

VoiceShell

Tomi Blinnikka

COLLABORATORS

	TITLE : VoiceShell		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY	Tomi Blinnikka	October 27, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	VoiceShell	1
1.1	VoiceShell table of contents	1
1.2	Contents of archive	2
1.3	Document version	2
1.4	Copyright information	3
1.5	Basics about VoiceShell	3
1.6	What is needed to use VoiceShell	3
1.7	What VoiceShell run on	4
1.8	Starting VoiceShell from a CLI/Shell	4
1.9	wbstart	5
1.10	Description of main window	6
1.11	menus	6
1.12	gadgets	8
1.13	How to teach words	8
1.14	Preferences and data file	9
1.15	ARexx commands	10
1.16	Multiple users	12
1.17	Hints & Tips	12
1.18	Example scripts to use with VoiceShell	13
1.19	Possible problems that You might encounter	13
1.20	Error messages displayed by VoiceShell	13
1.21	Known bugs	13
1.22	List of versions and changes	14
1.23	Future versions and ideas for them	15
1.24	VoiceShell shareware fee	15
1.25	Contacting the author	15

Chapter 1

VoiceShell

1.1 VoiceShell table of contents

22/10-93 VoiceShell V1.18

Table of contents

[Contents of archive](#)

[Document versions](#)

[Copyright](#)

[What does VoiceShell do?](#)

[What do You need?](#)

[Compatibility](#)

[Starting from CLI/Shell](#)

[Starting from Workbench](#)

[Main window](#)

[Menus](#)

[Gadgets](#)

[Learning words](#)

[Preferences & data file](#)

[ARexx commands](#)

[Multiple users](#)

[Hints and tips](#)

[Examples](#)

[Problems](#)

[Error messages](#)

[Bugs](#)

[Versions](#)

[Future versions](#)

[Shareware fee](#)

[Contacting the author](#)

1.2 Contents of archive

Contents of archive

This archive must include the following files:

File Description

VoiceShell/VoiceShell - the program itself
VoiceShell/VoiceShell.info - the program icon
VoiceShell/VoiceShell.guide - program documents (this file)
VoiceShell/VoiceShell.guide.info - icon file
VoiceShell/Libs/voice.library - library file
VoiceShell/Scripts/VSON - script file
VoiceShell/Scripts/VSOFF - script file
VoiceShell/Scripts.info - icon file
VoiceShell.info - icon file
Install - install script
Install.info - icon file
ReadMe - read this file first
ReadMe.info - icon file

1.3 Document version

DOCUMENT VERSIONS

Updated: 13/02-93 - for version 1.01
Updated: 14/02-93 - for version 1.02
Updated: 31/05-93 - for version 1.05
Updated: 02/06-93 - for version 1.06
Updated: 10/06-93 - for version 1.07
Updated: 19/06-93 - for version 1.08
Updated: 22/07-93 - for version 1.11
Updated: 08/08-93 - for version 1.12
Updated: 12/08-93 - for version 1.13
Updated: 14/09-93 - for version 1.14
Updated: 15/09-93 - for version 1.15
Updated: 15/09-93 - for version 1.16
Updated: 22/10-93 - for version 1.18

1.4 Copyright information

COPYRIGHT INFORMATION

Copyright (C) **Tomi Blinnikka** 1993, All Rights Reserved.

This program is . This means that you can copy it and give it to all your friends, upload it to a BBS or include it in a PD-library (FISH, Walnut Creek CDROMS etc.). BUT (!!!) if you use this program frequently you must pay \$5 - \$10. The other restrictions are: all of the files must be included in their original form without additions, deletions, or modifications of any kind and it IS NOT TO BE SOLD. THIS INCLUDES GERMAN PD DEALERS. If you wish to use this program for commercial purposes, you need written permission. The VoiceShell program is provided "AS IS" without warranty of any kind, and the is not responsible for any damage this software may or may not do.

1.5 Basics about VoiceShell

WHAT DOES IT DO?

This program is a 'replacement' for VCLI. It doesn't have the fancy graphics etc. but it seems to eat less CPU time and should be faster overall. It also has some extra options.

So what is VCLI I hear you ask. VCLI is a program by Richard Horne that uses his voice.library for recognizing speech. With VCLI and VoiceShell you can start programs by saying the program's name. You can VoiceShell 60 (VCLI allows 48) different words. You also have the possibility to load a new set of words, thus giving limitless possibilities. The more words you , the less accurate the result will be. Thus having more than one **data file** might be a good idea.

1.6 What is needed to use VoiceShell

WHAT DO YOU NEED?

You'll need an Amiga computer (see list), with Kickstart 2.04 (intuition.library and gadtools.library version 37 or higher). You'll also need voice.library version 8.2 or higher and a sampler (see list).

An accelerated Amiga (MC68020 or higher) is recommended for continuous usage.

1.7 What VoiceShell run on

COMPATIBILITY

VoiceShell will work on any Commodore-Amiga, running on Kickstart 2.04 or higher. This includes the following: A500(+), A600, A1000, A1200, A2000, A2500/20, A2500/30, A3000/16, A3000/25, A4000/40, A4000/30. The program **SHOULD** work on any upgrades (based on the Amiga Development Guidelines, see Amiga ROM Kernel Reference Manual (Kickstart 1.3 version): Libraries and Devices, introduction).

VoiceShell works with the following samplers:

PerfectSound

AudioMaster

GVP DSS 8

Generic samplers (inc. most homemade samplers)

NOTE! Most samplers will work with the generic sampler **setting**.

1.8 Starting VoiceShell from a CLI/Shell

STARTING FROM CLI/SHELL

You can start the program from both CLI/SHELL/etc. and .

To run the program, enter the following into your CLI/SHELL/etc.

```
1> VoiceShell FROM/K,I=ICONIFY/S,PS=PUBSCREEN/K,PRI=CX_PRIORITY/N/K,  
PK=CX_POPKEY/K,PU=CX_POPUP/K:
```

Where: ICONIFY iconifies program when run. The window won't be activated. Use 'I' as an abbreviation.

FROM allows you to specify the file (not the data file).

PUBSCREEN allows you to specify the name of the public screen you would like VoiceShell to open its window on. If the screen name isn't given, VoiceShell will use the default public screen. Use 'PS' as an abbreviation.

CX_PRIORITY allows you to set the priority of the commodity hotkey. A number from -128 to 127 is allowed. Use 'PRI' as an abbreviation.

CX_POPKEY allows you to specify the keystroke that brings up the VoiceShell window. NOTE: The hotkey doesn't work if VoiceShell is disabled!

The default hotkey is <control alt v>.

CX_POPUP specifies whether the window is to be opened when

the program is run.

Along with the GUI, you can also use the Break command or CTRL-C to quit the program. CTRL-D will turn VoiceShell off (disable), and CTRL-E will turn it back on (enable). You may also use **ARexx** commands to control the program.

You can also add this line into your S:User-StartUp to run the program when the machine is booted.

RUN >NIL: VoiceShell

You may want to add the ICONIFY option ...

RUN >NIL: VoiceShell ICONIFY

... to iconify the window, when the program starts.

1.9 wbstart

STARTING FROM WORKBENCH

From the Workbench, start the program like any other, by double-clicking on its icon. If, however, you would like to specify the file or iconification, use the tool types. This is done by clicking once on the icon and selecting 'Information...' from the 'Icons' menu in the Workbench. Then change the tool types accordingly (please refer to user manual, if needed). Please take care with upper- and lowercase letters! The inputs ARE case-sensitive. If the public screen name isn't given, VoiceShell will use the default public screen.

If you would like the program to run on every boot, move VoiceShell into the 'WBStartUp' drawer. If you do this, don't forget to add the tool type 'DONOTWAIT' (this applies to almost every program...). Below is a list of available tool types and there usage:

TOOL TYPE VALUE

FROM <filename>

ICONIFY YES

PUBSCREEN <public screen name>

CX_PRIORITY <Number>

CX_POPKEY <Keystroke to use as hotkey>

CX_POPUP <'NO' if you don't want the window to open when started>

DONOTWAIT <NONE> (needed if program is in the 'WBStartUp' drawer)

Examples:

FROM=ENV:VoiceShell.PREFS2

PUBSCREEN=My_Public_Screen_1

```
ICONIFY=NO
CX_POPUP=NO
CX_POPKEY=ctrl alt v
or
FROM=Work:VoiceShell/VS.PREFS
PUBSCREEN=Workbench
ICONFIY=YES
```

1.10 Description of main window

MAIN WINDOW

The main window has several , From the checkbox (On/Off), you can specify whether or not to listen to speech. When VoiceShell is off it will eat hardly any CPU time.

The string gadget 'Data file' holds the name of the **data file** which has the commands to be run and the speech samples.

The 'Vol Up' and 'Vol Dwn' are only active when a PerfectSound or DSS 8 sampler is used. With these two gadgets you can set the input gain of the sampler. Please refer to your samplers manual for more information. NOTE! The PerfectSound sampler will roll over from its maximum value to zero and from zero to its maximum value. The GVP DSS8 doesn't have this 'feature'. If the GVP DSS8 is being used, you can see the current gain setting next to the 'Vol Dwn' gadget.

The other button gadgets either 'Save' the (window place, gadget settings etc.) and the **data file** or 'Quit' the program. These two may also be selected from the . The last one, 'Load' will load ONLY the **data file**. NOTE! The 'Load' menuitem will load the AND **data file** !

The window has a Kickstart 2.04 Zoom gadget. When you click on this the window will go into its iconified state. Clicking on it again will open it up.

1.11 menus

MENUS

Project:

Load - Loads the (ENVARC:VoiceShell.PREFS or file specified on startup) AND the **data file**.

Save - Save the and date file.

About... - Gives some info on the program.

~~~~~

Quit - Quit program (NOTE! The and **data file** are not saved!).

Commands:

Edit/Learn - Allows you to VoiceShell the commands to recognize. You can also change the command without re-sampling. When you select this a window will open. In this window there are two columns of . The first column is the description of the command to learn. The second contains the command and its arguments to execute. Pressing return in any of the gadgets in the first column (Description) will cause VoiceShell to start learning the commands. Pressing return in the second column (Command to run) will only update the command to run. The Get-gadget allows you to specify the command to run with a file requester.

Setup:

Sampler - Allows you to set specify the type of sampler you have. If your sampler isn't mentioned, choose Generic.

Resolution - Sets the accuracy of the samples. Low uses less CPU time than High, but is less accurate.

Accuracy - Allows you to set how accurate the spoken words have to be. Set to a low value if you have extra background noise.

Channel - Allows you to select either the left or right channel to sample from. ALL BUT GENERIC.

Input - Allows you to set the input level, line or microphone. ALL BUT GENERIC.

CIA timer - Specifies the CIA timer to use. Other programs may use one or the other (especially music module players/editors).

Priority... - This sets the priority of the task that recognizes speech.

---

## 1.12 gadgets

### GADGETS

- Checkboxes

On/Off - Try to recognize speech? When this is off

VoiceShell won't use much CPU processing time.

- String gadgets

Data file - Path and name of the **data file**.

- Button gadgets

Vol Up - Increase the input gain. PerfectSound and GVP

DSS 8 only.

Vol Dwn - Decrease the input gain. PerfectSound and GVP

DSS 8 only.

Load - Load only the **data file** specified in the 'Data file' gadget.

Save - Save the and **data file**.

Quit - Quit program (NOTE! The and **data file** are not saved!).

## 1.13 How to teach words

### LEARNING WORDS

Select Edit/Learn from the to start learning new words. First type in the description for the word to be taught and press return. This will bring up the sampling window. Press 'Sample' to start learning this word and say the word. If the requester suggests you should accept, the input volume was fine and you may press the 'Accept' button. You are asked to do this three times. After the word has been taught, you are prompted to enter the command to be run when this word is recognized. Repeat the procedure for all the commands you wish to teach.

Pressing return in the first column will ALWAYS open the sampling window.

Be careful not to accidentally press return in the first column if you are editing the commands to be run and not the samples themselves. Pressing TAB in the first column will also start the sampler.

When the sampler window is open you will see two or four gadgets depending on your sampler setup. A short description follows:

Sample - Start sampling. The program will wait for input before starting.

Accept - Accept the previous sample and get the next one.

---

Volume Up - Turn the input gain up. PerfectSound and GVP DSS 8 only.

Volume Down - Turn the input gain down. PerfectSound and GVP DSS 8 only.

Closing the window with the window's close gadget will abort sampling, but destroy any previous sample.

You are asked to give 3 (three) good samples. These will be later used to recognize the word. See 'Hints & Tips' for some tips on teaching words. After three samples have been given the window will be closed and you can now enter the command to be run when this word is recognized. The command run will use the same input/output as VoiceShell unless it is redirected (><NIL: for example). If the command starts with a tilde ("~") VoiceShell assumes this command to be an internal one. The following internal commands are currently available:

~K - Allows you to simulate keypresses. See the [ARexx](#) command 'KEY' for a more detailed explanation.

The Get-gadget at the bottom of the window will open a file requester, which allows you to specify the command to run. To use this gadget, activate the desired string gadget and click on the Get-gadget.

As mentioned earlier, you can teach VoiceShell 60 words. In the window however, there are only twenty gadgets (10 for descriptions and 10 for the commands and arguments), use the menus to select which set of words to Edit/Learn.

## 1.14 Preferences and data file

### PREFERENCES & DATA FILE

When you select 'Save', either from the or by clicking on the gadget, VoiceShell will save all settings into a file called VoiceShell.PREFS, or the file specified with the FROM option on startup. It will be saved onto the assigned device ENVARC: where other system preferences are stored. It however, is not an IFF file. This is to save space, in both code size and disk space. The data file will also be saved to the file specified in the 'Data file' string gadget.

The window place will also be saved. If the Window was zoomed when 'Save' was chosen, the program will not however, zoom the window when the preferences are loaded or the program is started. On startup you must specify the option.

## 1.15 ARexx commands

### AREXX COMMANDS

VoiceShell can be controlled via ARexx. VoiceShell has an ARexx port with the name 'VOICESHELL'. By sending ARexx messages to this port you have almost full control over VoiceShell.

Examples:

To control VoiceShell via arexx you may use the shell command RX.

It is possible to control the program from an ARexx script or straight from the command line.

```
SYS:REXXC/RX "ADDRESS VOICESHELL COMMAND"
```

To load a prefs file named 'S:VoiceShell.PREFS2':

```
SYS:REXXC/RX "ADDRESS VOICESHELL 'LOAD S:VoiceShell.PREFS2'"
```

NOTE the double and single quotes. VoiceShell treats anything after the keyword 'LOAD ' (including a space) as a filename.

NOTE WELL! Sending ARexx commands to VoiceShell from ITSELF

will cause a LOCK UP in VoiceShell. This will not affect other programs, but you won't be able to quit VoiceShell. If you wish to send ARexx commands to VoiceShell from itself, YOU MUST USE THE 'RUN' COMMAND!

The ARexx command 'KEY' allows you to fake key presses as if you actually typed something. This allows you to cancel requesters, use menu/gadget short cuts, etc. The 'KEY' command is simple to use with single keypresses, but works quite oddly if you wish to wedge whole text strings to the machine.

Follow these few simple rules when using the 'KEY' command:

1. Don't forget the SPACE after the keyword 'KEY'!
2. To input text strings enter the string backwards.
3. Use the greater than/smaller than (">", "<") characters around qualifiers. Available qualifiers are, for example alt, shift, capslock, ctrl, lamiga, ramiga, etc. See your user's manual on commodities qualifiers for more information.
4. All keystrokes are converted to lower case. If your OS version knows how to, then all keystrokes are handled properly, even international characters.

For example:

```
RX "ADDRESS VOICESHELL 'KEY <LAMIGA B>'" - Cancel/Abort requesters
```

```
RX "ADDRESS VOICESHELL 'KEY <LAMIGA V>'" - OK/Retry requesters
```

```
RX "ADDRESS VOICESHELL 'KEY gnirts txet gnol'" - "long text string"
```

For more detailed information on using the ARexx language, please refer to

---

your user manuals.

Please see the [menu](#) and [gadget](#) for more detailed explanations on the following functions.

VoiceShell supports the following commands:

#### AREXX COMMAND RESULT

##### Window:

FRONT The main window is brought in front of all others.

BACK The main window is put behind all others.

ZOOM Zooms (zips) the window, i.e. iconifies & un-iconifies the main window.

HIDE Hides the main window. Same as pressing the 'Hide Interface' button in the Exchange program.

SHOW Shows the main window. Same as pressing the 'Show Interface' button in the Exchange program.

##### Project:

LOAD FILENAME Loads FILENAME as a preferences file and the data file set by these preferences.

ON Adds the VoiceTask. Start recognizing.

OFF Removes the VoiceTask (eats less CPU).  
Stop recognizing.

QUIT Quits VoiceShell.

##### Sampler:

PERFSND Use the Perfect Sound sampler.

SNDMSTR Use the Sound Master sampler.

DSS8 Use the GVP DSS 8 sampler.

GENERIC Use the generic sampler.

##### Resolution:

RESLO Set high resolution.

RESHI Set low resolution.

##### Accuracy:

LOWEST Set accuracy level.

LOWER Set accuracy level.

LOW Set accuracy level.

NORMAL Set accuracy level.

HIGH Set accuracy level.

HIGHER Set accuracy level.

EVENHIGHER Set accuracy level.

MAXIMUM Set accuracy level.

Input channel:

LEFT Listen to the left channel.

RIGHT Listen to the right channel.

Input gain:

LINE Set line input gain

MIC Set microphone input gain.

GAINUP Increase volume.

GAINDOWN Decrease volume.

Timer:

TIMERA Set CIA timer A.

TIMERB Set CIA timer B.

Misc:

KEY [KEYPRESS] Fakes [KEYPRESS] as a real keypress to the machine. See above for more details.

## 1.16 Multiple users

### MULTIPLE USERS

Because everybody has their own way of pronouncing words and simply because people have different voices, VoiceShell probably won't understand words said by person A, when person B has the words. In order to get around this problem, all the users must teach VoiceShell to understand their own voice and to save their personal files.

With a simple startup script for each user, multiple users can use VoiceShell on a single machine.

## 1.17 Hints & Tips

### HINTS & TIPS

1. When words, take the first sample a few times before accepting it. This way you can see what the waveform is supposed to look like and you'll get a cleaner sample. This increases accuracy when recognizing.

2. To load a new preference file by speech, teach VoiceShell the word and use the following as the CLI command:

```
RUN SYS:REXXC/RX "ADDRESS VOICESHELL 'LOAD FILENAME'"
```

VoiceShell will then send itself an ARexx message telling itself to load the file 'FILENAME'.

NOTE! You must use the 'RUN' command when sending ARexx message from VoiceShell to itself. See the section on [ARexx commands](#) for more information.

---

## 1.18 Example scripts to use with VoiceShell

### EXAMPLES

Here we have some example scripts to use with VoiceShell. The first one will disable VoiceShell by sending a CTRL-D to the program and the second one will enable VoiceShell.

NOTE! These two DO NOT WORK when VoiceShell is loaded from .

;Example 1, turn VoiceShell off

;---CUT---

Break `Status COM=VoiceShell` d

;---CUT---

;Example 2, turn VoiceShell on

;---CUT---

Break `Status COM=VoiceShell` e

;---CUT---

ARexx versions of these two scripts can be found in the VoiceShell/Scripts directory (they are not real ARexx scripts, but instead use the 'RX' command to send ARexx messages to VoiceShell.

## 1.19 Possible problems that You might encounter

### PROBLEMS

There can only be one voice.library open. If you try to run another copy VoiceShell along with itself or VCLI/AniMan you will receive an error message.

## 1.20 Error messages displayed by VoiceShell

### ERROR MESSAGES

The VoiceShell program will display necessary error messages. If a window can't be opened, the user will be informed. If a PublicScreen cannot be locked, the user will be informed etc. I HATE PROGRAMS THAT DON'T TELL WHAT-THE-HELL IS GOING ON! That clear? If VoiceShell doesn't give any output, output has been redirected to NIL: etc. Another case might be when VoiceShell has been started from Workbench. If nothing happens try starting it from a Shell.

## 1.21 Known bugs

### BUGS

No known bugs at the moment. VoiceShell has been run with enforcer & mungwall and nothing was found.

---

## 1.22 List of versions and changes

### VERSIONS

NOTE: The Size may vary a few bytes if the version is registered/not registered!

Version Size Date Description

-----

1.18 26244 22/10/93 Now correctly gets font sizes etc. everytime the window is opened (READ: bug!).

1.17 26024 05/10/93 Now handles international characters correctly, when faking keystrokes. Added 'HIDE' & 'SHOW' ARexx commands, forgot to add these in 1.16.

1.16 ~25000 26/09/93 VoiceShell was made a commodity.

1.15 23988 19/09/93 Added 'Get'-gadget into 'Edit Commands'-window.

1.14 23012 14/09/93 Added PUBSCREEN tooltype and CLI option. This version has VSConv in the archive.

1.13 22856 12/08/93 Added ability to wedge input events into the input stream. Added internal commands.

1.12 21804 08/08/93 Increased the size of the commands to run. Also made the edit window a little bit wider.

1.11 21724 22/07/93 Implemented ARexx 'LOAD' command. Bugs fixed. Added Install script.

1.10 ~21000 17/07/93 Slightly optimized. Bugs added (Vol Up/Dwn may crash).

1.09 ~22000 21/06/93 Added ARexx control.

1.08 20000 19/06/93 Searched for bug, but it was in the library. Version number bumped.

1.07 20000 10/06/93 Fixed stuff in Left/Right & Mic/Line selection.

1.06 19932 02/06/93 Added selection for Left/Right & Mic/Line. Added misc stuff. Fixed a few bugs.

1.05 18792 30/05/93 Set priority to 120 on startup and resets it to 0 when quit. Fixed some bugs. Added Accuracy, shows DSS gain.

1.04 ????? 25/05/93 Added support for GVP DSS8. Needs voice.lib version 7.1 or higher.

1.03 ????? 18/05/93 Fixed some font stuff.

1.02 15664 14/02/93 Added 'Load' gadget to load only the data file.

---

1.01 15588 11/02/93 Name changed from VShell to VoiceShell so it won't get mixed up with VisualShell or whatever it is.. :). Fixed a bug in the load routine. VS would load the wrong data file on startup.

1.00 ~15000 07/02/93 Changed parser. From option, generic sampler, timer added. Lotsa bugs fixed.

0.00a- ??? ?/?/92 Wrote the program basics and main routines.

0.99a

## 1.23 Future versions and ideas for them

### FUTURE VERSIONS

Future version will have the bug fixes for the bugs I haven't found. The ARexx mentioned in earlier versions has been implemented, what else do you need? :)

## 1.24 VoiceShell shareware fee

### SHAREWARE FEE

As of version 1.01 VoiceShell is ShareWare. I am not asking for a huge amount of money, so I hope I will get some response. The fee is only \$5 to \$10 (USD), or whatever that happens to make at the moment you're reading this, in german marks (DM). If you're from Finland a sum of 20 FIM to 50 FIM is great (Yeah, so what if I changed the \$10 to \$12 ? You Finnish guys don't have to pay so much for postage :)

## 1.25 Contacting the author

### CONTACTING THE AUTHOR

Please send donations, **bug** reports, and list of features you would like to have included in **future versions** to

Tomi Blinnikka

Syystie 10

00780 Helsinki

FINLAND

Contact me at the address above or you can also contact me on the following BBS and net(s):

MITS +358-0-458 2066

---

MIT5 +358-0-458 2077

MIT5 +358-0-458 2088 (Sec. level 40 or higher ONLY)

docbliny@mits.mdata.fi

"...trademark of Xerox Corporations. Reading legal mush can turn your  
brain to guacamole! Epson is registered trademark of Epson America  
..." -Amiga RKRM Includes & AutoDocs, fourth page.

Tomi Blinnikka

docbliny@mits.mdata.fi