruppen Interviews sind nicht schlecht um Gruppendynamik zu fördern. Tutorium Users study?

Methodik muss fest sitzen!

Eriksson Semantic document approach Knuth Literate Programming



## 7. Leichtgewichtige Generierung mit InstaGuide

Mit Hilfe von Modell-zu-Modell-Transformationen kann der Zustand eines Systems in einer anderen Form dargestellt werden.

Der Transformationsmechanismus ist automatisiert

Herausforderungen sind es eine Transformation zu erstellen die ein für den Menschen nutzbares Resultat liefert und auch schnell genug läuft um sie mit jeder Änderung neu generieren zu können.

Die Erkundungsphase, in der ein Überblick über das System und dessen Komponenten gegeben wird, kann mit Hilfe eine Online-Konfigurators ersetzt werden. Dem Benutzer wird das System im ganzen Präsentiert und er hat die Möglichkeit die Komponenten auszuwählen die ihn interessieren. Wird der Konfigurator benutzt kann automatisch für jeden Kunden eine Feature-Liste erstellt werden. Aus dieser Feature-Liste entsteht eine Variante des Produktes aus der das Handbuch generiert werden soll.

Relations between components on a meta level have to be defined, or do they? Consider having dependencies, they must be known within the system model (Uses relations)

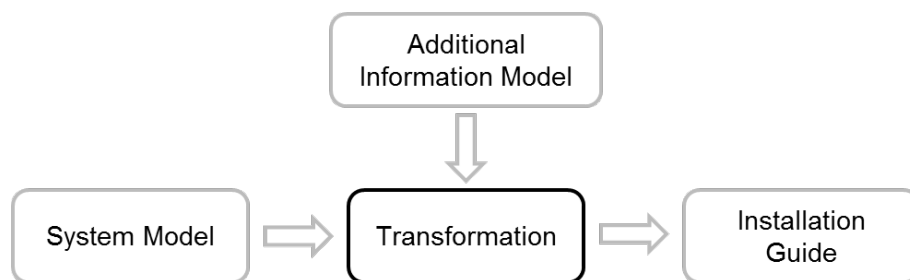


Figure 7.1.: Instaguide-Ansatz Überblick

In dem Beispielszenario

User stories: User buys some of the gizmos we offer In the background the configuration has to check the constraints User gets the package and a QR code to read in with the app(offline for mounting when data connection available)

Natürliche Sprache

Table 7.1.: Vergleich der Ansätze(unter dem Aspekt der Benutzbarkeit)

	Licht	
Heizkörper	×	✓
LED Strang		
Steckdosen		
Server		

# Bibliography

- [BCL04] C. Beckmann, S. Consolvo, and A. LaMarca, “Some Assembly Required: Supporting End-User Sensor Installation in Domestic Ubiquitous Computing Environments,” in *UbiComp 2004: Ubiquitous Computing SE - 7*, ser. Lecture Notes in Computer Science, N. Davies, E. Mynatt, and I. Siio, Eds., vol. 3205. Springer Berlin Heidelberg, 2004, pp. 107–124. [Online]. Available: [http://dx.doi.org/10.1007/978-3-540-30119-6\\_7](http://dx.doi.org/10.1007/978-3-540-30119-6_7)
- [Doca] “DocBook.org.” [Online]. Available: <http://www.docbook.org/>
- [Docb] “The DocBook Project.” [Online]. Available: <http://docbook.sourceforge.net/>
- [EG01] W. Edwards and R. Grinter, “At Home with Ubiquitous Computing: Seven Challenges,” in *UbiComp 2001: Ubiquitous Computing SE - 22*, ser. Lecture Notes in Computer Science, G. Abowd, B. Brumitt, and S. Shafer, Eds. Springer Berlin Heidelberg, 2001, vol. 2201, pp. 256–272. [Online]. Available: [http://dx.doi.org/10.1007/3-540-45427-6\\_22](http://dx.doi.org/10.1007/3-540-45427-6_22)
- [EJ13] J. Ellsworth and R. Johnson, “Augmented reality glasses: Ex-valve-employees castar,” 2013. [Online]. Available: <http://www.pcgamesn.com/valve-were-making-augmented-reality-glasses-ex-employees-carrying-project-forward-castar>
- [Nie13] P. Niels, “Intelligente Augmented Reality Handbücher Zeigen, wie’s geht - Werkerunterstützung für die Fabrik der Zukunft,” 2013. [Online]. Available: [http://av.dfki.de/projects\\_recent/ar-handbook](http://av.dfki.de/projects_recent/ar-handbook)
- [PBvdL05] K. Pohl, G. Böckle, and F. J. van der Linden, *Software Product Line Engineering: Foundations, Principles and Techniques*. Springer-Verlag New York, Inc., Sep. 2005. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1095605>
- [RHE<sup>+</sup>10] R. Rabiser, W. Heider, C. Elsner, M. Lehofer, P. Grünbacher, and C. Schwanninger, “A flexible approach for generating product-specific documents in product lines,” in *Proceedings of the 14th international conference on Software product lines: going beyond*, ser. SPLC’10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 47–61. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1885639.1885645>
- [Sta07] B. Stayton, *DocBook XSL: The Complete Guide 4. ed.* Sagehill Enterprises, 2007. [Online]. Available: <http://www.sagehill.net/docbookxsl/>
- [YNK<sup>+</sup>09] T. Yonezawa, J. Nakazawa, G. Kunito, T. Nagata, and H. Tokuda, “SOEML: Smart Object Events Modeling Language based on Temporal Intervals,” in *International Journal of Multimedia and Ubiquitous Engineering, Vol. 4, No. 3*, 2009, pp. 73–91.
- [YNT11] T. Yonezawa, J. Nakazawa, and H. Tokuda, “Retrieving Environmental Smartness with Second-sighted Glasses,” Keio University, Tech. Rep.

ITE Tech. Rep., vol. 35, no. 9, HI2011-9, 2011. [Online]. Available:  
<http://www.ite.or.jp/ken/paper/000000001ABy/eng/>



# Appendix

## A. First Appendix Section

ein Bild

Figure A.1.: A figure

...