

MobilisBuddy

Integration sozialer
Netzwerke in distanz-
abhängige Dienste auf
mobilen Endgeräten



Daniel Schuster, Thomas Springer, **Benjamin Söllner**, Dirk Hering,
Alexander Schill

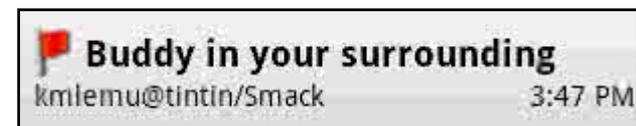
GeNeMe 2009

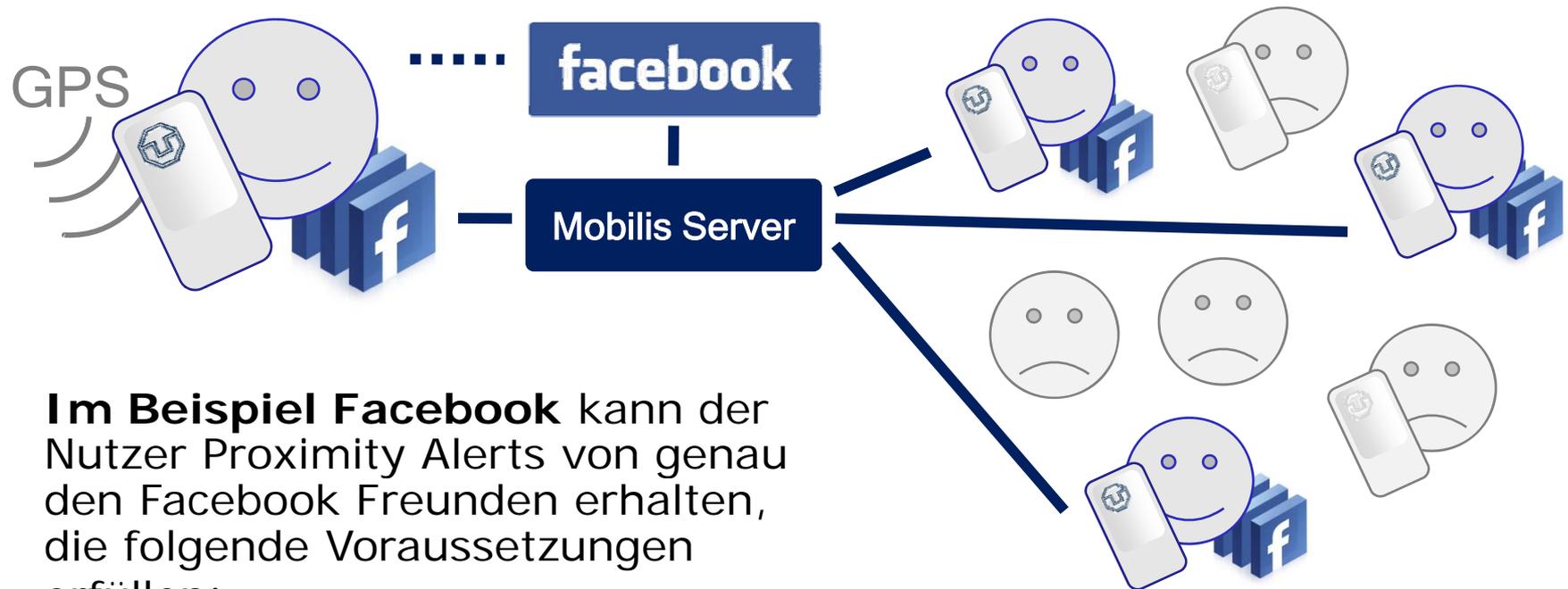


- Anwendungsfall
- Kommerzielle Relevanz
- Beiträge von MobilisBuddy
- Mobilis-Ansatz
 - Architektur
 - Integrationskonzept
 - Ortsinformationen und Proximity-Berechnung
- Validierung
 - Prototyp
 - Erfahrungen
- Zusammenfassung und Ausblick



- Ein spontanes, zufälliges Aufeinandertreffen zweier Menschen, die sich nahe stehen, am selben Ort.
- Wie kann ein mobiles verteiltes System dabei helfen, dass sich diese zwei Freunde nicht verpassen?





Im Beispiel Facebook kann der Nutzer Proximity Alerts von genau den Facebook Freunden erhalten, die folgende Voraussetzungen erfüllen:



Android-Handy
MobilisBuddy installiert und Facebook-Login definiert



MobilisBuddy Facebook-Applikation zum Facebook-Profil hinzugefügt



Google Latitude

Google Maps für Handys



Google Latitude Datenschutz

Standortfreigabe:

- Eigene Standort erkennen
Ihr Standort wird automatisch aktualisiert
- Eigene Standort festlegen
Einen Standort manuell auf der Karte wählen
- Eigene Position verbergen
Ihre Freunde können Ihre Position nicht sehen

Von Google Latitude abmelden



BuddyCloud

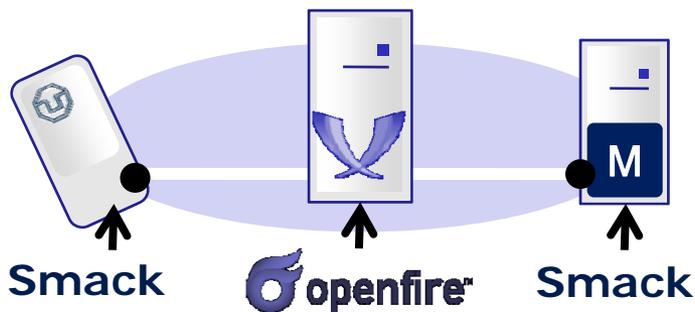
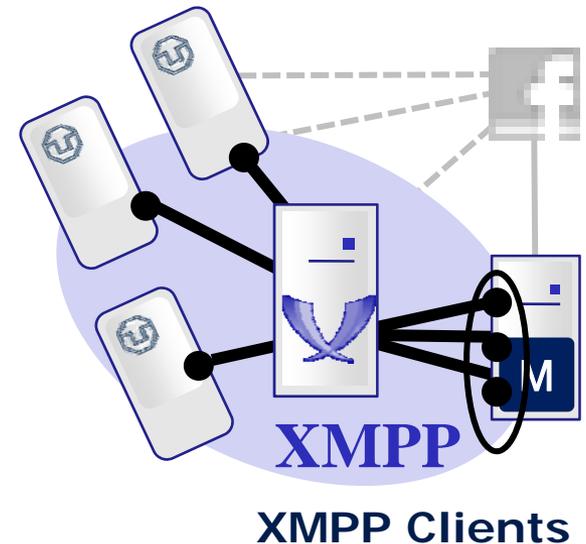




Was ist neu an MobilisBuddy?

- Social Networking Plattformen fördern die Vernetzung von Menschen über das Internet
- These: Vernetzung ist kein Selbstzweck sondern fördert Interaktionen
- Deshalb Nutzung der Funktionalität und Informationen bestehender Social Network Plattformen , um kollaborative Mehrwertdienste anzubieten
- Ziel ist eine offene Plattform für kollaborative Dienste in sozialen Netzwerken auf mobilen Geräten (Integration mit Web-basierten Plattformen)
- Explorative Studie zu kollaborativen Sozialen Netzwerken auf mobilen Geräte
- XMPP und Android werden hinsichtlich ihres Potentials für eine Umsetzung getestet

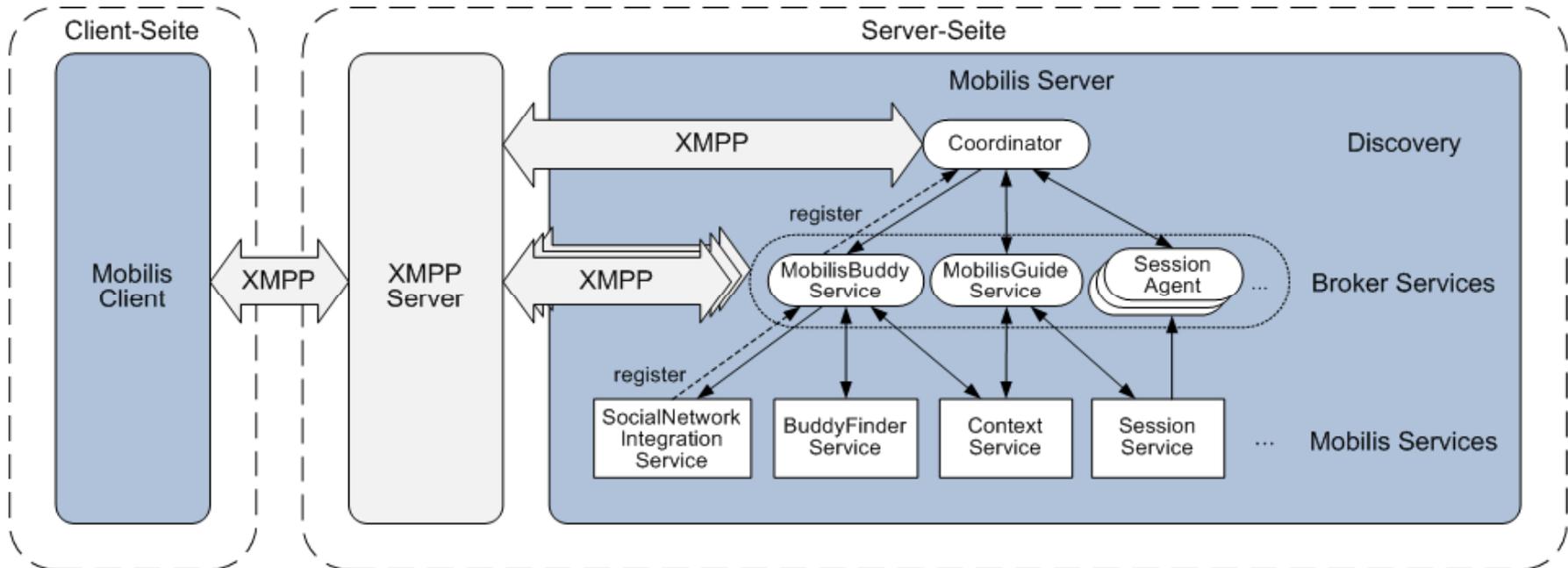
- Internetstandard für **XML-basierten Nachrichtenaustausch** (bidirektional)
- **XML**-basiert. Typen von Nachrichten: <message>, <presence>, <i q>
- Grundlage /Vorgänger: **Jabber**-Protokoll
- **XEP**: XMPP Extension Protocols
- Einsatz: **Instant Messanging**, Mehrbenutzerchats, Dateiübertragung, Presence, Location, Multi-User Chatroom, ...



Wir verwenden:

 **openfire**® als XMPP-Server

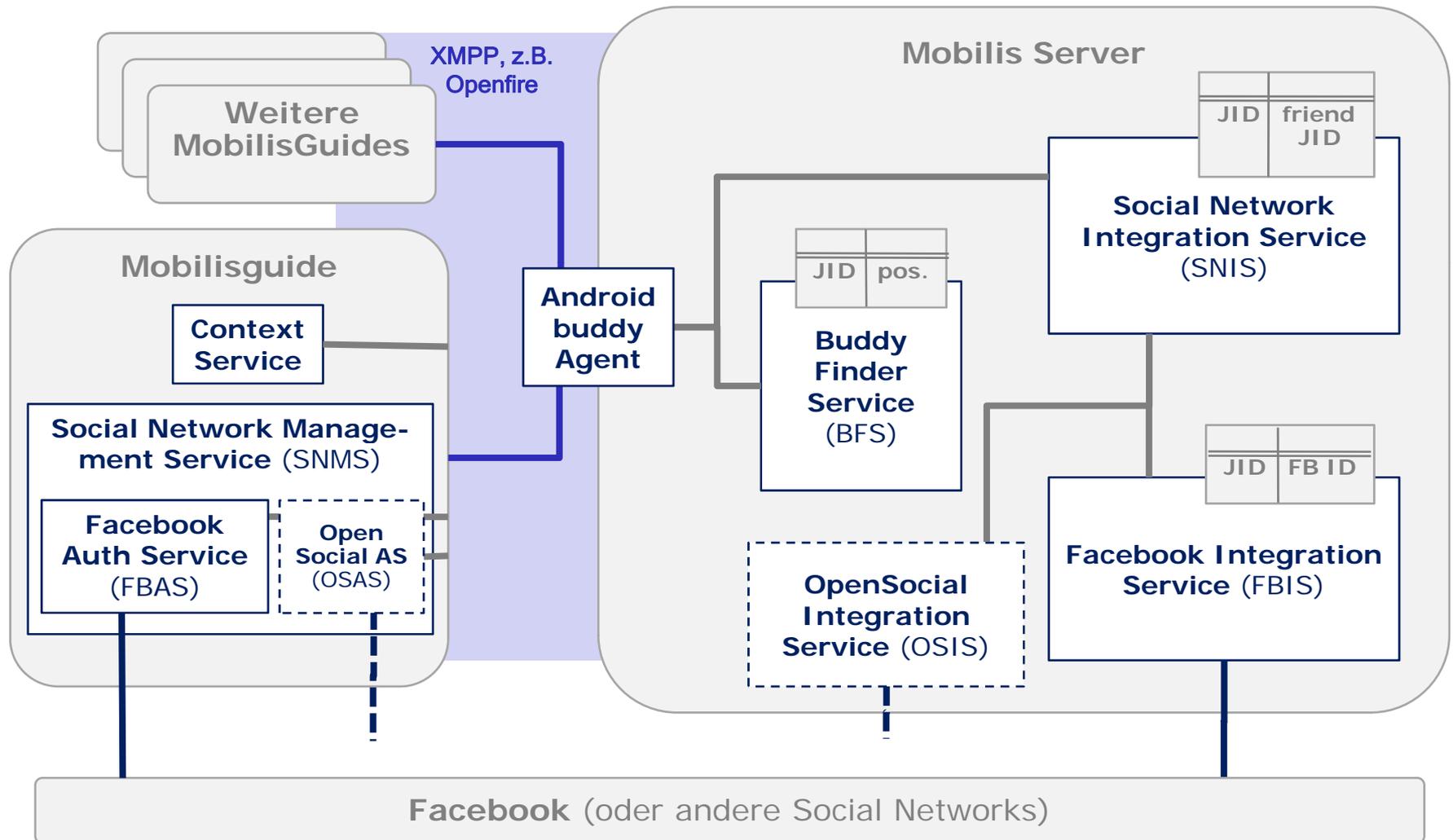
Smack als XMPP-Client API



- Kollaborative Anwendungen basierend auf XMPP
- XMPP-Server als Black Box (e.g., OpenFire)
- Coordinator and Broker Services sind ebenfalls XMPP clients
- Kollaborationssitzungen = Multi-User chat rooms

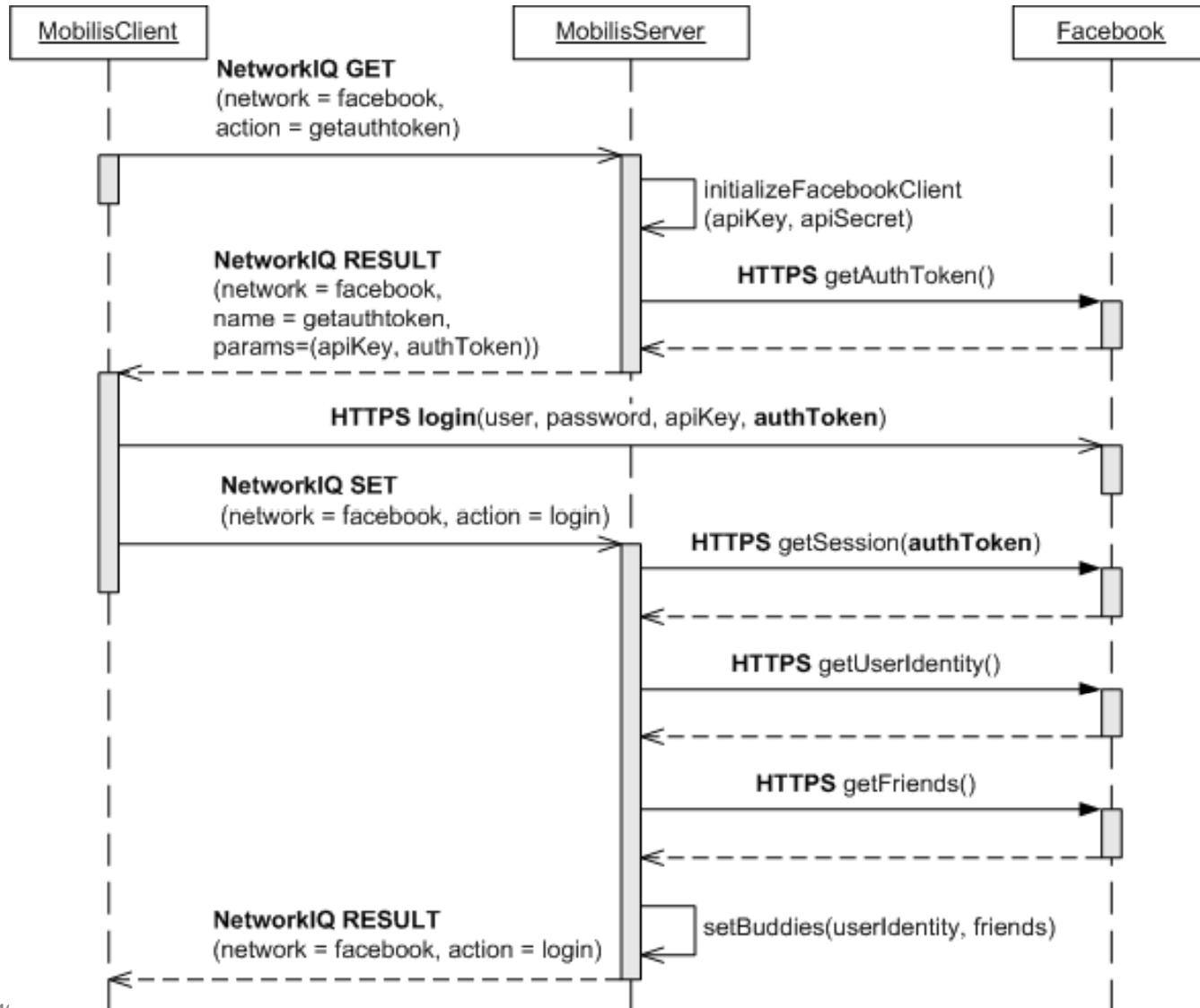


Integrationskonzept



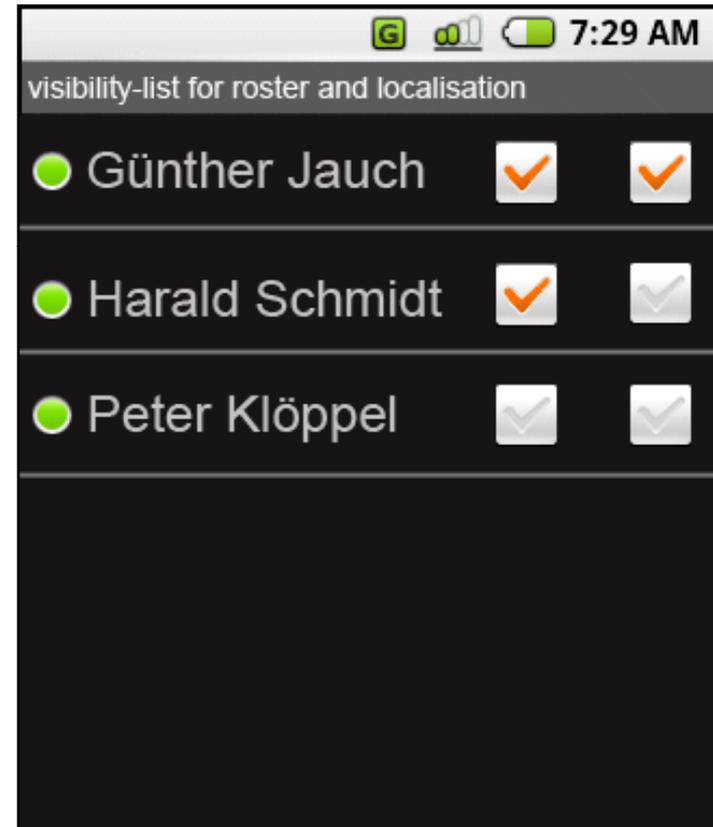


Integration von Facebook



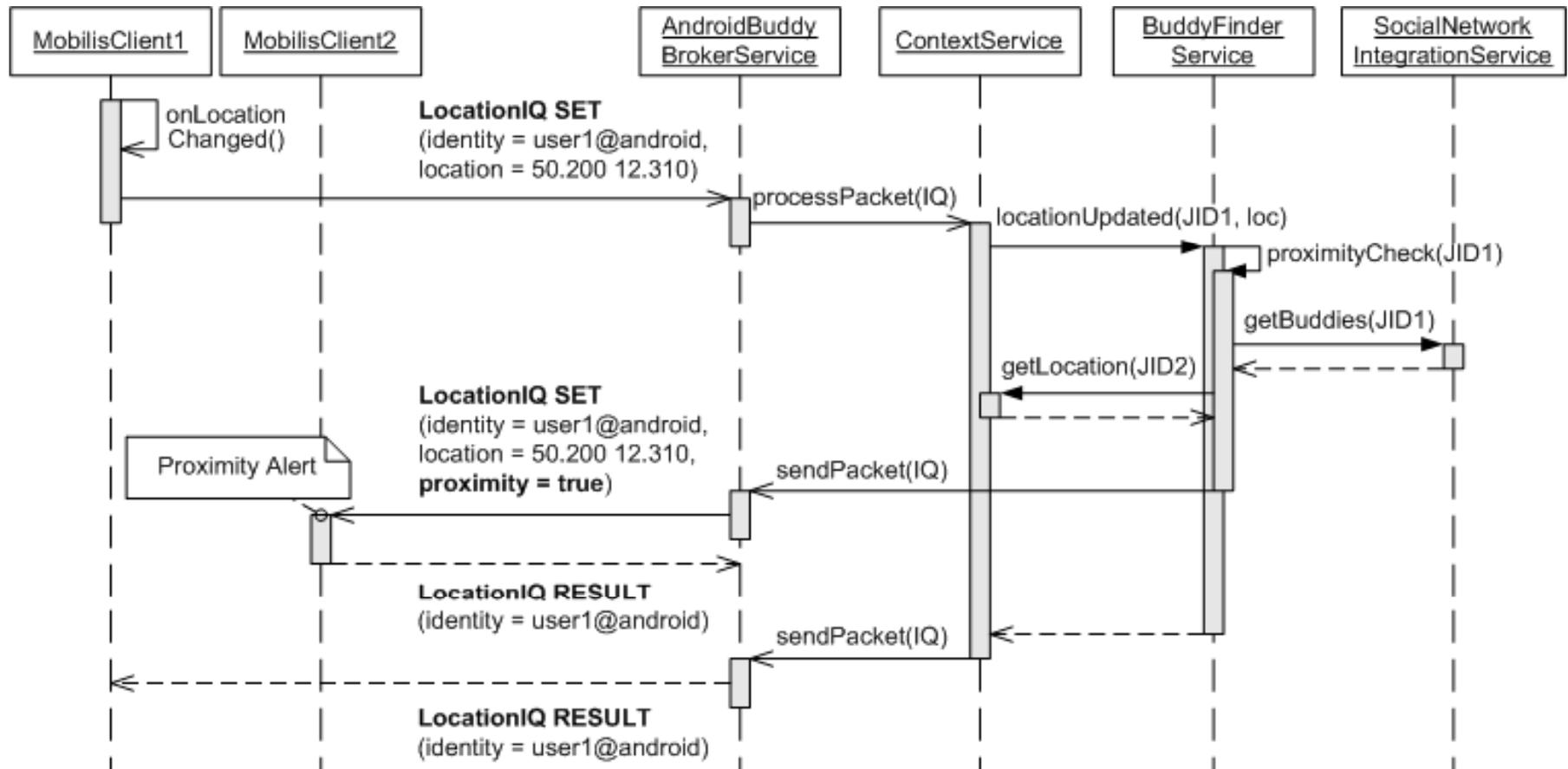


- Android SDK
 - API getLocation()
 - Transparent welcher „LocationProvider“ angesprochen wird (GPS, CellID)
- Benutzer bekommt nur Lokationen, wenn sich ein Buddy in einem bestimmten Radius von ihm befindet
- Location Updates konfigurierbar





Location-Updates und Proximity Checks





The screenshot displays an Android emulator window titled "Android Emulator (5554)". The main application, "AndroidBuddy", shows a map interface with a yellow route highlighted. The map includes street names like "Antonstraße", "Hansstraße", and "Theresienstraße". A status bar at the bottom of the emulator shows "klemu changed location".

Overlaid on the emulator is a "Mobilis Server" window showing a log of XMPP messages. The log contains the following entries:

```
(FINEST) BFS deliverLocationUpdate:
de.tudresden.inf.mobilis.xmpp.packet.LocationIQ@3c1
(FINER) BFS proximityCheck between: test@tintin/Android and
klemu@tintin/Smack
(FINE) distance by closeto(): 377.7851035795834
(FINEST) BFS deliverLocationUpdate:
de.tudresden.inf.mobilis.xmpp.packet.LocationIQ@3c1
(INFO) klemu@tintin/Smack updated location.
(FINE) Mapped: test@tintin = test@tintin/Android
(FINER) BFS proximityCheck between: klemu@tintin/Smack and
test@tintin/Android
(FINE) distance by closeto(): 391.77714599126364
(FINEST) BFS deliverLocationUpdate:
de.tudresden.inf.mobilis.xmpp.packet.LocationIQ@3c1
(FINER) BFS proximityCheck between: test@tintin/Android and
klemu@tintin/Smack
(FINE) distance by closeto(): 391.77714599126364
(FINEST) BFS deliverLocationUpdate:
de.tudresden.inf.mobilis.xmpp.packet.LocationIQ@3c1
(INFO) klemu@tintin/Smack updated location.
(FINE) Mapped: test@tintin = test@tintin/Android
(FINER) BFS proximityCheck between: klemu@tintin/Smack and
test@tintin/Android
(FINE) distance by closeto(): 405.76918713713536
(FINEST) BFS deliverLocationUpdate:
de.tudresden.inf.mobilis.xmpp.packet.LocationIQ@3c1
(FINER) BFS proximityCheck between: test@tintin/Android and
klemu@tintin/Smack
(FINE) distance by closeto(): 405.76918713713536
(FINEST) BFS deliverLocationUpdate:
de.tudresden.inf.mobilis.xmpp.packet.LocationIQ@3c1
(INFO) klemu@tintin/Smack updated location.
(FINE) Mapped: test@tintin = test@tintin/Android
(FINER) BFS proximityCheck between: klemu@tintin/Smack and
test@tintin/Android
(FINE) distance by closeto(): 419.76122714384337
(FINEST) BFS deliverLocationUpdate:
de.tudresden.inf.mobilis.xmpp.packet.LocationIQ@3c1
(FINER) BFS proximityCheck between: test@tintin/Android and
klemu@tintin/Smack
(FINE) distance by closeto(): 419.76122714384337
(FINEST) BFS deliverLocationUpdate:
de.tudresden.inf.mobilis.xmpp.packet.LocationIQ@3c1
(INFO) klemu@tintin/Smack updated location.
(INFO) klemu@tintin/Smack updated location.
```

To the right of the server log is a "KML-Routing-Simulator" window displaying a list of coordinates:

```
longitude: 13.7464 latitude: 51.06499
longitude: 13.7466 latitude: 51.06499
longitude: 13.7468 latitude: 51.06499
longitude: 13.747 latitude: 51.06499
longitude: 13.7472 latitude: 51.06499
longitude: 13.7474 latitude: 51.06499
longitude: 13.7476 latitude: 51.06499
longitude: 13.7478 latitude: 51.06499
longitude: 13.748 latitude: 51.06499
longitude: 13.7482 latitude: 51.06499
longitude: 13.7484 latitude: 51.06499
longitude: 13.7486 latitude: 51.06499
longitude: 13.7488 latitude: 51.06499
longitude: 13.749 latitude: 51.06499
longitude: 13.7492 latitude: 51.06499
longitude: 13.7494 latitude: 51.06499
longitude: 13.7496 latitude: 51.06499
longitude: 13.7498 latitude: 51.06499
longitude: 13.75 latitude: 51.06499
```

Below the list are two buttons: "open kml-file" and "simulate route".



- Black-Box Tests
 - separate Tests für Client- und Serverkomponenten
 - Funktionstests erfolgreich
 - Nutzerbewegungen mit KML simuliert
- Erfahrungen mit XMPP
 - Etabliert im Internet (Jabber, GoogleWave, GoogleTalk)
 - Zahlreiche Erweiterungen für kollaborative Anwendungen (Presence, Location, Chatroom)
 - Entwicklungsunterstützung mit Smack und Openfire
 - Eignung/Probleme in drahtlosen Infrastrukturen sind noch genauer zu untersuchen
- Erfahrungen mit Android
 - Plattform bietet einfache API zu grundlegenden Funktionen (GoogleMaps, Lokationsermittlung)
 - XMPP nicht direkt unterstützt, aber freie Bibliothek portierbar (im Gegensatz zu JavaME)
 - Innovatives Konzept zur Anwendungsentwicklung (Activities, Intents, Services, Content Provider)



- Social Networking Plattformen fördern die Vernetzung von Menschen über das Internet
- Vernetzung ist kein Selbstzweck sondern fördert Interaktionen
 - In der realen Welt (MobilisBuddy, OpenBC, LinkedIn, ...)
 - In der virtuellen Welt (Kommunikation, Austausch, Kollaboration)
- MobilisBuddy
 - Integration existierender Sozialer Netzwerke
 - Soziale Netzwerke als Grundlage kollaborativer Dienste
 - Experimente mit Android und XMPP
 - Plattform für Pervasive Collaboration
- Ausblick
 - weitere Anwendungen (MobilisTrader, MobilisCarShare, MobilisTraveler) und Basisdienste
 - Evaluation der Protokolle in heterogenen Netzwerkkumgebungen



- **deutsch-brasilianisches** Forschungsprojekt der TU Dresden
- Forschungsvorhaben: Entwicklung eines **Frameworks** für **kollaborative** Anwendung auf **mobilen** Endgeräten
- Mittel der Wahl: **Android, XMPP**
- **Anwendungen** selbst werden von Studenten, z.B. in Komplexpraktika entwickelt





Vielen Dank für die Aufmerksamkeit !

Fragen?