

Ein kontextbezogener Instant-Messaging-Dienst auf Basis des XMPP-Protokolls

**F. Dürr, J. Palauro, L. Geiger, R. Lange,
K. Rothermel**

Institut für Parallele und Verteilte Systeme
Universitätsstraße 38
70569 Stuttgart
GERMANY

E-Mail: {nachname}@ipvs.uni-stuttgart.de

Abstract

Dieser Beitrag beschreibt die Verteilung kontextbezogener Informationen mittels eines erweiterten Instant-Messaging-Dienstes. Dieser Dienst ermöglicht das Senden von Nachrichten an alle Teilnehmer, die einen bestimmten Kontext besitzen und sich z.B. an einem bestimmten Ort aufhalten oder bestimmte Interessen besitzen. Als Basis dient das Extensible Messaging and Presence Protokoll (XMPP) sowie ein XMPP-basierter Instant-Messaging-Dienst. Wir beschreiben Protokoll- und Architekturweiterungen für die Integration von Kontextinformationen in das XMPP-Protokoll und die Server-Infrastruktur. Der erweiterte Dienst unterstützt insbesondere die Server-seitige Filterung von Nachrichten aufgrund von Kontextinformationen und ermöglicht dadurch die effiziente Nachrichtenverteilung.

1. Einleitung

In diesem Artikel wird die Verteilung kontextbezogener Informationen mit Hilfe eines neuartigen kontextbezogenen Instant-Messaging-Dienstes betrachtet. Ziel dieses Dienstes ist die Implementierung des Contextcast-Kommunikationsparadigmas. Contextcast ermöglicht die Zustellung von Nachrichten an alle Nutzer, die einen bestimmten Kontext besitzen. Das heißt, im Gegensatz zu aktuellen Instant-Messaging-Diensten, bei denen die Empfänger einer Nachricht explizit durch ihren Benutzer-ID adressiert werden, soll es dieser kontextbezogene Instant-Messaging-Dienst ermöglichen, Benutzer implizit aufgrund ihres Kontexts zu adressieren. Der Sender muss dabei die Identität der Empfänger nicht kennen – im Unterschied zu herkömmlichen Instant-Messaging-Diensten, bei denen die Empfänger den Sendern in Form so genannter Buddy-Listen bekannt sind.

Der Kontext, der zur Adressierung eines Benutzers verwendet werden kann, ist dabei vielfältiger Natur. Eine wichtige Kontextinformation ist beispielsweise die aktuelle Position des Benutzers, an der er sich aufhält. Durch Berücksichtigung dieser Information können Nachrichten an alle Personen zugestellt werden, die sich an einem bestimmten Ort aufhalten. Die reine ortsbezogene Adressierung als eine Unterart der allgemeinen kontextbezogenen Adressierung wird dabei auch als Geocast bezeichnet. Neben dem Ort sind aber eine Vielzahl weiterer Benutzerkontexte für die Adressierung nutzbar, beispielsweise das verwendete Fortbewegungsmittel (Autofahrer, Fußgänger, usw.), sein Alter, sein Geschlecht, seine Interessen oder allgemein die Situation, in der sich der Benutzer gerade befindet.

Die Anwendungsmöglichkeiten von Contextcast sind dabei so vielfältig wie die Kontextinformationen, die zur Adressierung verwendet werden können. Typische Beispiele umfassen die Verteilung von Warnmeldungen an alle Gefährdeten in einem bestimmten geographischen Gebiet, oder die Zustellung von Informationen zu Sehenswürdigkeiten an alle Touristen mit bestimmten Interessen. Denkbar ist auch die selektive Verteilung von Veranstaltungsinformationen oder von Produktinformationen.

Die Motivation für die Verwendung von Instant-Messaging-Diensten für die Implementierung der Contextcast-Kommunikation ergibt sich aus zwei aktuellen Trends. Zum einen erfreuen sich entsprechende Dienste sehr hoher Beliebtheit. Sie sind somit praktisch universell verfügbar und bieten bekannte und akzeptierte Benutzungsschnittstellen. Zum anderen ermöglichen aktuelle Technologien zur mobilen Kommunikation die Möglichkeit, Instant-Messaging-Dienste auch auf mobilen Endgeräten zur Verfügung zu stellen. Entsprechende Kommunikationstechnologien wie drahtlose lokale Netze (WLAN) oder UMTS ermöglichen hohe Übertragungsraten und werden durch Flatrate-Tarife immer attraktiver.

Neben diesen Technologietrends wird die Verwendung von Instant-Messaging für die Implementierung von Contextcast aber auch durch die technischen Eigenschaften der Instant-Messaging-Protokolle motiviert. Wie später noch im Detail ausgeführt wird, ermöglichen diese Protokolle die für Contextcast typische Push-Kommunikation. Die Klienten müssen also nicht wie bei diversen anderen Protokollen wie RSS über HTTP ständig aktiv Änderungen abfragen (engl. polling), sondern sie bleiben passiv und werden bei Verfügbarkeit von entsprechenden Informationen durch die Instant-Messaging-Infrastruktur informiert. Ferner steht eine Infrastruktur von Instant-Messaging-Servern zur Verfügung, die Informationen aufgrund der Nutzerkontexts filtern können bevor sie an den Klienten zugestellt wird, im Gegensatz beispielsweise zu RSS wo ein substantieller Anteil der Filterung beim Endgerät erfolgt. Die infrastrukturbasierte Filterung entlastet somit die Endgeräte und die (drahtlose) Kommunikationsschnittstelle.

Der Rest dieses Beitrages ist wie folgt strukturiert. In Abschnitt 2 gehen wir auf die Erweiterung des XMPP-Instant-Messaging-Protokolls um Contextcast-Mechanismen ein. Abschnitt 3 stellt die Arbeit in den Kontext verwandter Arbeiten, bevor dieser Beitrag in Abschnitt 4 mit einer Zusammenfassung und einem Ausblick auf zukünftige Arbeiten abgeschlossen wird.

2. Erweiterung des XMPP-Protokolls und der XMPP-Infrastruktur um kontextbezogene Adressierungskonzepte

In diesem Abschnitt beschreiben wir die Erweiterungen des Extensible Messaging and Presence Protocols (XMPP) um kontextbezogene Adressierungsmechanismen. XMPP ist ein offenes Standardprotokoll für Instant-Messaging. Die Offenheit und das XMPP zugrunde liegende XML-basierte Nachrichtenformat ermöglichen eine einfache Erweiterung dieses Protokolls. Ferner wird dieses Protokoll von vielen Diensteanbietern unterstützt und es existieren diverse offene XMPP-Klienten und Server, die als Grundlage für die Erweiterung um kontextbezogene Konzepte dienen können. In dieser Arbeit verwenden wir den XMPP-Klienten Spark [1] und den XMPP-Server Openfire [2] als Grundlage für unsere Erweiterungen.

Zunächst geben wir in diesem Abschnitt einen Überblick über die erweiterte Funktionalität der kontextbezogenen Nachrichtenzustellung und das zugrunde liegende Konzept zur kontextbezogenen Adressierung von Nachrichten und zur Definition von Benutzerkontexten. Gefolgt wird dieser Abschnitt durch die Beschreibung der erweiterten verteilten Systemarchitektur der XMPP-Server-Infrastruktur und der lokalen Erweiterungen des XMPP-Servers Openfire.

2.1. Erweiterte Funktionalität und kontextbezogenes Adressierungskonzept

Die Möglichkeit zur kontextbezogenen Adressierung von Teilnehmern soll für die Implementierung der folgenden Funktionen genutzt werden:

Kontextbezogene Nachrichten: Hierbei handelt es sich um einzelne Nachrichten, die an Teilnehmer zugestellt werden, die einen bestimmten Kontext besitzen. Der Sender kennt hierbei die Identitäten der Empfänger nicht. Nachrichten besitzen dabei eine bestimmte Lebenszeit. Besitzt eine Teilnehmer zum Sendezeitpunkt der Nachricht den adressierten Kontext, oder ändert er während der Lebenszeit der Nachricht seinen Kontext, so dass er dem adressierten Kontext entspricht, so wird die Nachricht an ihn zugestellt.

Kontextbezogener Chat: Im Gegensatz zu einzelnen kontextbezogenen Nachrichten, die nur vom Sender zum Empfänger vermittelt werden, besteht beim kontextbezogenen Chat die Möglichkeit, Chat-Sitzungen mit Benutzern zu initiieren, die einen bestimmten Kontext besitzen. Die Teilnehmer, welche den durch die Chat-Einladung des Initiators adressierten Kontext besitzen, können dem Chat beitreten und bidirektional Nachrichten mit dem Initiator austauschen.

Kontextbezogene Newsletter: Beim kontextbezogenen Newsletter abonnieren Teilnehmer aufgrund ihrer Interessen bestimmte existierende Themengebiete, beispielsweise Nachrichten zum Thema Rock-Musik. Nachrichten werden aufgrund der Abonnements und weiterer Kontextinformationen an die Teilnehmer zugestellt. So kann ein Sender z.B. eine Nachricht über ein Rock-Konzert an alle Rock-Musik-Newsletter-Abonnenten versenden, die sich gerade im Umkreis des Veranstaltungsorts aufhalten.

Zur Adressierung von kontextbezogenen Nachrichten, Chat-Einladungen und Newslettern sowie zur Definition der Teilnehmerkontexte können verschiedene Kontextattribute eingesetzt werden. Grundsätzlich handelt es sich dabei um Schlüssel-Wert-Paare, die vom Empfänger und dem Sender frei definiert werden können. So definieren beispielsweise die Paare „alter = 30“ und „position = /de/stuttgart“ das Alter und den aktuellen Aufenthaltsort einer Person. Hierdurch können beliebige Kontextinformationen spezifiziert werden, so dass jeder Anwendungsentwickler den Kontext wählen kann, der für die entsprechende Anwendung relevant ist. Andererseits hat eine völlig freie Definition den Nachteil, dass die Kompatibilität verschiedener Anwendungen schwer zu garantieren ist. Wir gehen daher davon aus, dass die verwendbaren Kontextinformationen in entsprechenden Schemata definiert sind [3]. Allgemeine, anwendungsübergreifende Kontextattribute werden von einem Standardschema festgelegt, das jede Anwendung interpretieren kann; anwendungsspezifische Erweiterungen werden in erweiterten Schemata definiert, die u.U. nur von bestimmten Anwendungen verstanden werden.

Insbesondere Ortsinformationen sind für mobile Anwendungen von großer Bedeutung. Aufgrund der für den Benutzer intuitiven Verständlichkeit werden Ortsinformationen in Form hierarchischer symbolischer Lokationen spezifiziert. So steht beispielsweise die symbolische Lokation /de/stuttgart für die Stadt Stuttgart in Deutschland. Wie in Abschnitt 2.2 beschrieben wird, werden intern auch geometrische Koordinaten verwendet, die über ein entsprechendes hybrides Lokationsmodell [4] in symbolische Lokationen übersetzt werden.

2.2. Erweiterte XMPP-Systemarchitektur

Das Instant-Messaging-System besteht im Wesentlichen aus drei Hauptkomponenten (siehe Abbildung 1).

Der *Instant-Messaging-Klient* stellt die Benutzungsschnittstelle dar und wird für das Senden und Empfangen von Nachrichten verwendet. Wir verwenden als Grundlage den XMPP-Klienten Spark [1], der durch entsprechende Komponenten erweitert wurde (diese Komponenten sind in der Abbildung durch „Conny“ – dem Namen des Projekts, das diese Erweiterungen entworfen hat – bezeichnet). Jeder mobile Klient ist mit einem oder mehreren Positionierungssystemen ausgestattet. Integriert wurden das satellitengestützte GPS-System und das WLAN-basierte Positionierungssystem der Firma Ekahau. Beide Systeme geben Positionen in Form von geometrischen Koordinaten aus (Längen-Breitengrad bzw. Koordinaten in einem kartesischen Referenzsystem).

Jeder Klient ist mit dem *Instant-Messaging-Server* seines Anbieters verbunden. Die Server haben die Aufgabe, Nachrichten vom Sender zu den Empfängern der Nachricht weiterzuleiten. Entsprechend dem XMPP-Protokoll übergibt der Klient des Senders hierzu die Nachricht an seinen lokalen Server. Dieser Server leitet die Nach-

richt an alle Server weiter, bei denen Empfänger mit entsprechenden Kontexten registriert sind, die wiederum die Nachricht an ihre lokalen Klienten mit dem adressierten Kontext zustellen. Wir setzen als Grundlage für unsere Implementierung den XMPP-Server Openfire ein [2].

Die Server besitzen Verbindungen zu einem Dienst, der das Kontextmodell verwaltet. Insbesondere verwaltet dieser Dienst das oben angesprochene hybride Lokationsmodell. Die Instant-Messaging-Komponenten nutzen dieses Modell zur Ermittlung der symbolischen Lokationshierarchie, die dann dem Benutzer im Klienten angezeigt wird, sowie zur Umrechnung geometrischer Koordinaten in symbolische Lokationen. Als *Umgebungsmodelldienst* setzen wir auf der Nexus-Infrastruktur auf [5]. Diese Infrastruktur ermöglicht die verteilte und skalierbare Verwaltung von Umgebungsmodellaten. Einzelne Teilmodelle von verschiedenen Anbietern werden jeweils durch so genannten Spatial-Model-Server bereitgestellt, wobei es sich hierbei um ein offenes System handelt, bei dem jeder Anbieter seine Daten in das Gesamtsystem integrieren kann. Somit wird es möglich, globale Umgebungsmodelle bereitzustellen. Durch die Föderation der Teilmodelle werden die Teilmodelle den Instant-Messaging-Servern als ein durchgängiges Modell präsentiert.

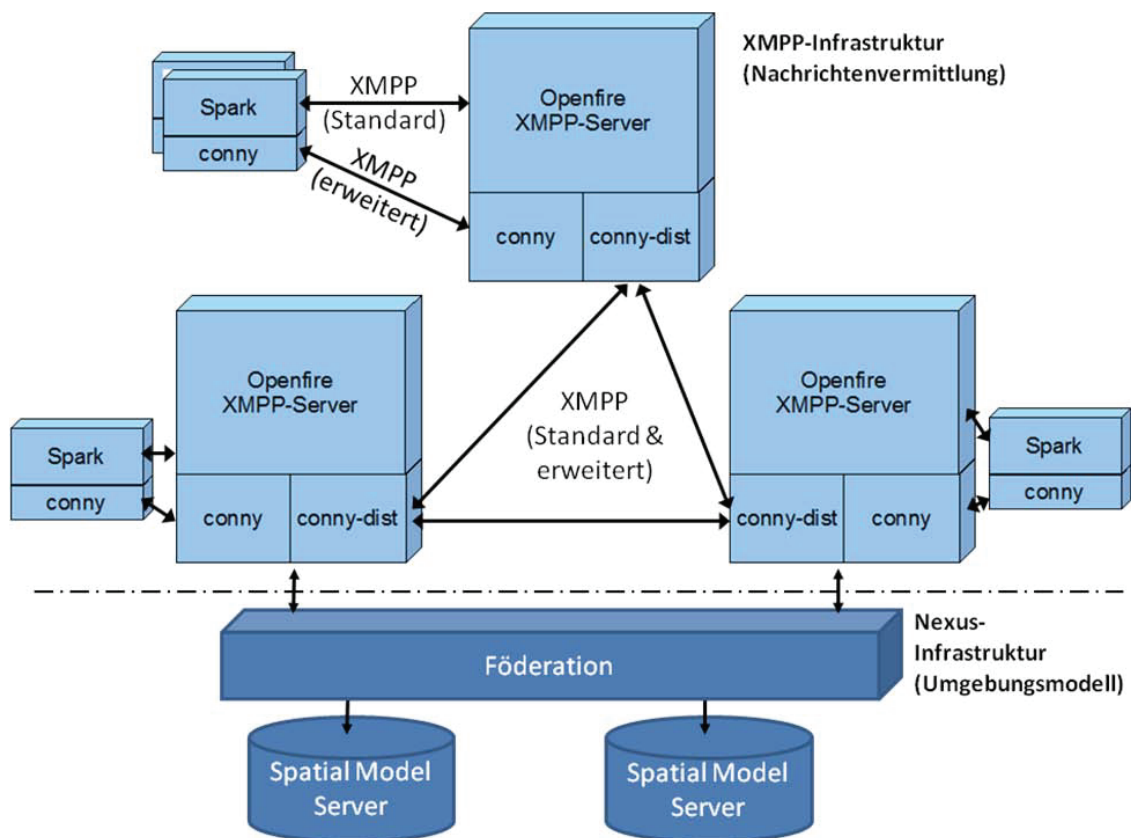


Abbildung 1: Systemarchitektur

2.3. Kontextbezogene Nachrichtenvermittlung

Die kontextbezogene Nachrichtenvermittlung erweitert die Standard-XMPP-Kommunikation. Das heißt, das System unterstützt neben der Weiterleitung kontextbezogener Nachrichten auch weiterhin herkömmliches Instant-Messaging. Die Erweiterungen gliedern sich in zwei Teile: erstens die erweiterte Klient/Server-Kommunikation, zweitens die Server/Server-Kommunikation.

Die Klient/Server-Kommunikation wurde um verschiedene Nachrichtentypen erweitert. So wurden zum einen Nachrichten eingeführt, um den Kontext eines Klienten, d.h. seine Kontextattribute einschließlich Ort, sowie seine Abonnements von kontextbezogenen Newslettern, an den Server zu übermitteln. Der Server verwaltet diese Informationen und verwendet sie für die Server-seitige Nachrichtenfilterung. Das heißt, wenn ein Server eine kontextbezogene Nachricht erhält, dann stellt er diese nur an diejenigen Klienten zu, deren Kontext dem adressierten Kontext entspricht. Für die verschiedenen Funktionen (kontextbezogene Nachricht, Newsletter und Chat-Einladung) wurden entsprechende XMPP-Nachrichtenformate definiert. Beispielhaft soll hier eine einfache kontextbezogene Nachricht dargestellt werden:

```

<message id="B97rz-72" to="conny.bar.com" from="foo@bar.com">
  <connymessage type="message" sender="foo@bar.com"
  creationdate="1206455896030" deliverdate="1206455896030"
  lifetime="2100000">
    <context>
      <location>/europa/deutschland/bw/stuttgart</location>
      <profile>
        <value key="Studiengang" value="Softwaretechnik"/>
      </profile>
    </context>
    <content>
      <subject>A Test Message</subject>
      <text>A simple test message.</text>
    </content>
  </connymessage>
</message>

```

Diese Nachricht definiert zum einen die bereits angesprochene Zeitspanne der Nachrichtauslieferung durch die Attribute creationdate und lifetime. Ferner wird hier ein einfacher Kontext adressiert, bestehend aus der Position der Empfänger (location) und einem weiteren Kontextattribut (Studiengang). Die XMPP-Nachricht ist an die Komponente conny des XMPP-Servers mit der Adresse bar.com adressiert. Ein entsprechend erweiterter XMPP-Server wird die Nachricht anstatt sie direkt an einen bestimmten Benutzer weiterzuleiten an diese Komponente übergeben, welche die kontextbezogene Nachrichtenfilterung und –weiterleitung implementiert.

Die Server/Server-Kommunikation wird primär für die Vermittlung von Nachrichten an die Server entfernter Empfänger eingesetzt, die nicht beim Server des Senders registriert sind. In der aktuellen Implementierung bilden die Server hierzu ein unstrukturiertes Netz. Nachrichten werden in diesem Netz durch Fluten (engl. flooding) weitergeleitet, d.h. der Server sendet eine weiterzuleitende Nachricht einfach an alle mit ihm verbundenen Server. Es findet also aktuell keine Filterung bei der Weiterleitung im Netz statt, sondern nur eine Server-seitige Filterung am „Rande“ des Netzes. Obwohl diese Server-seitige Filterung die Klient/Server-Verbindung entlastet, so wird doch das Netz u.U. stark belastet. Dieses einfache Verfahren erfüllt aber eine wichtige Anforderung an das System. Die Kontextinformationen der Teilnehmer werden nicht an andere fremde Server weitergeleitet, sondern verbleiben auf den als vertrauenswürdig angenommenen Servern des jeweiligen Anbieters des Teilnehmers, was einen wichtigen Vorteil bezüglich der Privatheit von Teilnehmerkontexten darstellt.

3. Verwandte Arbeiten

Heutige Web-Standards konzentrieren sich auf den Zugriff auf ortsbezogene Informationen. Hervorzuheben ist hier insbesondere der GeoRSS-Standard [6], mit dem es möglich ist, Informationen mit Orten zu verknüpfen. Obwohl RSS grundsätzlich die Idee einer Push-Kommunikation verfolgt – der Benutzer wird informiert, sobald sich relevante Änderungen ergeben – so ist die technische Umsetzung der Kommunikation über HTTP oft Pull-basiert realisiert. Wir argumentieren daher, dass ein Push-basiertes Protokoll wie XMPP für die Verbreitung orts- und kontextbezogener Informationen deutlich besser geeignet ist.

Die orts- und kontextbezogene Vermittlung von Informationen mittels spezieller Overlay-Netze ist derzeit aktiver Forschungsgegenstand [7, 8, 9]. Diese Arbeit kann von diesen Ansätzen profitieren, indem entsprechende Vermittlungsprotokolle im Netz aus XMPP-Servern implementiert werden. Hierdurch könnte die präsentierte Server-seitige Filterung am Rande des Netzes zu einer netzseitigen Filterung erweitert werden.

4. Zusammenfassung und zukünftige Arbeiten

In diesem Beitrag wurde Erweiterungen eines Instant-Messaging-Dienstes zur kontextbezogenen Vermittlung von Nachrichten vorgeschlagen. Durch diese Erweiterungen ist es möglich, Nachrichten an Benutzer zu senden, die sich in einem bestimmten Kontext, z.B. an einem bestimmten Ort, befinden, mit diesen Chat-Sitzungen aufzubauen oder kontextbezogene Newsletter zu abonnieren. Ermöglicht wurde diese kontextbezogene Kommunikation durch Erweiterung des XMPP-Protokolls, die in die XMPP-Klienten- und Server-Software Spark und Openfire integriert wurden. Durch die für XMPP typische Push-basierte Kommunikation und die Server-seitige Filterung in der XMPP-Infrastruktur konnte eine effiziente Nachrichtenverteilung implementiert werden, die insbesondere die Klienten und deren meist drahtlosen Kommunikationsverbindungen entlastet. Gleichzeitig stellt das Vermittlungsverfahren sicher, dass teilnehmerspezifische private Kontextinformationen

beim vertrauenswürdigen Server des Teilnehmers verbleiben und nicht an andere Server und Teilnehmer weitergeleitet werden.

In zukünftigen Arbeiten soll vor allem die Effizienz der Nachrichtenverteilung weiter gesteigert werden. Hierzu sind spezielle kontextbasierte Vermittlungsalgorithmen in das XMPP-Server-Netz zu integrieren, wodurch sich eine Nachrichtenfilterung im Netz anstatt an eine Server-seitige Filterung an den Rändern des Netzes ergibt. Hierzu können beispielsweise existierende Geocast-Protokolle integriert werden. Da diese Protokolle allerdings nur eine effiziente Vermittlung auf reinen Ortsinformationen ermöglichen, die nur einen Teil der relevanten Empfängerkontexte darstellen, ist es unser Ziel, spezielle Contextcast-Protokolle zu entwerfen und zu integrieren, um eine feingranulare Filterung aufgrund allgemeiner Kontextinformationen zu ermöglichen. Eine wesentliche Herausforderung besteht dabei darin, einerseits die Privatheit der Kontextdaten sicherzustellen und teilnehmerspezifische Kontextdaten nicht beliebig im Netz zu verteilen, andererseits aber auch mit den weitergegebenen Informationen eine effiziente Nachrichtenfilterung zu ermöglichen.

Denkbar sind auch Erweiterungen der zur Adressierung eingesetzten Kontextinformationen beispielsweise um zeitbezogene Kontexte, um hiermit gezielt Teilnehmer anzusprechen, die sich z.B. zu einem bestimmten Zeitpunkt an einem bestimmten Ort aufgehalten haben.

Danksagungen

Unser Dank gilt vor allem den Teilnehmern des Studienprojekts Advanced Instant Messaging Service – Sören Brunk, Hannes Mühleisen, Jonas Palauro, Andreas Poszlovszki, Michael Schäfer, Johannes Schneider, Katharina Wiesner und Alexander Wobser –, die das vorgestellte System umgesetzt haben.

Diese Arbeit wurde teilweise durch die Deutsche Forschungsgemeinschaft (DFG) innerhalb des Sonderforschungsbereichs 627 unterstützt.

Referenzen

- [1] <http://www.igniterealtime.org/projects/spark/index.jsp>
- [2] <http://www.igniterealtime.org/projects/openfire/index.jsp>
- [3] M. Bauer, F. Dürr, J. Geiger, M. Grossmann, N. Hönle, J. Joswig, D. Nicklas, T. Schwarz: „Information Management and Exchange in the Nexus Platform“, Technischer Bericht Nr. 2004/04.
- [4] F. Dürr, K. Rothermel: „On a location model for fine-grained geocast“. UbiComp 2003: 5th International conference on ubiquitous computing, 2003.
- [5] D. Nicklas, M. Großmann, T. Schwarz, B. Mitschang: „A Model-Based, Open Architecture for Mobile, Spatially Aware Applications“. Proceedings of the 7th International Symposium on Spatial and Temporal Databases, 2001.
- [6] C. Reed, R. Singh, R. Lake, J. Lieberman, M. Maron: „An introduction to georss: A standards based approach for geo-enabling rss feeds“. Open Geospatial Consortium Inc., White Paper OGC 06-050r3, 2006. <http://www.opengeospatial.org/pt/06-050r3>
- [7] F. Dürr, C. Becker, K. Rothermel: „An Overlay Network for Forwarding Symbolically Addressed Geocast Messages“. Proceedings of the 15th International Conference on Computer Communications and Networks (ICCCN '06), 2006.
- [8] L. Geiger, F. Dürr: „Kontextbezogene Kommunikation“. 4. GI/ITG KuVS Fachgespräch Ortsbezogene Anwendungen und Dienste, 2007.
- [9] G. Cugola, J. E. M. de Cote: „On Introducing Location Awareness in Publish-Subscribe Middleware“. ICDCSW '05: Proceedings of the 4th International Workshop on Distributed Event-Based Systems (DEBS) (ICDCSW'05), 2005.