

# Mobile Devices for Interacting with Tagged Objects

## Development Support and Usability

**Christof Roduner**

Doktorprüfung, 17. Februar 2010



# Hintergrund

- Eine zentrale Motivation im Pervasive Computing:
  - Brückenschlag zwischen **realer Welt** und **digitaler Welt** durch **Anreicherung** von physischen Objekten mit Informationen und Diensten.



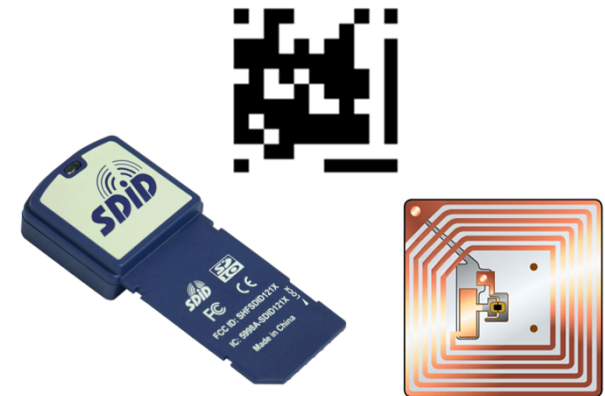
Bilder: Steve Hinske, ETH Zürich



# Elemente zur Umsetzung

## ■ Identifikation

- Meist mittels **Tags** und Lesegerät
- RFID-Etiketten, Barcodes



## ■ Interaktion

- Oft mittels mobilen Geräten
- **Mobiltelefon** als Benutzerschnittstelle vieler Ubicomp-Systeme



# Mobile Geräte und Tagging-Technologien

## ■ Entwicklung:

- Erkennung von EAN13-Barcodes
- Bidirektionale Kommunikation mittels NFC
- Verschmelzung von Identifikation und Interaktion



## ■ Konsequenz:

- Digitale Anreicherung von Objekten im grossen Stil
- Nicht beschränkt auf ausgesuchte Objekte
- Zahlreiche Akteure
  - Hersteller, Händler und Dritte können Dienste an Objekte „anheften“

# Herausforderungen: Usability und Umsetzung

- Wo ist die digitale Anreicherung sinnvoll?
  - **Benutzbarkeit** liegt nicht überall auf der Hand
- Wie können Dienste rasch implementiert und den Benutzern zugänglich gemacht werden?
  - Fehlende **Abstraktionen** für Entwickler
  - Fehlende **Infrastruktur** zum Auffinden von Informationen und Diensten
  - **Monolithische** Anwendungen auf heutigen Mobiltelefonen

# Beiträge der Dissertation

## 1. Nutzbarkeitsstudie

- Mobiltelefone zur Bedienung von Geräten
- Nachweis der Über- bzw. Unterlegenheit nach Anwendungsfällen

## 2. Offene Infrastruktur

- Anbieten von Informationen und Diensten für physische Objekte
- Architektur und Umsetzung

## 3. Software-Framework

- Konzepte zur Unterstützung der Entwicklung
- Umsetzung im „Browser for the Internet of Things (BIT)“



## Teil 1

# Benutzbarkeit von Mobiltelefonen zur Bedienung von Geräten

# Mobiltelefone zur Bedienung von Geräten

- Idee:
  - Ablösung der traditionellen haptischen Benutzerschnittstelle
  - Auslagerung aufs Mobiltelefon
- Vorgeschlagen für:
  - Getränkeautomaten, Lichtschalter, Video-Recorder, DVD-Spieler, Fernseher, Anrufbeantworter, Projektoren, Stereoanlagen, Geldautomaten, Aufzüge, Heizungssteuerungen, Kopiergeräte, Autos, Mikrowellengeräte, Waschmaschinen, ...





# Wünschbar oder lediglich machbar?

- Was ist der Nutzen?
- Mit dem Mobiltelefon die Raumtemperatur steuern:
  - Sinnvoll fürs Ferienhaus
  - Aber für den Raum, in dem ich mich gerade befinde?
- Präsenz des Nutzers nötig:
  - Mikrowellenöfen
  - Getränkeautomaten



# Benutzerstudie

- 23 Teilnehmer
- 4 Geräte in unterschiedlichen Situationen, z.B.:
  - Einstellung am Gerät ändern
  - Problem lösen
- Teilnehmer in **zwei Gruppen**:
  1. Herkömmlich Bedienung der Geräte  
Problemlösung mit Hilfe von gedrucktem Handbuch
  2. Bedienung der Geräte mit Mobiltelefon  
Problemlösung mit Hilfe von Mobiltelefon
- Erhobene Daten:
  - **Benötigte Zeit** zur Erfüllung einer Aufgabe
  - **Subjektive Beurteilung** durch Benutzer

# Appliance Interaction Device (AID)



(a) Radio main menu

(b) Adj. water hardness

(c) Dishwasher user manual

(d) Printer troubleshooter

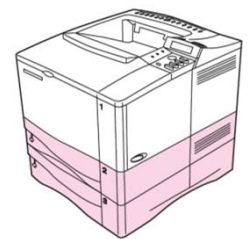
- Klassisches Layout mit Tastatur
- Simulierte Kommunikation
- Handbuchinhalt unverändert übernommen

# Hypothesen

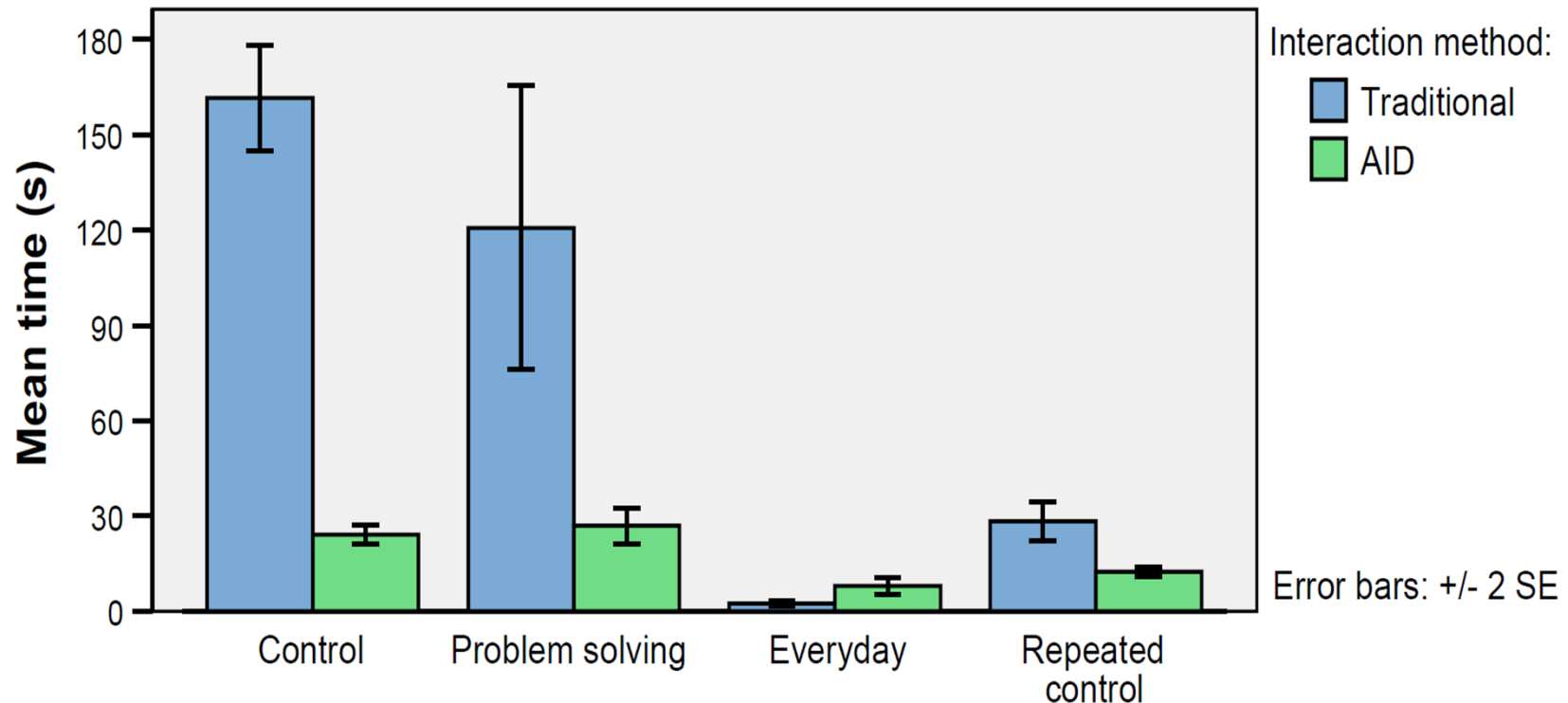
1. In **alltäglichen Situationen** können Geräte schneller auf herkömmliche Art bedient werden.
2. In **aussergewöhnlichen Situationen** ist die Interaktion mittels Mobiltelefon schneller.
3. Das manuelle **Beheben von Fehlern** wird durch das Mobiltelefon beschleunigt, wenn der Gerätestatus übermittelt wird. (**Ersatz des Handbuchs**)

# Geräte und Aufgaben

- Waschmaschine, Kaffeemaschine, Drucker, Radio
- 4 Aufgabentypen:
  - “Control tasks”:
    - Wasserhärte einstellen, Reinigungsseite drucken
  - “Problem solving tasks”:
    - Kaffeefilter wechseln, Belag auf Geschirr
  - “Everyday tasks”:
    - Sender wechseln, Druckjob abbrechen, Kaffee brauen
  - “Repeated control tasks”:
    - Erneutes Lösen eines bereits bekannten “control tasks”
- Insgesamt 18 Aufgaben
  - Ausschluss offensichtlich schlecht unterstützter Tasks



## Quantitative Ergebnisse: Zeit



- Statistisch **signifikanter** Unterschied bei bei allen Task-Typen (ANOVA,  $p < 0.001$ ), **Bestätigung der Hypothesen**
- AID auch schneller, wenn Benutzer **bereits mit Gerät vertraut**
- Problemlösung mit AID signifikant schneller, **auch ohne Gerätestatus**

# Qualitative Ergebnisse: Wettbewerb

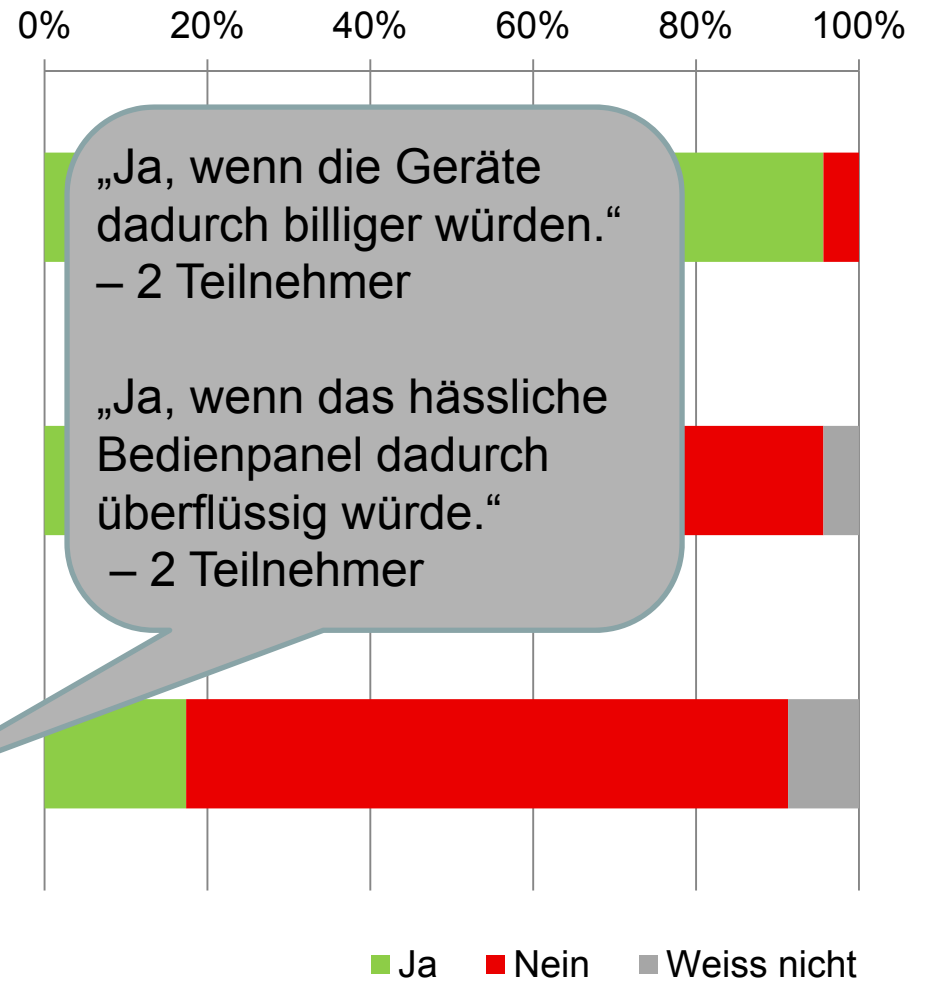
- Zum Abschluss **freie Wahl** der Interaktionsmethode
- In **69%** aller “control”- und “problem solving”-Tasks **Entscheidung für AID**
- Auch wenn Vorteile nicht von vornherein klar
- Äusserungen der Teilnehmer:
  - **Verunsicherung** führt zu Bevorzugung des AID
  - Direkte Bedienung des Radios ist **natürlicher**

# Qualitative Ergebnisse: Fragebogen

Benutzen Sie Geräte, für die Sie sich eine Bedienung mit dem AID wünschen?

„Akku leer?“  
„Telefon verlegt?“  
„Keine Netzabdeckung?“  
„Abhängigkeit von Technik?“  
... direkt?

Könnten Sie sich vorstellen, für alle Funktionen nur noch das AID zu verwenden?





# Zusammenfassung

- Sehr **positiv aufgenommen** von Benutzern
  - “Ich habe den AID gern benutzt.” → Zustimmung von 4.5 (Skala: 1-5)
- Schneller und einfacher in **aussergewöhnlichen** Situationen
- Keine Vorteile in **alltäglichen** Situationen
- Kostengünstiger Weg zu besserem **Display** und intuitiven **Eingabemöglichkeiten**
- Implikationen:
  - Bessere Benutzbarkeit auch ohne **grössere Bedienpanels**
  - Kein universeller Einsatz – stattdessen **hybride** Schnittstellen



## Teil 2

# Die *Open Lookup Infrastructure* zum Anbieten von Diensten für physische Objekte

# Szenario und Anwendungen

- Herkunfts-, Inhalts-, Preisinformation
- Hilfe bei Problemlösung basierend auf
- Auffinden des nächsten Service-Cent
- Empfehlungen (z.B. aus Umwelt-, Eth
- Qualitätsüberlegungen)
- ...
- Mögliche Stakeholder:
  - Hersteller
  - Andere Firmen
  - Konsumentenschutzorganisation
  - Individuen
  - ...



# Zentrale Anforderungen

- Finden von Informationen und Diensten basierend auf:
  - Thema (z.B. „alles zum Thema Gesundheit“)
  - Typen (z.B. strukturierte Angaben zu Inhaltsstoffen)
  - Kontext (z.B. Ort, Zeit, Gerätestatus)
- Offenheit für alle Stakeholder
- Erweiterbarkeit
  - Für neue Anwendungen
  - Für Standardisierung
  - Für existierende Informationssysteme
- Leichtgewichtig

# Resources und Resource Descriptions

- Resources:
  - Information
  - Beispiel: S
- Resource Description
  - Link zwisc

resource id	f5f7305bf097af39c68b790d817d7889f788f222
tag id	urn:epc:id:sgtin:640130.319951.7362
url	(empty)
data	2010-02-17
profile	http://foodindustry.org/profiles/exp-date/
context	(empty)
title	Expiration date
description	Expiration date for OrganicMilk, 1 liter
signature	(empty)



For: 7610200337481  
At: http://www...

**Nutrition Facts**  
For: 7610200337481  
At: http://www...



Resources

Resource Descriptions

Tagged Product

Res. Consumer

# Resource Repositories

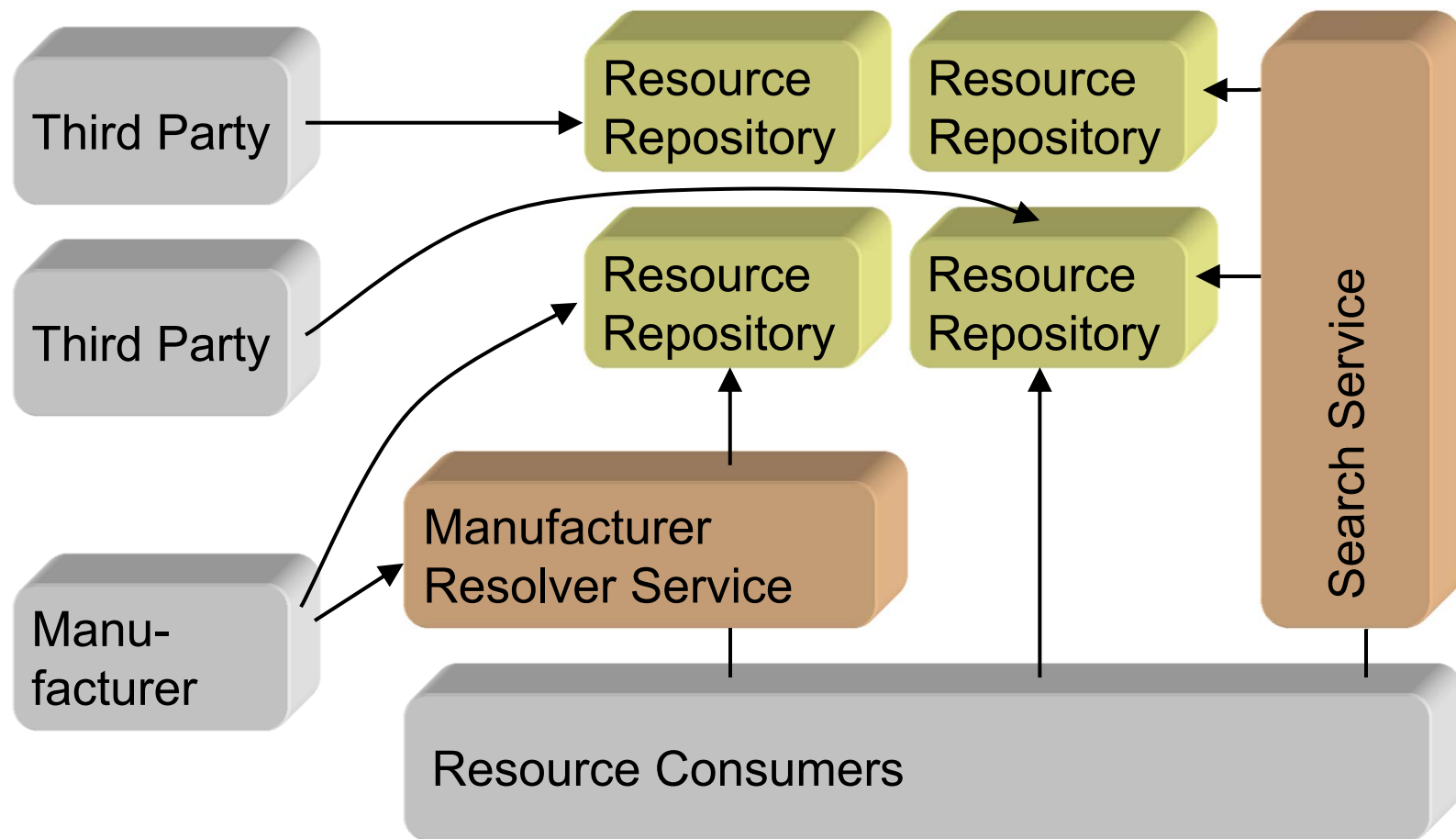
- Speichern Resource Descriptions, oft thematisch ähnlich
- Durch Anbieter von Resources betrieben oder gehostet
- Operationen:
  - RegisterResource, RemoveResource, LookupResource
- Offen für Community oder zugangsbeschränkt
  
- Beispiel einer Lookup-Anfrage:

tag id	urn:epc:id:sgtin:0652642:800031.400
profile	<a href="http://appliances.org/troubleshooting-hints/">http://appliances.org/troubleshooting-hints/</a>
search term	(empty)
context	Status=E683[hint]

# Finden von Resource Repositories

- Unterschiedliche Mechanismen, abhängig von Use-Case
- Vier Arten von Use-Cases
- Benutzer ist interessiert an...
  - Repository des **Herstellers**
    - Beispiel: Bedienungsanleitung
  - Einzellinem, **bestimmtem** Repository
    - Beispiel: Preisvergleichsdienst
    - URL von Freunden erhalten, aus Werbung erfahren etc.
  - **Mehreren**, vordefinierten Repositories
    - Beispiel: mehrere Konsumentenorganisationen
  - Resource Descriptions aus **irgendeinem** Repository
    - Beispiel: Volltextsuche
- Notwendigkeit für **Manufacturer Resolver Service** und **Search Service**

# Architekturübersicht





# Prototyp

- Mit Bindings für REST, Web Services und XML-RPC
- Manufacturer Resolver Service:
  - Integration mit **ONS**-Dienst von EPCglobal
- Search Service:
  - Basierend auf Apache Lucene
  - Mit **Web-Erweiterung**, in Anlehnung an Technorati:  

```
<a href="http://open-lookup.org/epc/0614141/100932" />  
<a href="http://open-lookup.org/context/status/JAM_A1" />
```
- Mehrere Dienste darauf aufbauend implementiert



# Zusammenfassung

- Prototyp deckt formulierte Anforderungen ab:
  - Offenheit
  - Erweiterbarkeit
  - Unterstützung von Kontext
- Infrastruktur erlaubt...
  - **Benutzern** das Finden von Informationen und Dienste,
  - **Diensteanbietern** den Zugang zu potenziellen Kunden und
  - **Herstellern** das Anbieten von digitalem Mehrwert.



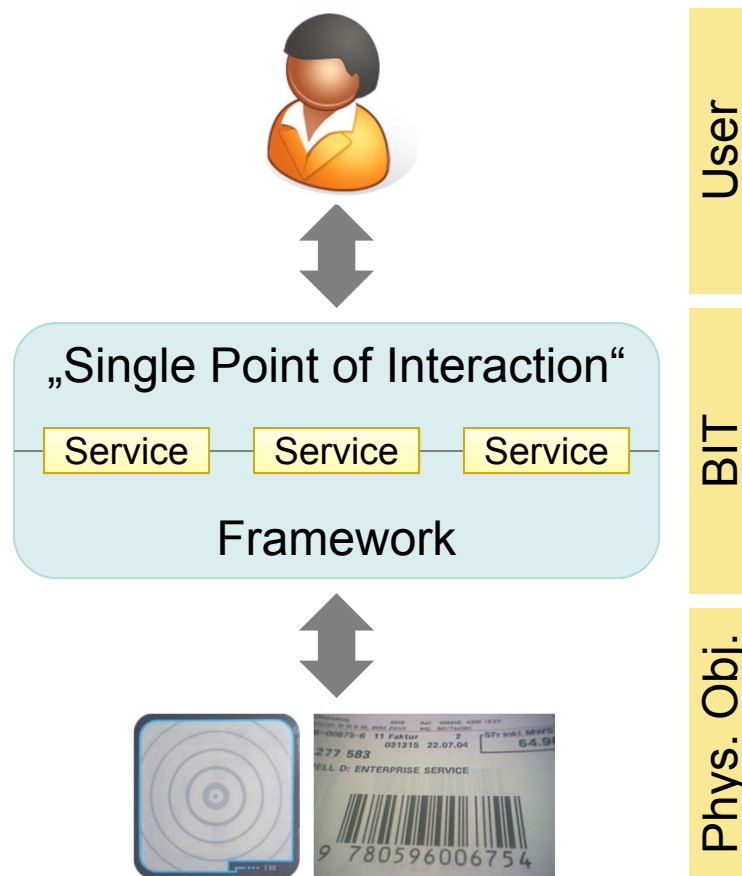
## Teil 3

# Software-Framework und Browser for the Internet of Things (BIT)

# Herausforderungen im mobilen Umfeld

- Softwareentwicklung **umständlich** – verglichen mit Web
  - Vielzahl von **Plattformen**
  - Web-Applikationen im Browser des Telefons
    - Fehlender **Zugriff auf Hardware**
    - Schlechtere **Reaktionszeit**
- Dienste für physische Produkte jedoch oft einfach
  - Erwünscht: **Agilität**
  - Ziel: Entwicklung nicht schwieriger als Erstellung einer Website
- Anwendungszentrierung
  - **Geeigneter** für physische Objekte: Dienstzentrierung
  - Interaktion mit Objekten ist oft **spontan**
  - Manuelle **Installation** einer Anwendung ist hinderlich

# Browser for the Internet of Things (BIT)



## Hauptziele:

- **Navigation** von Diensten
  - Gibt Überblick
  - Erleichtert Auswahl
  - Ermöglicht Wechsel
- **Koordination** von Diensten
- **Entwicklung** von Diensten

# Hauptkonzepte des Frameworks (1)

- Applets
  - Implementieren Dienste
  - Transparenter Download und automatische Ausführung
- Endpoints
  - Logische Verbindung zu physischem Objekt
- Runlists
  - Liste von Applets
  - Applets werden ausgeführt, wenn Objekt erkannt wird:
    - Jedes Applet erhält Kontrolle und kann kurze Ausgabe produzieren (Text und Icon)
    - Ergebnis der Runlist: aneinandergereihte Ausgaben

## Hauptkonzepte des Frameworks (2)

- Aggregation Perspective
  - Zur Anzeige von Runlist-Ergebnissen
  - Zum Start eines Applets durch Auswahl seiner Ausgabe
- Exclusive Perspective
  - Zur Anzeige eines vom Benutzer ausgewählten Applets
  - Bildschirm und User-Input zur freien Verfügung des Applets



# Entwicklung von Diensten mit BIT

- BIT Markup Language
  - Definition des GUIs in [Views](#), die Widgets enthalten
  - Layout über CSS
- Skriptsprache Lua
  - Skripts in [separaten Files](#) oder [eingebettet in Views](#)
  - Erinnert an JSP, PHP etc., aber Ausführung auf Mobiltelefon
- API
  - Benutzerschnittstelle
  - Persistenter Speicher
  - Datenaustausch
    - HTTP-Requests, Serialisierung nach JSON und XML
  - Open Lookup Infrastructure



overview.bit:

# Ein einfaches Beispiel:

## Übersichts-View eines Review-Applets

```

<view title="ReviewCentral.com"
  <list>
    reviews = {}
    for i = 1, #review_resources do
      reviews[i] = bit_idcp_decode(review_resources[i].data.value)
      bit.out('<
        reviews[
      bit.out(re
      bit.out('<
    end
  ?>
  <menu>
    <menu_item a
      Show detail
    </menu_item:
  </menu>
</list>
</view>

```



ma  
fu

```

local repository =
review_resources =
  {tag_id = tagged
bit.show_view("ove
end

```

```

= reviews[index]})">

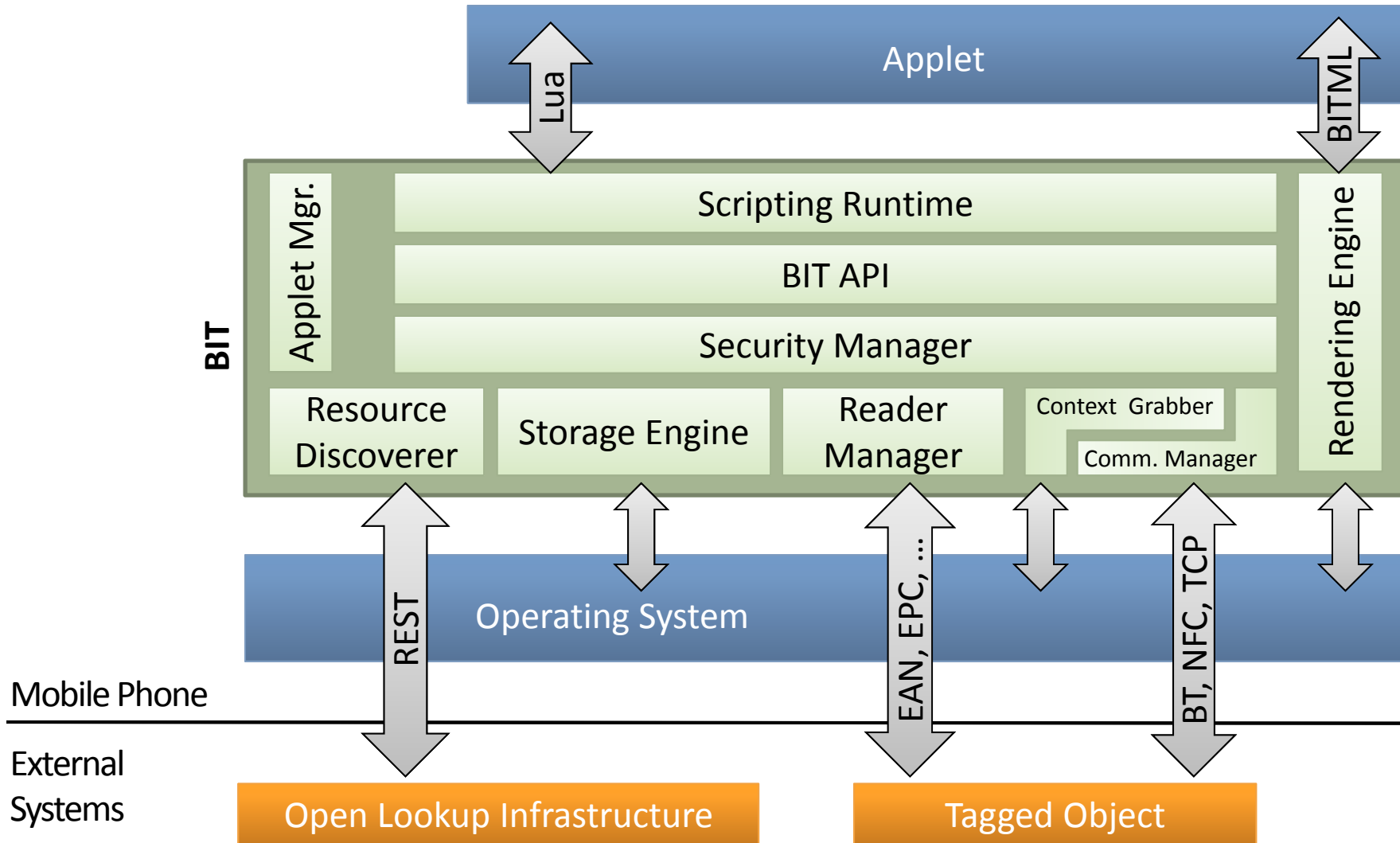
```

```

VIEW} )

```

# Architektur von BIT



# BIT-Implementation

- In Python auf Symbian S60 Plattform
- Lua-Interpreter portiert auf Symbian S60
- Modifiziertes Mobiltelefon Nokia E61i
  - Integrierter Gen2-RFID-Reader
- Barcodeerkennung mittels Batoo



# Praktischer Einsatz von BIT

## Implementation von neun Diensten:

Dienst	Lines of Code
Produktbewertungen	
Konsumentenschutz	
CO <sub>2</sub> -Rechner	
Allergieassistent	
Preisvergleich	102
Einkaufsliste	126
Suche über Open Lookup Infr.	53
Kaffeemaschinensteuerung	260
Self Checkout	158

Zum Vergleich J2ME-Implementation:

- 931 LOC
- Keine Bluetooth-Kommunikation
- Beide mit Standard-Layout

# Zusammenfassung

- Ähnlich einfach wie Web-Applikationen
- Abstraktion der Komplexität der Plattform (C++, Kommunikation etc.)
- Plattformunabhängig
- Unterstützung für breite Palette von Diensten

# Schluss

- Vermehrt Interaktion mit Gegenständen via Mobiltelefon
- Herausforderungen
  - Usability
  - Effiziente Umsetzung von Diensten
- Beiträge dieser Dissertation auf drei Ebenen:
  - **Nutzerstudie** über Vorteile und Grenzen bei der Bedienung von Geräten
  - **Infrastruktur** zum Veröffentlichen und Finden von Diensten
  - **Framework** zur Vereinfachung der Entwicklung von Diensten

# Vielen Dank fürs Zuhören!