

# XMPP 协议在流媒体互动平台中的扩展与实现

杜 威<sup>1,2</sup>, 林 浒<sup>2</sup>, 杨海波<sup>2</sup>

<sup>1</sup>(中国科学院研究生院, 北京 100049)

<sup>2</sup>(中国科学院沈阳计算技术研究所, 沈阳 110168)

**摘 要:** 随着网络技术的飞速发展, 基于互动交流的流媒体视听播控平台将成为未来流媒体产业的发展趋势。XMPP 协议满足了播控平台进行即时交流的业务需求, 但在对于互动平台中的消息过滤和隐身可见支持不够完善。本文通过扩展 XMPP 协议, 设计并实现了即时消息的人工审核和系统自动审核机制、隐私列表的基本单元和隐私列表的使用机制、针对群组管理员的隐身可见机制, 并通过实验验证了该设计方案的正确性和可行性。

**关键词:** XMPP; 消息过滤; 隐身; 隐身可见; 隐私列表

## Extension and Implementation of XMPP on Streaming Media Interactive Platform

DU Wei<sup>1,2</sup>, LIN Hu<sup>2</sup>, YANG Hai Bo<sup>2</sup>

<sup>1</sup>(Graduate University, Chinese Academy of Sciences, Beijing 100049, China)

<sup>2</sup>(Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang 110168, China)

**Abstract:** With the rapid development of the Internet technology, based on the interactive video-audio streaming-media studio control platform will become the trend. XMPP protocol meets the requirement of the instant message communication. However, in the case of message filtering and visible to someone while invisible to others are not work well. In the paper, extended the XMPP protocol, the instant message filtering mechanism has been designed and implemented in the artificial or automatic way. The basic elements of the privacy list, the rules of the privacy lists, and the mechanism of visible to the group administer while invisible to others in the group have been designed and implemented. The experimental results have proved the validity and the feasibility of the design.

**Key words:** XMPP; message filtering by user hierarchy; invisible; visible to someone and invisible to others; privacy lists

## 1 引言

随着互联网与 3G 为代表的移动互联网的飞速发展, 流媒体作为集音频、视频于一体的新型媒体传播方式, 已经广为人们所接受, 基于网络的流媒体打破了以往的单项收听、收看模式, 为用户之间进行互动交流提供了可能。因此, 基于实时在线互动交流的流媒体视听播控平台将成为未来流媒体产业的发展趋势。与传统的广播和电视相比, 实时流媒体互动播控平台的优势在于能够在“视听”流媒体的同时进行实时消息互动交流, 形成边收听、边观看、边讨论的模式, 如何实现这样一种边收听观看边讨论的互动交流方式

是本文研究的重点。

由 IETF (Internet Engineering Task Force) 组织提出的 RFC3920<sup>[1]</sup>和 RFC3921 协议<sup>[2]</sup>支持点对点媒体会话、服务发现、结构化数据格式、信道加密、群组会话、Presence 状态显示、联系人列表、授权等服务, 在即时通信、群组聊天、系统控制、地理定位、VoIP、游戏等方面有独特的优势<sup>[3]</sup>, 能够满足流媒体互动播控平台的关于即时互动的业务需求。由于播控平台存在实时性强、受众广、影响范围大的特点, 因此保证平台中用户聊天内容不含不健康、广告等信息, 并为主持人提供特殊功能, 使得主持人、导播可以在节目

收稿时间:2011-07-20;收到修改稿时间:2011-09-11

直播时查看到每一位听众的真实在线状态等信息至关重要。因此,播控平台存在对即时消息进行过滤和栏目群管理员需要观察用户实时在线状态的业务需求。XMPP 协议在消息过滤和隐身可见机制方面支持不够完善,目前仅有 XEP-0045 协议<sup>[4]</sup>支持群组聊天室, XEP-0126 协议<sup>[5]</sup>和 XEP-0016 协议<sup>[6]</sup>对隐身机制有所涉及,但与播控平台的业务需求还有差距,因此需要通过扩展 XMPP 协议来实现该目标。

本文在研究互动播控平台业务需求的基础上,指出了 XMPP 协议在消息过滤和隐身可见机制的不足,研究了如何扩展 XMPP 协议,设计了即时消息的人工审核和系统自动审核的消息审核机制,设计了隐私列表的基本单元和隐身列表的使用机制,设计了针对群组管理员的隐身可见机制,并通过实验验证了设计方案的正确性和可行性。

## 2 XMPP协议扩展

### 2.1 基于 XMPP 协议的消息传输流程

用户通过 IM 客户端以 XML 文件流方式与服务器进行交互,XML 文件发送到服务器端以后,服务器首先由会话池为该客户端同服务器的连接在会话池里分配一个会话,然后 XML 文件将被转发到 XML Parser 模块,在此模块中完成对 Packet 的抽取,接着 Packet 将进入到一个由 Packet 组成的 Queue 里,再依次发送到 Packet Handlers 模块里,通过与数据库的连接,后的 Packet 发送到会话连接池中,再发送到用户端。共

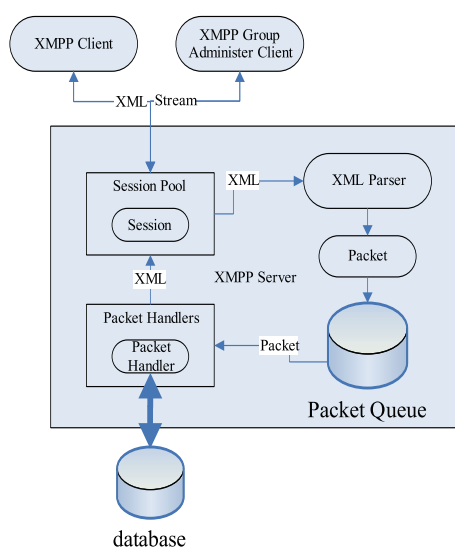


图 1 客户端与服务器通信模型架构图

同完成对 Packet 的基于规则的过滤处理,当处理过相关的协议扩展工作主要是在群组管理员客户端(Group Administer Client)和服务器的 Packet Handler 部分,通过对不同用户等级的消息 packet 设计对不同的处理流程,进而完成消息的过滤。系统架构如图 1 所示。

### 2.2 消息审核的设计思想

针对群组聊天消息处理,本文从用户客户端、服务器端、群组管理员客户端三个方面来说明:

#### 2.2.1 用户客户端:

当用户发送消息时,用户自己的聊天记录里添加自己的发言,客户端软件将消息以 Message 的格式发送到服务器端。

#### 2.2.2 服务器端处理:

服务器端接收到信息包,判断消息类型:

如果消息类型 type='chat' 时,表示该消息为普通的 1 对 1 用户聊天信息,则把该消息发送到 to 属性所指示的目标用户;

如果消息类型 type='groupchat' 时,表示该消息为用户发的群组聊天信息,则对消息进行判别:

如果消息含有 flag 标记位,且服务器检测到该消息是来自 to 属性所指示的栏目群的群管理员用户,则系统将该消息体发到 to 属性指定的聊天室中,群组内有权接收该消息的用户客户端均应该收到该消息。

如果消息体中没有 flag 标记为,系统将依据消息的发送者身份做如下判断:

(a) 发送者为白名单中用户,即用户属性为主持人(moderator)和房间所有者(owner)则系统直接把消息发送到聊天室里,即系统转发该信息到群内所有客户端。

(b) 发送者为黑名单中用户,即用户属性为被逐出的(outcast),则系统将按照当前服务器处理消息的模式进行处理,即把该消息作为用户的发言保存在服务器的聊天记录数据库中或者直接丢弃该消息。

(c) 发送者为普通用户,即用户属性为参与者(participant),则系统首先将该消息保存在数据库中,然后检查当前的消息审核模式:

如果为系统自动审核,则消息需要经过自动审核模块处理:若消息通过审核,则系统将该消息转发到 to 属性所指示的栏目群聊天室里;若没有通过审核,则系统将丢弃该消息。

如果为群组管理员人工审核模式,则系统将消息包上添加 flag 元素,并将 flag 的子元素 to 属性值设为

目标地址 to 属性所指示的栏目群聊天室名称, flag 子元素 result 属性设置为 unknown, 消息 type 属性为 chat, 修改消息的 to 属性为群组管理员, 然后系统将该消息按照 1 对 1 聊天的方式发送到群管理员客户端。具体流程如图 2 所示。

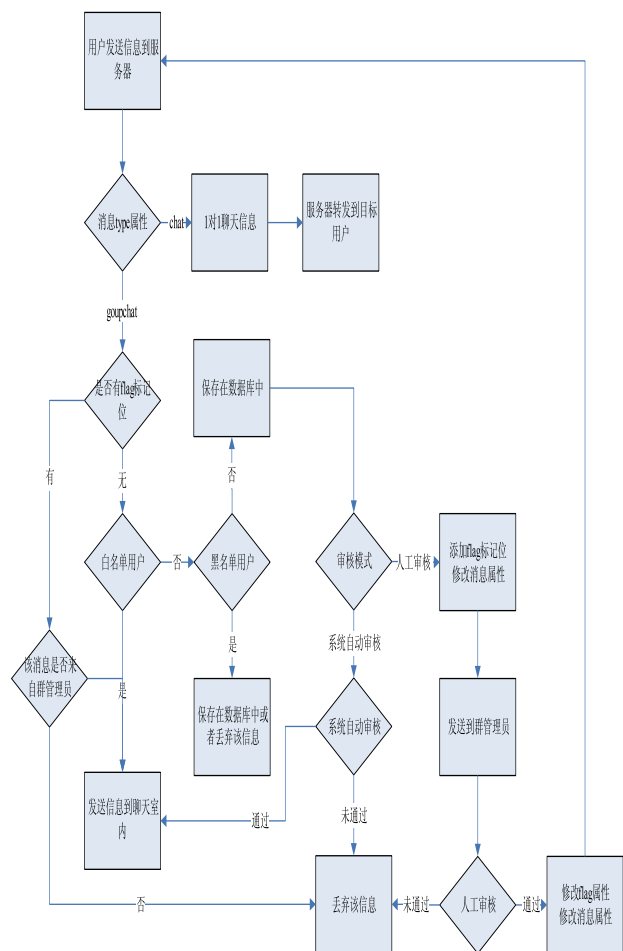


图 2 消息分级过滤流程图

### 2.2.3 群组管理员客户端：

当客户端检查到一个含有 flag 标记位的聊天信息后, 该消息将显示在人工审核的窗口, 管理员通过阅读该消息完成人工审核, 如果该信息通过审核, 则管理员点击发送按钮。此时, 管理员客户端修改消息的类型为 groupchat, 修改消息的目的地址为从 flag 的子元素 to 中取出来的用户发送的目标聊天室地址, 修改 flag 子元素 result 属性值为 true, 去掉 flag 子元素 to 及其属性, 消息发送者为原来发送消息的用户名。其他属性不变, 发送该消息到服务器端。如果该消息未通过管理员的人工审核, 则管理员客户端点击丢弃按

钮, 管理员客户端软件将丢弃该消息。

### 2.3 用户状态变化设计思想

本文首先给出隐身可见定义。

隐身可见：当用户处于在线隐身状态时, 用户或者服务器端通过设置特定状态显示信息可以让某些特定用户知道该用户当前的在线状态信息, 则称用户对那些可以知道用户当前在线隐身状态信息的用户为隐身可见。

对于用户从登录系统到通过 IM 终端同好友交互, 再到退出终端时不同的状态变化, 共有两种可能, 一种是在线状态登录, 一种是隐身状态登录, 登录到客户端后, 用户可以从在线变为隐身, 也可以从隐身变为在线, 退出客户端时, 根据用户当前状态直接退出客户端。

#### 2.3.1 用户在线状态

当用户使用在线方式登录 IM 时, 用户使用缺省隐私列表, 此时用户的状态消息直接发送到服务器, 用户的当前状态对于所有订阅了用户状态的好友都可以见到, 对于用户所在群组的其他人也可以看到用户当前的状态信息。用户可以进行同好友聊天, 接收好友发送的聊天信息等。当用户希望退出 IM 系统时, 用户向服务器发送 unavailable 的 presence 消息, 当服务器接收到该消息后, 服务器将向所有订阅了用户状态的用户发送该用户 unavailable 的 presence 消息, 然后断开连接。其状态变化如图 3 所示。

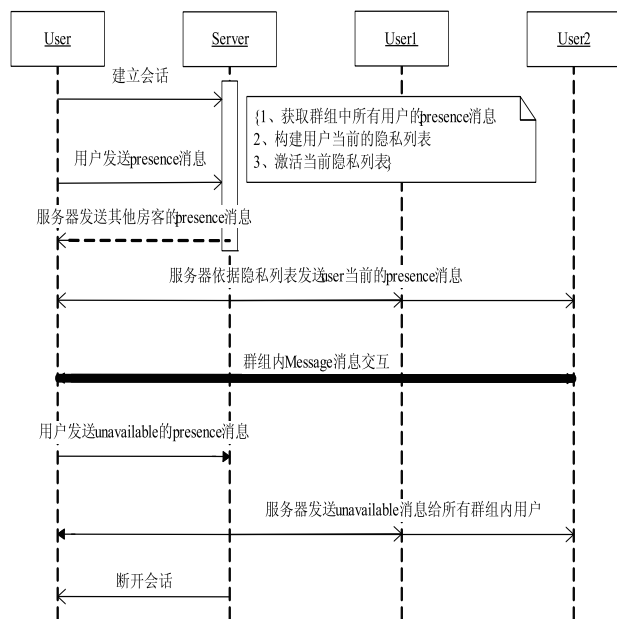


图 3 用户状态变化的 presence 消息流程

### 2.3.2 用户由在线变为隐身状态

当用户选择隐身时,用户首先向服务器发送一个 invisible 类型的 presence 消息,当服务器接收到该消息后,服务器将向所有订阅了该用户状态消息的用户发送该用户 unavailable 的 presence 消息,服务器端检查用户的隐私列表是否需要更新,即用户所在群组中的用户及针对这些用户的隐身可见属性是否有变化,若无变化,则不需要进行更新,服务器从数据库中读取出用户的隐私列表;若有变化,则根据当前的群组内用户信息生成新的隐私列表。撤销当前使用的隐私列表,激活用户用于隐身的隐私列表,然后用户将发送 presence 消息到服务器端,由服务器完成依据隐私列表规则的用户好友 presence 消息通知。如图 4 所示。

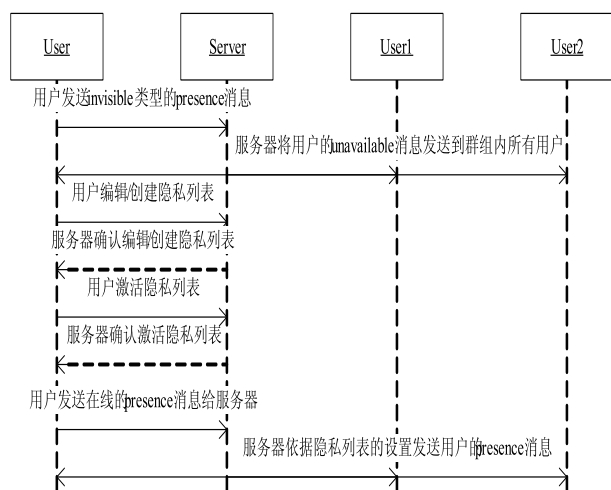


图 4 用户由在线变为隐身状态变迁图

### 2.3.3 用户隐身登录服务器

当用户使用隐身状态登录服务器时,在建立会话、服务器完成对用户的授权后,用户发送 available 的状态消息之前,用户激活了一个隐私列表,该隐私列表中规定用户屏蔽对非隐身可见的好友<presence-out/>消息,用户允许对隐身可见好友栏目群管理员和群所有者的<presence-out/>消息,用户屏蔽对群组内除管理员和所有者以外的其他人员<presence-out/>消息。系统自动激活隐私权限设置,在该设置中对管理员设置了可见的权限。

### 2.3.4 用户从隐身转为在线状态

用户不再使用任何的隐私列表,然后用户发送 available 的 presence 状态消息给服务器,服务器将用户消息转发给群组内所有用户。

## 3 XMPP协议扩展实现

使用 smack 库<sup>[7]</sup>作为基本的 XMPP 通信基础,使用 openfire 作为服务器端软件。并从以下几个方面来完成系统的具体实现:

### 3.1 消息的自动过滤模块

当服务器软件启动后,服务器首先把所有的规则从数据库中取出来,并将这些规则添加到一个规则链表里。当用户消息过来后,取得信息的消息体,并保存为字符串的格式。遍历这些规则,并将每个规则都匹配一下该字符串。如果发现有匹配成功的,则表示该消息内含有敏感的文字信息,退出匹配过程并给出提示,服务器将丢弃该消息。如果把所有的规则都匹配结束后,仍然没有发现与字符串匹配相符合的规则,则该消息内没有敏感信息。服务器将该消息发送到所指示的聊天室里。

### 3.2 人工审核模块

群组管理员的客户端软件处理来自服务器端的消息包时,首先检查消息的类型

如果为基本的聊天消息,类型 type 为 chat,则显示在管理员与该用户的聊天记录窗口里。

如果消息类型为 groupchat,检查该消息体的头部是否含有 flag 标签,如果有 flag 标签,则表示该消息为服务器发来的等待用户进行人工检查的消息,客户端软件将该消息显示在用于消息人工检测的窗口里,当消息通过审核后,管理员点击 send 按钮,此时,客户端完成该消息的 to、result、type 属性的修改,to 属性修改为从 flag 位取出的目标聊天室地址,result 修改为 true,表示已经通过人工审核,type 修改为 groupchat。客户端将该消息发送到服务器端;如果没有通过人工审核,则管理员点击删除按钮,客户端将丢弃该消息。否则,该消息为普通的群组聊天消息,客户端软件将该消息显示在对应的群聊天室窗口里。

### 3.3 隐私列表的管理

#### 3.3.1 隐私列表的创建和添加

当用户加入一个群组时,系统隐性地创建一个隐私列表;如果当前用户已经有一个隐私列表时,则系统检查该隐私列表是否处于激活状态,如果处于激活状态,则系统撤销当前隐私列表的激活状态,并激活系统默认的 default 隐私列表,即数据库中 isDefault 属性为 1 所对应的隐私列表。当用户拥有一个可以操作的隐私列表时,客户端软件可以操作该隐私列表。

当用户成功添加进入一个新的群组后，系统添加群组内所有用户该用户的隐私列表项。

隐私列表添加结束后，当用户选择隐身时，系统将激活用户自己设置的隐私列表。

### 3.3.2 隐私列表的修改和删除

当用户退出某个群组时，客户端软件需要隐性地修改包含了该组用户隐身属性设置的列表，用户首先需要遍历地取得该隐私列表，然后把所有涉及到该群组的用户的相关属性设置给取消。当群组内添加人员或者删除人员信息时，或者群组内的管理员信息发生改变时，用户的客户端软件需要进行自动地更新相关的隐私属性设置。

## 4 测试

### 4.1 测试场景

为了测试系统的具体实现，构建了一个名为“duweiroom”的聊天室。聊天室有三个用户，包括 duwei、jessica、jack 和 dewey，其中 dewey 是聊天室内管理员。其中用户之间的订阅关系如表 1 所示。其中，dewey 用户为群组管理员，duwei 与 jessica、jack

与 jessica 用户双方互相订阅，jack 与 duwei 用户没有任何订阅关系，dewey 用户作为群组管理员强制地订阅了 duwei、jack 和 jessica 用户的 Presence 信息。通过 openfire 作为服务器，结合 Smack 库作为基本的 XMPP 协议实现，修改相应函数实现，完成相关测试工作。

表 1 聊天室内用户之间的订阅关系

订阅者	duwei	jessica	jack	dewey
duwei	——	——	——	——
jessica	both	——	——	——
jack	none	both	——	——
dewey	from	from	from	——

### 4.2 测试结果与分析

本测试主要针对群组聊天时，普通用户发送从在线状态转变到隐身状态的消息流程来完成测试。本测试使用 Smack 下的 debug 模式获取到 XML 文件。

当用户状态消息变化时，服务器将按照该隐私列表进行状态更新，通知用户的好友进行对应的状态更新。整个流程如图 5 所示。

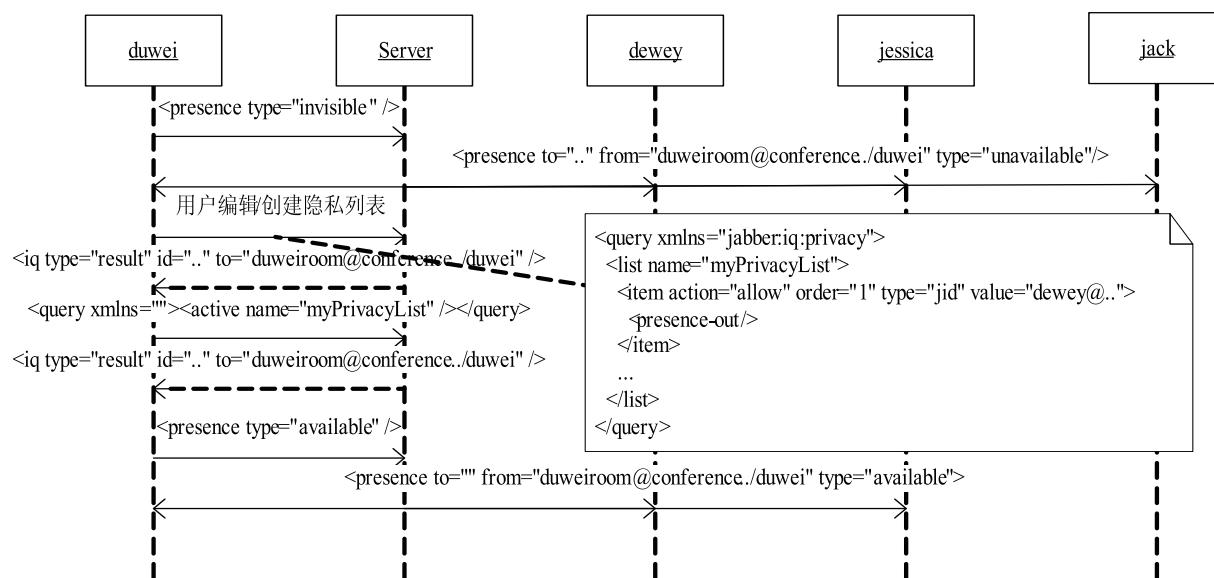


图 5 用户从在线变为隐身状态 XML 描述

通过使用 Smack 的 debug 模式进行了抓包分析，从基本的 XML 数据信息来看，当用户激活群组聊天室时，系统将搜索当前群组的所有用户，并把群管理员添加为隐私列表中对其显示 presence 状态消息

的项。

当用户状态从在线变为隐身时，客户端请求服务器激活用户所设置的隐私列表，服务器激活隐私列表。图 6 说明用户请求激活隐私列表的 xml 片段。

```
<iq id="NLz49-14" from="duwei@10.0.0.3/Smack" type="set">
  <query xmlns="jabber:iq:privacy">
    <active name="myPrivacyList"/>
  </query>
</iq>
```

图 6 用户请求激活隐私列表

通过以上信息分析可知,该隐身可见机制符合最初设计目标。

## 5 总结

在研究流媒体互动播控平台的基础上,针对其中的消息过滤和隐身可见机制进行了业务需求分析,结合 RFC3921、XEP-0045、XEP-0126 和 XEP-0016 等协议,设计了基于 XMPP 协议的人工审核和系统自动过滤的消息审核机制、隐私列表的基本单元和使用机制、针对群组管理员的隐身可见机制,并使用 smack 库验证了该设计方案。

(上接第 258 页)

本,可以建立文本矩阵。依据距离矩阵和文本矩阵,对于任一可在标准键盘输入的语言,其键盘布局优化设计的信息已经足够。最后本文给出了针对中文拼音输入的新键盘布局设计,并通过几份实际的文本测试,进行了相关效率比较。而要使新键盘布局方案得到推广,根本上还是改变人们长久以来的习惯,如果要广泛普及还有很多工作要做。

## 致谢

在此,向中文词库提供者郭家宝表示感谢。

## 参考文献

- 1 Liebowitz SJ, Margolis SE. The Fable of the Keys. Journal of Law and Economics, 1990,33(1):1-25.
- 2 Anderson AM, Mirka GA, Sharon M, Joines B, Kaber DB. Analysis of Alternative Keyboards Using Learning Curves. Human Factors, 2009,51(1):35-45.
- 3 Land RI. Keyboard Entry-Can it be Simplified. ACM SIGSOC Bulletin, 1982,13(3).
- 4 O'Brien MA, Rogers W, Fisk AD, Richman M. Assessing

## 参考文献

- 1 Peter SA. RFC3920: Extensible Messaging and Presence Protocol (XMPP): Core. Jabber Software Foundation http://www.ietf.org/rfc/rfc3921.txt.October 2004.
- 2 Peter SA. RFC3921: Extensible Messaging and Presence Protocol Protocol(XMPP): Instant Messaging and Presence. Jabber Software Foundation. http://www.ietf.org/rfc/rfc3921.txt. October 2004.
- 3 Peter SA, Smith K, Troncon R. XMPP: The Definitive Guide, O'Reilly: USA, April 2009.
- 4 Peter SA. XEP-0045: Multi-User Chat. Jabber Software Foundation. http://xmpp.org/extensions/xep-0045.html 2008
- 5 Peter SA. XEP-0126:Invisibility. XMPP Standards Foundation.http://xmpp.org/extensions/xep-0126.html. August 2005.
- 6 Peter M, Peter SA.XEP-0016:Privacy Lists. XMPP Standards Foundation. http://xmpp.org/extensions/xep-0016.html. Feb 2007.
- 7 Smack. Ignite Realtime: Smack API. http://www.Ignitereal-time.org/projects/smack/2009.

design features of virtual keyboards for text entry. Human Factors, 2008,50(4):680-698.

- 5 Zhai SM, Sue A, Accot J. Movement model, hits distribution and learning in virtual keyboarding. Dennis Wixon. The SIGCHI Conference on Human Factors in Computing Systems: Changing Our World, Changing Ourselves. NY: ACM, 2002:17-24.
- 6 Francis G, Oxtoby C. Buiding and testing optimized keyboards for specific text entry. Human Factors, 2006, 48(2):279-287.
- 7 Goettl JS, Brugh AW, Julstrom BA. Call me e-mail: Arranging the keyboard with a permutation-coded genetic algorithm. In: Liebrock LM, ed. The 2005 ACM symposium on Applied Computing. NY: ACM, 2005:947-951.
- 8 卢亚军.藏文计算机通用键盘布局与输入法研究.中文信息学报,2006,20(2):78-86.
- 9 郝玉哲.人机工程型键盘.中国安全科学学报,1995,5(6):40-44.
- 10 方敏,金一庆,刘丹凤.英文输入用键盘布局的改进.计算机工程与设计,2004,25(1):153-160.