

# PROJECT REPORT

## Customization of Ekiga for use with GatorPhone

*Mahesh Sukumar*

*UF ID: 1156-2993*

**Abstract**—Currently, In order to connect to the Gatorphone.com phone system, the recommended soft-phone is X-Lite which is a free (but closed-source) application from CounterPath Corporation. This project proposes to make use of the Open-Source Soft phone Ekiga (Previously called “Gnome-Meeting”) and customize it for use with Gatorphone.com. The customization will reduce the setup time required for using the soft-phone with the Gatorphone service by creating a new menu entry for Gatorphone.com in the Account Creation menu. Having this menu entry will ease the configuration of the Gatorphone account as the user will have to only enter his Gatorphone.com user id and password only, instead of entering all the other sip server related details as is done with X-Lite. This project provides the modified installers for both Linux and Windows platforms. As there is no native build process for Windows, we use the cross-compilation support available in Linux to build the Windows version too.

**Index Terms**—[Software](#), [Software Programming](#), [Software Tools](#).

### I.INTRODUCTION

Ekiga is licensed under the GNU General Public License (GPL) and is a SIP and H.323 compatible (SIP and H.323 are two protocols used in telephony over Internet) VoIP (Voice over IP, or IP Telephony) and Video Conferencing application that allows you to make audio and video calls to remote users with SIP or H.323 hardware and software. It supports all modern VoIP features for both SIP and H.323. Ekiga is the first Open Source application to support both H.323 and SIP, as well as audio and video. Ekiga was formerly known as GnomeMeeting.

Ekiga Ver.3.0.0 SoftPhone has the following features:

#### GUI:

- **HIG Compliant GUI:** Ekiga looks right, behaves properly, and fits into the GNOME user interface as a whole.
- **Message Waiting Indications Support:** Ekiga can tell you how much Voice Mails are waiting.
- **Contact centric interface:** preferred contacts can be listed in the main user interface.
- **Contact's presence display:** Offline, Away, Do Not

disturb, Online status (SIP/SIMPLE)

- Automatic addition of network neighbors contacts in the main user interface (using the Bonjour/ZeroConf protocol)
- Optional call panel
- Connection quality meter in the status bar
- Easy to use shortcuts
- Tray icon with call-in notification
- Easy account selection when dialing for a contact: auto-completion

#### CALL:

- **Call Hold (SIP and H.450.1):** This effectively pauses Video & Audio transmission.
- **Call Transfer (SIP and H.450.2):** You can transfer the remote user to another H.323 or SIP user.
- **Call Forwarding on No Answer, on Busy, Always (SIP and H.450.3):** This allows you to configure Ekiga to forward incoming calls to a specified SIP or H.323 user.
- **Do Not Disturb Modes**
- **Dialpad:** This allow you to dial numbers.
- **DTMFs Support:** This feature is necessary when using services asking you to dial numbers.
- **Calls History:** This is a convenient way about all outgoing and incoming calls.
- **Call Monitoring:** Statistics about the network traffic caused by Ekiga are displayed in the status bar.
- **Enum Support:** Enum is a method to provide a unified numbering system between the public switched telephone network (PSTN) and various VoIP providers.

#### VIDEO:

- Configurable Full-Screen Video Conferencing
- Quality Versus speed slider
- Support for XVideo (hardware rendering)
- High Frame rate (up to 30 fps)
- Configurable Resolution (from 176x144 to 704x576)
- SIP Capabilities exchange

#### TEXT CHAT:

- Instant Messaging with built-in smiley support (SIP)
- H.245 Text Chat during calls (H323)

Manuscript received February 26, 2009.

Mahesh Sukumar is a Masters Student in the Electrical and Computer Engineering Department at The University of Florida, USA. phone: 352-246-7890; (e-mail: smahesh321@ufl.edu).

- Graphical selection of smiley for easy inclusion in text message
- Presence display: display your peer chat status

#### CONTACTS:

- Advanced Address book: The Address Book is a feature which allows you to find users to call and/or to save locally your list of persons that you call on a regular basis.
- LDAP Support: Ekiga can load the list of users from a remote LDAP directory
- Status with custom message (SIP): The status of your contacts are displayed in the main User Interface with a custom message.

#### CONFIGURATION:

- New Configuration Assistant: The Configuration Assistant is a 8 step-by-step questionnaire that will guide you through all the steps involved in creating the basic configuration you will need to operate Ekiga.
- In call Instant-Apply Support for Settings
- GConf and External Configuration: Ekiga can store its settings in Gnome's GConf application or in a standalone file.
- Advanced Personal Settings Management
- Configurable Sound Events: You can customize sounds and select a special sound device for them to play; you're able to have the ring in your Hi-Fi hardware and the call in the headset.
- Configurable Port Ranges (SIP and H323): Ekiga uses standard ports; in rare you might change the ports. (Advanced setting)
- Better NAT support; automatic selection of STUN

#### DEVICES:

- OSS and ALSA Compatible Sound cards Support
- Video4Linux, Video4Linux 2 and Firewire Cameras Support
- Devices Auto-Detection
- Automatic detection of hotplugging of alsa audio devices and video4linux video devices on Linux

#### Misc:

- SIP re-INVITE support
- Translated into Many Languages
- Plug-in support for audio and video codecs

#### CODECS FEATURES:

- Dynamic Jitter Buffer

#### AUDIO:

- iLBC, GSM-06.10, MS-GSM, G.711-Alaw, G.711-uLaw, G.726, G.721 and Speex Audio Codecs
- Dynamic Threshold Algorithm for Silence Detection

- Echo Cancellation
- Wideband Codec Support

#### VIDEO:

- H.261 Video Codec
- H.264 Video Codec
- H.263+ Video Codec
- MPEG4 Video Codec
- THEORA Video Codec
- Video Bandwidth Limitation

#### VoIP SERVICES:

- Possibility to Simultaneously Register to Several Accounts: You can register as many SIP or H.323 provider accounts as you want, and you're able to use them simultaneously.
- SIP Compliant: You can use any SIP compliant VoIP provider. They may provide you address for VoIP similar to e-mail address, PC-to-Phone calls, Phone-to-PC calls, Voice mails, ...
- Registrar Support: You can register to SIP compliant VoIP providers.
- H.323v4 Compliant: You can use any H323v4 compliant VoIP provider.
- Gatekeeper (RAS) Support: You can register a VoIP service using H.323.
- Outbound Proxy Support: Some providers use a relay for your communications and require this setting.
- H.235 Annex D. Support: Security of calls.
- SIP: SIMPLE presence support
- Better SIP Outbound Proxy support : As recommended in the RFC
- Better handling of SIP registrations : More robust code
- SIP Presence subscriptions : SIP SUBSCRIBE/NOTIFY
- SIP Presence publishing : SIP PUBLISH
- SIP Presence document : SIP PIDF

#### INTEGRATION:

- Part of Gnome
- Integration with Novell Evolution 2.00: You can share your contacts with the groupware client for Linux, "Novell® Evolution™".
- KDE and GNOME Compatibility
- Remote LDAP server integration for network address-book integration
- Interoperability with PBX like Asterisk(tm)
- Local network integration using the Bonjour/ZeroConf protocol

#### WINDOWS FEATURES:

- Support for DirectShow Video Capture
- Support for DirectDraw Video Output

- Installer

#### NETWORK:

- Transparent NAT Support, Assisted NAT Support (STUN, IP Translation): Ekiga has extensive and improved NAT support thanks to STUN. In 99% of the cases, you do not have any configuration to do, and you can even be reachable from the outside without any port forwarding.
- Rendez-Vous Support: This system facilitates service discovery on a local network. Ekiga users in the local network will show up automatically.
- Gateway/Proxy Support
- H.245 Tunneling and Fast Start
- Unique port : Use only one port for all outgoing SIP requests going to the same destination
- Detection of dynamic IP address changes and coming up and going down of network interfaces

#### For Developers:

- GUI and Engine separation to allow GUI customisation/port to other toolkits and integration in other projects.
- Use of sigc++ in a signal-based organization.

## II. WHY EKIGA?

Ekiga was selected for this customization project in comparison to other Voip Clients based on the following criteria:

1. Is fully Open Source and free.
2. Is currently being actively developed and will be actively developed in the foreseeable future.
3. Works on both Linux and Windows platforms.
4. Has properly working build mechanisms for both Linux and Windows.

The other voip clients like X-Lite, miniSIP, YATE and LinPhone do not meet atleast one of the above listed criteria.

There are a few minor disadvantages of using Ekiga. They are:

1. Ekiga is not skinnable yet.
2. As it is being actively developed/modified, it will be difficult to keep synchronized with its frequent release cycles.

## III. PROJECT OBJECTIVES

### A. Study the Ekiga Open-Source Software and its dependencies

Building Ekiga from source depends on a number of other softwares/projects. Given below is a partial list -

#### 1)PTLib

- flex: compile-time dep.
- bison: compile-time dep.
- OpenLDAP: compile-time and run-time dep. It needs the headers (-devel packages)
- libexpat1-dev: compile-time dep, needed for presence indication.

PTLib is a moderately large C++ class library that originated many years ago as a method to produce applications that run on both Microsoft Windows and Unix X-Windows systems. It also was to have a Macintosh port as well, but this never eventuated. In those days it was called the PWLib the Portable Windows Library.

Since then, the availability of multi-platform GUI toolkits such as KDE and wxWindows, and the development of the OpenH323 and OPAL projects as primary user of the library, has emphasised the focus on networking, I/O portability, multi-threading and protocol portability. Mostly, the library is used to create high performance and highly portable network-centric applications. So all the GUI abstractions have been dropped and it was renamed the Portable Tools Library that we see today.

#### 2)Opal

- PTLib.
- for some codecs.

Opal is a full-featured library that supports a variety of protocols and multi-media formats. Features include:

- Low latency RTP stack designed specifically for real-time multimedia
- Full featured H.323, SIP and IAX2 protocol stacks
- Audio codecs including G.711, GSM06.10, Speex and iLBC.
- Video codecs including H.261 and H.263
- Run-time loadable codec interface for proprietary or codecs such as G.729, H.263, H.264 and MPEG4
- Completely Open Source using the commercially friendly MPL 1.1 license

Through the use of a platform isolation library called PTLIB, Opal can run on tiny embedded systems or multi-processor servers.

#### 3)Ekiga

- GLib: GIMP library
- GTK+: GIMP toolkit 2.6 or higher
- libgnome: GNOME library

- gnome-common: gnome-common module to support GNU autotools with some scripts
- GNOME UI lib: GNOME UI library
- GConf: GNOME config database
- libsdl: SDL library, for full screen support
- libebook: evolution library, for Evolution mail client integration
- dbus-glib: GLib DBUS bindings, for basic control of Ekiga using DBUS
- autotools: GNU, aka *automake*, *autoconf*

#### B. Add Gatorphone.com specific functionality to Ekiga

This includes adding a GatorPhone.com menu entry in the “Add Account” menu item under “Accounts” menu. The “Add a GatorPhone.com

#### C. Test the changes

The changes done to the software shall be tested on both Linux and Windows. The modified software shall also undergo a regression testing phase to make sure that our changes have not broken any of the original functionalities of Ekiga.

#### D. Create Linux and Windows specific installer packages

Installer packages for Linux in .deb format and for Windows in either .msi or .exe format shall be created.

### IV. RESEARCH RESOURCES

As this is a programming project, resources available on the internet will be heavily used. Some of the web resources that will be used are listed below -

<http://wiki.ekiga.org>

### V. WORK PLAN

The work plan is divided into 4 stages -

1. Design – Creation of High Level Design, Low Level Design.
2. Coding and Unit Testing – Coding the application in the IDE and basic Unit testing.
3. Comprehensive Testing and Debugging – End to End testing of the application and Fixing of the errors found.
4. Logging and Creation of the project report (this is a continuous process step which starts right at the Design stage).

### VI. DELIVERABLES

#### A. Installable Application

Installers for Linux and windows shall be provided.

#### B. Project Report

The project report in .doc format shall be submitted.

#### C. The work log

The log of the work done during various stages in .doc format.

### VII. MODIFICATIONS DONE TO EKIGA SOURCES FOR THIS CUSTOMIZATION

The source code of Ekiga was studied in detail and it was found that the changes to incorporate GatorPhone.com specific functionality into Ekiga, a total of 3 files had to be changed. Those files are:

1. opal-account.cpp.patch
2. opal-account.h.patch
3. opal-bank.cpp.patch

These files are present in the src/endpoints directory in the Ekiga source directory.

Opal-account.h: Included “GatorPhone” to the enumeration “Type”.

Opal-account.cpp: Added sip.gatorphone.com to be of type SIP.

Opal-bank.cpp:

- 1) Modified/Corrected the keyboard shortcuts for all the listed account types.
- 2) Added a new “action” to include addition of GatorPhone.com accounts.
- 3) Included a new case to handle the scenario when a new Gatorphone.com account gets created. This “case” construct takes care of abstracting the low level details of configuring the GatorPhone.com SIP account.

### VIII. EKIGA BUILD PROCESS FOR LINUX

For the build process to run smoothly, it has to be made sure that the development libraries for features like openLDAP, ALSA are installed. These can be easily installed using the Synaptic Package Manager, if using Ubuntu or any Debian based Linux (Synaptic Package Manager can be manually installed even on non-debian based Linux systems using the distribution's native package management system, for eg., YUM in RedHat/Fedora).

a) Compiling PTLIB:

v4l is disabled by default; if you need it, add `--enable-v4l` to `./configure`.

For Optimized Build:

```
./configure --enable-sunaudio=no --enable-oss=yes --prefix=/usr --disable-sdl --disable-avc --disable-dc --enable-v4l
```

```
make
make install
```

#### b) Compiling OPAL:

Opal compilation needs very much RAM to build the optimized version. *make debug* can be used to not optimize it, though this will make a very big and slow library.

If you have installed only the debug library of ptlib, only the debug version of opal can be build. The same applies to the release version. In case both versions of the ptlib library are installed, both versions of the opal library can be built and installed.

#### For Optimized Build:

```
aptitude install libspeex-dev libx264-dev
libtheora-dev libavcodec-dev libavformat-dev
./configure --prefix=/usr --enable-localspeex
make
make install
```

#### c) Compiling EKIGA:

If you have downloaded Ekiga from SVN repository, then you don't have a `configure` script. You should use `./autogen.sh` instead of `./configure` below (giving the same parameters), and you need complete GNU autotools-support and intltool.

The following options (among others) are recognized:

- `--disable-doc` disables the generation of the documentation files (readable with GNOME yelp). Saves hassle with `gnome-doc-utils`.
- `--disable-gnome` disables entirely GNOME support
  - it still depends on `gnome-doc-utils`, use `--disable-doc` for that
  - it still needs GLib and GTK+ of course...
  - it's experimental and was introduced to make the WIN32-port possible
  - it saves it config AND addressbooks into a FILE (usually `~/config/ekiga.conf` or whatever was reported by GLib)
  - for the moment, expect some weirdness with the UI and some config keys
- `--disable-avahi` disables the generation of zeroconf Avahi code ("Contacts near me")
- `--disable-dbus` disables generation of Ekiga's DBUS interface (yes, Ekiga can be remote-controlled a bit)

Like for opal, ekiga has to be told whether to use the debug or optimized versions of the ptlib and opal library. Thus if you have compiled the debug versions of the libraries, you will have to use debug build of ekiga.

#### For Optimized Build:

```
aptitude install gnome-common libsasl2-dev
gettext libgnome2-dev libldap2-dev libgconf2-
dev autoconf libgnomeui-dev libxv-dev
intltool automake1.8 scrollkeeper libxml-
parser-perl evolution-data-server-dev
```

```
libavahi-common-dev libavahi-client-dev
libavahi-glib-dev gnome-doc-utils libsigc+
+-2.0-dev libdbus-glib-1-dev libebook1.2-dev
./configure --prefix=/usr
make
make install
```

To build the .deb package, issue the command:  
`sudo checkinstall -D --fstrans=no`

#### d) Additional Dependencies:

FFMPEG: FFMPEG (SVN) is required for the following video codecs:

- MPEG 4 Part 2 Video via [RFC 3016](#)
- H.264 via [RFC 3984](#)
- H.263 + (H.263-1998) via [RFC 2429](#) / [RFC 4629](#)

In order to install the latest version of FFMPEG, check out the SVN repository  
`svn co svn://svn.mplayerhq.hu/ffmpeg/trunk`  
`ffmpeg`

afterwards:

```
./configure --enable-shared
make
sudo make install
```

x264: x264 (GIT) is required for the following video codecs:

- H.264 via [RFC 3984](#)

In order to install the latest version of x264 check out the GIT repository

```
git clone git://git.videolan.org/x264.git
```

afterwards:

```
./configure --enable-shared
make
sudo make install
```

#### IX. PROBLEMS ENCOUNTERED WHILE COMPILING FOR LINUX

Downloaded the sources for Ekiga, PTlib and OPAL from [http://www.ekiga.org/index.php?rub=5&path=sources/ekiga\\_3.0.0](http://www.ekiga.org/index.php?rub=5&path=sources/ekiga_3.0.0)

1. While creating the .deb package, ie., using just "sudo checkinstall" results in the following error -

```
/usr/share/icons/hicolor/16x16/apps/ No such
file or directory
```

Using the option "`-D --fstrans=no`" with checkinstall works though:

```
sudo checkinstall -D --fstrans=no
```

## X.EKIGA BUILD PROCESS FOR WINDOWS

The pre-requisites that are needed for cross-compiling Ekiga for Win32 on the Linux System are installed using the following commands:

```
apt-get install mingw32 mingw32-binutils
mingw32-runtime nsis gnome-doc-utils
```

```
aptitude install build-essential make
automake bison flex
```

```
aptitude install zip unzip git-core
subversion
```

```
aptitude install gnome-common libgtk2.0-dev
xmlto libgconf2-dev libavahi-glib-dev
libavahi-client-dev
```

The cross-build on Linux for Win32 needs to produce binaries of every library that Ekiga needs and which are not part of the Windows API. Other than in the native build for Linux, pre-installed Linux binaries of libraries such as OpenLDAP etc. are useless for cross-build for the Win32 target. Therefore, as part of the cross-build process, these dependencies are obtained as sources and cross-compiled for Win32 before the script compiles Ekiga itself against these dependencies.

Get the Ekiga Win32 build script for latest stable:

```
svn co
http://svn.gnome.org/svn/ekiga/branches/gnome-2-24/win32
```

or get the Ekiga Win32 build script for trunk:

```
svn co
http://svn.gnome.org/svn/ekiga/trunk/win32
```

Execute the following command to fetch the dependencies (sources/libraries) from the Internet:

```
make update-sources
```

```
make
```

If everything works out well, the cross-compiled windows version will be located in dist/Ekiga. We can transfer this directory to a win32 system and simply run ekiga.exe.

## XI. PROBLEMS ENCOUNTERED WHILE COMPILING FOR WINDOWS

1. Win32 Makefile does not fetch libXML source:

Error: The Win32 Makefile is not able to fetch the ver.26 of libxml from the default server.

Solutions Tried:

Change the Makefile source fetch server for libxml to <ftp://fr.rpmfind.net/pub/libxml>

This solution solves the problem.

2. Compilation issue with H.264 in Opal:

Error: Undefined references to various H.264 routines while compiling for Win32 using Ekiga sources ver. 3.2.

```
undefined reference to `_x264_nal_encode'
undefined reference to `_x264_encoder_close'
undefined reference to `_x264_encoder_open'
etc.,
```

S

Solutions Tried:

Instead of using the latest version of sources ie., ver. 3.2, I used the last stable version ie., 3.0.0. This version (3.0.0) compiles properly after applying the usual PtimeParse and SvcProc patches.

3. Undefined reference to PtimeParse:

Error: While compiling the win32 version of Ekiga, we get the following error:

```
Undefined reference to `_PtimeParse'
```

Solutions Tried:

In the definition of PtimeParse function in [src/]ptlib/src/ptlib/common/ptime.cxx, modified it to return a value of 0.

This solution solves the problem.

4. Undefined reference to `SvcProc':

Error: While Compiling Ekiga for win32, we get the error - "Undefined references to `SvcProc'.

Solutions Tried:

Modified the Makefile at ptlib/src/ to include \$(PLATFORM\_SRC\_DIR)/svcproc.cxx  
This solution solves the problem.

5. Win32 Console Issue: Ekiga for Windows starts a console window along with its GUI and this console window remains open even after closing the Ekiga GUI. This problem is not fixed yet.

Solutions Tried:

Checked out Ekiga ver 3.0.2 sources using the command-  
svn co [http://svn.gnome.org/svn/ekiga/tags/EKIGA\\_3\\_0\\_2](http://svn.gnome.org/svn/ekiga/tags/EKIGA_3_0_2)

This version is supposed to have the fix for the win32 console issue.

Fixed the svcproc.cxx patch on ptlib src Makefile.

Updated the win32 Makefile with DEBUG=0

```
Ran "make update-sources"
```

```
Ran "make"
```

Build succesful. But even this results in the console window problem.

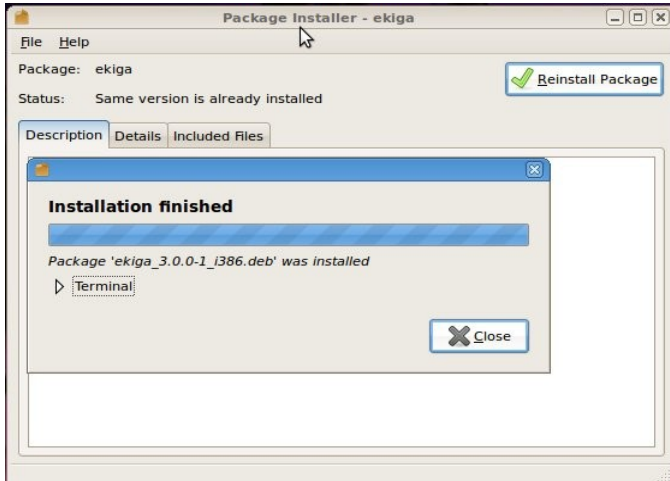
Thus, this solution does not solve the problem. It is hoped that the future versions will have fixes for this issue.

Even sources of Ver.3.0.3 and Ver.3.2 do not solve this issue.

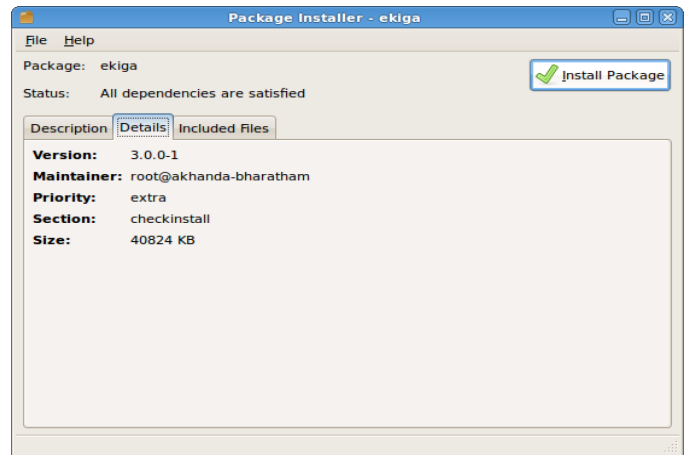
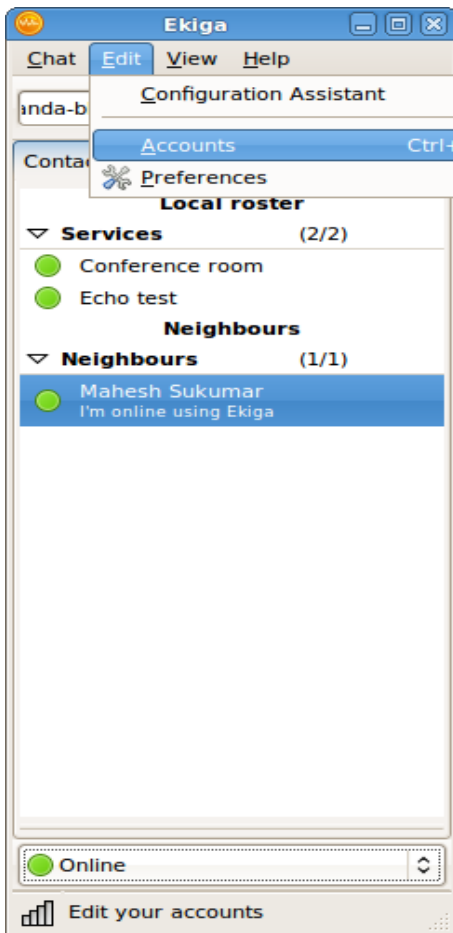


## XII.INSTALLER, SETUP AND CONFIGURATION FOR LINUX

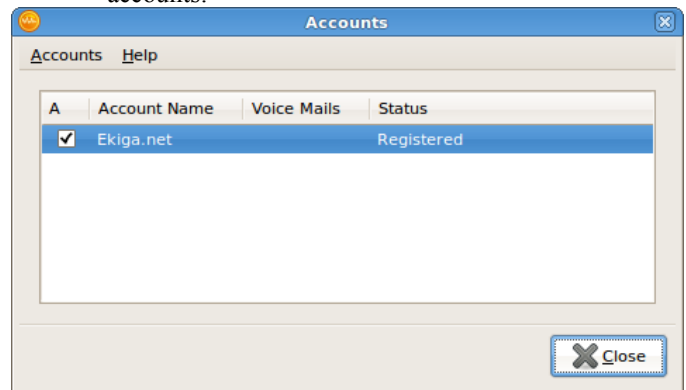
1. Run the .deb package by double clicking on it.
2. Click on “close” once the installation is done.



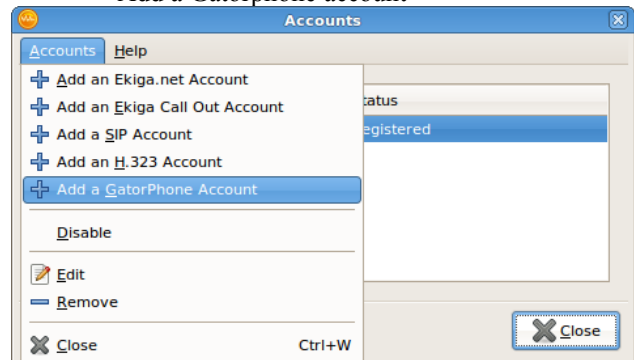
3. The Ekiga Main Window with Accounts Menu. (Ekiga can be started by either typing Alt+F2 → ekiga “or” by clicking on the Ekiga menu from the main “Applications” menu → Internet → Ekiga.



4. Clicking on “Accounts” Menu will open up the following window which will list the registered accounts.



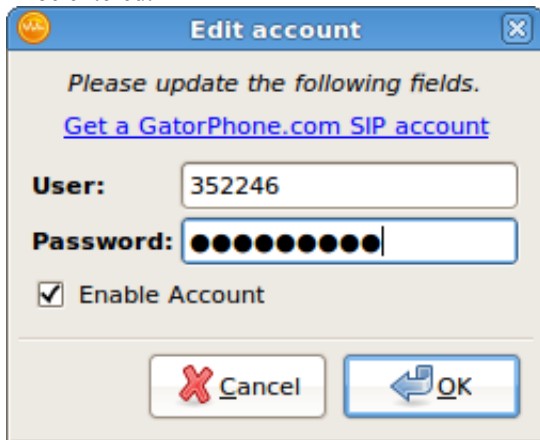
5. To register for GatorPhone account, click “Accounts → Add a Gatorphone account”



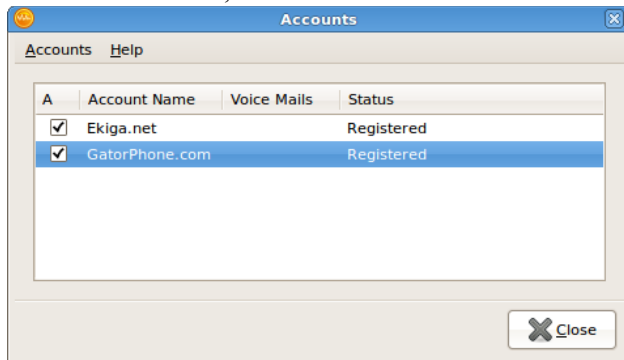
6. Enter the Gatorphone.com user-id and password. If one does not already have a Gatorphone.com account, this window also has a link item which when clicked opens up the Gatorphone.com web site on the default web browser.



7. Now the Gatorphone.com user-id and password can be entered:

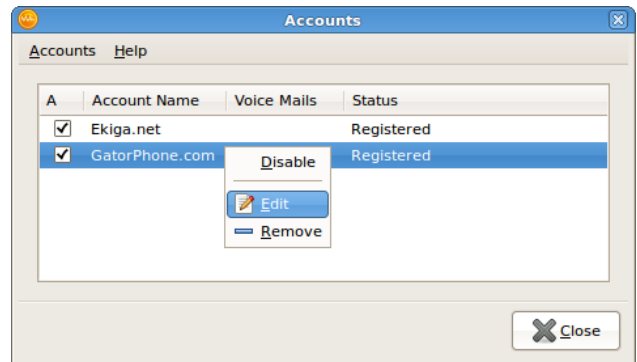


8. After entering the account details and clicking on the "OK" button, we will find that the

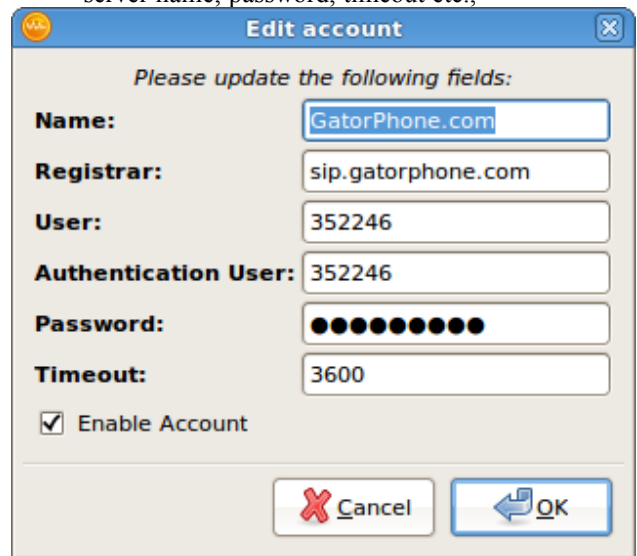


Gatorphone.com account is listed in the Accounts window:

9. After clicking on "Close", we can either start using Gatorphone with Ekiga or if we want to make any changes to the account that we just now setup, we can do that by right-clicking on the account and clicking on "Edit".



10. This opens up the "Edit Account" window in which we can modify the parameters such as user-id, SIP server name, password, timeout etc.,



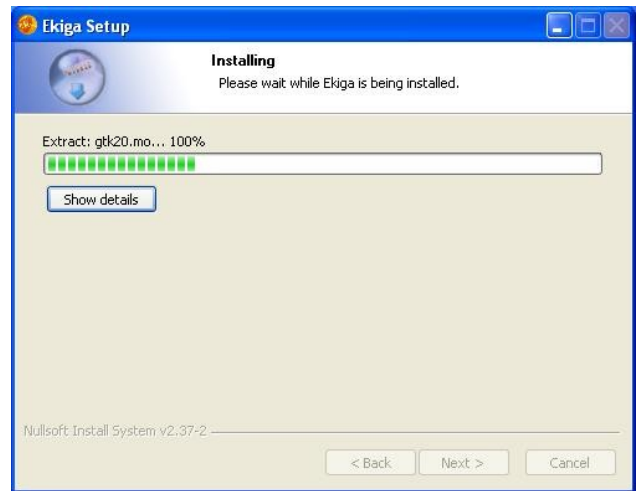
### XIII.INSTALLER, SETUP AND CONFIGURATION FOR WINDOWS

1. Double click on the installer to get the language selection window:

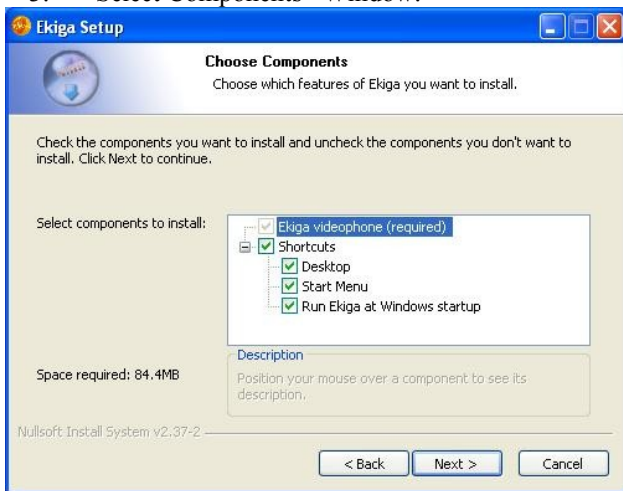


2. The Welcome Window:





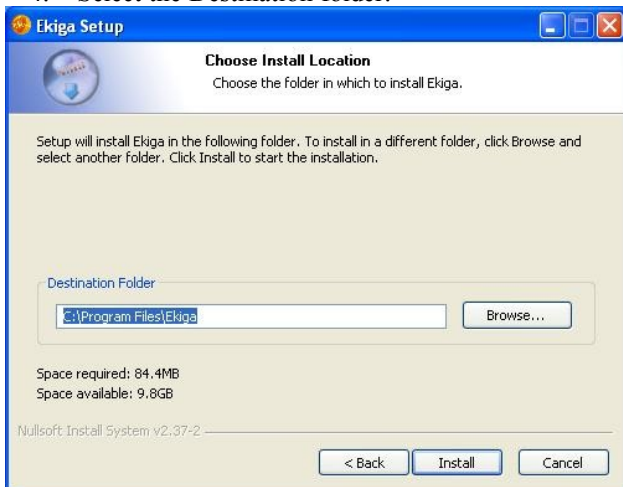
### 3. "Select Components" Window:



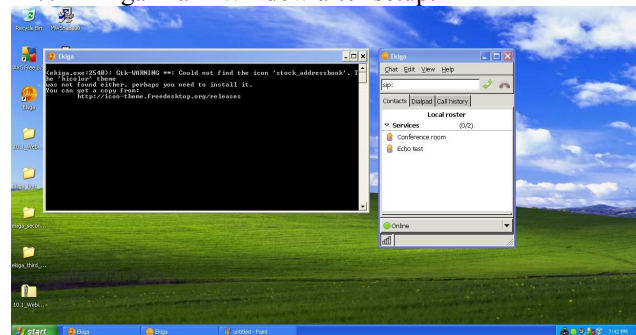
### 6. "Installation Complete" Window:



### 4. Select the Destination folder:



### 7. Ekiga Main Window after setup:



The steps for Configuration of Ekiga in Windows are the same as that for LINUX.

### 5. Installation progress window:

## XIV.PATCHES FOR GATORPHONE

The patches that were developed for including Gatorphone related functionality in Ekiga are three in number:

4. opal-account.cpp.patch
5. opal-account.h.patch
6. opal-bank.cpp.patch

These patches can be applied to all future versions of Ekiga sources by using the following commands:

```
patch -p0 < opal-account.cpp.patch
patch -p0 < opal-account.h.patch
patch -p0 < opal-bank.cpp.patch
```

in the folder ekiga/src/endpoints.

## APPENDIX

1. ALSA – Advanced Linux Sound Architecture
2. Bonjour – Apple's Implementation of ZeroConf.
3. DBUS – Desktop BUS.
4. dep. – Dependency (Software).
5. DTMF – Dual Tone Multi Frequency
6. Evolution – Open Source Email Client from Novell Inc.
7. EXE – Executable file.
8. FLEX – Fast Lexical Analyser Generator
9. G.711 - G.711 is an ITU-T standard for audio companding. It is primarily used in telephony.
10. G.729 - G.729 is an audio data compression algorithm for voice that compresses voice audio in packets of 10 milliseconds duration.
11. Glade – User Interface designer for GTK+ and GNOME.
12. GLIB – Gnome LIBrary.
13. GNOME – GNU Object Model Environment
14. GNU – GNU's Not Unix
15. GTK+ – The GIMP Tool Kit
16. GIMP – GNU Image Manipulation Program.
17. GSM – Global System for Mobile Communications.
18. H.235 -H.235 covers security and encryption for H.323 and other H.245 based terminals.
19. H.245 -It is a control channel protocol used with[in] e.g. H.323 and H.324 communication sessions, and involves the line transmission of non-telephone signals. It also offers the possibility to be tunneled within H.225.0 call signaling messages. This eases firewall traversing.
20. H.261 -It is a 1990 ITU-T video coding standard originally designed for transmission over ISDN lines on which data rates are multiples of 64 kbit/s.
21. H.263+ - It is a video codec standard originally designed as a low-bitrate compressed format for videoconferencing.
22. H.264 - H.264 is a standard for video compression, and is equivalent to MPEG-4 Part 10, or MPEG-4 AVC (for Advanced Video Coding).
23. H.323 - The H.323 standard addresses call signaling and control, multimedia transport and control, and bandwidth control for point-to-point and multi-point conferences.
24. IAX2 – It is the Inter-Asterisk eXchange protocol native to Asterisk PBX and supported by a number of other softswitches and PBXs. It is used to enable VoIP connections between servers as well as client-server communication.
25. IDE – Integrated Development Environment.
26. ILBC – It is a royalty free[1] narrowband speech codec, developed by Global IP Solutions (GIPS) formerly Global IP Sound.
27. IP – Internet Protocol.
28. MPEG4 – Moving Picture Experts Group - 4 is a collection of methods defining compression of audio and visual (AV) digital data.
29. MPL – Mozilla Public License.
30. MS-GSM – It s a Microsoft version of the ETSI GSM 06.10 Full Rate (FR) speech codec. MS-GSM is compatible with GSM 6.10 but uses Microsoft frame packing.
31. NAT – Network Address Translation.
32. NSIS - Nullsoft Scriptable Install Systems
33. OpenLDAP – Open Light-weight Directory Access Protocol
34. OSS – Open Source Software.
35. OSS – Open Sound System.
36. RFC – Request For Comments.
37. RTP – Real-time Transport Protocol.
38. SDL – Simple Direct-Media Layer
39. SIMPLE - the *Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions*, is an instant messaging (IM) and presence protocol suite based on Session Initiation Protocol (SIP) managed by the IETF.
40. Sigc++ - libsigc++ implements a typesafe callback system for standard C++. It allows you to define signals and to connect those signals to any callback function, either global or a member function, regardless of whether it is static or virtual.
41. SIP – Session Initiation Protocol
42. SIP PIDF – SIP Presence Information Data Format.
43. Speex – It is a free software speech codec that may be used on VoIP applications and podcasts. Speex claims to be free of any patent restrictions and is licensed under the revised (3-clause) BSD license. It may be used with the Ogg container format or directly transmitted over UDP/RTP.
44. STUN - Simple Traversal of User Datagram Protocol through Network Address Translators (NATs) is a standards-based set of methods and a network protocol used in NAT traversal for applications of real-time voice, video, messaging, and other interactive IP communications. In the original specification in RFC 3489
45. Theora – It is an open and royalty-free lossy video compression technology being developed by the Xiph.Org Foundation as part of their Ogg project.
46. UI – User Interface.

47. URL – Uniform Resource Locator
48. Video4Linux - or V4L is a video capture application programming interface for Linux. Several USB webcams, TV tuners, and other devices are supported. Video4Linux is closely integrated with the Linux kernel.
49. win32 – Windows 32 Bit.
50. Xvideo - The X video extension, often abbreviated as XVideo or Xv, is a video output mechanism for the X Window System.
51. ZeroConf - Zero Configuration Networking (Zeroconf), is a set of techniques that automatically creates a usable IP network without configuration or special servers. This allows inexperienced users to connect computers, networked printers, and other network devices and expect a functioning network to be established automatically.

#### ACKNOWLEDGMENT

Mahesh Sukumar thanks Damien Sandras <dsandras\_at\_seconix.com>, who is the initial creator of the Ekiga Project and others who are/were associated with the Ekiga Project.

#### REFERENCES

- [1] GatorPhone - <http://www.gatorphone.com>
- [2] IPKall - <http://www.ipkall.com>
- [3] Compiling Ekiga on Linux - [http://wiki.ekiga.org/index.php/Compiling\\_Ekiga](http://wiki.ekiga.org/index.php/Compiling_Ekiga)
- [4] Cross Compiling Ekiga on Linux for Windows - [http://wiki.ekiga.org/index.php/Cross-compile\\_Win32\\_3.0](http://wiki.ekiga.org/index.php/Cross-compile_Win32_3.0)
- [5] Ekiga win32 console issue - <http://osdir.com/ml/ekiga-list/2009-04/msg00036.html>
- [6] Ekiga builds for Windows users - [http://wiki.ekiga.org/index.php/Windows\\_Users](http://wiki.ekiga.org/index.php/Windows_Users)
- [7] X-Lite - <http://www.counterpath.net/X-Lite-Download.html>
- [8] YATE - Yet Another Telephony Engine. <http://yate.null.ro/pmwiki/>
- [9] MINISIP - <http://www.minisip.org/>
- [10] LinPhone - <http://www.linphone.org/>
- [11] ptlib and opal - <http://www.opalvoip.org/>
- [12] x264 – free library for encoding H264/AVC video streams - <http://www.videolan.org/developers/x264.html>
- [13] GIT version control System - <http://git-scm.com/>
- [14] Subversion Version control system - [http://en.wikipedia.org/wiki/Subversion\\_\(software\)](http://en.wikipedia.org/wiki/Subversion_(software))
- [15] MINGW – Minimalistic GNU for Windows - <http://www.mingw.org/>
- [16] GNU Make - <http://www.gnu.org/software/make/>
- [17] libxml – The XML Library for GNOME - <http://xmlsoft.org/>
- [18] checkinstall - <http://asic-linux.com.mx/~izto/checkinstall/>
- [19] GTK+ 2.0 Tutorial - <http://library.gnome.org/devel/gtk-tutorial/stable/>
- [20] GTK Documentation - <http://www.gtk.org/documentation.html>
- [21] Building GTK apps for Windows on Linux - <http://ricardo.ecn.wfu.edu/~cottrell/cross-gtk/>
- [22] Patches for building the win32 version - <http://www.mail-archive.com/ekiga-devel-list@gnome.org/msg02325.html>
- [23] Ffmpeg - <http://www.ffmpeg.org/>
- [24] EXPAT - <http://expat.sourceforge.net/>
- [25] GNOME - <http://library.gnome.org/devel/>
- [26] NSIS Installer- [http://nsis.sourceforge.net/Main\\_Page](http://nsis.sourceforge.net/Main_Page)
- [27] GNU - <http://www.gnu.org/philosophy>
- [28] SIP Protocol - <http://www.ietf.org/rfc/rfc3261.txt>
- [29] Damien's blog - <http://blog.ekiga.net/>
- [30] Ubuntu - <http://www.ubuntu.com/>
- [31] Kubuntu - <http://www.kubuntu.org/>
- [32] GNU Configure - <http://www.airs.com/ian/configure/>
- [33] GCC Configure - <http://gcc.gnu.org/install/configure.html>
- [34] GNU Patch - <http://www.gnu.org/software/patch/>
- [35] GNU Diff - <http://www.gnu.org/software/diffutils/>
- [36] VirtualBox virtualization Software - <http://www.virtualbox.org/>
- [37] DBUS – Message Bus System - <http://dbus.freedesktop.org/>

**Mahesh Sukumar** joined the University of Florida as a Masters student in September 2008 in the Electrical and Computer Engineering Department.

Before that, From September 2004 till August 2008, he worked as a Software Engineer with Wipro Technologies, India.

From October 2008 to 'till date' he works part-time as a Computer Technician at the Muscle Physiology Laboratory at the University of Florida.