Zephyr on Athena

Zephyr on Athena

On this page:

- Overview
- Dealing With Harassing Zephyr Messages
- Zephyr concepts
- Specifying Classes, Instances and Recipients
  - Using * to specify instance or recipient
  - Written conventions
- Using Zephyr
  - Sending Messages (zwrite)
  - Formatting Text in Zephyr Messages
  - Locating Other Users (zlocate and znol)
  - Login Notification (znol)
  - Subscribing to Zephyr Messages (zctl)
    - Adding subscriptions for the duration of your login
    - Adding Subscriptions Permanently Until You Remove Them
  - Removing Subscriptions
  - Common Zephyr Subscriptions
  - Setting Zephyr Exposure (zctl)
  - Sending Automatic Replies With zaway
  - Sending Automatic Messages to Yourself (zleave)
  - Using Zephyr Over Dialup
  - Copying the Text of a Zephyr Message
- Customizing Zephyr
  - Setting Your Zephyr Signature
  - Changing the Default Font for Zephyr Messages
  - Filtering Out Unwanted Zephyr Messages
  - Filtering messages by adding a line to your ~/.zephyr.subs file
  - Positioning Zephyr Windowgrams on Your Screen
- Other Sources of Information About Zephyr
- Related Links

Overview

Zephyr is a communication service for sending and receiving messages interactively. It's faster than an email system; messages are delivered instantaneously, however you can converse only with other users who are logged on and running Zephyr at the same time you are.

Zephyr is a protocol, and there are many Zephyr clients. The default client on Athena workstations is the Zephyr Windowgram Client (zwgc), and that client will be covered in this document. Another popular client is BarnOwl. For more information on getting started with BarnOwl, please see the SIPB document Using Zephyr.

This document makes use of the command line. If you are unfamiliar with the command line or Terminal window, you may wish to read Working with the Command Line first.

Dealing With Harassing Zephyr Messages

If someone is using Zephyr to harass you, send email to stopit@mit.edu. If you believe you are in danger, call the Campus Police immediately at x3-1212.

Zephyr concepts

Zephyr allows users to conduct one-on-one and group conversations and to receive notices, e.g., notification of incoming email. Zephyr makes use of an identifying sequence for users to specify with whom they want to converse, or what kinds of notices they want to receive. The identifying sequence is made up of three components: class, instance and recipient.

- **Class**: Used to specify a subject or message type. There are also other standard classes in use which you can find out about in Common Zephyr Subscriptions.
- **Instance**: A subdivision of class that narrows down the subject or type. There are also other standard instances in use which you can find

out about in Common Zephyr Subscriptions.
- **Recipient**: The user or users to whom the message is sent. You can specify either a Kerberos username for an individual, or "*" to specify everyone who has specified receipt of messages sent to the indicated class and recipient.

To carry on conversations with Zephyr, you need to tell the zephyr server what type of messages you want to receive. This is called "subscribing", and each subscription is a combination of class, message and recipient (known as a "triplet"). By default, you are subscribed to any messages sent directly to you. Such personal messages are actually a triplet: class "message", instance "personal", recipient `joeuser@ATHENA.MIT.EDU`.

Specifying Classes, Instances and Recipients

In many cases, you don't need to specify all three components, i.e., class, instance and recipient, when sending or subcribing to messages because Zephyr uses a number of defaults. It's useful to understand the system of defaults so you can make sure that you're sending and/or subscribing to the right messages. The following is a list of the defaults:

- If you don't specify class, Zephyr will default to the class called **message**.
- If you don't specify instance, Zephyr will default to the instance called **personal**.
- If you don't specify recipient, Zephyr will default to the recipient called *****.

Using * to specify instance or recipient

It is possible to specify "*" for instance or recipient. The character has different meanings depending on how it is used:

- **Using * for an instance:** This indicates all instances that are used on the class that you specify. For example, if you subscribe to foo * *, you would receive all messages sent to foo and any instance that is specified with it.
- **Using * for recipient:** This indicates anything sent to the class and instance specified that does not have a recipient specified. For instance, if you subscribe to foo * *, you'd receive anything sent to foo, unless it had a recipient specified, such as "foo * joeuser".

Written conventions

Around campus, you might see the shorthand notation "-c sipb" or "-i white-magic" to describe the zephyr class "sipb" or the zephyr instance "white-magic". The shorthand comes from the arguments to the zwrite command (described) below. If no class is specified (in the case of "white-magic"), the class is assumed to be "message".

Using Zephyr

As noted earlier, this document covers the Zephyr Windowgram Client (zwgc). There are three primary commands you'll use when sending and receiving messages using zwgc:

- **zwgc:** This program displays incoming Zephyr messages in what are referred to as "windowgrams." By default, the **zwgc** program is started up when you log in.
- **zctl:** This program is used to subscribe and unsubscribe to classes and instances, and for additional configuration of **zwgc**.
- **zwrite:** Although not technically a part of zwgc, this program is used to send messages to other users, classes, or instances.

Sending Messages (zwrite)

The **zwrite** program is used to send messages to both inidivudal users and classes or instances. To send a message to an individual user, use the following command:

```
joeuser@athena$ zwrite username
```

If the user is not logged in, you'll see something like this:

```
joeuser@athena$ zwrite jruser
jruser: Not logged in or not subscribing to messages
```

If the user is logged in, you'll be prompted to enter your message and end with a single period (.) on a line by itself. The **zwrite** program does not automatically insert line breaks, so you'll need to press Return when you reach the end of a line. In the following example, the paragraph symbol (¶) is used to indicate where the user pressed Return:

```
joeuser@athena$ zwrite joeuser
Type your message now. End with control-D or a dot on a line by itself.
Hey, can I borrow the 18.03 notes from you? I can't find my copy, ¶
I think I left it in the library.¶
.¶
Message queued for joeuser... sent
joeuser@athena$
```

As you can see, zwrite will tell you that the message was sent successfully. (If the network is exceptionally busy, there may be a delay of several seconds between when it displays "Message queued for joeuser..." and when it displays "sent".

You can also send a message to multiple individuals by specifying multiple usernames on the command line:

```
joeuser@athena$ zwrite joeuser jruser
Type your message now. End with control-D or a dot on a line by itself.
Want to grab some lunch?¶
.¶
Message queued for joeuser... sent
Message queued for jruser... sent
joeuser@athena$
```

If any of the users in the recipient list were not logged in, the zwrite command would inform you of this, but would still send the message to the other recipients. You can use the **-C** option to zwrite to cause it to automatically include a "CC:" (carbon copy) line in the zephyr indicating all the recipients of the message. Without that option, each recipient will have no way of knowing whether the message was sent to other users or not.

Sending Messages to a Class or Instance
You can send messages to groups of users who are subscribing to classes and/or instances. To send a message to a specific class, use the **-c** option. To send a message to a specific instance, use the **-i** option with **zwrite**. The syntax is:

```
joeuser@athena$ zwrite -c class -i instance
```

Generally, you would not specify a recipient (a username) when sending zephyrs to a class or instance. Remember that *class* will default to "message" and *instance* will default to "personal". The following example sends your message to everyone who is subscribing to white-magic instance:

```
joeuser@athena$ zwrite -i white-magic
```

Since a class isn't specified, the default is used which is "message" and since a recipient isn't specified, the default "*" is used.

Note: Generally, you should not send zephyrs to a class or instance to which you are not subscribed, so that you can participate in the discussion. See the section on subscribing for more information.

There are many other options available with **zwrite**. See the man page for more information.

Formatting Text in Zephyr Messages

Zephyr lets you include formatting commands in the text of your messages to control how the text will appear on the recipient's screen. (The recipient can override these formatting commands; see Customizing Zephyr.)

A formatting command can be embedded anywhere in your text. Formatting commands use the following syntax:

```
@format(formatted text)
```

In the example above, *format* is the name of a Zephyr formatting command and *formatted text* is the text to be formatted. You can use delimiters other than parentheses to enclose the text to be formatted, including (), {}, [], and <>.

These formatting commands can be used:

@roman

@bold
@b

@italic
@i

@left
@l

@center
@c

@right
@r

@small

@medium

@large

@huge

@(@color(*colorname*) *text***)**

@(@font(*fontname*) *text***)**

regular typeface (default)

boldface

italics

left-aligned text(default)

centered text

right-aligned text

small type size

medium type size (default)

large type size

very large type size

colored text
(see below for information colors)

use a specific font

The following example demonstrates how you might use some of these commands. (Again, the paragraph symbol (¶) is used to indicate where the user would press Return when typing the body of the message):

```
joeuser@athena$ zwrite joeuser
Type your message now. End with control-D or a dot on a line by itself.
This is normal text. @b(But this is in bold). And this is @italic(italicized)¶
@center(this line is centered)¶
and @(@color(red) this is in red)¶
But this isn't in red.¶
.¶
Message queued for joeuser... sent
joeuser@athena$
```

The following commands set characteristics that relate to the entire Zephyr message:

@beep()

@font(*fontname*)

@color(*colorname*)

Causes the recipient's terminal to beep. (Including this command multiple times does not cause additional beeps).

Set the current font e.g., courier, times, or helvetica.
(Use "xlsfonts | more" to list other possibilities.)

Set the color (e.g. "red", "blue", "yellow"). Most color names are supported.

Locating Other Users (zlocate and znol)

Zephyr lets you locate other users if they are logged on and using Zephyr. To do this, use the **zlocate** command:

```
zlocate username
```

If a user is logged in and is not "hidden", (see Setting Zephyr Exposure for information) a message is displayed indicating the machine name in which the user is logged in, and the date. For example:

```
joeuser@athena:~$zlocate jruser
W20-575-20.MIT.EDU unix:0 Thu Feb 18 10:43:01 2010
```

If the user were not logged in, or had chosen not to receive messages, the zlocate command would display "Hidden or not logged-in". You can specify multiple users for one zlocate command:

```
joeuser@athena:~$ zlocate jruser someotheruser
jruser@ATHENA.MIT.EDU:
W20-575-20.MIT.EDU unix:0 Thu Feb 18 10:43:01 2010
someotheruser@ATHENA.MIT.EDU:
Hidden or not logged-in
```

Login Notification (znol)

The command **znol** ("Zephyr Notify On Login") accomplishes two tasks. When first run, the **znol** command will go through a list of usernames you've created and locate anyone who is logged in and not hidden. Throughout your login session, you'll be notified, via Zephyr messages, when users in this same list log in or out, as long as they're not hidden.

Before using znol, you'll need to create the list of usernames of people you want to locate. To do this, create a text file called **~/.anyone**, and enter a list of usernames, one per line. For example, your file might look like this:

```
joeuser@athena:~$ cat ~/.anyone
jruser
anotheruser
someotheruser
```

Once your file has been created, you can use the znol command:

```
joeuser@athena:~$ znol
jruser: W20-575-20.MIT.EDU :0.0 Thu Feb 18 10:43:01 2010
someotheruser:M12-182-3.MIT.EDU :0.0 Thu Feb 18 11:31:45 2010
```

Only two lines are displayed because only two out of the three users in the file were logged in at the time.
As **znol** tries to locate each user, it subscribes you to the login messages from that user. See Setting Zephyr Exposure and Subscribing to Zephyr Messages for more information.

If you decide you do not want to receive a message every time a person logs in or out, you can disable **znol** for the rest of your session by entering:

```
joeuser@athena:~$ znol off
```

And if you just want to locate the users in your ~/.anyone file, without subscribing to login messages, you can use the **-l** (for "list") option to znol, which will only list the users.

Subscribing to Zephyr Messages (zctl)

Subscriptions are handled through the **zctl** program, which is short for Zephyr control. You use **zctl** to subscribe to classes and instances and to save the subscriptions in a file (by default the file ~/.zephyr.subs).

In addition to the syntax shown below, zctl has an "interactive" mode. To enter interactive mode, simply type "zctl" at your shell prompt and press enter. You will then get a new prompt that looks like "zctl:". You can type **?** at the zctl prompt to get a list of **zctl** options.

There are two ways that you can subscribe: for the duration of the current login session, or permanently until you cancel the subscription.

Adding subscriptions for the duration of your login

Use the **sub** (for "subscribe") option to add subscriptions that are in effect until you log out:

```
zctl sub class instance recipient
```

For example, if you want to subscribe to the "weather" zephyr instance:

```
zctl sub message weather *
```

Note: The backslash (\ ) before the asterisk (*)is required because the asterisk is a shell wildcard. If you enter the **sub** command directly at the **zctl:** prompt, or if you edit your ~/.zephyr.subs file directly, you should *not* include the backslash).

Subscribing to a class uses a similar command, but you would also specify an asterisk for the instance name. To subscribe to class "sipb":

```
zctl sub sipb * *
```

Adding Subscriptions Permanently Until You Remove Them

The **add** option adds a new subscription to a subscriptions file called ~/.zephyr.subs so that you will be subscribed every time you log in. The syntax is the same as the **sub** command, but with **add** in place of **sub**. For example, to permanently subscribe to the "weather" instance:

```
zctl add message weather *
```

Removing Subscriptions

To remove a subscription for the duration of your login session, you would use the "unsub" command, which has a syntax similar to "sub" and "add":

```
zctl unsub class instance recipient
```

The class, instance and recipient you specify must match the ones in the subscription. For example, to unsubscribe from the instance "weather" we used in the first example, you would need to use this command:

```
zctl unsub message weather *
```

You could not simply unsubscribe from class "message", instance "*", because you had not previously subscribed to that class/instance combination.

To remove a subscription from your subscriptions file, you use the "del" command instead of the "unsub" command:

```
zctl del class instance recipient
```

Common Zephyr Subscriptions

By default, you will be subscribed to the following triplets when you log in:

message,personal,*yourname*@ATHENA.MIT.EDU

message,urgent,*yourname*@ATHENA.MIT.EDU

operations,message,*

filsrv,*lockername*,*

login,*username*,*

Allows you to receive personal messages

Allows you receive urgent personal messages

Used by IS&T staff to send announcements about cluster closings or service outages

If you use the "attach" command to attach the locker *lockername*, you will receive notifications of services outages that might affect that locker

If *username* is in your ~/.anyone file, the **znol** will subscribe you to this triplet

There are many zephyr classes and instances in use, but a small selection of popular ones is listed below. . Here is a small selection of popular ones:

message,weather,*

message,printadm,*

white-magic,**,**

message,baseball,*

message,olympics,*

Weather reports on the half hour|

Used to request print job removal or clearing of stalled print queues.

Random discussions, often about current events.

Discussion of baseball, typically only used during the playoffs or world series.

Occasionally used for discussion of the Olympic games, when the games are in session.

The following classes use instances for "threading" as there may be several unrelated discussions simultaneously. For example, if you were asking for help printing your document you might use the instance "printing". Please only use letters, numbers, and periods or hyphens in instance names. Once you pick an instance, please continue to use it for the duration of the conversation.

help,*,*

sipb,*,*

Ask other MIT users for help on virtually any topic. Please use an instance name appropriate for your topic.

Ask SIPB members about SIPB projects or computing questions. Please use an instance name appropriate for your topic.

Setting Zephyr Exposure (zctl)

Zephyr lets you control whether or not people can **zlocate** you. To set an exposure level you use the **set exposure** option:

```
zctl set exposure exposure-level
```

Where *exposure-level* can be one of the following (listed from widest exposure to narrowest exposure):

net-announced

net-visible

realm-announced

realm-visible

opstaff

none

Anyone can locate you and receive login messages about you.

Anyone can locate you. Only users within the same Kerberos realm (@mit.edu) ** can receive login notices about you.

Anyone within the same Kerberos realm ** can locate you or receive login messages about you.

(default) Anyone within the same Kerberos realm ** can locate you. No login or logout messages will be sent.

Only Athena operations staff can locate you, and no login messages will be sent.

You are not registered with Zephyr at all: No one can locate you, and you will not receive any messages (including personal messages).

Sending Automatic Replies With zaway

To send an automatic reply to users who send you a message, use the **zaway** command. In its simplest form, you can simply run "zaway":

```
joeuser@athena:~$ zaway
```

Anyone who sends you a zephyr will receive an automated reply saying "I'm sorry, but I'm away from my computer and am not able to receive your message." You can change the text by creating a file called ~/.away and putting your away message text into that file, or you can specify it on the command line:

```
joeuser@athena:~$ zaway -m "I'm at lunch, be back soon."
```

The ~/.away file also allows you to specify different messages for different users. The general format of this file is:

```
>username
message
>username
>username
message
```

In addition, there are two options that add to the ability to specify users:

- "%" the specified message is sent to all other users that have not been explicitly named in the file
- "*" the specified message is sent to all users.

Note: If a user name does not match any of those listed in the file, and no "*" or "%" field is specified, no return message is sent.

Here is an example of a ~/.away file:

```
>jruser
Hi jruser, I went to class, see you later.
>joeuser
Hey Joe, i'll see you later
>%
I don't know who you are, but I'm not here.
>*
I'll read your message when I get back.
```

In the example above, jruser and joeuser would receive customized replies. Anyone else would see "I don't know who you are, but I'm not here". And everyone (including jruser and joeuser) would see "I'll read your message when I get back."

Sending Automatic Messages to Yourself (zleave)

The **zleave** command sets an "alarm." You supply the time you want to leave, then Zephyr sends you reminders 5 minutes before, 1 minute before, at the actual time, and every minute for ten minutes thereafter (or until you log out or cancel **zleave**).

You can set the time as a specific time within the next 12 hours, using the format *hhmm*, or you can set a relative time with the format *+hhmm*. The zleave command will print the time of the alarm for confirmation. The examples below include the date at the time for

```
joeuser@athena:~$ date
Thu Feb 18 22:10:08 EST 2010
joeuser@athena:~$ zleave 0700
Alarm set for Fri Feb 19 07:00:26 2010
```

To set a relative time:

```
joeuser@athena:~$ date
Thu Feb 18 22:11:18 EST 2010
joeuser@athena:~$ zleave +0020
Alarm set for Thu Feb 18 22:31:18 EST 2010
```

The default messages are:

You have to leave in 5 minutes

Just one more minute!

Time to leave!

You're going to be late!

5 minutes before specified time

1 minute before

Specified time

Each subsequent minute

If you want to add more text to the default message, use the **-m** option. For example:

```
joeuser@athena:~$ zleave +0020 -m "Go to class"
```

In this case, the words "Go to class" would be appended to each of the messages listed above.

To cancel the **zleave** messages, use the following command:

```
joeuser@athena:~$ zleave cancel
```

Using Zephyr Over Dialup

When you log in without using the Xwindows system (such as using a terminal style session, or connecting with ssh), you need to run Zephyr in "tty" mode. This means that incoming messages will appear as regular text on your screen. If you want to do this on an as-needed basis, you can use the "-ttymode" option to zwgc:

```
joeuser@athena:~$ zwgc -ttymode
```

Alternatively, you can configure Zephyr to start automatically with every login session. To do this, you need to set the "fallback" zephyr variable:

```
joeuser@athena:~$ zctl set fallback true
```

Copying the Text of a Zephyr Message

You can save information from a Zephyr windowgram by using cut and paste. This text can be pasted into another non-windowgram window, such as your Email client or an Emacs window. Use the following mouse techniques to highlight text and put it in the X buffer.

Shift-Left click

Shift-Right click

Shift-Left click and drag over text

Shift-Right click and drag over text

Mark one end of the region to be copied

Mark other end of the region to be copied and highlighted

"Dragged" text is highlighted and copied

Extends highlighted text

Then click the middle mouse button in the window into which you would like to paste the text.
Note: There is no way to cut-and-paste "special effects" such as boldface, italics, or large print. This special text will get pasted as normal text.)

Customizing Zephyr

The first topic covered in this section describes how to make changes to the .zwgc.desc and .Xresources files since customizing Zephyr often requires modifying one of these files.

The subsequent topics cover the following ways to customize Zephyr:

- Setting your Zephyr signature
- Changing the default font for Zephyr messages
- Filtering out unwanted Zephyr messages
- An example of formatting incoming messages
- Using the ~/.Xresources file to position Zephyr windowgrams on your screen

Setting Your Zephyr Signature

When you send a Zephyr message to someone, your *signature* appears in the message. If you do not explicitly set it to something else, it will be your name as it appears in your **finger** information. You can pick a new Zephyr signature on a per-message basis, or set a new one for all messages:

To change your signature for all messages, use this command:

```
joeuser@athena$ zctl set zwrite-signature "new-signature"
```

In the example above, *new-signature* is the signature you want to use. In general, you should limit your signature to 70 characters or so to prevent sending excessively wide zephyrs to other users.

To change your signature for a specific Zephyr message, use the "-s" option to the **zwrite** command:

```
joeuser@athena$ zwrite -s "new-signature" ...
```

In the example above, *new-signature* is the signature you want to use (it will override any signature you set with the **zwrite-signature** variable described above). You can use the "-s" option with any other options you might use with the zwrite command.

Changing the Default Font for Zephyr Messages

If you want to change the default font that is used when displaying Zephyr messages, you need to include the following line in your ~/.Xresources file:

```
zwgc*style*substyle.default.fontfamily: font-name
```

where *font-name* is the name of the font that you want to use. If you don't have a ~/.Xresources file, you can create one.

You can use one of the predefined fonts: times, helvetica, or courier (which is the Athena default) or, if you have defined your own font families in your ~/.Xresources file, you can use one of those names.

Any changes to your ~/.Xresources file will take effect the next time you log in. If you want them to take effect for the current login session, you'll need to reload the X resource database with the following command:

```
joeuser@athena:~$ xrdb -load ~/.Xresources
```

Filtering Out Unwanted Zephyr Messages

You can selectively filter the display of Zephyr messages that you receive by using the Filter feature. For example, if you subscribe to the triplet <message,,> and do not want to see messages from the "white-magic" instance, you can filter out <message,white-magic,*>. You still receive the <message white-magic *> messages but they are not displayed. You can also redisplay messages you had previously filtered out.

To filter messages for the duration of your login session, you can use the **zpunt** command located in the **sipb** locker. (You will need to **add** the locker before using the command for the first time):

```
joeuser@athena$ add sipb
joeuser@athena$ zpunt instance
```

In the example above, *instance* is the name of the instance that you want to punt. The **zpunt** command can also take a class name. For example:

```
joeuser@athena$ zpunt white-magic
joeuser@athena$ zpunt help boring-topic
```

In the example above, the first command filters class "messsage", instance "white-magic". The second command filters class "help", instance "boring-topic".

To resume receiving messages on a previously punted class or instance, use the **zunpunt** command. The syntax is identical to the **zpunt** command.

Filtering messages by adding a line to your ~/.zephyr.subs file

To permanently filter a specific class or instance combination, you can add the following line to your ~/.zephyr.subs file for each message that you want to filter:

```
-class,instance,recipient
```

For example, to ignore class message, instance "white-magic" notices, you would insert this line:

```
-message,white-magic,*
```

Positioning Zephyr Windowgrams on Your Screen

To change where Zephyr notices appear on your screen you need to edit the ~/.Xresources file. You specify the x and y coordinates for a corner of the window by using positive or negative numbers to specify relative positions.

The syntax for a line that positions windowgrams is:

```
zwgc.style.class.instance*geometry: position
```

In the example above, *class.instance_are the specific class and instance for which you want to position the windowgram, and _position* is two positive or negative numbers specifying the X and Y positions. Positive coordinates are on the left or top of the screen, and negative coordinates are on the right or bottom of the screen. For example, to configure personal messages to appear in the lower right corner, you

```
zwgc.style.message.personal*geometry: -0-0
```

That configures zephyrgrams of class "message", instance "personal" to appear 0 pixels away from the bottom right corner of the screen.

You can specify a default location for all of your Zephyr notices (which normally appear in the top left) as follows:

```
zwgc*geometry: +0-0
```

The example above would cause all zwgc windows to appear 0 pixels from the left and 0 pixels from the bottom (i.e. the bottom left corner).

As before, any changes to your ~/.Xresources file will take effect the next time you log in. If you want them to take effect for the current login session, you'll need to reload the X resource database with the following command:

```
joeuser@athena:~$ xrdb -load ~/.Xresources
```

Other Sources of Information About Zephyr

For answers to some commonly asked questions about Zephyr, see: Zephyr Stock Answers.

You may also wish to consult the SIPB document: Inessential Zephyr, or the manual pages for any of the commands listed in this document. For more information on viewing manual pages, please see Working With the Command Line.

Related Links

Athena at MIT
Athena On-Line Help
Getting Started with Athena
Zephyr Stock Answers