

Multuser 3.0 System

Text protocol and API

[illegible]

1	INTRODUCTION	1
1.1	General	1
1.2	Multiserver Solutions	1
2	BASICS	2
2.1.1	Identifier	2
2.1.2	Alias	3
2.1.3	Magic command	4
2.1.4	Structure of text commands	5
2.1.5	Sources	5
3	TEXT PROTOCOL COMMANDS	6
3.1	Room and zone commands	6
3.2	Configuring zones	7
3.2.1	User commands	8
3.2.2	Control commands	8
3.2.3	User Favorite commands	9
3.2.4	User Favorites Folder Commands	9
3.2.5	Trigger commands	10
3.2.6	Proxy commands	10
3.2.7	Timer commands	11
4	ACTION SERVER	12
4.1	Help list	13
4.2	Device list	14
4.3	Service list	14
4.4	Status messages	15
4.4.1	Source status (Source)	15
4.4.2	Room status (Room)	15
4.4.3	User status (User)	16
4.5	User favorites list, content queries	16
4.5.1	Number of user favorites	16
4.5.2	User favorite type 1: single source	16
4.5.3	User Favorit Typ 2: Container	16
4.5.4	User Favorit Typ 3: Ordner	17

1 Introduction

The Multiuser 3.0 system is a Multiuser / Multiroom system that operates predominantly in the background. Revox has developed the text protocol for optimal integration into building automation or Smarthome systems.

The text protocol is an easily readable and learnable interface with which the basic functions of the multi-user system can be controlled simply and effectively. The text protocol commands are also used in the Multiuser Configurator for trigger actions.

1.1 General

This description assumes that a multiuser system with all necessary peripheral devices (router, switch, multiuser server) is available and correctly configured.

The higher-level control system communicates with the Multiuser Server, which can be in the form of a Revox V400 or Revox M300/M500, via the Multiuser text protocol. All configuration data of the entire Multiuser system is also stored there.

1.2 Multiserver Solutions

In the Multiuser 3.0 system, several multiuser servers can be used. This is called a multiserver solution. Since the servers constantly exchange all configuration data (manifests) with each other, communication from the higher-level control system can take place on a single multiuser server. With multiserver solutions, it is particularly advisable to address services specifically by identifier or alias.

2 Basics

With the multiuser text protocol, all services of the system can be addressed, retrieved or controlled. A service can be addressed in three different ways:

- Identifier
- Alias
- Magic Command

The differences including advantages and disadvantages are shown below.

2.1.1 Identifier

In the multiuser world, all services have a unique identifier ex works, which begins with a prefix and is uniquely determined via the MAC address of the amplifier / server including the final index number.

Identifier	P 00 04 A3 D2 3C 27 2
Prefix	D = Device U = User R = Room Z = Zone S = Source Y = Favorite (User Favorite) X = Timer P = Proxy T = Trigger K = KNX Gateway
MAC address	00 04 A3 D2 3C 27
Index number (hex)	0 = First service of this type 1 = Second service of this type ... F = Sixteenth service of this type

In the example above therefore, the identifier **P 00 04 A3 D2 3C 27 2** is the 2nd service of a proxy of device 00 04 A3 D2 3C 27.

Identifier in text protocol

In the multiuser text protocol, the identifier is called with a preceding "#" character.

#R0004A3D23C272:room:select: #U0008B1E23AA73

Advantages and disadvantages: Identifier

- + An identifier cannot be influenced by configurations via app or Configurator. Thus, it remains constant in a hardware-bound manner.
- For simple and manual configurations, the identifier is awkwardly long and therefore carries a high risk of errors when "writing".

2.1.2 Alias

To simplify the programming of the many parameters in the Multiuser world as much as possible, you can give an alias name to each Multiuser element (User, Room amplifier, Service) in the Multiuser Configurator. The alias represents the element and should be given a name that can be assigned quickly and uniquely.

User1, user2, etc. are certainly universal but are not very helpful in a project as you cannot immediately identify which user is meant. The alias is limited to 16 characters in the ASCII format.

IMPORTANT: An alias can never be assigned twice!

We also recommend assigning a prefix to the alias. This makes the alias uniquely assignable to a service and minimizes the risk of duplicate name assignment.

Example

- **r.living** for the name of the room
- **s.living** for the sources (locale source of the room)
- **t.living** for the trigger

Alias in the text protocol

If you use an alias in the text protocol, it must always be preceded by a "\$" character.

\$r.living:room:select:\$u.peter

Advantages and disadvantages: Alias

- + An alias can be freely defined, and the service is then uniquely addressable.
- The alias can be changed at any time via App or Configurator.

2.1.3 Magic command

In the Multiuser System, Magic commands are used to guarantee simple basic functions ex works, such as default source and audio triggers as well as RC5 triggers for wall and remote controls. The Magic call always refers to the services of the device on which the action was triggered. The Magic command is therefore poorly suited for control by higher-level control systems.

Magic Identifier list:

- @room
- @zone
- @user
- @favorite
- @local
- @timer
- @trigger
- @proxy

Example:

- With **@room** the wall control connected to the amplifier addresses its own room
- With **@user.1** the first user from the server in the system is addressed

Magic command in the text protocol

The Magic command in the text protocol is always preceded by an "@" character.

@room:room:select:@user.1

Advantages and disadvantages: Magic command

- + Preconfigured at the factory and easy to duplicate and use.
- The Magic command is not suitable for module or plugin communications from external systems, because a Magic command received from the Multiuser Server cannot be uniquely assigned to a service.

2.1.4 Structure of text commands

Basically, a Multiuser text always consists of four elements. In the example below, a user takes over a room and thus streams his music:

@room:room:select:@user.1

what is addressed: (room, user, source, proxy, timer, trigger, knx)

in which dependency: (room = local and user independent / user = user dependent and therefore effective in all rooms used by the user)

command type: (select, revert, volume, mute etc.)

what is selected: (user, source etc.)

Subsequent commands are separated with a semicolon ;

@user.1:user:select:@favorite;stream:1

Here the user favorites of user 1 are called and after the semicolon the first user favorite is selected from the list.

2.1.5 Sources

User Favoriten (Favroite Service)

The Multiuser 3.0 system is geared towards the user and his freely definable user favorites.

All sources provided by the Multiuser Server and the amplifiers with local inputs are available to the user. These sources can be saved individually per user under the User Favorites (Service Y=Favorite).

If you want to call up and control sources in the Multiuser 3.0 system, the easiest way is to communicate with the User Favorites. Conversely, this means that all sources (including local inputs) that you want to address are first added to the User Favorites.

Example:

The desired iRadio station is stored in the fourth position under the User Favorites of User 1. The call works as follows:

\$u.user1:user:select:\$y.favorite;stream:4 (*Alias*)

#U-ID:user:select:#Y-ID;stream:4 (*Identifier*)

Local sources (Source Service)

Local sources, such as analog and digital audio inputs or the HDMI/ARC input as well as Bluetooth, can also be called up directly. It makes sense to save these local inputs under the User Favorites and then call them up as described above. However, there are also constellations where a direct source call can be helpful, for example with audio or source triggers.

For this, the corresponding local input is directly addressed with the Alias, Identifier or Magic call.

M500/M300

stream :analog
:analog2
:coaxial
:optical
:tuner
:bluetooth
:arc

V219/M30

stream :analog
:coaxial
:optical

Example:

The Analog 2 input from a STUDIOMASTER M500 is called user-independently:

\$r.m500:room:select:\$s.m500;stream:analog2 (*Alias*)

#R-ID:room:select:#S-ID;stream:analog2 (*Identifier*)

3 Text protocol commands

3.1 Room and zone commands

For rooms and zones, although separate services, the same commands apply.

All commands are written one after the other without spaces; e.g.: `$r.room:room:volume:save:9`

The commands below are in alias format. Instead of the alias, the identifier or, depending on the constellation, the magic placeholder could also be used.

<i>\$r.room/\$z.zone</i>	:room	:select	:\$u.user/\$y.favorite/\$s.source
		:revert	:[\$y.favorite/\$s.source/\$u.user]
		:refer	:\$y.favorite/\$s.source/\$u.user
		:off/on	
<hr/>			
	:volume	:set	:0..40
	:volume	:save	:1..9
	:volume	:load	:1..9
	:volume	:plus	
	:volume	:minus	
	:volume	:auto ¹	:plus
	:volume	:auto ¹	:minus
	:volume	:auto ¹	:stop
	:volume	:relative	:plus
	:volume	:relative	:minus
<hr/>			
	:mute	:on	
	:mute	:off	
	:mute	:toggle	
<hr/>			
	:balance	:set	:-10..10
	:balance	:save	:1..9
	:balance	:load	:1..9
	:balance	:plus	
	:balance	:minus	
<hr/>			
	:bass	:set	:-12..12
	:bass	:save	:1..9
	:bass	:load	:1..9
	:bass	:plus	
	:bass	:minus	
<hr/>			
	:treble	:set	:-12..12
	:treble	:save	:1..9
	:treble	:load	:1..9
	:treble	:plus	
	:treble	:minus	

\$ = alias
 # = numeric identifier
 @ = magic identifier
 [] = optional parameter
 ⚙ = user defined command, e.g. from proxy server

¹ **volume:auto:plus/minus** starts an incremental/decremental change of the volume from the current value, till the maximum/ minimum is achieved or the command **volume:auto:stop** is set.

Example:

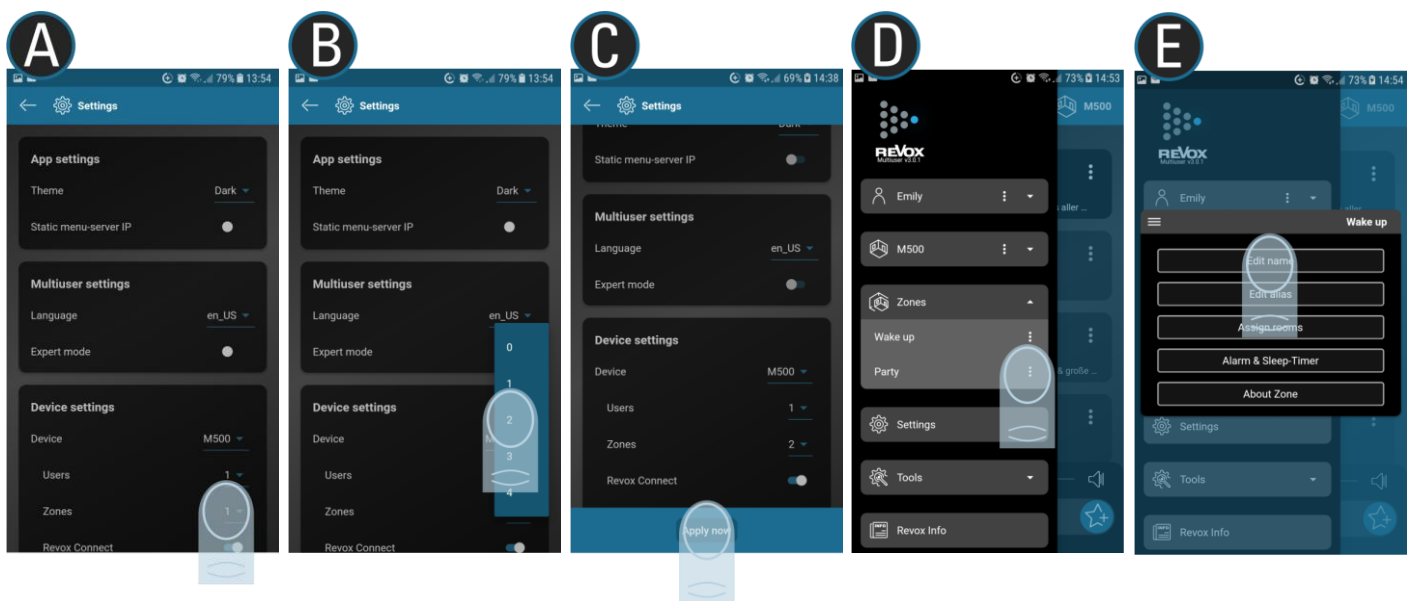
(The Room Identifier can also be replaced by a Zone Identifier or an Alias)

Aktion	Textbefehl	Beschreibung
Volume up	#R-ID-:room:volume:plus	increases the room volume by one step (0-40)
Volume down	#R-ID-:room:volume:minus	decreases the room volume by one step (0-40)
Set volume	#R-ID-:room:volume:set:0-40	sets the room volume to a fixed value (0-40)
Mute on	#R-ID-:room:mute:on	Mute on
Mute off	#R-ID-:room:mute:off	Mute off

3.2 Configuring zones

In the Multiuser system, rooms or amplifiers can be grouped into zones. Zones can also overlap, i.e. a room can belong to several zones. The zones are created in the Multiuser 3.0 App and their number depends on the device. Currently, a STUDI-OMASTER M300/ M500 can provide a maximum of 4 zones, a Multiuser Server V400 a maximum of 16 zones. The zones can be addressed within the app, but also via the Multiuser text protocol. For this purpose, each zone receives a **name** and an **alias**. The assignment of the rooms is also done in the app; so you have the possibility to change the associated rooms of a zone quickly.

The figures show the zone definition in the Multiuser app.



Calling zones: #Z-ID / \$z.zonealias / @zone.1 – 9

The zone call forwards the command that follows it to all rooms that belong to the corresponding zone. The rooms do not react directly to the command, but the system first translates the command into the respective room commands, which are then executed and confirmed.

Example

Action	Text command	Description
User 1	#Z-ID-:room:select:@user.1	all rooms of this zone are assigned to user 1
Set volume	#Z-ID-:room:volume:set:10	all rooms of zone2 are set to the volume value 10

3.2.1 User commands

The following commands are shown in alias format. Instead of the alias, the identifier or, depending on the constellation, the magic placeholder could also be used.

```
$r.room/$z.zone/$u.user :user :select :$y.favorite/$s.source
                        :user :revert1 :[$y.favorite/$s.source]
                        :user :refer2 :$y.favorite/$s.source/$u.user
                        :user :off
                        :volume :plus
                        :volume :minus
```

¹revert : this command always selects the previous source that was active in a room. If no other source was selected before the currently active source, the amplifier goes into standby.

²refer : Assigns the user or source currently assigned to room **B** to room **A**.

Example:

Action	Textbefehl	Description
Stream favorite	#R-ID-:user:select:@favorite	Room selects the user favorites from the already logged in user
Revert	#R-ID:user:revert	The room returns to the previously heard source of the user

3.2.2 Control commands

The commands below are shown in alias format. Instead of the alias, the identifier or, depending on the constellation, the magic placeholder could also be used.

```
$r.room/$z.zone/$u.user
$y.favorite/$s.source :stream :$y.favorite/$s.source/1..99/$alias
                     :play :1...999/$alias
                     :pause :on/off/toggle
                     :next :[favorite]1
                     :previous :[favorite]1
                     :on
                     :off
                     :system :off
```

Beispiele:

Action	Text command	Description
Next	#R-ID-:next	selects next track (playlist) / next source (local inputs or user favorites without list content)
Previous	#R-ID-:previous	selects the previous track (playlist) / source (local inputs or user favorites without list content)
Next favorite	#R-ID:next:favorite	selects the next User Favorite (folder and playlist contents are ignored)
Previous favorite	#R-ID:previous:favorite	selects the previous User Favorite (folder and playlist contents are ignored)

¹The next/previous standard command selects the next or previous entry of a list (user favorite or content of an album/playlist or local input). If the last entry of a list (playlist or user favorite folder) is playing, the playback jumps to the next user favorite with a "next" command.

If you want to jump forward or backward in the User Favorites list, regardless of whether a User Favorite has list content or not, use the following command: **\$r.room:next:favorite**

3.2.3 User Favorite commands

The following commands are shown in alias format. Instead of the alias, the identifier or, depending on the constellation, the magic placeholder could also be used.

Simple User Favorite call, the last heard User Favorite plays

```
$r.room/$z.zone/$u.user :user :select :$y.favorite
```

Targeted user favorites call

```
$r.room/$z.zone/$u.user :user :select :$y.favorite ;stream :1..99
$r.room/$z.zone/$u.user :user :select :$y.favorite ;stream :$alias
```

Example:

Action	Text command	Description
User play favorite	#U-ID-:user:select: @favorite.?	User selects his User Favorites, the last heard User Favorite plays
User play favorite no 9	#U-ID-:user:select : @favorite;stream:9	User selects his User Favorites, User Favorite number 9 plays
User play favorite pop	#U-ID-:user:select: @favorite;stream:\$pop	User selects his User Favorites, User Favorite with the alias pop plays

3.2.4 User Favorites Folder Commands

The following commands are shown in alias format. Instead of the alias, the identifier or, depending on the constellation, the magic placeholder could also be used.

User favorites folder call, the last listened content in the folder plays

```
$r.room/$z.zone/$u.user :user :select :$y.favorite ;stream :1-99
$r.room/$z.zone/$u.user :user :select :$y.favorite ;stream :$alias
```

Targeted user favorites folder and content call:

```
$r.room/$z.zone/$u.user :user :select :$y.favorite ;stream :1-99 ;play :1..64
$r.room/$z.zone/$u.user :user :select :$y.favorite ;stream :$alias ;play :$alias
```

Example:

Action	Text command	Description
User play favorite folder	#U-ID-:user:select :@favorite;stream:1	User selects his User Favorites, User Favorites folder plays (if the first entry is a folder)
User play favorite 1 (folder) and content no 3	#U-ID:user:select:@favorite ;stream:1;play:3	User selects his User Favorites, User Favorites folder is selected, content number 3 plays
User play favorite folder pop and content dj	#U-ID:user:select:@favorite;stream:\$pop;play:\$dj	user selects his user favorites, user favorites folder with alias pop is selected, content with alias dj is playing

3.2.5 Trigger commands

The following commands are shown in alias format. Instead of the alias, the identifier or, depending on the constellation, the magic placeholder could also be used.

```
$t.trigger          :trigger      :on
                   :trigger      :off
                   :trigger      :toggle
                   :trigger      :pulse      : [seconds]
                   -----
                   :[🌀]
                   :[🌀]          :break
                   :[🌀]          :step      :1..16
```

🌀 Each amplifier has 4 action sequencer slots in the trigger settings of the Configurator. The action sequencers allow a time-defined sequence of Multiuser text commands (sequences/macros). Each Action Sequencer can be assigned an individual name.

Example:

Action	Text command	Description
Start sequence	\$t.trigger:tv_on	Starts the individually created sequencer with the name tv_on
Break sequence	\$t.trigger:tv_on:break	Cancels the individually created sequencer with the name tv_on

3.2.6 Proxy commands

The following commands are shown in alias format. Instead of the alias, the identifier or, depending on the constellation, the magic placeholder could also be used.

```
$p.proxy           :[🌀]
                   :[🌀]          :break
                   :[🌀]          :step      :1..16
```

🌀 Each amplifier in the multiuser system provides as many proxy places as the number of local inputs. The proxy service in the Multiuser system allows the control of external devices that can be addressed via IP command, such as TV sets, BluRay players or IP-IR /IP-Serial converters. If you send a proxy command, it will be forwarded via TCP/ UDP to the corresponding IP address and port and interpreted there. In addition, each proxy place provides a total of 4 additional action sequencers.

Example:

Action	Text command	Description
Start proxy	\$p.cd:on	Triggers the IP command "on" stored in the proxy
Break sequence	\$p.cd:cd_off:break	Cancels the individually created sequencer with the name cd_off

3.2.7 Timer commands

The commands below are shown in alias format. Instead of the alias, the identifier or, depending on the constellation, the magic placeholder could also be used.

```
$x.timer      :[🕒]
               :[🕒]      :break
               :[🕒]      :step   :@1..16
               :[🕒]      :on
               :[🕒]      :off
               :[🕒]      :test
```

Multuser servers have 4 action timers, each of which also contains 4 action sequencers that can be set individually in the Configurator. In the action input field of the Configurator, a multitude of options are available with the multuser text configuration. These timers are not suitable for daily wake-up, as they can only be operated in the Configurator. Rather, they are to be used for general, time-dependent events.

For the morning wake-up, each multuser amplifier has room-specific, individual alarm clocks that can be set via the multuser app. These alarms are not displayed in the Configurator and cannot be called up using the text log.

Example:

Action	Text command	Description
Activate event1	<code>\$x.timer:event_1:on</code>	Activates the first timer
Break sequence	<code>\$x.timer:music_off:break</code>	Cancels the individually created timer sequencer with the name music_off

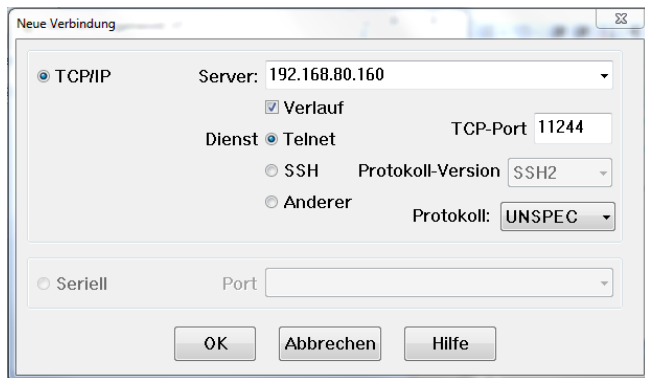
4 Action Server

The Multiuser Action Server runs on port 11244 of the Multiuser Server. Via this Action Server, third-party devices or external controllers can control the Multiuser system using the Multiuser text protocol (IP telegrams). In addition to the Multiuser Server, which provides the fully comprehensive Action Server for the entire Multiuser, each Multiuser amplifier also has a small Action Server.

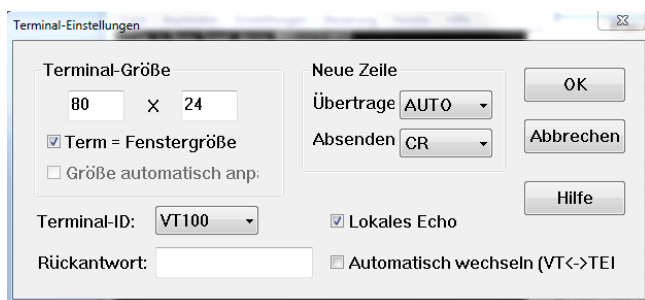
A terminal program is used to connect to the Action Server via TCP/IP and Telnet. Recommended terminal programs are PuTTY and TeraTerm.

In the following description TeraTerm is used.

To connect to the server via TCP/IP, enter the corresponding IP address of the server in the terminal program, in this case 192.168.80.160, and the port of the ActionServer. The port number is set to port 11244.



Once the connection is successfully established, you go to the terminal settings and set "Transmit" (Übertrage) to AUTO, in order to achieve the correct line formatting.



The Return key has to be pressed once to access the Communication menu.

```

Welcome to Revox Multiuser device #DC0EE403AE13A0
-----
? = help
1 = list:devices
2 = list:services
8 = status:on
9 = status:off

```

? → Creates a line of all available actions
 1 → Creates a list of all devices
 2 → Creates a list of all services
 8 → Switches the status messages on
 9 → Switches the status messages off

4.1 Help list

? → Creates a line of all available actions.

```
Multiuser actions (please read the Revox Multiuser manual for more information)
-----
? - #Identifier, $Alias or @Magic
@ - Numeric value or string
-> - See below
[] - Optional parameter
:* - User defined action

?room/?user -----+ :room -----+ :select ----- :?favorite/?user/?source
|                                     +- :refer ----- :?user/?room
|                                     +- :revert ----- :[?user/?source]
|                                     +- :off ----- :[?user/?source]
|                                     +- :on
|                                     +- :volume ----- :->parameter 0..40
|                                     +- :balance ----- :->parameter -10..10
|                                     +- :bass ----- :->parameter -12..12
|                                     +- :treble ----- :->parameter -12..12
|                                     +- :mute ----- :on/off/toggle
|
|                                     +- :user -----+ :select ----- :?favorite/?source
|                                     +- :refer ----- :?user/?room
|                                     +- :revert ----- :[?source]
|                                     +- :off ----- :[?source]
|                                     +- :volume ----- :plus/minus

?room/?user/?source +- :stream ----- :->service
|                                     +- :play ----- :[@track]
|                                     +- :pause ----- :on/off/toggle
|                                     +- :next/previous
|                                     +- :system ----- :off

?favorite           +- :stream      -- :->[@favorite-index]
|                                     +- :play ----- :->[@favorite-index]/[@playlist_index]
|                                     +- :next/previous

?trigger -----+ :trigger ----+ :on/off/toggle
|                                     +- :pulse ----- :@seconds
|
|                                     +- :* -----+ :[run]
|                                     +- :break
|                                     +- :step:@1..16

?proxy -----+ :* -----+ :[run]
|                                     +- :break
|                                     +- :step:@1..16

?timer -----+ :* -----+ :[run]
|                                     +- :break
|                                     +- :step:@1..16
|                                     +- :on/off
|                                     +- :test

-> parameter
-----
+- :set ----- :@min..max
+- :save ----- :@1..9
+- :load ----- :@1..9
+- :plus/minus
+- :auto ----- :plus/minus/stop
+- :relative ----- :plus/minus

-> source-service (V219)
-----
+- :analog
+- :coaxial
+- :optical

-> source-service (M500)
-----
+- :coaxial
+- :optical
+- :tuner
+- :bluetooth
+- :analog
+- :analog2
+- :arc
```

4.2 Device list

[1] → Creates a list of all devices

	id	ip address	name	version
1	#0C0EE403AE13A0	192.168.13.178	STUDIOMASTER	138-03.00.15
2	#0C0EE408A5D7A0	192.168.13.184	STUDIOMASTER	138-03.00.15
3	#000224DD062760	192.168.13.149	V400	138-03.00.15
4	#0C0EE408C0A240	192.168.13.181	M30-AMP	030-03.00.15
5	#0C0EE408C09860	192.168.13.119	M30-AMP	030-03.00.15

4.3 Service list

[2] → Creates a list of all available services and their identifier, alias and name

	id	alias	name	type
1	#DC0EE403AE13A0	\$DC0EE403AE13A0	STUDIOMASTER	500.82.device
2	#RC0EE403AE13A0	\$r.sz	Sitzung	500.82.room
3	#SC0EE403AE13A0	\$s.sz	Sitzung	500.82.local
4	#SC0EE403AE13A1	\$SC0EE403AE13A1	Peter's Musik	500.82.server
5	#YC0EE403AE13A1	\$YC0EE403AE13A1	Peter's Favoriten	500.82.favorite
6	#SC0EE403AE13A2	\$SC0EE403AE13A2	Sarah's Musik	500.82.server
7	#YC0EE403AE13A2	\$YC0EE403AE13A2	Sarah's Favoriten	500.82.favorite
8	#UC0EE403AE13A1	\$u.user1	Peter	500.82.user
9	#UC0EE403AE13A2	\$UC0EE403AE13A2	Sarah	500.82.user
10	#XC0EE403AE13A0	\$XC0EE403AE13A0	Timer	500.82.timer
11	#XC0EE403AE13A1	\$XC0EE403AE13A1	Timer	500.82.timer
12	#PC0EE403AE13A0	\$PC0EE403AE13A0	Proxy	500.82.proxy
13	#PC0EE403AE13A1	\$PC0EE403AE13A1	Proxy	500.82.proxy
14	#PC0EE403AE13A2	\$PC0EE403AE13A2	Proxy	500.82.proxy
15	#PC0EE403AE13A3	\$PC0EE403AE13A3	Proxy	500.82.proxy
16	#PC0EE403AE13A4	\$PC0EE403AE13A4	Proxy	500.82.proxy
17	#PC0EE403AE13A5	\$PC0EE403AE13A5	Proxy	500.82.proxy
18	#PC0EE403AE13A6	\$PC0EE403AE13A6	Proxy	500.82.proxy
19	#PC0EE403AE13A7	\$PC0EE403AE13A7	Proxy	500.82.proxy
20	#TC0EE403AE13A0	\$t.sz	Trigger	500.82.trigger
21	#SC0EE403AE13AF	\$SC0EE403AE13AF	AirPlay	500.82.airplay
22	#DC0EE408A5D7A0	\$DC0EE408A5D7A0	STUDIOMASTER	300.82.device
23	#RC0EE408A5D7A0	\$RC0EE408A5D7A0	Laden M300	300.82.room
24	#SC0EE408A5D7A0	\$SC0EE408A5D7A0	Laden M300	300.82.local
25	#XC0EE408A5D7A0	\$XC0EE408A5D7A0	Timer	300.82.timer
26	#XC0EE408A5D7A1	\$XC0EE408A5D7A1	Timer	300.82.timer
27	#PC0EE408A5D7A0	\$PC0EE408A5D7A0	Proxy	300.82.proxy
28	#PC0EE408A5D7A1	\$PC0EE408A5D7A1	Proxy	300.82.proxy
29	#PC0EE408A5D7A2	\$PC0EE408A5D7A2	Proxy	300.82.proxy
30	#PC0EE408A5D7A3	\$PC0EE408A5D7A3	Proxy	300.82.proxy
31	#PC0EE408A5D7A4	\$PC0EE408A5D7A4	Proxy	300.82.proxy
32	#PC0EE408A5D7A5	\$PC0EE408A5D7A5	Proxy	300.82.proxy
33	#PC0EE408A5D7A6	\$PC0EE408A5D7A6	Proxy	300.82.proxy
34	#PC0EE408A5D7A7	\$PC0EE408A5D7A7	Proxy	300.82.proxy
35	#ZC0EE408A5D7A0	\$ZC0EE408A5D7A0	Zone	300.82.zone
36	#TC0EE408A5D7A0	\$TC0EE408A5D7A0	Trigger	300.82.trigger
37	#SC0EE408A5D7AF	\$SC0EE408A5D7AF	AirPlay	300.82.airplay

4.4 Status messages

[8] → Switches on the status messages

[9] → Switches off the status messages

```

8
$u.sarah:status:name=Sarah:source=$YC0EE403AE13A2:stream=favorite:info=Meine Favoriten
$SC0EE403AE13AF:status:name=AirPlay:stream=unknown:info=:cover=http://192.168.13.178//pictures/none.png
$u.v400:status:name=Test V400:source=$Y00224DD062760:stream=favorite:info=Meine Favoriten
$r.sz:status:name=Sitzung:source=$unknown:user=$unknown:power=off:mute=off:volume=0:stream=unknown:info=
$YC0EE403B384D1:status:name=M500's Favoriten:source=$unknown:stream=favorite
$s.sz:status:name=Sitzung:stream=coaxial:info=Coaxial Input:cover=http://192.168.13.178//coverart/local/local_icon_1.png
$Z00224DD062760:status:name=EG:source=$unknown:user=$unknown:power=off:mute=off:volume=13:stream=unknown:info=
$m.500:status:name=M500:source=$YC0EE403B384D1:stream=favorite:info=Meine Favoriten
$SC0EE408A5D7A0:status:name=Laden M300:stream=coaxial:info=Coaxial Input:cover=http://192.168.13.184//coverart/local/local_icon_1.png
$S00224DD062760:status:name=Test V400's Musik:stream=unknown:info=:cover=http://192.168.13.149//pictures/none.png
$SC0EE408C0A240:status:name=M30 Testrack:stream=analog:info=Analog Input:cover=http://192.168.13.181//coverart/local/analog_in.png
$SC0EE403AE13A1:status:name=Peter's Musik:stream=airable.radios:info=Radio Swiss Pop:cover=http://static.airable.io/29/07/458975.png
$SC0EE403B384D1:status:name=AirPlay:stream=unknown:info=:cover=http://192.168.13.139//pictures/none.png
$YC0EE408A5D7A0:status:name=Zone:source=$unknown:user=$unknown:power=off:mute=off:volume=13:stream=unknown:info=
$Y00224DD062760:status:name=Test V400's Favoriten:source=$unknown:stream=favorite
$r.m500:status:name=M500:source=$YC0EE403AE13A1:user=$u.peter:power=on:mute=off:volume=5:stream=airable.radios:info=Radio Swiss Pop
$SC0EE408A5D7AF:status:name=AirPlay:stream=unknown:info=:cover=http://192.168.13.184//pictures/none.png
$RC0EE408C09860:status:name=Büro Christian:source=$unknown:user=$unknown:power=off:mute=off:volume=13:stream=unknown:info=
$s.sz:status:name=M500:stream=coaxial:info=Coaxial Input:cover=http://192.168.13.139//coverart/local/local_icon_1.png
$SC0EE408C0A24F:status:name=AirPlay:stream=unknown:info=:cover=http://192.168.13.181//pictures/none.png
$RC0EE408A5D7A0:status:name=Laden M300:source=$YC0EE403AE13A1:user=$u.peter:power=on:mute=off:volume=10:stream=airable.radios:info=Radio Swiss
Pop
$SC0EE403B384D1:status:name=M500's Musik:stream=unknown:info=:cover=http://192.168.13.139//pictures/none.png
$YC0EE403AE13A1:status:name=Peter's Favoriten:source=$SC0EE403AE13A1:stream=airable.radios
$SC0EE408C09860:status:name=Büro Christian:stream=analog:info=Analog Input:cover=http://192.168.13.119//coverart/local/analog_in.png
$SC0EE403AE13A2:status:name=Sarah's Musik:stream=unknown:info=:cover=http://192.168.13.178//pictures/none.png
$YC0EE403AE13A2:status:name=Sarah's Favoriten:source=$unknown:stream=favorite
$r.m30:status:name=M30 Testrack:source=$unknown:user=$unknown:power=off:mute=on:volume=2:stream=unknown:info=
$u.peter:status:name=Peter:source=$YC0EE403AE13A1:stream=airable.radios:info=Radio Swiss Pop
$SC0EE408C0986F:status:name=AirPlay:stream=airplay:info=AirPlay:cover=http://192.168.13.119//coverart/airplay/airplay.png

```

4.4.1 Source status (Source)

Example: **\$SC0EE403AE13A1:status:name=Peter'sMusik:stream=airable.radios:info=Radio Swiss Pop:cover= http\://static.airable.io/29/07/458975.png**

- status = alias name
- name = source name
- stream = Info about favorite source or local input
- info = Plain text name as short info
- URL = URL of Covers

4.4.2 Room status (Room)

Example: **\$r.m500:status:name=M500:source=\$YC0EE403AE13A1:user=\$u.peter:power=on:mute=off:volume=5:stream=airable.radios:info=Radio Swiss Pop**

- status = alias name
- name = room name
- source = favorites or source service
- user = logged in user
- power = on / off
- mute = on/ off
- volume = 0 - 40 volume units
- stream = Info about favorite source or local input
- info = Plain text name as short info

4.4.3 User status (User)

Example: `$u.peter:status:name=Peter:source=$YC0EE403AE13A1:stream= airable.radios:info=Radio Swiss Pop`

- status -> Alias name
- name -> User name
- source -> favorites or source service
- stream -> Info about favorite source or local input
- info -> Plain text name as short info

4.5 User favorites list, content queries

If you want user-specific information about the User Favorites list (including folders), this can also be queried via the Action Server. To do so, call the desired User Favorites Service Identifier of the user and complete it with `:menu:list`

`[#Y-ID:menu:list]` → lists the information of the user favorites list

```
#YC0EE403AE13A2:menu:list
$YC0EE403AE13A2:favorite:list:count=3
$YC0EE403AE13A2:favorite:list:index=1:type=Internet:name=BBC Radio 1:cover=http://static.airable.io/11/37/265191.png
$YC0EE403AE13A2:favorite:list:index=2:type=Internet-container:name=Studiomaster:cover=http://resources.tidal.com/images/e8f241b2/f038/4046/8f30/499bdd630ee8/160x160.jpg
$YC0EE403AE13A2:favorite:list:index=3:type=Folder:name=Radio
$YC0EE403AE13A2:favorite:folder-list:count=2
$YC0EE403AE13A2:folder:list:index=1:type=Internet:name=Radio 1:cover=http://static.airable.io/27/16/676505.png
$YC0EE403AE13A2:folder:list:index=2:type=Internet:name=BBC Radio 1:cover=http://static.airable.io/11/37/265191.png
```

4.5.1 Number of user favorites

`#YC0EE403AE13A2:favorite:list:count=3`

- count=3 -> 3 User favorite entries

4.5.2 User favorite type 1: single source

`:index=1:type=Internet:name=BBC Radio 1:cover:http`

- index=1 -> Position, at which position is this entry in the User Favorites list: 1. position
- type=internet -> User favorites type: single Internet source from multiuser server
- name=BBC -> Name of user favorite
- cover=http -> URL of Icon

4.5.3 User Favorit Typ 2: Container

`:index=2:type=Internet:name=BBC Radio 1:cover:http`

- index=2 -> Position, at which position is this entry in the User Favorites list: 2. position
- type=internet-container -> User favorite type: Internet Container, as TIDAL Playlist with content
- name=Studiomaster -> Name of user favorites
- cover=http -> URL of Icon

4.5.4 User Favorit Typ 3: Ordner

:index=3:type=Folder:name=Radio

:folder-list:count=2

:folder:list:index=1:type=Internet:name=Radio 1:cover:http

- index=3 -> Position, at which position is this entry in the User Favorites list: 3. position
- type=Folder -> User Favorites Type: Folder with content
- name=Radio -> Name of the user favorite folder
- folder-list:count:2 -> Folder with content: two entries in the folder
- folder:list:index=1:type... -> Detailed information about the respective sources in the folder



REVOX
Studio Sound Quality

Revox Deutschland GmbH | Am Krebsgraben 15 | D-78048 VS-Villingen | Tel.: +49 7721 8704 0 | info@revox.de | www.revox.com
Revox (Schweiz) AG | Eichwatt 5 | CH-8105 Regensdorf | Tel.: +41 44 871 66 11 | info@revox.ch | www.revox.com
Revox Handels GmbH | Josef-Pirchl-Straße 38 | AT-6370 Kitzbühel | Tel.: +43 5356 66 299 | info@revox.at | www.revox.com