Ri.

NAME
pidgin - Instant Messaging client

SYNOPSIS
pidgin *[options]*

DESCRIPTION
pidgin is a graphical modular messaging client based on libpurple which is capable of connecting to XMPP, IRC, SILC, Novell GroupWise, Lotus Sametime, Zephyr, Gadu-Gadu, and QQ all at once. It has many common features found in other clients, as well as many unique features.

Pidgin can be extended by plugins written in multiple programming languages and controlled through DBus or **purple-remote**.

OPTIONS
The following options are provided by Pidgin using the standard GNU command line syntax:

-c, --config=*DIR*
> Use *DIR* as the directory for config files instead of *~/.purple*.

-d, --debug
> Print debugging messages to stdout. These are the same debugging messages that are displayed in the **Debug Window**.

-f, --force-online
> Try to be online even if the network is reported (by Windows, or NetworkManager on Linux) to be unavailable.

-h, --help
> Print a summary of command line options and exit.

-m, --multiple
> Allow multiple instances of Pidgin to run.

-n, --nologin
> Don't automatically login when Pidgin starts. Sets the global status to **Offline**.

-l, --login[=*NAME*,*NAME*,...]

Enable the comma-separated list of accounts provided, disabling all other accounts. If the user does not specify such a comma-separated list, the first account in accounts.xml will be enabled.

-v, --version

Print the current version and exit.

TERMS

Pidgin uses a few terms differently from other applications. For convenience they are defined here:

Buddy List

The list of other users who the user wants to see status information for and have quick access to for messaging.

Buddy

A user who has been added to the Buddy List.

Contact

A grouping of more than one buddy who are all the same person. A contact may contain buddies from any protocol and may contain as many buddies as the user desires. Contact arrangements are stored locally only.

Alias

A private "nickname" that may be set for Buddies or the user himself. On some protocols, aliases are saved on the server but not visible to other users. On other protocols, aliases are saved only locally.

Protocol

A messaging service. XMPP, Zephyr, etc. are protocols. Others may call these "service types," "account types," "services," and so on.

BUDDY LIST

The **Buddy List** window is Pidgin's main interface window. Using this window a user can see which of his/her buddies is online, away, idle, etc. The user can also add buddies to and remove buddies from the buddy list.

The **Buddy List** window contains a list of the user's buddies who are online and have allowed the user to be notified of their presence. The icon to the left of each buddy indicates the buddy's current status.

Double clicking a buddy will open a new **Conversation** window. Right clicking will pop up a menu:

Get Info
  Retrieves and displays information about the buddy. This information is also known as a Profile.

IM Opens a new **Conversation** window to the selected buddy.

Send File
  Sends a file to the selected buddy (only available on protocols that support file transfer).

Add Buddy Pounce
  A Buddy Pounce is a configurable automated action to be performed when the buddy's state
  changes. This will open the **Buddy Pounce** dialog, which will be discussed later.

View Log
  Pidgin is capable of automatically logging messages. These logs are either plain text files (with a
  .txt extension) or html files (with a .html extension) located under the *~/.purple/logs* directory.
  This menu command will display Pidgin's log viewer with logs loaded for that buddy or chat.

Alias
  Create an alias for this buddy. This will show an editable text field where the buddy's name was
  displayed. In this field one can give this buddy an alternate, more friendly name to appear on the
  buddy list and in conversations.

  For example, if a buddy's name was jsmith1281xx and his real name was 'John Q. Smith,' one
  could create an alias as to identify the buddy by his common name.

The remainder of the menu will consist of protocol specific commands. These commands vary
depending on the protocol.

Status Selector
  At the bottom of the **Buddy List** is a status selector which allows one to change his/her status. This
  will be discussed further in the **STATUS MESSAGES** section below.

ACCOUNT EDITOR
  The account editor consists of a list of accounts and information about them. It can be accessed by
  selecting **Manage** from the Accounts menu. Clicking *Delete* will delete the currently selected account.
  Clicking *Add* or *Modify* will invoke a **Modify Account** window. Here, the user can add or alter
  account information. When creating a new account, the user will submit a username and password.

The user will also choose the protocol for the account.

If *Remember Password* is chosen, the password will be saved in Pidgin's *~/.purple/accounts.xml* configuration file.

If *Enabled* is checked in the accounts dialog, this account will follow the status currently selected in the status selector. If it is not checked, the account will always be offline.

Each protocol has its own specific options that can be found in the modify screen.

PREFERENCES
All options take effect immediately.

Interface
Show system tray icon
Specifies when to show a Pidgin icon in the notification area of the user's panel (commonly referred to as the System Tray).

Hide new IM conversations
Specifies when to hide new IM messages. Messages will queue under the specified condition until shown. Clicking the Pidgin icon in the notification area or system tray will display the queued messages. An icon also appears in the buddy list's menu bar; this icon may also be used to display queued messages.

Show IMs and chats in tabbed windows
When checked, this option will cause IM and chat sessions to appear in windows with multiple tabs. One tab will represent one conversation or chat. Where tabs are placed will be dictated by the preferences below.

Show close buttons on tabs
When checked, this option will cause a clickable "U+2715 MULTIPLICATION X" unicode character to appear at the right edge of each tab. Clicking this will cause the tab to be closed.

Placement

Specifies where to place tabs in the window. Some tab orientations may allow some users to fit more tabs into a single window comfortably.

New conversations

Specifies under which conditions tabs are placed into existing windows or into new windows. For a single window, select *Last created window* here.

Conversations

Enable buddy icon animation

If a buddy's icon happens to be animated, this option will enable the animation, otherwise only the first frame will be displayed.

Notify buddies that you are typing to them

Some protocols allow clients to tell their buddies when they are typing. This option enables this feature for protocols that supports it.

Default Formatting

Allows specifying the default formatting to apply to all outgoing messages (only applicable to protocols that support formatting in messages).

Smiley Themes

Allows the user to choose between different smiley themes. The "none" theme will disable graphical emoticons - they will be displayed as text instead. The **Add** and **Remove** buttons may be used to install or uninstall smiley themes. Themes may also be installed by dragging and dropping them onto the list of themes.

Sounds

Method

Lets the user choose between different playback methods. The user can also manually enter a command to be executed when a sound is to be played(*%s* expands to the full path to the file name).

Sounds when conversation has focus

When checked, sounds will play for events in the active conversation if the window is focused. When unchecked, sounds will not play for the active conversation when the window is focused.

Enable Sounds
> Determines when to play sounds.

Sound Events
> Lets the user choose when and what sounds are to be played.

Network

> **STUN server**
> > This allows specifying a server which uses the STUN protocol to determine a host's public IP address. This can be particularly useful for some protocols.

> **Autodetect IP address**
> > When checked, causes Pidign to attempt to determine the public IP address of the host on which Pidgin is running and disables the **Public IP** text field listed below.

> **Public IP**
> > If **Autodetect IP address** is disabled, this field allows manually specifying the public IP address for the host on which Pidgin is running. This is mainly useful for users with multiple network interfaces or behind NATs.

> **Manually specify range of ports to listen on**
> > Specify a range ports to listen on, overriding any defaults. This is sometimes useful for file transfers and Direct IM.

> **Proxy Server**
> > The configuration section to enable Pidgin to operate through a proxy server. Pidgin currently supports SOCKS 4/5 and HTTP proxies.

Browser

Browser

Allows the user to select Pidgin's default web browser. Firefox, Galeon, Konqueror, Mozilla, Netscape and Opera are supported natively. The user can also manually enter a command to be executed when a link is clicked (*%s* expands to the URL). For example, *xterm -e lynx "%s"* will open the link with lynx.

Open link in

Allows the user to specify whether to use an existing window, a new tab, a new window, or to let the browser to decide what to do when calling the browser to open a link. Which options are available will depend on which browser is selected.

Logging

Log format

Specifies how to log. Pidgin supports HTML and plain text, but plugins can provide other logging methods.

Log all instant messages

When enabled, all IM conversations are logged. This can be overridden on a per-conversation basis in the conversation window.

Log all chats

When enabled, all chat conversations are logged. This can be overridden on a per-conversation basis in the conversation window.

Log all status changes to system log

When enabled, status changes are logged.

Status / Idle

Report idle time

Determines under which conditions to report idle time. **Based on keyboard and mouse use** uses keyboard and mouse activity to determine idle time. **From last sent message** uses the time at which the user last sent a message in Pidgin to determine idle. **Never** disables idle reporting.

Change status when idle
> When enabled, this uses the **Minutes before becoming idle** and **Change status to** preferences described below to set status on idle.

Minutes before becoming idle
> Specifies how many minutes of inactivity are required before considering the user to be idle.

Change status to
> Specifies which "primitive" or "saved" status to use when setting status on idle.

Use status from last exit at startup
> If this is checked, Pidgin will remember what status was active when the user closed Pidgin and restore it at the next run. When disabled, Pidgin will always set the status selected in **Status to apply at startup** at startup.

Status to apply at startup
> When **Use status from last exit at startup** is disabled, this specifies which "primitive" or "saved" status to use at startup.

CONVERSATIONS
When starting a new conversation, the user is presented with the **Conversation** window. The conversation appears in the upper text box and the user types his/her message in the lower text box. Between the two is a row of options, represented by icons. Some or all buttons may not be active if the protocol does not support the specific formatting. From left to right:

Font
> This menu provides font control options for the current conversation. Size, style, and face may be configured here.

Insert
> This menu provides the ability to insert images, horizontal rules, and links where the protocol supports each of these features.

Smile!
> Allows the insertion of graphical smileys via the mouse. This button shows the user a dialog with

the available smileys for the current conversation.

CHATS

For protocols that allow it, **Chats** can be entered through the *Buddies* menu.

Additional features available in chat, depending on the protocol are:

Whisper

The text will appear in the chat conversation, but it will only be visible to the sender and the receiver.

Invite

Invite other people to join the chat room.

Ignore

Ignore anything said by the chosen person

Set Topic

Set the topic of the chat room. This is usually a brief sentence describing the nature of the chat--an explanation of the chat room's name.

Private Message (IM)

Send a message to a specific person in the chat. Messages sent this way will not appear in the chat window, but instead open a new IM conversation.

STATUS MESSAGES

Most protocols allow for status messages. By using status messages, a user can leave an informative message for others to see. Status and status messages are configured via the status selector at the bottom of the Buddy List window. By default the menu shown here is divided into sections for "primitive" status types, such as *Available*, *Away*, etc.; a few "popular" statuses (including "transient" statuses) which have been recently used, and a section which shows **New Status...** and **Saved Statuses...** options for more advanced status manipulation.

Primitive Statuses

A primitive status is a basic status supported by the protocol. Examples of primitive statuses would be Available, Away, Invisible, etc. A primitive status can be used to create a **Transient Status or a Saved Status**, both explained below. Essentially, primitive statuses are building blocks

of more complicated statuses.

Transient Statuses

When one of the statuses from the topmost section of the status selector's menu is selected, this creates a transient, or temporary, status. The status will show in the "popular statuses" section in the menu until it has not been used for a sufficiently long time. A transient status may also be created by selecting *New Status...* from the status selector's menu, then clicking *Use* once the user has entered the message.

Saved Statuses

Saved statuses are permanent--once created, they will exist until deleted. Saved statuses are useful for statuses and status messages that will be used on a regular basis. They are also useful for creating complex statuses in which some accounts should always have a different status from others. For example, one might wish to create a status called "Sleeping" that has all accounts set to "Away", then create another status called "Working" that has three accounts set to "Away" and another account set to "Available."

New Status Window

When the user selects *New Status...* from the status selector menu, Pidgin presents the user with a dialog asking for status-related information. That information is discussed below:

Title - The name of the status that will appear in the status selctor's menu. If the user clicks the *Save* or *Save & Use* button, this name will also be shown in the **Saved Status Window**. The title should be a short description of the status.

Status - The type of status being created, such as Available, Away, etc.

Message - The content of the status message. This is what is visible to other users. Some protocols will allow formatting in some status messages; where formatting is not supported it will be stripped to the bare text entered.

Use a different status for some accounts - This allows the creation of complex statuses in which some accounts' status differs from that of other accounts. To use this, the user will click the expander to the left of the text, then select individual accounts which will have a different status and/or status message. When the user selects an account, Pidgin will present another status dialog asking for a status and a message just for the selected account.

Saved Status Window

When the user selects *Saved Statuses...* from the status selector's menu, Pidgin presents a dialog that lists all saved statuses. "Transient" statuses, discussed above, are *NOT* shown here. This window provides the ability to manage saved statuses by allowing the creation, modification, and deletion of saved statuses. The *Use*, *Modify*, and *Delete* buttons here allow operation on the status selected from the list; the dd button allows creation of a new saved status, and the *Close* button closes the window.

BUDDY POUNCE

A Buddy Pounce is an automated trigger that occurs when a buddy returns to a normal state from an away state. The **Buddy Pounce** dialog box can be activated by selecting the *Buddy Pounce* option from the **Tools** menu. From this dialog, new pounces can be created with the **Add** button and existing pounces can be removed with the **Delete** button. A pounce can be set to occur on any combination of the events listed, and any combination of actions can result. If *Pounce only when my status is not Available* is checked, the pounce will occur only if the user is set to a non-available status, such as invisible, do not disturb, away, etc. If *Recurring* is checked, the pounce will remain until removed by the **Delete** button.

CUSTOM SMILIES

Pidgin 2.5.0 introduced support for custom smilies on those protocols for which interested contributors have developed support. The custom smiley manager can be accessed by selecting *Smiley* from the *Tools* menu. From here, custom smilies may be added, edited, or deleted by clicking the *Add*, *Edit*, or *Delete* buttons, respectively.

During a conversation with another user, that user's custom smileys may be added to the user's own custom smiley list directly from the conversation window by right-clicking the new custom smiley and selecting *Add Custom Smiley...*

PLUGINS

Pidgin allows for dynamic loading of plugins to add extra functionality to Pidgin. See *plugins/HOWTO* or *http://developer.pidgin.im/wiki/CHowTo* for information on writing plugins.

The plugins dialog can be accessed by selecting *Plugins* from the *Tools* menu. Each plugin available appears in this dialog with its name, version, and a short summary of its functionality. Plugins can be enabled with the checkbox beside the name and short description. More information on the currently selected plugin is available by clicking the expander beside the text *Plugin Details*. If the selected plugin has preferences or configuration options, the *Configure Plugin* button will present the plugin's

preferences dialog.

PERL

Pidgin allows for plugins to be written in the perl scripting language. See *Perl Scripting HOWTO* in the Pidgin documentation for more information about perl scripting.

TCL

Pidgin allows for plugins to be written in the Tcl scripting language. See *plugins/tcl/TCL-HOWTO* for more information about Tcl scripting.

D-Bus

Pidgin allows for interaction via D-Bus. Currently very little documentation about this interaction exists.

FILES

/usr/local/bin/pidgin: Pidgin's location.
~/.purple/blist.xml: the buddy list.
~/.purple/accounts.xml: information about the user's accounts.
~/.purple/pounces.xml: stores the user's buddy pounces.
~/.purple/prefs.xml: Pidgin's configuration file.
~/.purple/status.xml: stores the user's away messages.
~/.purple/logs/PROTOCOL/ACCOUNT/BUDDYNAME/DATE.{html,txt}: conversation logs.

DIRECTORIES

/usr/local/lib/pidgin/: Pidgin's plugins directory.
/usr/local/lib/purple-2/: libpurple's plugins directory.
~/.purple: users' local settings
~/.purple/plugins/: users' local plugins

BUGS

The bug tracker can be reached by visiting *http://developer.pidgin.im/query*

Before sending a bug report, please verify that you have the latest version of Pidgin. Many bugs (major and minor) are fixed at each release, and if yours is out of date, the problem may already have

been solved.

PATCHES

If you fix a bug in Pidgin (or otherwise enhance it), please submit a patch (using **mtn diff > my.diff** against the latest version from the Monotone repository) at *http://developer.pidgin.im/simpleticket*

You are also encouraged to drop by at **#pidgin** on *irc.libera.chat* to discuss development.

SEE ALSO

http://pidgin.im/
http://developer.pidgin.im/
purple-remote(1)
finch(1)

LICENSE

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but **WITHOUT ANY WARRANTY**; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02111-1301 USA

AUTHORS

Pidgin's active developers are:
  Daniel 'datallah' Atallah (developer)
  Paul 'darkrain42' Aurich (developer)
  John 'rekkanoryo' Bailey (developer and bugmaster)
  Ethan 'Paco-Paco' Blanton (developer)
  Thomas Butter (developer)
  Ka-Hing Cheung (developer)

   Sadrul Habib Chowdhury (developer)
   Mark 'KingAnt' Doliner (developer) *<thekingant@users.sourceforge.net>*
   Sean Egan (developer) *<seanegan@gmail.com>*
   Casey Harkins (developer)
   Ivan Komarov
   Gary 'grim' Kramlich (developer)
   Richard 'rlaager' Laager (developer) *<rlaager@pidgin.im>*
   Sulabh 'sulabh_m' Mahajan (developer)
   Richard 'wabz' Nelson (developer)
   Christopher 'siege' O'Brien (developer)
   Bartosz Oler (developer)
   Etan 'deryni' Reisner (developer)
   Tim 'marv' Ringenbach (developer) *<marv_sf@users.sf.net>*
   Michael 'Maiku' Ruprecht (developer, voice and video)
   Elliott 'QuLogic' Sales de Andrade (developer)
   Luke 'LSchiere' Schierer (support)
   Megan 'Cae' Schneider (support/QA)
   Evan Schoenberg (developer)
   Kevin 'SimGuy' Stange (developer and webmaster)
   Will 'resiak' Thompson (developer)
   Stu 'nosnilmot' Tomlinson (developer)
   Nathan 'faceprint' Walp (developer)

  Our crazy patch writers include:
   Marcus 'malu' Lundblad
   Dennis 'EvilDennisR' Ristuccia
   Peter 'fmoo' Ruibal
   Gabriel 'Nix' Schulhof
   Jorge 'Masca' Villaseñor

  Our artists are:
   Hylke Bons *<h.bons@student.rug.nl>*

  Our retired developers are:
   Herman Bloggs (win32 port) *<herman@bluedigits.com>*
   Jim Duchek *<jim@linuxpimps.com>* (maintainer)
   Rob Flynn *<gaim@robflynn.com>* (maintainer)

Adam Fritzler (libfaim maintainer)
Christian 'ChipX86' Hammond (developer & webmaster) *<chipx86@chipx86.com>*
Syd Logan (hacker and designated driver [lazy bum])
Jim Seymour (XMPP developer)
Mark Spencer (original author) *<markster@marko.net>*
Eric Warmenhoven (former lead developer) *<eric@warmenhoven.org>*

Our retired crazy patch writers include:
Felipe 'shx' Contreras
Decklin Foster
Peter 'Bleeter' Lawler
Robert 'Robot101' McQueen
Benjamin Miller

This manpage was originally written by Dennis Ristuccia *<dennis@dennisr.net>*. It has been updated
and largely rewritten by Sean Egan *<seanegan@gmail.com>*, Ben Tegarden
<tegarden@uclink.berkeley.edu>, and John Bailey *<rekkanoryo@pidgin.im>*.