VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

DATAMINING Z JABBERU

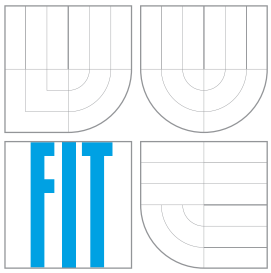
BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

JAROSLAV SENDLER

BRNO 2011

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

DATAMINING Z JABBERU

DATAMINING FROM JABBERU

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — JAROSLAV SENDLER
AUTHOR

VEDOUCÍ PRÁCE — Ing. JOZEF MLÍCH
SUPERVISOR

BRNO 2011

Abstrakt

Výtah (abstrakt) práce v českém jazyce.

Abstract

Výtah (abstrakt) práce v anglickém jazyce.

Klíčová slova

Klíčová slova v českém jazyce.

Keywords

Klíčová slova v anglickém jazyce.

Citace

Jaroslav Sendler: Datamining z jabberu, bakalářská práce, Brno, FIT VUT v Brně, 2011

Datamining z jabberu

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana ...

.....
Jaroslav Sendler
2. února 2011

Poděkování

Zde je možné uvést poděkování vedoucímu práce a těm, kteří poskytli odbornou pomoc.

Obsah

Kapitola 1

Úvod

Dnes mezi velmi se rozšiřující technologie na poli síťo

Kapitola 2

XMPP

Pro usnadnění pochopení budou v následující kapitole rozebrány základní stavební kameny protokolu Extensible Messaging and Presence Protocol (XMPP). Konkrétně jsou zde popsány stávající vlastnosti implementace [odkud se čerpalo], architektura protokolu XMPP obecně [9, 10] a další detaily protokolu [1, 11, 7], které se vztahují k požadavkům na data mining popisovaný v této práci [8, 6]. Další informace byly čerpány z [12, 5, 2, 4, 3].

Samotný protokol je datován do roku 2004 (březen), kdy na něj byl přejmenován Jabber. Původní projekt Jabber byl vytvořen roku 1998 autorem Jeremie Miller, jež ho založil na popud nesvobodných uzavřených IM služeb. Měl mít tři základní vlastnosti -jednoduchost a srozumitelnost pro implementaci, jednoduše rozšiřitelný a otevřený. Základní vlastnosti a výhody klientů a serverů budou popsány níže. Roku 1999, 4.ledna byl vytvořen první server se jménem Jabber. Komunita vývojářů se chopila iniciativy a napsala klienty pro různé platformy (Linux, Macintosh, Windows), kteří dokázali se serverem komunikovat. Roku 2004 byl přidán mezi RFC (request of comments - žádost o komentáře) dokumenty. Základní normy jsou RFC 3920 (obecná specifikace protokolu) a RFC 3921 (samotný instant messaging a zobrazení stavu). Další zdokumentovaná rozšíření jsou vydávána v podobě tzv. XEP (XMPP Extension Protocol) dokumentů, starším jménem JEP (Jabber Enhancement Proposal). Dnešní počet těchto norem se blíží k číslu 300. Každý XEP obsahuje status, stav vývoje (schválení), ve kterém se zrovna nachází. Jako bezpečnostní prvky jsou zde podporovány SASL, TLS a GPG. XMPP protokol je postaven na obecném značkovacím jazyce XML, proto vlastnosti popsané v kapitole 2.3 na straně 6 platí i pro tento protokol.

2.1 Architektura

Dobře navržená architektura tvoří základ pro správně fungující internetovou technologii. XMPP protokol využívá decentralizované klient-server složení. Tato struktura se nejvíce podobá struktuře posílání e-mailů. V tomto případě je decentralizace sítě chápána jako inteligentní nezávislost mezi vývojáři klientů a serverů. Každý z nich se může zaměřit na důležité části svého vývoje. Server na spolehlivost a rozšiřitelnost, klient na uživatele. Každý server pracuje samostatně, chod ani výpadek jiné datové stanice žádným způsobem neovlivní jeho běh, pouze bude nedostupný seznam kontaktů a služby, kterými server disponuje.

V tabulce č.2.1 jsou shrnuty rozdíly v architektuře Jabber, WWW a e-mail[1]. S každou zde jmenovanou službou má Jabber něco společného. Co se týče charakteristiky se vydal

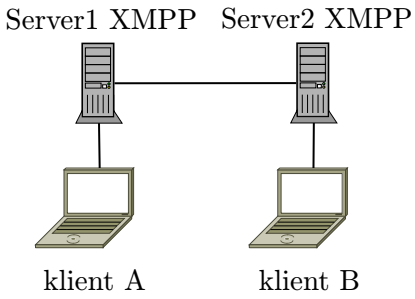
[1] internetový systém elektronické pošty

střední cestou. Na rozdíl od e-mailu, nepoužívá vícenásobný hops a v porovnáním s WWW využívá mezi-doménové připojení.

Charakteristika	WWW	Email	Jabber
mezi-doménové připojení	Ne	Ano	Ano
vícenásobný hops	N/A	Ano	Ne

Tabulka 2.1: Srovnání služeb WWW, Email a Jabber

Tyto vlastnosti jsou zárukou pro bezpečný přenos zpráv, znemožňují "krádeže"JID[2], který je popsán v podkapitole Jabber ID 2.4, a spamování. Obrázek 2.1 zobrazuje přenos zprávy mezi klientem *A* jehož účet vlastní *server1* a klientem *B* s účtem na *serveru2*.



Obrázek 2.1: Přenos zprávy

Klient

Klient je především plně ovládatelný grafický program podporující jednoduché odesíláni zpráv, ale v této práci jej zastupuje bot s konzolovým rozhraním. XMPP svou architekturou vnucuje, aby byl co nejjednodušší. Vlastnosti, které by měl mít jsou shrnuty do tří bodů:

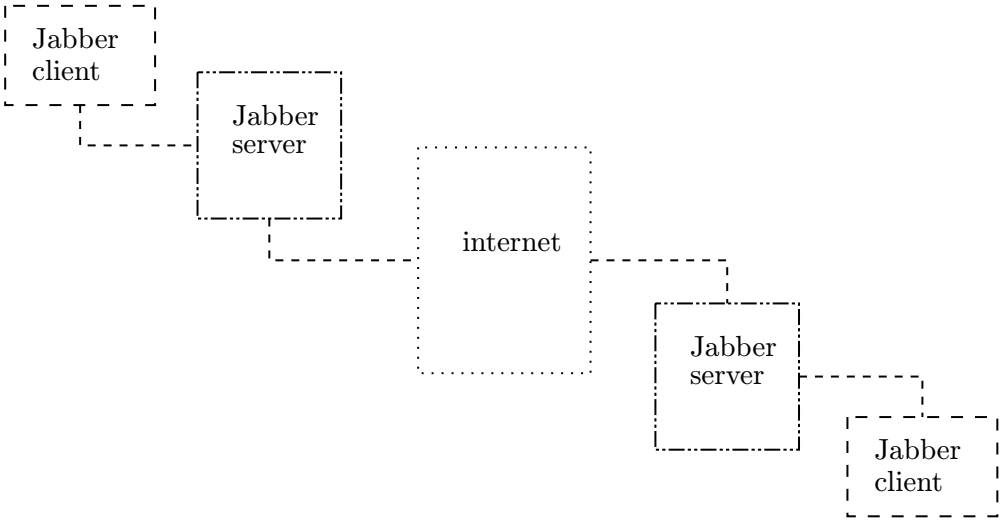
1. komunikace se serverem pomocí TCP soketu
2. rozparsování a následná interpretace příchozí XML zprávy „stanza“(kapitola 2.4)
3. porozumění sadě zpráv z Jabber jádra

Server

Hlavní vlastnost již není jako u klienta jednoduchost, ale stabilita a bezpečnost. Standardně běží na TCP portu 5222. Komunikace mezi servery je realizována přes port 5269. Každý server uchovává seznam zaregistrovaných uživatelů, kteří se do sítě mohou připojovat pouze přes něj. Tento seznam nemá žádný jiný server. To zajišťuje nemožnost „krádeže“ účtu. Protože XMPP komunikace probíhá přes síť, musí mít každá entita adresu, tady nazvána JabberID. XMPP spoléhá na DNS tudíž používá jména na rozdíl od IP protokolu.

Obrázek 2.2 znázorňující distribuovanou architekturu Jabberu byl převzat z [7].

[2] uživatelské jméno



Obrázek 2.2: Distribuovaná architektura Jabberu

2.2 Rozšíření

XMPP protokol je možné rozšířit o další vlastnosti. Pro jejich popis slouží XEP. Pro tuto práci jsou nepostradatelné rozšířené statusy, které popisují standardy XEP-0060 a XEP-0163. Obě tato rozšíření jsou důležitá.....

V tabulce 2.2 jsou shrnuty informace o serverech Jabberu. První sloupec tvoří jméno, následuje programovací jazyk v němž je napsán. Většina je vydávána pod licencí GPL[3], kromě ejabbred a Prosody. Ejabbred používá GPLv2, což je GPL licence druhé verze a Prosody licenci MIT/X11. U všech software byla zkoumána nejaktuálnější verze. Její číslo naleznete ve třetím sloupci. Všechny servery lze provozovat na operačním systému Linux a Windows. Na platformě Mac OS mohou být použity všechny zde jmenované vyjma jabberd2. Pět z šesti zde představených software pro server Jabber jsou ve stále vyvíjeny, tedy kromě jabberd14. Tabulka taktéž shrnuje důležité vlastnosti serverů v oblasti podpory rozšířených statusů o standardy XEP-0060 a XEP-0163.

Server	Jazyk	Verze	XEP-0060	XEP-0163
ejabberd	Erlang/ Top	2.1.6	ANO	ANO
Openfire	java	3.6.4	ANO	ANO
jabbred2	c	2.2.11	NE	NE
jabbred14	c, c++	1.6.1.1	ANO[4]	NE
Prosody	lua	0.7.0	NE [5]	ANO
Tigase	java	5.0.0	ANO	ANO

Tabulka 2.2: Přehled Jabber serverů

[3]General Public License-všeobecná veřejná licence GNU

2.3 XML

Jazyk XML (eXtensible Markup Language), metajazyk pro deklaraci strukturovaných dat, je jádrem protokolu XMPP. Samotný jazyk vznikl rozšířením metajazyka SGML, jež slouží pro deklaraci různých typů dokumentů. Základní vlastností je jednoduchá definice vlastních značek (tagů). Dokument XML se skládá z elementů, jež můžeme navzájem zanořovat. Vyznačujeme je pomocí značek - počáteční a ukončovací. Pomocí tohoto jazyka je tvořena stanza popsaná v kapitole 2.4.

Základní struktura dokumentu psaného jazykem XML je ukázána na obrázku 2.3. Každý dokument začíná XML deklarací a informací o kódovaní, ve kterém je dokument psán (1.řádek obrázku). Následuje kořenový element, jež je uzavřen na samotném konci dokumentu. 4. řádek ukazuje možnost použití prázdného elementu, který obsahuje jeden atribut s názvem zkratky fakulty.

```
1. <?xml version="1.0" encoding="utf-8"?>          // XML deklarace, kódování
2. <fakulta>                                       // kořenový element
3.  <název>Fakulta informačních technologií</název> // obsah elementu název
4.  <zkratka fakulty="FIT"/>                       // prázdný element
5.  <typy studia>                                  // počáteční tag
6.   <bakalářské titul="Bc."></bakalářské>        // název a hodnota atributu
7.   <magisterské></magisterské>
8.   <doktorské></doktorské>
9.  </typy studia>                                 // ukončovací tag
10.</fakulta>
```

Obrázek 2.3: Příklad základního XML dokumentu.

2.4 Stanza

Základní jednotkou pro komunikaci založenou na XML je stanza. Skládá se ze tří elementů *message*, *presence* a *iq*, jež každý má svůj jednoznačný význam.

Message

XML element prvního zanoření sloužící k posílání zpráv všeho druhu. Je to základní metoda pro rychlý přenos informací z místa na místo. Zprávy jsou typu „push“, tedy jsou odeslány a neočekává se žádná aktivita od příjemce, která by přijetí potvrdila. Zprávy jsou používány pro IM, skupinový chat a pro oznámení nebo upozornění. Pod elementy a atributy tvořící nutné minimum zprávy jsou *to* (příjemce zprávy), *from* (odesílatel zprávy) a *body* (obsah zprávy). Základní používané typy zpráv jsou *normální* (zpráva bez kontextu, vyžaduje odpověď), *chat* (komunikace mezi dvěma entitami), *groupchat*(skupinový chat) *headline* (upozornění) *error* (chybová zpráva). Celá struktura elementu <*message/*> je zobrazena v příloze E.2 strana 19.

Základní použití elementu <*message/*> je ukázáno na obrázku 2.4. Zobrazuje strukturu zprávy. Na prvním řádku je element *to* – příjemce zprávy, druhý *from* – odesílatel, následuje *type* – typ zprávy a nakonec samotný obsah.

```
1.          <message from="uzivatel@jabbim.com"
2.                   to="jabinfo@jabbim.com/bot"
3.                   type="chat"
4.              <body> Kolik je hodin? </body>
5.          </message>
```

Obrázek 2.4: Příklad použití elementu <*message/*>.

IQ

IQ nebo–li *Info/Query* poskytuje strukturu pro *request–response* (žádost–odpověď) vazbu a wokrflows, podobný metodám GET, POST a PUT z protokolu HTTP. Na rozdíl od *message* je *iq* spolehlivější přenos optimalizovaný pro výměnu dat (binární data). Příjemce musí na každou přijatou zprávu odpovědět, neboli potvrdit přijetí. Žádosti na proces nebo akci jsou posílány jednotlivě [**?**]. Celá struktura elementu <*iq/*> je zobrazena v příloze E.1 strana 18.

Obrázek 2.5 znázorňuje základní použití elementu <*iq/*>. Uživatel *uzivatel* posílá dotaz na získání kontakt listu (řádek 5.).

```
1.          <iq from="uzivatel@jabbim.com/doma"
2.              to="uzivatel@jabbim.com"
3.              id="uhhfw23648"
4.              type="get"
5.             <query xmlns="jabber:iq:roster"/>
6.          </iq>
```

Obrázek 2.5: Příklad použití elementu <*iq/*>.

Presence

Presence nebo–li informace o stavu (přítomnost) rozesílá dostupnost ostatních entit v síti. Struktura elementu <*presence/*> je zobrazena v příloze E.3 strana 19.

Základní použití <*presence/*> je zobrazeno na obrázku 2.6. Kontakt *jabinfo@jabbim.com/bot* (1.řádek) posílá informace o svém stavu (řádek č.2) a svůj status (č.3).

```
1.          <presence from="jabinfo@jabbim.com/bot"
2.             <show> online </show>
3.             <status> Jsme zde. </status>
4.          </presence>
```

Obrázek 2.6: Příklad použití elementu <*presence/*>.

Jabber ID

Jabber ID (JID) je jednoznačný virtuální identifikátor uživatele na síti. Není case-sensitive a je složen ze dvou částí. Takzvané *Jabber bare* neboli čisté ID a *resource*. První je část

na první pohled připomíná e-mailovou adresu *user@server*. Druhá část slouží k přesné identifikaci jednotlivých spojení. Je použit ke směrování trafiku s uživateli v případě otevření vícero spojení pod jedním uživatelem. Společně Jabber bare a resource tvoří tzv. *full JID* — *user@server/resource*.

2.5 Knihovny

Jabber je realizován jako otevřený XML standart pro instant messaging formát, proto existuje mnoho programovacích jazyků. Většina z nich disponuje několika knihovnami, usnadňující práci s protokolem XMPP. V tabulce 2.3 jsou pro nejznámější programovací jazyky zobrazeny dostupné knihovny.

Programovací jazyk	**knihovna**
C	iksemel, libstrophe, Loudmoutn
C++	gloox, Iris
JAVA	JabberBeans, Smack, JSO, Feridian, Emite, minijingle
.NET	Jabber-Net, agsXMPP SDK
Python	JabberPy, PyXMPP, SleekXMPP, Twisted Words
Perl	Net-Jabber
Ruby	XMPP4R, Jabber4R, Jabber::Simple, Jabber::Bot

Tabulka 2.3: Přehled Jabber knihoven

gloox

Gloox je stabilní Jabber/XMPP knihovna vydávána pod licencí GNU GPL. Je určena pro vývoj klienta a komponent. Jelikož je psána v ANSCI C++ mezi podporované platformy patří Linux, Windows, Mac OS X, Symbian/Nokia S60, FreeBSD a další systémy podporující ANSI C++ kompilátor.

Pomocí knihovny gloox je psán bot v této práci. Byla vybrána na základě požadavku psaní programu v jazyce C/C++ a operačním systému Linux. V porovnání s jinými knihovnami pro jazyk C nebo C++ disponuje lepší podporovou a dokumentací. Gloox plně podporuje standart XMPP Core [9] a z větší části i standard XMPP IM [10]. Dodatečně je plně podporováno kolem 30 XEP standardů například XEP-0054 vcard-temp a další.

Kapitola 3

Dataming

3.1 Metody dolovaní dat

3.2 Dolovaní znalosti z databází

3.3 Programy

Kapitola 4

Developer

4.1 Databáze

PostgreSQL

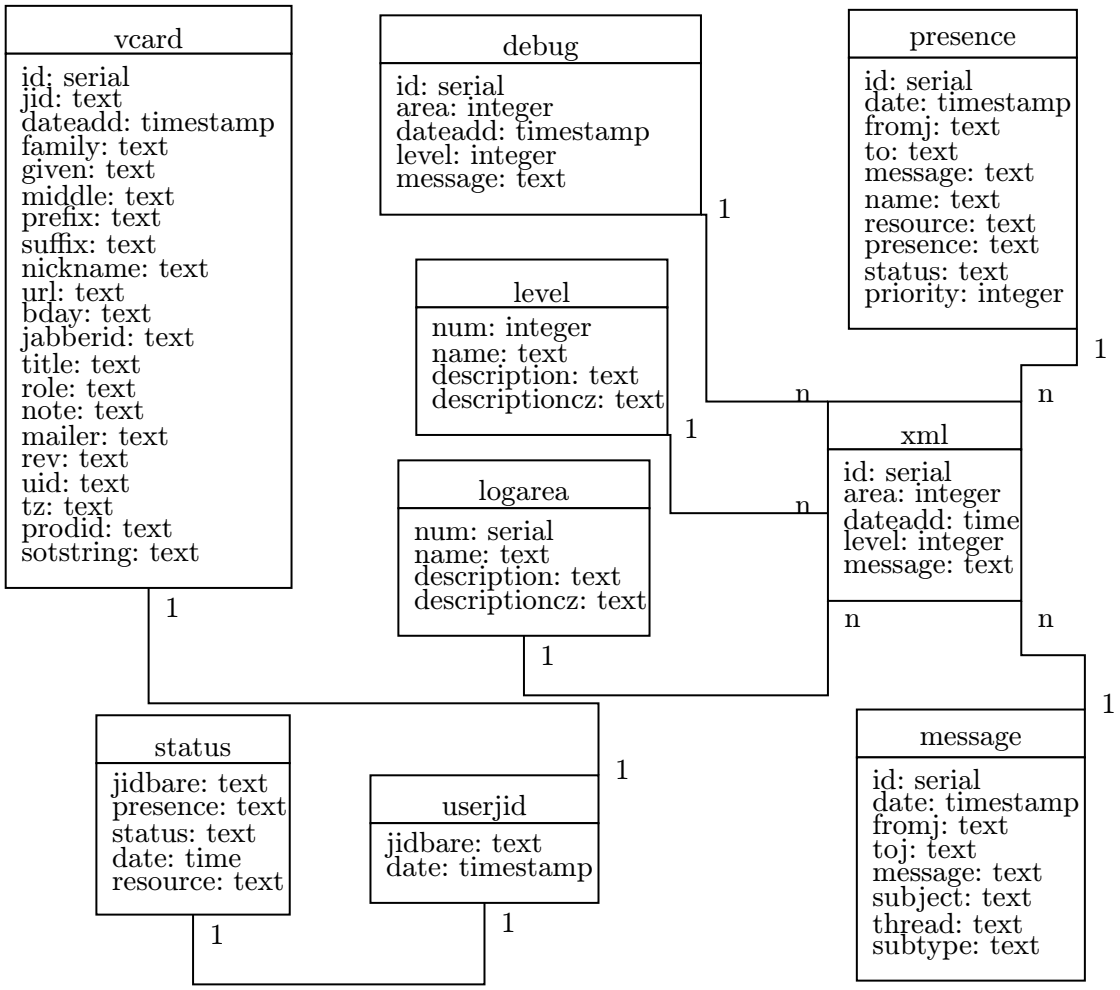
Návrh databáze

4.2 Bot

Návrh bota

4.3 Knihovny, jazyk

4.4 Jiné produkty



Obrázek 4.1: Struktura databáze

Kapitola 5

Závěr

Literatura

[1] Adams, D.: *Programming jabber*. Sebastopol: O'Reilly, první vydání, 2002, 455 s., iSBN 05-960-0202-5.

[2] Fred, H.: *Computer networking and the internet*. Edinburg: Addison-Wesley Publishing Company, první vydání, 2005, 803 s., iSBN 03-212-6358-8.

[3] Kolektiv autorů: Extensible Markup Language (XML) 1.0. [online], 26-11-2008, [cit. 2. února 2011].
URL http://www.w3.org/TR/2008/REC-xml-20081126/

[4] Kosek, J.: *XML pro každého : podrobný průvodce*. Praha: Grada, první vydání, 2000, 163 s., iSBN 80-716-9860-1.

[5] Kurose, J. F.; Ross, K. W.: *Computer networking : top-down approach featuring the internet*. Boston: Addison-Wesley Publishing Company, druhé vydání, 2003, 752 s., iSBN 03-211-7644-8.

[6] Millard, P.; Saint-Andre, P.; Meijer, R.: XEP-0060: Publish-Subscribe. [online], 12-07-2010, [cit. 2. února 2011].
URL http://xmpp.org/extensions/xep-0060.html

[7] Moore, D.; Wright, W.: *Jabber developer's handbook*. Indianapolis: Sams Publishing, první vydání, 2004, 487 s., iSBN 06-723-2536-5.

[8] Saint-Andre, P.; Smith, K.: XEP-0163: Personal Eventing Protocol. [online], 12-07-2010, [cit. 2. února 2011].
URL http://xmpp.org/extensions/xep-0163.html

[9] Saint-André, P.: Extensible Messaging and Presence Protocol (XMPP): Core. [online], 10-2004, [cit. 2. února 2011].
URL http://tools.ietf.org/html/rfc3920

[10] Saint-André, P.: Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence. [online], 10-2004, [cit. 2. února 2011].
URL http://tools.ietf.org/html/rfc3921

[11] Saint-André, P.; Smith, K.; Troncon, R.: *XMPP : the definitive guide : building real-time applications with jabber technologies*. Sebastopol: O'Reilly, první vydání, 2009, 287 s., iSBN 978-059-6521-264.

[12] Stevens, W.; Fenner, B.; M.Rudoff, A.: *UNIX Network Programming*. Boston: Addison-Wesley Publishing Company, třetí vydání, 2004, 991 s., iSBN 01-314-1155-1.

Příloha A

Obsah CD

Příloha B

Manual

Příloha C

Konfigrační soubor

Příloha D

Slovník výrazů

DNS — Domain Name System

GPG — dkshckdsjvlsdjvodsvjdfokj

IM služby — dkshckdsjvlsdjvodsvjdfokj

IP — Internet Protocol

JEP — dkshckdsjvlsdjvodsvjdfokj

JID — dkshckdsjvlsdjvodsvjdfokj

SASL — dkshckdsjvlsdjvodsvjdfokj

TCP — dkshckdsjvlsdjvodsvjdfokj

TLS — dkshckdsjvlsdjvodsvjdfokj

WWW — dkshckdsjvlsdjvodsvjdfokj

XEP — dkshckdsjvlsdjvodsvjdfokj

XML — dkshckdsjvlsdjvodsvjdfokj

XMPP — dkshckdsjvlsdjvodsvjdfokj

e-mail — dkshckdsjvlsdjvodsvjdfokj

jabber — dkshckdsjvlsdjvodsvjdfokj

klient — dkshckdsjvlsdjvodsvjdfokj

presence — dkshckdsjvlsdjvodsvjdfokj

server — dkshckdsjvlsdjvodsvjdfokj

stanza — dkshckdsjvlsdjvodsvjdfokj

vCard — dkshckdsjvlsdjvodsvjdfokj

Příloha E

Stanza - základní schéma

E.1 iq

```
<iq from=""
    to=""
    type="[get,set,result,error]"
    id=""
  Namespace
</ig>
```

Obrázek E.1: Popis elementu <*iq/*>.

jabber:client	jabber:server	jabber:iq:auth	jabber:iq:register
jabber:iq:roster	jabber:x:offline	jabber:iq:agent	jabber:iq:agents
jabber:x:delay	jabber:iq:version	jabber:iq:time	vcard-temp
jabber:iq:private	jabber:iq:search	jabber:iq:oob	jabber:x:oob
jabber:iq:admin	jabber:iq:filter	jabber:iq:auth:0k	jabber:iq:browse
jabber:x:event	jabber:iq:conference	jabber:x:signed	jabber:x:encrypted
jabber:iq:gateway	jabber:iq:last	jabber:x:envelope	jabber:x:expire
jabber:xdb:ginsert	jabber:xdb:nslist	texthttp://www.w3.org/1999/xhtml	

Tabulka E.1: Přehled Namespace elementu <*iq/*>.

E.2 Message

```
<message from=""
         to=""
         type="[normal,chat,groupchat, headline, error]"
         id=""
  <body>  </body>
  <x xmlns='jabber:x:event'>
     <[Offline, Delivered, Displayed, Composing]/>
  <subject>  </subject>
  <thread>  </thread>
  <error>   </error>
  <x>   </x>
</message>
```

Obrázek E.2: Popis elementu <*message/*>.

E.3 Presence

```
<presence from=""
          to=""
          type="[available, unavailable, probe, subscribe,
                 unsubscribe, subscribed, unsubscribed, error]"
          id=""
  <show>
    [away, chat, dnd, normal, xa]
  </show>
  <status>   </status>
  <priority>   </priority>
  <error>   </error>
</presence>
```

Obrázek E.3: Popis elementu <*presence/*>.