

# Kostenmodell der Kontextkommunikation und Kontextverarbeitung in Ubiquitären Informationsumgebungen

Tobias Zimmer  
Telecooperation Office (TecO)  
Universität Karlsruhe

## 1 Einführung

Die Grundidee ubiquitärer Informationssysteme ist es, dass Kleinstcomputer mehr und mehr in die Arbeits- und Lebensumgebung sowie alltägliche Gegenstände dieser Umgebungen eingebettet werden und dort dem Benutzer Dienste anbieten, die ihn bei der Erledigung seiner täglichen Aufgaben unterstützen. Dabei ist eines der primären Ziele diese Dienste und die Geräte, welche die Dienste erbringen so unaufdringlich wie möglich zu gestalten.

Kontext bildet dabei die Basis der Interaktion zwischen Benutzer und ubiquitärer Informationsumgebung. Kontext bezeichnet im Forschungsbereich Ubiquitous Computing Informationen über die Situation und die Umstände, die Grundlage für die Erbringung eines Dienstes sind. Kontextsensitive Anwendungen sind in der Lage auf diese Kontexte zu reagieren und ihr Verhalten entsprechend anzupassen.

Aus dem in [24] eingeführten, systemnahen Kontextmodell sowie bekannten Modellen der Kommunikation in Netzwerken und bereits in der Literatur beschriebenen Kontextmodellen, wird in diesem Artikel ein Kostenmodell für die Kommunikation und Kontextverarbeitung in ubiquitären Informationsumgebungen abgeleitet. Dieses Kostenmodell hat zum Ziel, die verschiedenen möglichen Modellierungsansätze hinsichtlich ihrer Kosten vergleichbar zu machen und so Hinweise für die effiziente Modellierung der Kommunikation und Verarbeitung von Kontexten zu geben.

## 2 Kostenklassen und Kostenentstehung

Kosten können auf unterschiedlichen Wegen entstehen, sie entstehen aber immer dort, wo Ressourcen verbraucht werden. In ubiquitären Informationsumgebungen können alle im Betrieb entstehenden Kosten auf eine zentrale Ressource zurückgeführt werden: die Energie [21]. Sowohl die Kommunikation von Nachrichten zwischen den Teilnehmern einer ubiquitären Informationsumgebung, als auch die Verarbeitung von Daten durch die Teilnehmer verbraucht Energie. Der Energieverbrauch kann als Schranke für die Nützlichkeit von Ubiquitous Computing Technologien betrachtet werden. Die zur Verfügung stehende Energie begrenzt Zeit und Intensität in der eine Umgebung aktiv sein kann und damit den Nutzen, den sie erbringen kann. Daraus ergibt sich ein Optimierungsproblem der effizienten Nutzung der verfügbaren Energie. Dennoch soll hier nicht der tatsächliche Energieverbrauch betrachtet, sondern ein abstraktes Kostenmaß zu Grunde gelegt werden. Durch die große Anzahl verschiedener Artefakte ist es nicht möglich den konkreten Energiebedarf einer bestimmten Konfiguration einer ubiquitären Informationsumgebung zu berechnen.

Kosten entstehen in einer ubiquitären Informationsumgebung durch

**Kommunikation** Kommunikation verbraucht Energie, aus diesem Grunde trägt jede Kommunikation zu den entstehenden Kosten bei. Prinzipiell kann Kommunikation in einer ubiquitären Informationsumgebung direkt zwischen Artefakten stattfinden (peer-to-peer), sie kann zwischen einem Artefakt und einem Infrastrukturobjekt stattfinden oder zwischen Infrastrukturobjekten. Für dieses Kostenmodell werden alle drei Arten der Kommunikation gleich behandelt, außerdem soll die Größe des kommunizierten Datenpaketes bei der Kostenberechnung vernachlässigt werden. Kommunikationskosten werden also nur anhand der Anzahl der kommunizierten Datenpakete berechnet.

**Datenverarbeitung** Solange Daten verarbeitet werden, können Artefakte nicht in einen energiesparenden Schlafzustand versetzt werden. Obwohl unterschiedliche Berechnungen verschiedene Laufzeiten haben und sich der Energiebedarf der Hardware verschiedener Artefakte unterscheidet, soll hier lediglich die Anzahl der Verarbeitungen in die Kostenberechnung einfließen. Datenverarbeitung umfasst die Erfassung von Sensorwerten, Berechnungen zur Generierung, Fusion und Aggregation von Kontexten und falls vorhanden, die Präsentation an einer Benutzerschnittstelle.

Bedingt durch diese beiden Kostenklassen entstehen in ubiquitären Informationsumgebungen unterschiedliche Kosten in Abhängigkeit von der Modellierung der Kommunikation und Verarbeitung von Kontexten.

### 3 Netzwerktopologien

In ubiquitären Informationsumgebungen existieren zwei relevante Netzwerktopologien: segmentierte Netzwerke und unsegmentierte Netzwerke. Für die weitere Modellierung sei angenommen, dass das Kommunikationsmedium innerhalb eines jeden Segmentes rundruffähig ist.

Unsegmentierte Netzwerke sind Netzwerke, die aus nur einem einzigen Netzwerksegment bestehen. Sie zeichnen sich durch die Möglichkeit der direkten Kommunikation aller Teilnehmer aus. Segmentierte Netzwerke bestehen aus mehreren Segmenten, die durch Übergangsobjekte verknüpft sind. Teilnehmer in verschiedenen Segmenten können nur mit Hilfe von Übergangsobjekten miteinander kommunizieren. Übergangsobjekte sind Geräte, die selbst Teilnehmer mehrerer Netzwerksegmente sind oder Daten an Übergangsobjekte eines anderen Netzwerksegmentes weiterleiten können. Übergangsobjekte können sowohl Infrastrukturobjekte als auch spezielle Artefakte sein. Als Beispiel für segmentierte Netzwerke in denen ausgezeichnete Artefakte die Rolle eines Übergangsobjektes übernehmen können, sei hier das Bluetooth Scatternet genannt [22].

Nun werden die Grundkosten der Kommunikation in ubiquitären Informationsumgebungen definiert. Als Basis dient dabei eine abstrakte Kosteneinheit, deren Wert den Energiekosten für die Kommunikation einer Nachricht entspricht.

Sei  $\mathcal{A}$  die Menge aller Artefakte,  $\mathcal{I}$  die Menge aller Infrastrukturobjekte und  $\mathcal{G} \subset (\mathcal{A} \cup \mathcal{I})$  die Menge aller Übergangsobjekte. Dann sei  $\kappa$  eine Kostenfunktion mit:

$$\kappa : (\mathcal{A} \cup \mathcal{I}) \times (\mathcal{A} \cup \mathcal{I}) \mapsto \mathbb{N} \quad (1)$$

und

$$\kappa(x, x) := 0 \quad \forall x \in (\mathcal{A} \cup \mathcal{I}) \quad (2)$$

die Kosten für die interne Kommunikation eines Objektes. Sei weiter  $\tilde{S}$  die Menge aller Netzwerksegmente und  $S \in \tilde{S}$  eines dieser Segmente, dann gilt:

$$S \subset (\mathcal{A} \cup \mathcal{I}) \quad (3)$$

und

$$\kappa(x, y) := 1 \quad \forall x, y \in S \quad \text{mit} \quad S \in \tilde{S} \quad (4)$$

für die Kommunikationskosten von Objekten innerhalb desselben Segmentes. Die Übergangskosten zwischen benachbarten Segmenten seien definiert als:

$$\kappa(g_1, g_2) := 1 \quad \forall g_1 \in S_1 \wedge \forall g_2 \in S_2 : \quad g_1, g_2 \in \mathcal{G} \wedge S_1 \cap S_2 = \emptyset \quad (5)$$

Benachbarte Segmente sind Segmente, deren Übergangsobjekte direkt miteinander kommunizieren können.

Die Übergangskosten zwischen Segmenten, die nicht benachbart sind hängen davon ab, ob das Übergangsobjekt im durchleitenden Segment direkt mit dem nächsten Übergangsobjekt kommunizieren kann, oder ob dies nicht der Fall ist und die Daten stattdessen im durchleitenden Segment zuerst zu einem anderen Übergangsobjekt weitergeleitet werden müssen.

Seien  $g_s$  und  $g_z$  zwei beliebige Übergangsobjekte in disjunkten Segmenten  $S_s$  und  $S_z$  und sei  $n$  die Anzahl der Segmente, zwischen  $S_s$  und  $S_z$  und  $m$  die Anzahl der Segmente zwischen  $S_s$  und  $S_z$  in denen eine Durchleitung der Daten nötig ist, dann gilt für die allgemeinen Übergangskosten von  $S_s$  nach  $S_z$ :

$$\kappa(g_s, g_z) = 1 + n + m \quad (6)$$

Für alle Kommunikationen gelte weiter bezüglich der entstehenden Kosten das Kommutativgesetz:

$$\kappa(x, y) = \kappa(y, x) \quad \forall x, y \in (\mathfrak{A} \cup \mathfrak{J}) \quad (7)$$

Damit sind die Grundkosten der Kommunikation für das Kostenmodell definiert. Auf der Basis dieser Grundkosten können verschiedene konkrete Netzwerkkonfigurationen hinsichtlich der in ihnen entstehenden Kommunikationskosten analysiert und verglichen werden.

#### 4 Datenverarbeitung

Neben den Kosten für die Kommunikation entstehen in einer ubiquitären Informationsumgebung Kosten für die Verarbeitung von Daten. Daher sei eine Kostenfunktion für die Kontextverarbeitung definiert mit:

$$v : (\mathfrak{A} \cup \mathfrak{J}) \times \mathbb{N} \mapsto \mathbb{N} \quad (8)$$

Um verschiedene Ansätze bei der Verarbeitung von Kontexten vergleichen zu können, werden die Kosten der Verarbeitung hier stark abstrahiert modelliert. Unterschiedliche Laufzeiten verschiedener Algorithmen bleiben hier vorerst unberücksichtigt ebenso wie der variierende Energiebedarf der Hardware verschiedener Artefakte. In die Berechnung eingehen soll lediglich die Anzahl  $n$  der Verarbeitungsprozesse, wobei ein Verarbeitungsprozess die Erfassung von Sensorwerten und die algorithmischen Berechnungen zur Generierung, Fusion und Aggregation von Kontexten umfasst. Daher seien die Kosten für eine Verarbeitung definiert als:

$$v(x, 1) := 1 \quad \forall x \in (\mathfrak{A} \cup \mathfrak{J}) \quad (9)$$

und die Kosten für  $n$  Kontextverarbeitungen ergeben sich zu:

$$v(x, n) := n \quad \forall x \in (\mathfrak{A} \cup \mathfrak{J}) \wedge \forall n \in \mathbb{N} \quad (10)$$

Dieses vereinfachte Modell der Verarbeitungskosten hat den Vorteil, dass damit konzeptionelle Verarbeitungs- und Kommunikationsmodelle für Kontexte vergleichbar werden, ohne dass die Einflüsse einer konkreten Implementierung dieser Systeme die Ergebnisse verfälschen.

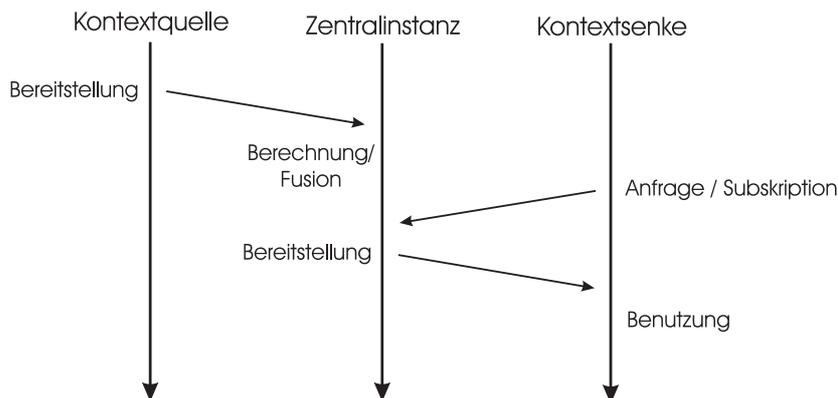
## 5 Alternative Kommunikations- und Verarbeitungsmodelle

In der Fachliteratur existiert eine große Anzahl von Kommunikations- und Verarbeitungsmodellen für kontextsensitive Systeme [19, 18, 13, 7, 5, 21, 12, 23, 4, 24, 16, 11]. In diesem Abschnitt werden aus diversen in der Literatur beschriebenen Modellen und weitergehenden theoretischen Überlegungen alternative Grundkonzepte für die Modellierung der Kontextkommunikation und -verarbeitung destilliert.

In der Literatur zu Kontextmodellen lassen sich drei unterschiedliche Ansätze für die Umsetzung der Kommunikation von kontextsensitiven Anwendungen identifizieren. Diese drei Grundmodelle bilden die Basis für eine beispielhafte Kostenrechnung. Das Ziel ist es, hier ein optimales Basismodell für die Modellierung der Kommunikation in ubiquitären Informationsumgebungen hinsichtlich der zu erwartenden Kosten zu finden.

### 5.1 Vollständig zentralisierte Systeme

Zentralisierte Kontextverarbeitungsmodelle zeichnen sich vor allem dadurch aus, dass sie eine zentrale Instanz oder einen Service besitzen, der die Verarbeitung aller Kontexte übernimmt. Klienten in diesen Systemen agieren als Kontextquellen oder Kontextsenken. Abbildung 1 zeigt schematisch den Ablauf der Kommunikation und Verarbeitung von Kontexten in diesen zentralisierten Systemen.



**Abbildung 1.** Zentralisiertes System

Hier sendet eine Kontextquelle ihre Kontextinformation an die Zentralinstanz. Dort werden Kontexte gesammelt, gespeichert und weiterverarbeitet. Eine Kontextsenke kann nun eine Anfrage an diese zentrale Instanz stellen und erhält als Antwort den angefragten Kontext, falls dieser vorhanden ist.

Neben dem Anfragebetrieb kann das zentralisierte System auch in einem Subskriptionsbetrieb arbeiten. Dabei sendet die Kontextsenke nicht einzelne Anfragen für jeden Kontext, sondern abonniert einen Kontext für eine gewisse Zeitspanne oder bis zum Widerruf. Die Zentralinstanz schickt dann selbständig neue Kontexte an die subskribierten Klienten sobald diese verfügbar werden.

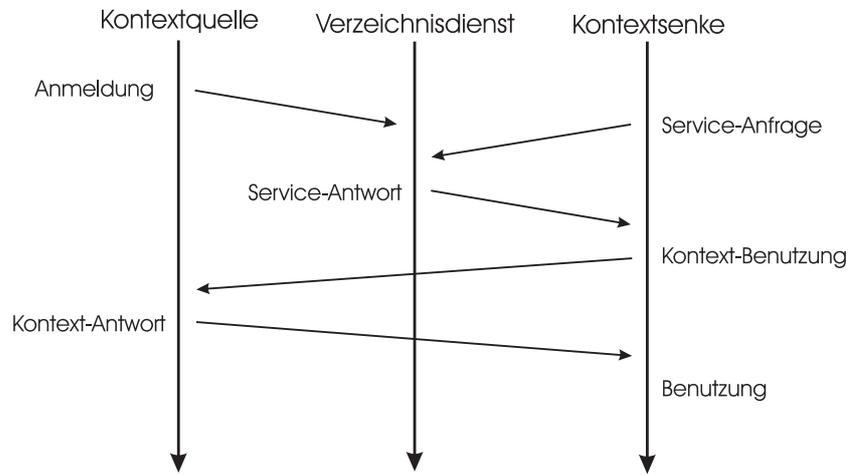
Obwohl ein zentralisierter Ansatz angesichts hoher Dynamik in ubiquitären Informationsumgebungen und der Tendenz schlecht zu skalieren, wenig geeignet scheint, um die Kommunikation und Kontextverarbeitung in diesen Umgebungen zu organisieren, finden sich in der Literatur viele Beispiele von Kontextmodellen, die diesem Modell folgen. Dazu gehören unter anderem die *Reconfigurable Context-Sensitive Middleware (RCSM)* von Stephan Yau et al. [23], das *Java Context Awareness Framework (JCAF)* von Jakob Bardram [1], *Context Fabric (ConFab)* von Jason Hong [12], das *Pervasive Autonomic Context-aware Environment (PACE)* von Karen Henriksen et al. [10, 11] und die *Context Broker Architecture (CoBrA)* von Harry Chen et al. [3].

### 5.1.1 Zentralisierte Vermittlung

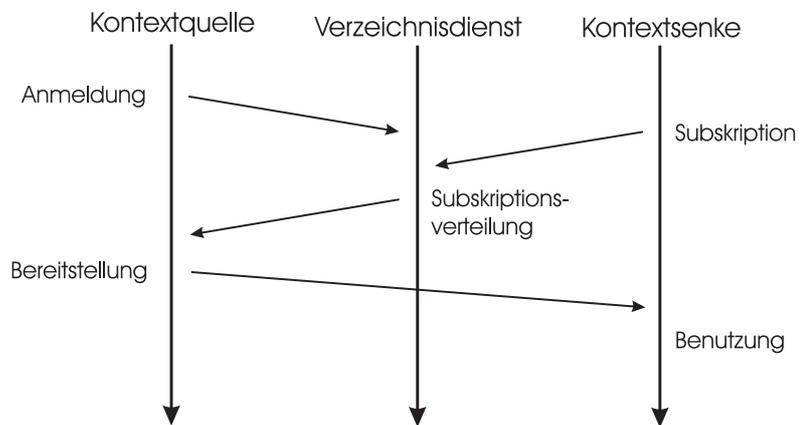
Im Gegensatz zu zentralisierten Systemen bieten Systeme mit zentraler Vermittlung lediglich eine zentrale Dienstfindung an. Die Kontextverarbeitung liegt hier bei den Klienten. Abbildung 2 zeigt die schematische Darstellung eines Systems, das auf der Basis von einzelnen Kontextanfragen operiert.

Um ihren Kontextservice in einer Umgebung bereitzustellen, muss sich jede Kontextquelle bei dem zentralen Verzeichnisdienst registrieren. Dabei teilt sie mit, welche Kontexte sie bereitstellen kann und wie sie im Netzwerk zu erreichen ist. Eine Kontextsenke kann nun den Vermittlungsdienst für einen bestimmten Kontext anfragen. Die Vermittlungsinstanz beantwortet die Anfrage positiv, wenn der angefragte Kontext von einer registrierten Kontextquelle geliefert werden kann. Die Antwort enthält Angaben, wie die Quelle zu erreichen ist (im Allgemeinen die Adresse der Kontextquelle). Daraufhin kann die Kontextsenke die Quelle direkt anfragen und bekommt von dieser auch direkt die benötigten Kontextinformationen übermittelt.

Wie oft eine neue Anfrage an den zentralen Verzeichnisdienst gestellt werden muss, hängt von der Dynamik der ubiquitären Informationsumgebung ab. Verlassen registrierte Kontextquellen das System nur selten, kann eine Kontextsenke die Vermittlungsinformationen lange nutzen. Ist die Dynamik dagegen hoch, kann für jede Kontextanfrage eine erneute Anfrage des Verzeichnisdienstes nötig werden.



**Abbildung 2.** Zentralisierte Vermittlung mit Anfrage



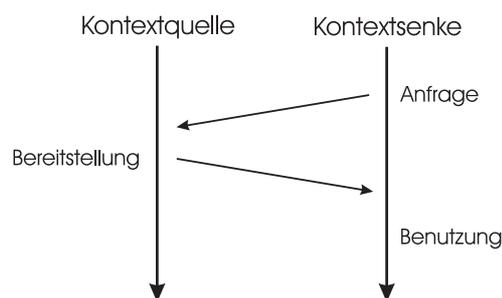
**Abbildung 3.** Zentralisierte Vermittlung mit Subskription

Kommt in zentral vermittelten ubiquitären Informationsumgebungen ein Subskriptionsmechanismus zum Einsatz, ändert sich das Kommunikationsschema (Abbildung 3). Auch hier muss sich jede Kontextquelle bei der Vermittlungsinstanz anmelden. Kontextsenken senden eine Subskription, die dann vom Vermittlungsdienst an alle Kontextquellen weitergeleitet wird, die den abonnierten Kontext bereitstellen können. Sind neue Kontextdaten verfügbar, werden diese direkt von der Kontextquelle an die subskribierten Senken übermittelt.

Nach diesem Prinzip arbeiten zum Beispiel das *Context Toolkit* von Anind Dey [7, 6], die *Service-Oriented Context-Aware Middleware (SOCAM)* von Tao Gu et al. [9], *Gaia* von Manuel Roman [17] und das *SOLAR* System, einer Implementierung des Context Fusion Network (CFN) von Guanling Chen [2].

## 5.2 Dezentrale Systeme

Dezentral organisierte ubiquitäre Informationsumgebungen verzichten vollständig auf zentrale Verwaltungskomponenten (siehe Abbildung 4). Die Kontextbereitstellung geschieht direkt von der Kontextquelle zur Kontextsenke. Wie im Falle der zentralisierten Vermittlung, liegt auch hier die Verarbeitung der Kontextinformationen bei den Klienten.



**Abbildung 4.** Vollständig dezentrales System

In dezentralen Systemen besteht die besondere Herausforderung einer effizienten Dienstfindung. Der Vorteil einer rein dezentralen Architektur für ubiquitäre Informationsumgebungen besteht in der direkten Abbildung des Ad-hoc-Charakters vieler Anwendungen.

Ein Beispiel eines Modells, das diesem Prinzip folgt, ist der *Gadgetware Architectural Style (GAS)* von Achilles Kameas et al. [14]. In diesem System liegen die Dienstbeschreibungen verteilt auf den einzelnen Netzwerkknoten vor. Eine Kontextquelle fragt einen Kontext bei einem anderen Knoten an, kann dieser den Kontext nicht bereitstellen, leitet er die Anfrage an andere Knoten weiter. Wird so eine Kon-

textquelle gefunden, die den angefragten Kontext liefern kann, sendet sie diesen an die Kontextsenke, welche die Anfrage initiiert hat. In der technischen Umsetzung des Systems senden die Artefakte Gruppenrufe (engl. multicast), um Netzwerkressourcen zu schonen. Eine ausführliche Beschreibung der Funktion der peer-to-peer Komponente eComP des GAS-Systems findet sich in [15].

Ein weiteres System, das auf die Unterstützung einer infrastrukturlosen Kontextverarbeitung und Kommunikation abzielt, ist der FuzzySpace [20]. Der FuzzySpace stellt eine Erweiterung des TupleSpace [8] dar, die den Erfordernissen ubiquitärer Informationsumgebungen angepasst wurde. Der FuzzySpace kann grundsätzlich auch als zentralisiertes System betrieben werden, wenn nur eine einzige Implementierung in einer ubiquitären Informationsumgebung aktiv ist. Die Funktion unterscheidet sich dann nicht grundlegend von der eines Broker-Systems. Allerdings kann der FuzzySpace auch als verteilte Datenstruktur implementiert werden. Dann ähnelt die Arbeitsweise der des GAS-Systems. Der verteilte FuzzySpace wird aber im Gegensatz zu den GAS-Klienten Anfragen nicht an andere Artefakte weiterleiten, sondern Daten in den Implementierungen unabhängig voneinander repliziert halten. Um diese Replikation zu gewährleisten kann es allerdings nötig sein, dass der verteilte FuzzySpace mittels zusätzlicher Datagramme die enthaltenen Kontexte synchronisiert.

Auch für ein dezentrales Modell ist eine auf Subskriptionen basierende Kontextkommunikation möglich. Hierbei wird die Anfrage einer Kontextsenke durch die Subskription ersetzt. Diese müssen von den einzelnen Kontextquellen unabhängig verwaltet werden. Wird dann ein neuer Kontext verfügbar, kann die Quelle diesen direkt an die registrierte Kontextsenke verschicken.

## **6 Kostenrechnung**

Basierend auf dem vorgestellten Kostenmodell wird nun beispielhaft für drei unterschiedliche Kommunikations- und Verarbeitungsmodelle für Kontextinformationen eine Kostenrechnung durchgeführt.

### **6.1 Grundannahmen für die Kostenrechnung**

Nachfolgend werden einige Grundannahmen diskutiert, welche die Basis für die Kostenrechnung bilden: Da das Kostenmodell ein rundruffähiges Kommunikationsmedium voraussetzt (vgl. Abschnitt 3), soll für die beispielhafte Kostenrechnung von Kommunikationskosten für die Verwaltung abgesehen werden, die durch Partitionierung der Netzwerksegmente hervorgerufen werden könnten. Im Falle der zentralisierten Organisation und der zentralisierten Vermittlung betrifft dies zum Beispiel Kosten für Mehrfachanfragen, falls die zentrale Instanz nicht erreichbar ist. Bei einer dezentralen Organisation können hier, wie beim FuzzySpace, Kosten zum Beispiel für die Synchronisation anfallen.

Grundlage der Kostenrechnung ist ein Basisprozess der Kontextverarbeitung. Der Basisprozess  $B = \sum b$  sei definiert als die Aggregation zweier Kontexte und die Bereitstellung des aggregierten Kontextes. Die Kostenrechnung ermittelt dann die durchschnittlichen Kosten für einen Basisprozess  $\bar{K}(B) = \sum \bar{v}(b) + \sum \bar{\kappa}(b)$ . Diese Grundkosten fallen separat für jedes kontexterzeugende Objekt und jeden Erzeugungsprozess in einer ubiquitären Informationsumgebung an.

Die Kosten der Kommunikation und Kontextverarbeitung skalieren mit drei Einflussfaktoren. Diese sind die Anzahl der Artefakte  $n_{art}$ , die Anzahl der Netzwerksegmente  $n_{seg}$  und die Dynamik in der ubiquitären Informationsumgebung. Die Dynamik in der Umgebung beruht auf der Mobilität der Artefakte, sie repräsentiert die sich ändernde Dienst- und Verarbeitungstopologie durch das mobilitätsbedingte Hinzukommen und Verschwinden von aktiven Objekten. Die Verwendung eines Basisprozesses für die Kostenrechnung normalisiert die Kosten gegen die Anzahl der Artefakte in der ubiquitären Informationsumgebung. In allen betrachteten Modellen skalieren die Gesamtkosten somit linear mit der Anzahl der aktiven Artefakte. Für diese beispielhafte Kostenrechnung soll weiter davon ausgegangen werden, dass alle Netzwerksegmente eine Durchleitung der Daten benötigen. Die Dynamik sei durch einen Faktor  $d \in [0; 1]$  repräsentiert. Der Faktor  $d$  gibt an, bei wie viel Prozent der Basisprozesse eine erneute Dienstfindung nötig ist. In allen anderen Fällen wird davon ausgegangen, dass bereits alle Klienten über die nötigen Informationen zur Dienstnutzung verfügen.

Der Basisprozess  $B$  als Berechnungsgrundlage stellt die minimalen Kosten dar, die für einen Kontextableitungsprozess angenommen werden können. Die obige Definition abstrahiert vollständig von Skalierungseffekten der Kontextverarbeitung, die erfahrungsgemäß in großen ubiquitären Informationsumgebungen zu erwarten und auch gewünscht sind: werden in einer ubiquitären Informationsumgebung Kontexte produziert, können diese weiterverarbeitet und daraus neue Kontexte produziert werden. Man muss daher annehmen, dass die Anzahl, der in einer ubiquitären Informationsumgebung vorhandenen Artefakte, die Kontexte produzieren und die Anzahl der Kontextableitungen, die durch  $B$  beschrieben werden, in einem exponentiellen Zusammenhang stehen. Die hier vorgenommene lineare Definition stellt damit eine untere Schranke der zu erwartenden Kosten dar.

Um eine übersichtliche Darstellung zu ermöglichen, werden die beiden Faktoren  $n_{seg}$  und  $d$  separat variiert. Dabei wird der jeweils andere Faktor mit einem festen Grundwert versehen ( $n_{seg} = 1$  und  $d = 0.3$ ). Betrachtet werden die drei zuvor identifizierten Modellierungen der Kommunikation und Verarbeitung von Kontexten, die sich hinsichtlich der zu erwartenden Kommunikationskosten unterscheiden. Für jeden dieser Fälle werden die Kosten für den anfragebasierten und den subskriptionsbasierten Betrieb gegenübergestellt.

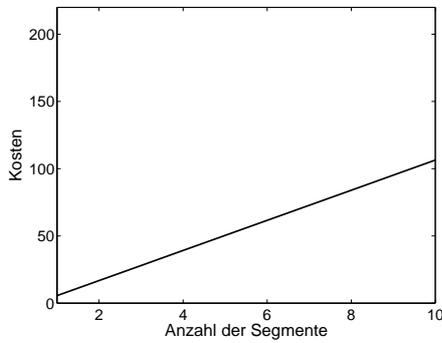
## 6.2 Fall I: Vollständig zentralisiertes System

Zunächst wird der Fall einer zentralen Systemorganisation betrachtet. Die zentrale Instanz  $z \in I$  dient der Speicherung und Verarbeitung der Kontexte. Neben der Zentralinstanz werden zwei Kontextquellen  $q_1, q_2 \in A$  betrachtet die Kontexte erzeugen, sowie eine Kontextsenke  $s \in A$ , die einen Kontext aus  $z$  benötigt. Mit den getroffenen Grundannahmen gilt dann für die Kommunikationskosten:

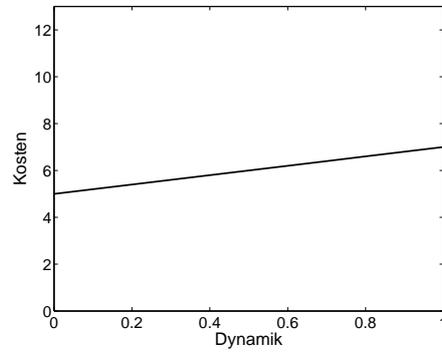
$$\kappa(q_1, z) = \kappa(q_2, z) = \kappa(z, s) = (2 \cdot n_{seg}) - 1$$

Und für die Kosten für die Fusion eines Kontextes in  $z$ :

$$v(z, 1) = 1$$



**Abbildung 5.** Zentralisiertes System mit Anfrage: variable Segmentanzahl



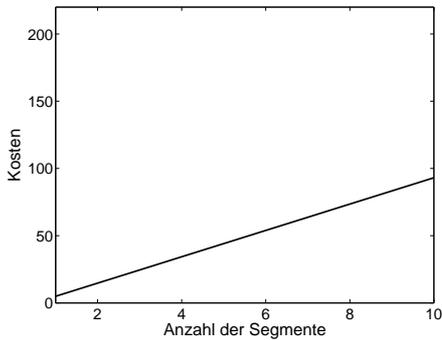
**Abbildung 6.** Zentralisiertes System mit Anfrage: variable Dynamik

Im anfragebasierten Fall setzen sich die Kosten des Basisprozesses zusammen aus den Kosten für die Anmeldung von  $q_1$  und  $q_2$ , den Kosten für die Kommunikation zweier Kontexte von  $q_1$  und  $q_2$  nach  $z$  und den Verarbeitungskosten für die Erzeugung eines neuen Kontextes. Hinzu kommen die Kosten für die Anfrage des neuen Kontextes von  $s$  und die Übermittlungskosten von  $z$  nach  $s$ . Daraus ergeben sich die durchschnittlichen Gesamtkosten eines Basisprozesses zu:

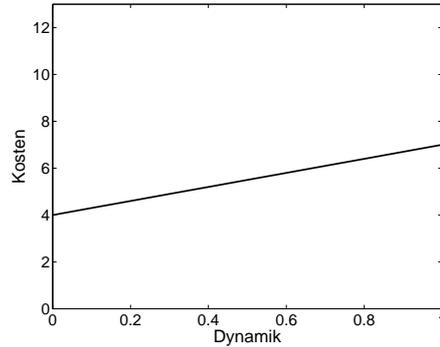
$$\bar{K}(B) = d \cdot (((2 \cdot n_{seg}) - 1) \cdot 7) + (1 - d) \cdot (((2 \cdot n_{seg}) - 1) \cdot 5) \quad (11)$$

Abbildung 5 zeigt die Kosten für ubiquitäre Informationsumgebungen mit 1 bis 10 Netzwerksegmenten. In Abbildung 6 sind die Kosten mit variabler Dynamik aufgetragen.

In beiden Fällen zeigt sich ein linearer Zusammenhang. Die erste Ableitung  $\bar{K}'$  ist ein Maß für die Adaptivität des Modells. Je kleiner der Wert von  $\bar{K}'$  ausfällt, um so geringer wird das System durch die Skalierung beeinflusst.



**Abbildung 7.** Zentralisiertes System mit Subskription: variable Segmentanzahl



**Abbildung 8.** Zentralisiertes System mit Subskription: variable Dynamik

Im Falle subskriptionsbasierter Kommunikation entstehen Kosten für die Anmeldung der Kontexte von  $q_1$  und  $q_2$  bei  $z$ , sowie deren Übermittlung und Verarbeitung in  $z$ . Dazu kommen die Kosten für die Subskription von  $s$  auf den neuen Kontext und dessen Kommunikation von  $z$  nach  $s$ . Die durchschnittlichen Gesamtkosten eines Basisprozesses sind dann:

$$\bar{K}(B) = d \cdot (((2 \cdot n_{seg}) - 1) \cdot 7) + (1 - d) \cdot (((2 \cdot n_{seg}) - 1) \cdot 4) \quad (12)$$

Die Kosten in Abhängigkeit von der Anzahl der Netzwerksegmente und der Dynamik sind in den Abbildungen 7 und 8 dargestellt. Tabelle 1 fasst die Ergebnisse für vollständig zentralisierte Systeme zusammen. Es zeigt sich, dass die Grundkos-

	$\bar{K}(B)$	$\bar{K}'_{n_{seg}}$	$\bar{K}'_d$
Anfrage	5,6	11,2	2,0
Subskription	4,9	9,8	3,0

**Tabelle 1.** Kosten in zentral organisierten Systemen

ten für die subskriptionsbasierte Kommunikation in zentral organisierten ubiquitären Informationsumgebungen geringer sind als für den Fall einer anfragebasierten Kommunikation. Subskriptionsbasierte Systeme skalieren besser mit der Anzahl der Netzwerksegmente, die an einer Kommunikation beteiligt sind, wogegen anfragebasierte Systeme besser mit der Dynamik in der ubiquitären Informationsumgebung skalieren.

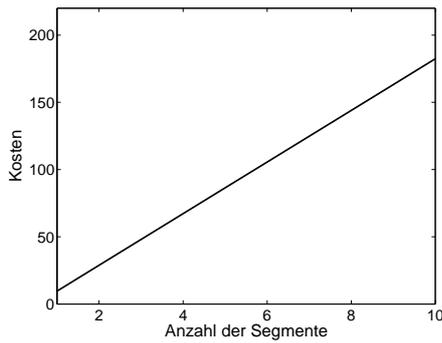
### 6.3 Fall II: Zentralisierte Vermittlung

Im Falle zentral vermittelter Systeme dient die zentrale Instanz lediglich der Dienstfindung. Sie übernimmt nicht, wie in vollständig zentralisierten Systemen, die Verarbeitung von Kontexten und deren Speicherung. Diese Aufgaben bleiben den Artefakten überlassen. Die zentrale Vermittlungsinstanz  $z \in I$  stellt Informationen über die Verfügbarkeit von Kontexten bereit und speichert, von welchen Artefakten  $q_1, q_2 \in A$  diese bereitgestellt werden können. Artefakte  $s \in A$ , die Kontexte weiterverarbeiten, können diese Informationen abfragen und dann die nötigen Kontexte direkt bei den  $q_1$  und  $q_2$  beziehen. Mit den getroffenen Grundannahmen gilt dann für die Kommunikationskosten:

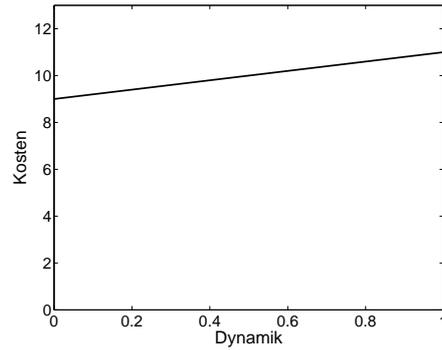
$$\kappa(q_1, z) = \kappa(q_2, z) = \kappa(z, s) = \kappa(q_1, s) = \kappa(q_2, s) = (2 \cdot n_{seg}) - 1$$

Die Kosten für die Weiterverarbeitung eines Kontextes in  $s$  betragen:

$$v(s, 1) = 1$$



**Abbildung 9.** Zentrale Vermittlung mit Anfrage: variable Segmentanzahl

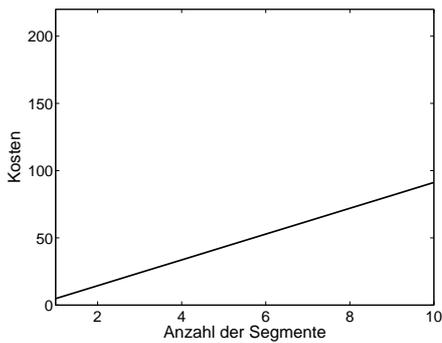


**Abbildung 10.** Zentrale Vermittlung mit Anfrage: variable Dynamik

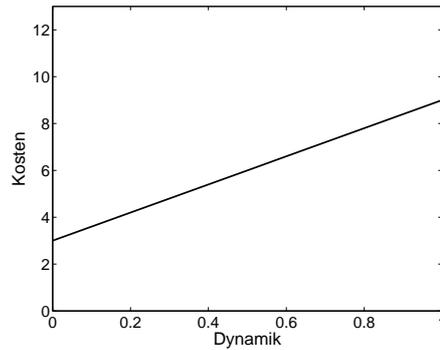
Auch hier sollen der Fall eines rein anfragebasierten und der Fall eines subskriptionsbasierten Systems unterschieden werden. In einem anfragebasierten System setzen sich die Kosten aus den Dienstanmeldungen von  $q_1$  und  $q_2$  bei  $z$ , den Dienstanfragen für die beiden zu verarbeitenden Kontexte von  $s$  an  $z$ , sowie die darauf folgende Dienstantwort von  $z$  an  $s$  zusammen. Hinzu kommen die Kontextabfragen von  $s$  an  $q_1$  und  $q_2$  und deren Antworten an  $s$ . Daraus ergeben sich die durchschnittlichen Gesamtkosten eines Basisprozesses zu:

$$\bar{K}(B) = d \cdot (((2 \cdot n_{seg}) - 1) \cdot 11) + (1 - d) \cdot (((2 \cdot n_{seg}) - 1) \cdot 9) \quad (13)$$

Das Skalierungsverhalten in diesem Fall ist den Abbildungen 9 und 10 zu entnehmen. Auffallend sind hier hohe Grundkosten und ein extrem schlechtes Skalierungsverhalten mit der Anzahl der Netzwerksegmente im System (vergleiche auch Tabelle 2).



**Abbildung 11.** Zentrale Vermittlung mit Subskription: variable Segmentanzahl



**Abbildung 12.** Zentrale Vermittlung mit Subskription: variable Dynamik

Für den subskriptionsbasierten Fall ergeben sich Kosten aus der Anmeldung der Kontextquellen  $q_1$  und  $q_2$  an  $z$ , der Subskription von  $s$  für die beiden benötigten Quellkontexte bei  $z$  und der Verteilung dieser Subskriptionen durch  $z$  an alle registrierten Kontextquellen (durch Rundruf). Hinzu kommen auch hier die Kosten für die Übertragung der Kontexte von  $q_1$  und  $q_2$  an  $s$  und die Verarbeitungskosten, die in  $s$  bei der Fusion der Kontexte anfallen. Die durchschnittlichen Gesamtkosten eines Basisprozesses belaufen sich dann auf:

$$\bar{K}(B) = d \cdot (((2 \cdot n_{seg}) - 1) \cdot 9) + (1 - d) \cdot (((2 \cdot n_{seg}) - 1) \cdot 3) \quad (14)$$

Abbildung 11 zeigt die Kostenentwicklung bei der Skalierung der Anzahl der Netzwerksegmente, das Skalierungsverhalten mit variierender Dynamik in der ubiquitären Informationsumgebung ist in Abbildung 12 dargestellt. Es zeigt sich hier, dass bei moderaten Grundkosten die Skalierung mit der Anzahl der Netzwerksegmente mit der bei zentralisierten Organisationen vergleichbar ist. Die Skalierung mit der Dynamik im System fällt dagegen schlechter aus. Tabelle 2 fasst die Ergebnisse für zentral vermittelte Systeme zusammen. Deutlich zu erkennen sind die unterschiedlichen Grundkosten und das gegenläufige Skalierungsverhalten der beiden Varianten zentral vermittelter Systeme.

	$\bar{K}(B)$	$\bar{K}'_{n_{seg}}$	$\bar{K}'_d$
Anfrage	9,6	19,2	2,0
Subskription	4,8	9,6	6,0

**Tabelle 2.** Kosten in zentral vermittelten Systemen

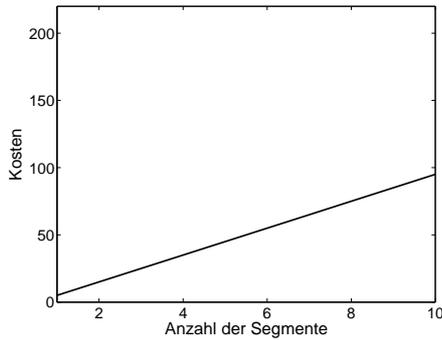
#### 6.4 Fall III: Dezentrales System

Rein dezentral organisierte ubiquitäre Informationsumgebungen beinhalten im Gegensatz zu zentralisierten oder zentral vermittelten Systemen keine zentrale Komponente, welche die Kommunikation und den Datenaustausch organisiert. Im Falle dezentraler Systeme muss also sowohl die Kontextverarbeitung, als auch die Dienstfindung von den Artefakten  $q_1, q_2, s \in A$  selbst übernommen werden. Mit den getroffenen Grundannahmen gilt dann für die Kommunikationskosten:

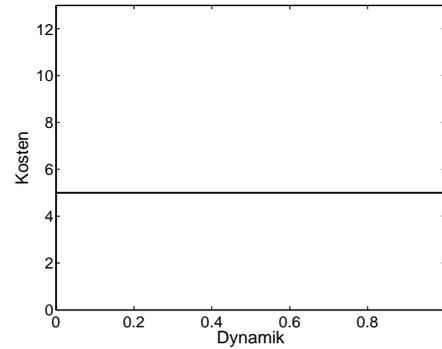
$$\kappa(q_1, s) = \kappa(q_2, s) = (2 \cdot n_{seg}) - 1$$

Die Kosten für die Weiterverarbeitung eines Kontextes in  $s$  betragen:

$$v(s, 1) = 1$$



**Abbildung 13.** Dezentrales System mit Anfrage: variable Segmentanzahl

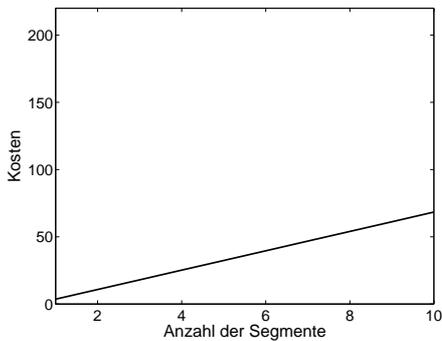


**Abbildung 14.** Dezentrales System mit Anfrage: variable Dynamik

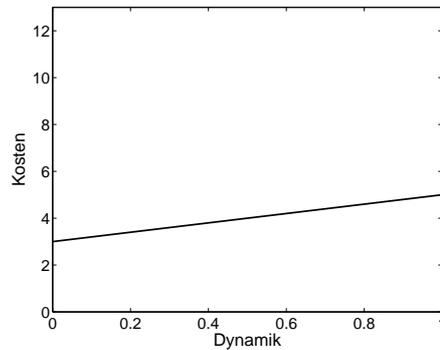
Auch hier sollen anfragebasierte Lösungen solchen gegenübergestellt werden, die auf der Basis von Subskriptionen arbeiten. In anfragebasierten Systemen entstehen Kosten für die Anfragen der benötigten Kontexte von  $s$  an  $q_1$  und  $q_2$ , sowie deren Antworten an  $s$  mit den entsprechenden Kontexten. Zusätzlich entstehen Kosten für die Verarbeitung dieser beiden Quellkontexte in  $s$ . Daraus ergeben sich die durchschnittlichen Gesamtkosten eines Basisprozesses zu:

$$\bar{K}(B) = d \cdot (((2 \cdot n_{seg}) - 1) \cdot 5) + (1 - d) \cdot (((2 \cdot n_{seg}) - 1) \cdot 5) \quad (15)$$

Abbildung 13 zeigt ein moderates Skalierungsverhalten bei einer steigenden Anzahl von Netzwerksegmenten. In Abbildung 14 ist zu erkennen, dass dieses System offenbar unabhängig von der Dynamik in einer ubiquitären Informationsumgebung arbeitet, also somit ideal in dieser Hinsicht skaliert.



**Abbildung 15.** Dezentrales System mit Subskription: variable Segmentanzahl



**Abbildung 16.** Dezentrales System mit Subskription: variable Dynamik

Für den Fall eines subskriptionsbasierten Systems entstehen Kosten für die Subskriptionen von  $s$  bei  $q_1$  und  $q_2$ . Weitere Kosten fallen an für die Bereitstellung der Kontexte von  $q_1$  und  $q_2$ , sobald diese verfügbar werden und die Kosten für die Weiterverarbeitung dieser Daten in  $s$ . Daraus ergeben sich die durchschnittlichen Gesamtkosten eines Basisprozesses zu:

$$\bar{K}(B) = d \cdot (((2 \cdot n_{seg}) - 1) \cdot 5) + (1 - d) \cdot (((2 \cdot n_{seg}) - 1) \cdot 3) \quad (16)$$

Abbildung 15 zeigt das sehr gute Skalierungsverhalten dezentraler Systeme mit Subskription gegenüber der Anzahl der Netzwerksegmente. Aus Abbildung 16 ist zu entnehmen, dass die Kosten in einem subskriptionsbasierten System von der Dynamik in der Umgebung abhängen, die Grundkosten sind allerdings gering.

	$\bar{K}(B)$	$\bar{K}'_{n_{seg}}$	$\bar{K}'_d$
Anfrage	5,0	10,0	0,0
Subskription	3,6	7,2	2,0

**Tabelle 3.** Kosten in dezentralen Systemen

Tabelle 3 stellt die Ergebnisse für dezentral organisierte ubiquitäre Informationsumgebungen gegenüber. Trotz des idealen Verhaltens anfragebasierter Systeme in dynamischen Umgebungen ist hier zu erkennen, dass die durchschnittlichen Gesamtkosten der Kontextverarbeitung in subskriptionsbasierten Systemen geringer sind.

## 6.5 Fazit

Grundsätzlich muss bei dieser Kostenbetrachtung beachtet werden, dass reale ubiquitäre Informationsumgebungen zwar einer der drei vorgestellten Modellierungen folgen, innerhalb dieser aber meist einen Mix der betrachteten Fälle darstellen. Zum einen erlauben verschiedene Systeme (zum Beispiel PACE [11] und das Context Toolkit [7]) sowohl Subskriptionen als auch direkte Anfragen, zum anderen wird die Dynamik in einer ubiquitäre Informationsumgebung niemals völlig stabil bleiben.

Systeme Organisation	$\bar{K}(B)$	$\bar{K}'_{n_{seg}}$	$\bar{K}'_d$
dezentral subskriptionsbasiert	3,6	7,2	2,0
zentral vermittelt subskriptionsbasiert	4,8	9,6	6,0
zentralisiert subskriptionsbasiert	4,9	9,8	3,0
dezentral anfragebasiert	5,0	10,0	0,0
zentralisiert anfragebasiert	5,6	11,2	2,0
zentral vermittelt anfragebasiert	9,6	19,2	2,0

**Tabelle 4.** Kosten im Vergleich

Das Ergebnis der Kostenbetrachtung ist in Tabelle 4 zusammengefasst. Die einzelnen Systemkonfigurationen sind aufsteigend nach den durchschnittlichen Kosten eines Basisprozesses sortiert. Es zeigt sich, dass offenbar Systeme, die vorwiegend subskriptionsbasiert arbeiten, solchen, die anfragebasiert arbeiten, hinsichtlich der durchschnittlichen Kosten überlegen sind. Als kostengünstigste Modellierung einer ubiquitäre Informationsumgebung zeigt sich das dezentrale subskriptionsbasierte System. Die ermittelten Grundkosten eines Basisprozesses wurden ermittelt für ein Netzwerksegment  $n_{seg} = 1$  und bei einem Dynamikfaktor von  $d = 0.3$ .

In einer dezentral organisierten ubiquitären Informationsumgebung mit Subskriptionsmechanismus für Kontexte, entstehen die geringsten Kosten pro Basisprozess. Darüber hinaus verfügt diese Konfiguration mit einem Wert von  $\bar{K}'_{n_{seg}} = 7,2$  über das beste Skalierungsverhalten bei einer steigenden Anzahl von Netzwerksegmenten (siehe Abbildung 17).

Abbildung 18 zeigt, dass in hochdynamischen Umgebungen nur das dezentrale anfragebasierte System besser skaliert, allerdings auf höherem Niveau. In beiden vergleichenden Grafiken bildet das dezentrale subskriptionsbasierte Modell eine untere Schranke für die Grundkosten eines Basisprozesses.

Neben diesem Kostenvorteil bietet die dezentrale Lösung weitere Vorteile für die Realisierung von ubiquitären Informationsumgebungen. Einer der wichtigsten dieser Vorteile ist, dass es in einem dezentral organisierten System keine zentrale Instanz gibt, deren Ausfallen das Gesamtsystem funktionsunfähig machen kann. Hinzukommt, dass nur in dezentral organisierten Systemen die Möglichkeit eines echten peer-to-peer Betriebes besteht, für den keinerlei Infrastruktur nötig ist. Das heißt

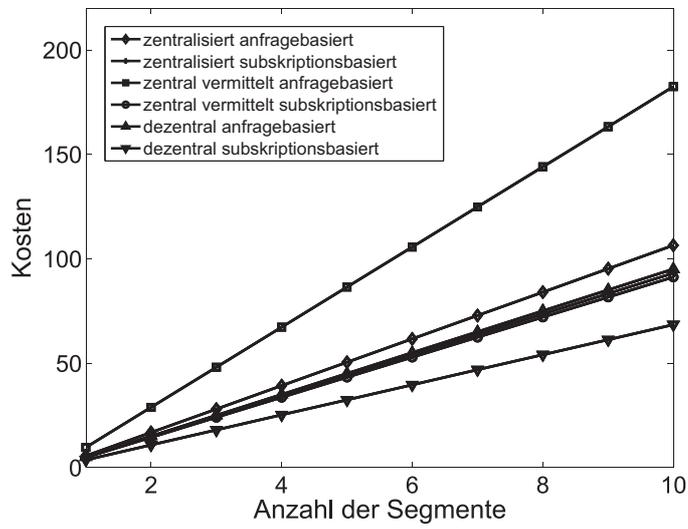


Abbildung 17. Übersicht: Skalierung bei variabler Segmentanzahl

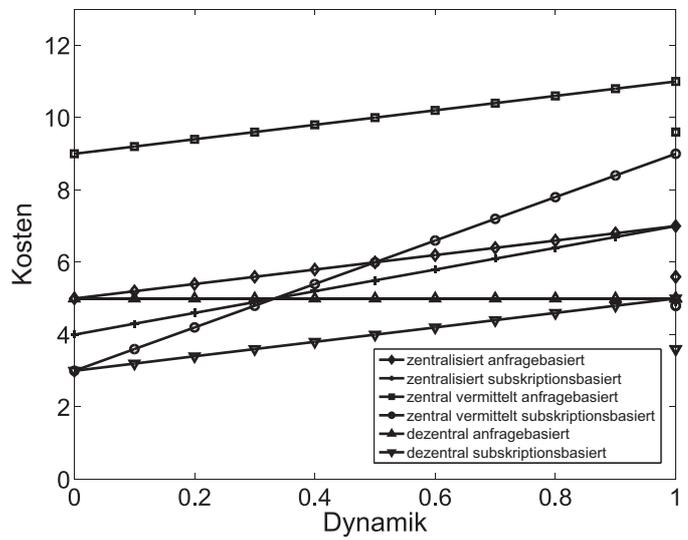


Abbildung 18. Übersicht: Skalierung bei variabler Dynamik

nur dezentrale Systeme sind in der Lage die Ad-hoc-Ausbildung von ubiquitären Informationsumgebungen zu unterstützen, wo immer Artefakte aufeinandertreffen und miteinander kommunizieren können.

## Literatur

- [1] BARDRAM, Jakob E.: The Java Context Awareness Framework (JCAF) - A Service Infrastructure and Programming Framework for Context-Aware Applications. In: *Proceedings of the 3rd International Conference on Pervasive Computing (Pervasive 2005)*, 2005
- [2] CHEN, Guanling: *Solar: Building A Context Fusion Network for Pervasive Computing*, DARTMOUTH COLLEGE Hanover, New Hampshire, Diss., 2004
- [3] CHEN, Harry: *An Intelligent Broker Architecture for Pervasive Context-Aware Systems*, University of Maryland, Diss., December 2004
- [4] COUTAZ, Joelle ; CROWLEY, James L. ; DOBSON, Simon ; GARLAN, David: Context is key. In: *Communications of the ACM* 48 (2005), Nr. 3, S. 49–53. – ISSN 0001–0782
- [5] DEY, A. K.: Understanding and using context. In: *Personal and Ubiquitous Computing Personal and Ubiquitous Computing, Special issue on Situated Interaction and Ubiquitous Computing* 5, 1 (2001), S. 4–7
- [6] DEY, A. K. ; ABOWD, G. D. ; SALBER, D.: A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. In: *Human-Computer Interaction* 16 (2-4) (2001), S. 97–166
- [7] DEY, A.K.: *Providing Architectural Support for Building Context-Aware Applications*, Georgia Institute of Technology, Diss., December 2000
- [8] GELERNTER, David ; BERNSTEIN, Arthur J.: Distributed communication via global buffer. In: *PODC '82: Proceedings of the first ACM SIGACT-SIGOPS symposium on Principles of distributed computing*. New York, NY, USA : ACM Press, 1982. – ISBN 0–89791–081–8, S. 10–18
- [9] GU, T. ; PUNG, H. K. ; ZHANG, D. Q.: A Middleware for Building Context-Aware Mobile Services. In: *Proceedings of IEEE Vehicular Technology Conference (VTC 2004)*, 2004
- [10] HENRICKSEN, K. ; INDULSKA, J. ; RAKOTONIRAINY, A.: Modeling context information in pervasive computing systems. In: MATTERN, F. (Hrsg.) ; NAGHSHINEH, M. (Hrsg.): *Proceedings of 1st International Conference on Pervasive Computing*. Zurich, Switzerland : Springer, 2002, S. pp. 167–180

- [11] HENRICKSEN, Karen ; INDULSKA, Jadwiga ; MCFADDEN, Ted ; BALASUBRAMANIAM, Sasitharan: Middleware for Distributed Context-Aware Systems. In: *Lecture Notes in Computer Science* Bd. 3760. Springer, Oct 2005, S. 846 – 863
- [12] HONG, J.I.: The Context Fabric: An Infrastructure for Context-Aware Computing. In: *Human Factors in Computing Systems: CHI, 2002*
- [13] INDULSKA, Jadwiga ; ROBINSON, Ricky ; RAKOTONIRAINY, Andry ; HENRICKSEN, Karen: Experiences in Using CC/PP in Context-Aware Systems. In: *MDM '03: Proceedings of the 4th International Conference on Mobile Data Management*. London, UK : Springer-Verlag, 2003. – ISBN 3–540–00393–2, S. 247–261
- [14] KAMEAS, A. ; BELLIS, S. ; MAVROMMATI, I. ; DELANEY, K. ; COLLEY, M. ; POUNDS-CORNISH, A: An architecture that treats everyday objects as communicating tangible components. In: *Pervasive Computing and Communications*, IEEE, March 2003, S. 115 – 122
- [15] KAMEAS, A. ; MAVROMMATI, I. ; RINGAS, D. ; WASON, P.: eComP: an architecture that supports P2P networking among ubiquitous computing devices. In: *Peer-to-Peer Computing, 2002. (P2P 2002)*, 2002, S. 57 – 64
- [16] KHEDR, M. ; KARMOUCH, A.: Negotiating context information in context-aware systems. In: *IEEE Intelligent Systems and Their Applications* 19(6) (2004), S. 21– 29
- [17] ROMAN, M. ; HESS, C. ; CERQUEIRA, R. ; RANGANATHAN, A. ; CAMPBELL, R.H. ; NAHRSTEDT, K.: A middleware infrastructure for active spaces. In: *Pervasive Computing, IEEE* 1(4) (2002), S. 74 – 83
- [18] SCHILIT, B. ; ADAMS, N. ; WANT, R.: Context-aware computing applications. In: *1st International Workshop on Mobile Computing Systems and Applications*, 1994
- [19] SCHILIT, Bill N. ; ADAMS, Norman ; GOLD, Rich ; TSO, Michael ; WANT, Roy: The PARCTAB mobile computing system. In: *Proceedings Fourth Workshop on Workstation Operating Systems (WWOS-IV)*, 1993
- [20] SCHMIDT, Albrecht ; GELLERSEN, Hans-Werner: Modell, Architektur und Plattform für Informationssysteme mit Kontextbezug. In: *Informatik - Forschung und Entwicklung* 16 4 (2001), November, S. 213 – 224
- [21] SCHMIDT, Albrecht ; LAERHOVEN, Kristof V.: How to build smart appliances. In: *IEEE Personal Communications* 8(4) (2001), S. 66 – 71

- [22] SIG, Bluetooth: *The official Bluetooth Web site*, 2006
- [23] YAU, S. S. ; KARIM, F. ; WANG, Y. ; WANG, B. ; S.GUPTA: Reconfigurable Context-Sensitive Middleware for Pervasive Computing. In: *IEEE Pervasive Computing* 1(3) (2002), S. 33–40
- [24] ZIMMER, Tobias: Towards a Better Understanding of Context Attributes. In: *Proceedings of PerCom 2004*. Orlando, USA, März 2004, S. 23–28

### Abbildungsverzeichnis

1	Zentralisiertes System . . . . .	5
2	Zentralisierte Vermittlung mit Anfrage . . . . .	7
3	Zentralisierte Vermittlung mit Subskription . . . . .	7
4	Vollständig dezentrales System . . . . .	8
5	Zentralisiertes System mit Anfrage: variable Segmentanzahl . . . . .	11
6	Zentralisiertes System mit Anfrage: variable Dynamik . . . . .	11
7	Zentralisiertes System mit Subskription: variable Segmentanzahl . . . . .	12
8	Zentralisiertes System mit Subskription: variable Dynamik . . . . .	12
9	Zentrale Vermittlung mit Anfrage: variable Segmentanzahl . . . . .	13
10	Zentrale Vermittlung mit Anfrage: variable Dynamik . . . . .	13
11	Zentrale Vermittlung mit Subskription: variable Segmentanzahl . . . . .	14
12	Zentrale Vermittlung mit Subskription: variable Dynamik . . . . .	14
13	Dezentrales System mit Anfrage: variable Segmentanzahl . . . . .	15
14	Dezentrales System mit Anfrage: variable Dynamik . . . . .	15
15	Dezentrales System mit Subskription: variable Segmentanzahl . . . . .	16
16	Dezentrales System mit Subskription: variable Dynamik . . . . .	16
17	Übersicht: Skalierung bei variabler Segmentanzahl . . . . .	18
18	Übersicht: Skalierung bei variabler Dynamik . . . . .	18

### Tabellenverzeichnis

1	Kosten in zentral organisierten Systemen . . . . .	12
2	Kosten in zentral vermittelten Systemen . . . . .	15
3	Kosten in dezentralen Systemen . . . . .	16
4	Kosten im Vergleich . . . . .	17