

XMPP - Jabber

Noch ein IM



Agenda

- Motivation
 - Das Kaffee Problem
 - Andere ...
 - AGBs
- Was ist Jabber?
- Warum ist Jabber „cool“?
- Beispiel Architektur
- Features

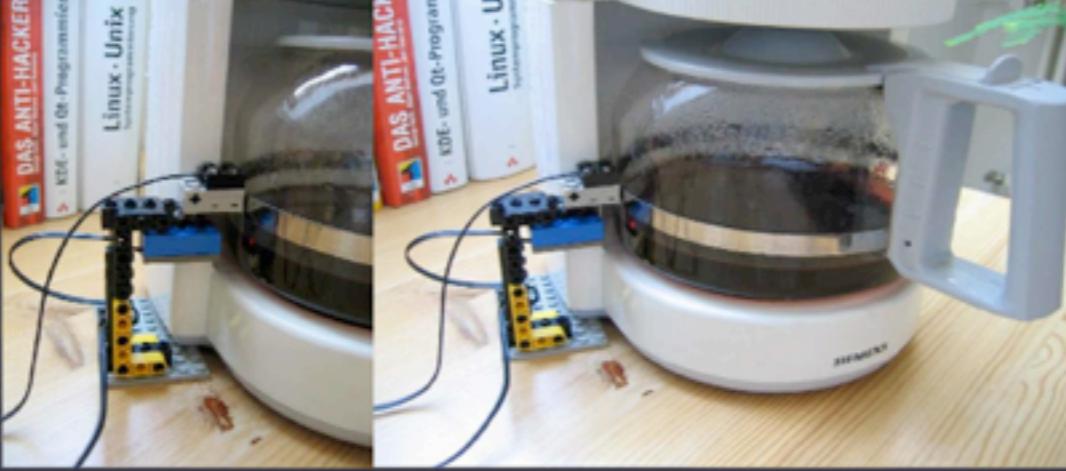
Motivation

- Instant Messaging
 - Nachrichten in Echtzeit austauschen
 - Video-, Audiokonferenzen
 - Multi-User Chat
 - Kollaboratives Arbeiten (Whiteboard)
- Schneller, direkter, „persönlicher“ als Email
- Spam, Fishing, ...

Das Kaffee Problem

- Kaffee ist das Benzin der Entwickler
- Problem:
 - Weg zur Kaffee Maschine zu weit
 - Kaffee meistens leer
 - Niemand setzt eine neue Kanne auf
 - Kaffee ist nie frisch gebrüht
- Lösung?

Kaffee Überwachung



The image shows a LEGO Mindstorms setup for coffee monitoring. A Siemens coffee maker is connected to a LEGO IR Tower and a Mindstorms robot. The setup is placed on a wooden desk next to several books about Linux and programming.

Three screenshots of the Psi instant messaging client are shown, illustrating the status changes:

- Coffeepot is absent:** The coffee pot is empty, so the status message is "coffeepot is absent".
- Coffee is available:** The coffee pot has been filled, so the status message is "coffee is available!".
- Coffeepot is empty:** The coffee pot has been drained again, so the status message is "coffeepot is empty".

- 100g LEGO-Bausteine
- 1 LEGO Mindstorms Rick
- 1 LEGO Mindstorms IR Tower
- 20 Zeilen Perl Code

Andere

- Jabber
- Skype
- ICQ
- MSN
- Google Talk
- AIM
- Yahoo!



AGBs - MSN

”[...] Wir sind jedoch berechtigt, Ihre Mitteilungen zu überwachen und Informationen über Sie, einschließlich des Inhalts Ihrer Mitteilungen, offen zu legen, wenn wir dies als notwendig erachten, um:

(1) das Gesetz einzuhalten oder auf ein Gerichtsverfahren zu reagieren, (2) Ihre Einhaltung dieses Vertrags zu gewährleisten oder (3) die Rechte, das Eigentum oder die Interessen von Microsoft, ihrer Angestellten, ihrer Kunden oder der Öffentlichkeit zu schützen.

Seien Sie stets vorsichtig, wenn Sie persönliche Angaben oder Angaben zu Ihrer Familie machen.“

<http://messenger.msn.com/Help/Terms.aspx>

AGBs - ICQ

*"You agree that by posting *any material or information* anywhere on the ICQ Services and Information you surrender your copyright and any other proprietary right in the posted material or information.*

*You further agree that ICQ Inc. is entitled to use at its own discretion *any of the posted material or information in any manner it deems fit, including, but not limited to, publishing the material or distributing it.*"*

<http://www.icq.com/legal/policy.html>

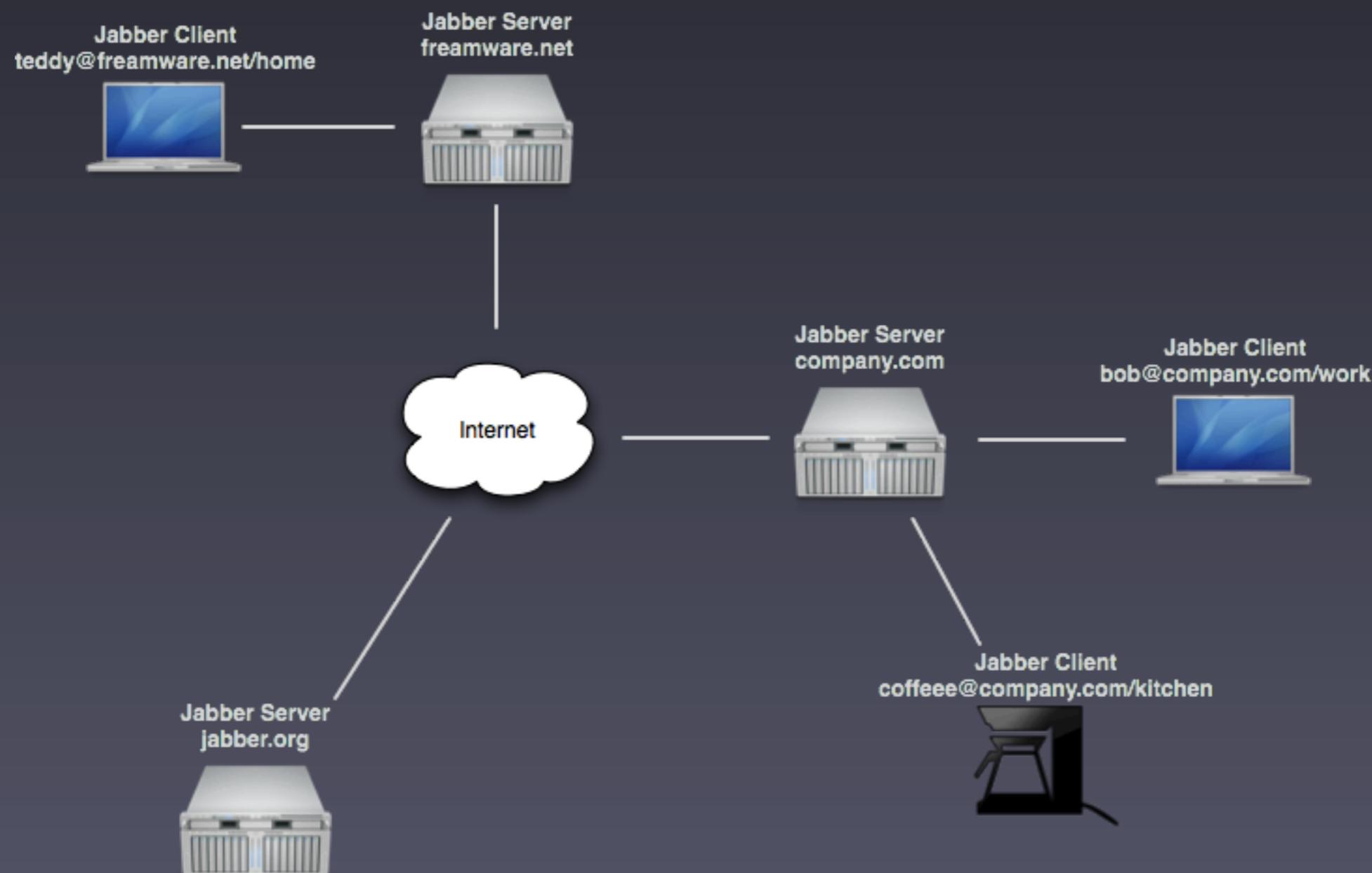
Was ist Jabber?

- „Jabber is best known as “the Linux of instant messaging“ – an open, secure, ad-free alternative to consumer IM services“
- 1999 von Jeremie Miller entwickelt
- **eXtensible Messaging and Presence Protocol**
- 2004 IETF Standard

Warum ist Jabber „cool“?

- Open Source, Offener Standard
- XML Datenströme
- Dezentral, dynamisches Netz - Server2Server
Verbindung on demand
- Jeder kann einen Server betreiben
- Client, Server für fast jede
Programmiersprache und jedes OS

Beispiel Architektur



Features (I)

- JID: dr scream@freamware.net/home
- Mehrere „Resourcen“ (Clients) gleichzeitig Online
- Serverseitige Kontaktliste (Roster)
- Jingle: Audio-, Videokonferenzen
- MUC: Multi-User-Chat
- Jabber Support: Nagios, Flyspray, ...

Features (2)

- Remote Controlling via Client
- Services zu anderen Diensten können angeboten werden
 - Gateway zu ICQ, MSN, ...
 - User Directory
- Verschlüsselte Kommunikation

Verschlüsselte Kommunikation

- TLS (Verschlüsselung)
 - Server zu Server
 - Client zu Server
- Ist TLS Ausreichend?
 - TLS bietet Transport Layer Security
 - Nachrichten sind unverschlüsselt auf dem Server
 - Nachrichten sind kopier- / speichbar

PANIC

- Ich hab noch so viele Fragen?!
- Informationen:
 - [http://de.wikipedia.org/wiki/
Extensible_Messaging_and_Presence_Protocol](http://de.wikipedia.org/wiki/Extensible_Messaging_and_Presence_Protocol)
 - <http://xmpp.org>
 - <http://jabber.org>

Jabber Implementierungen

Jabber Server & Programmieren mit Jabber Services

Jabber Server

- Vorhandene Server Infrastruktur
 - freamware.net
 - jabber.org
 - amessage.info
 - gmail.com
- Server Software
 - ejabberd, Erlang
 - OpenFire, Java
 - jabberd1, C
 - jabberd2 rewrite, C

Server: ejabberd

- Programmiersprache: Erlang
- Open Source (GPL-2)
- Plattform unabhängig
- „leichte“ Einrichtung
- ungewöhnliche Syntax (Konfiguration, Sprache)
- GMX, GoogleTalk

Programmieren: Was brauchen wir?

- Viel Zeit zur Implementierung von RFC3920 und RFC3921
- Fertige Bibliotheken
 - Python, C, Perl, Java, ...

Programmieren: Bibliotheken

Bibliothek	Sprache	Client	Server	Anmerkeung
smack	Java	X		High Level Bibliothek Grundlegende Funktionen
JSO	Java	X	X	Low Level Bibliothek Zugriff auf alle Details des Jabber Protokolls
class.jabber	PHP	X		High Level Bibliothek Jabber Dienst für Webanwendungen
Net::Jabber	Perl	X		High Level Bibliothek Schlechte Dokumentation
pyxmpp	Python	X		High Level Bibliothek Ausgereift, gute Dokumentation
iksemel	C	X		XML Parser Geeignet für Embedded Systeme

Programmieren: Beispiel (I)

- CVS Notification
 - CVS, Versionskontrolle für Teamarbeit
 - Information über Status von Dateien im CVS (commit, edit, unedit)
 - Mitteilung abhängig vom Jabber „Online“ Status

Programmieren: Beispiel (2)

- Verbindungsaufbau

```
con = jabber.Client(host=Server,port=5223, connection=xmlstream.TCP_SSL)
try:
    con.connect()
except xmlstream.error, e:
    print "Couldn't connect: %s" % e
    sys.exit(0)
```

- Authentifizieren beim Server

```
con.auth(username,password,resource)
con.setDisconnectHandler(disconnectCB)
con.registerHandler('presence', presenceCB)
con.getRoster()
con.sendInitPresence()
```

Programmieren:

Beispiel (3)

- Versenden der Nachricht(en)

```
def presenceCB(con, prs):
    type = prs.getType()
    who = prs.getFrom().getStripped()

    if type == 'subscribe':
        con.send(jabber.Presence(to=who, type='subscribed'))
        con.send(jabber.Presence(to=who, type='subscribe'))
    elif type == 'available':
        prs.getStatus()

    if (prs.getShow() != 'dnd' and who == cvsuser):
        Con.send(jabber.Message(cvsuser, message, subject="CVS Watch Alarm"))
```

Projekt Beispiele

- **Mimír** is a Jabber enabled news service. It collects news from various sources and notifies its users when new news items have been found.
- **Volity** is a platform for creating and playing Internet-based multiplayer games, with a focus on board and card games.
- **Poncho** is a Library that provides a simple but high-level Network Interface implemented in Java and C++. It allows you to quickly network integrate devices like temperature sensors as well as server monitoring system.

Noch mehr PANIC

- Wahhh, noch viel mehr Fragen :-)