

# SAGE2 Command-line Operation

[Starting](#)

[Commands](#)

[External scripts](#)

[Shell](#)

[Command](#)

[Upload](#)

## Starting

When using the source code version of SAGE2, you can start the SAGE2 server with the following options:

- **Minimal command**
  - `node server.js`
- **Options**
  - `-h, --help` outputs usage information
  - `-V, --version` outputs the version number
  - `-i, --no-interactive` non-interactive prompt
  - `-f, --configuration <file>` specifies a configuration file
  - `-l, --logfile [file]` specifies a log file
  - `-q, --no-output` quiet mode, no output
  - `-s, --session [name]` loads a session file (last session if omitted)
  - `-t, --track-users [file]` enables user interaction tracking (specified file indicates users to track)
  - `-p, --password <pwd>` sets the password to connect to SAGE2 session

## Commands

Once the server is started, it enters a read–eval–print loop (REPL), which accepts commands with arguments. It supports history (arrow keys up and down) and navigation on the line (arrow keys left and right, or Emacs-style shortcuts). Commands are: *help*, *kill*, *apps*, *clients*, *streams*, *clear*, *tile*, *fullscreen*, *save*, *load*, *open*, *resize*, *moveby*, *moveto*, *assets*, *regenerate*, *hideui*, *sessions*, *update*, *version*, *exit*.

- **help** list available commands
- **kill** close application: argument is *appid*
  - `> kill app_0`

- **apps** list running applications (index, appid, name, position, description, version and author)
  - > apps
  - 0: **app\_0** welcome [737x491 +112+32] welcome (v1.0.0) by Luc Renambot <renambot@gmail.com>
  - 1: **app\_1** photos [655x491 +153+558] Photo Slideshow (v1.3) by Andrew Johnson <lenzman@me.com>
  - 2: **app\_2** movie\_player [873x491 +1004+32] SAGE2\_Trailer.mp4 (v2.0.0) by SAGE2
- **clients** list connected clients and type
  - > clients
  - Clients (6)
  - -----
  - 0: 127.0.0.1:51520 (sageUI)
  - 1: 127.0.0.1:51530 (display -1) master
  - 2: 127.0.0.1:51749 (sageUI)
  - 3: 127.0.0.1:51751 (consoleManager)
  - 4: 127.0.0.1:51754 (audioManager)
  - 5: 127.0.0.1:51756 (display 0)
- **streams** list media and block streams
  - > streams
  - Block streams (0)
  - -----
  - Media streams
  - -----
  - 0: 127.0.0.1:53246|0 media\_stream #Luc2: Shared Screen
- **clear** close all running applications
  - > clear
- **tile** layout all running applications
  - > tile
- **fullscreen** maximize one application: appid
  - > fullscreen app\_0
- **save** save state of running applications into a session: name optional
  - > save
    - saved session file to default.json
  - > save toto
    - saved session file to toto.json
- **load** load a session and restore applications: name optional
  - > load
    - reading session from default.json
  - > load toto
    - reading session from toto.json
- **open** open a file or an application (see 'assets' for urls): open file\_url [0.5, 0.5]

- > open /uploads/apps/googlemaps
  - opens at origin, top left
- > open /uploads/apps/googlemaps 0.5 0.5
  - opens in center of wall
- > open /uploads/apps/googlemaps 200 200
  - opens with center of window at 200,200 pixels
- **resize** resize a window: appid width height. Values between > 0 and < 1 are used as percentage of whole wall. Otherwise in pixels.
  - > resize app\_0 0.5 0.5
    - resize application to be half width and half height
  - > resize app\_0 200 300
    - resize application to be 200x300 pixels
  - > resize app\_0 200
    - resize application to be width 200 and height derived from aspect ratio
- **moveby** shift a window: appid dx dy
  - > moveby app\_0 200 200
    - move application left and down by 200 pixels
- **moveto** move a window: appid x y
  - > moveto app\_0 200 200
    - move application to 200,200 pixels absolute coordinate
- **assets** list the assets in the file library (name, type, URL, size in bytes)
  - > assets
    - Assets> 0 car\_threejs application/custom /uploads/apps/car\_threejs
    - Assets> 1 googlemaps application/custom /uploads/apps/googlemaps
    - Assets> 2 QR.png image/png /uploads/images/QR.png 1198
    - Assets> 3 SAGE2\_collaborate\_com2014.pdf application/pdf /uploads/pdfs/SAGE2\_collaborate\_com2014.pdf 2382611
  - URL can be used in 'open' command
- **regenerate** regenerates the assets icons (used in case of update or corruption)
  - > regenerate
- **hideui** hide/show/delay the user interface
  - > hideui
    - hide / restore user interface elements on the wall
  - > hideui 20
    - hide user interface elements after 20 seconds of inactivity
- **sessions** list the available sessions
  - > sessions
    - Sessions
    - 0: Name: **42**                      Size: 56KB      Date: 2015/10/19 18:34:19
    - 1: Name: **default**                  Size: 19KB      Date: 2015/11/02 14:25:56
    - 2: Name: **test**                      Size: 19KB      Date: 2015/10/22 18:08:32
  - the 'name' can be used in the 'load' command
- **update** run a git update on the current branch (default 'master')

- > update
  - GIT> Update success - Already up-to-date
- > update
  - GIT> Update success - Updating 931c4da..9e1ed16
  - Fast-forward
  - public/src/radialMenu.js | 160+++++++-----
  - public/src/ui\_builder.js | 2 +
- 
- **version** print SAGE2 version
  - > version
    - Version> base: 0.3.0 branch: master commit: 931c4da 2015/11/01
- **exit** stop SAGE2
  - > exit
    - Session> saved session file default.json
  - session saved automatically in 'default.json' file. Can be reloaded with '-s' parameter at startup or using command 'load default'.

## External scripts

The 'bin' folder contains several scripts to interact with a currently running SAGE2 server. The command are run with the 'node' Javascript interpreter (node scripts.js) or directly executed from the current folder (./script.js).

## Shell

- Equivalent to the REPL mechanism from the server
- Usage:
  - **sage\_shell.js** <url>
  - url is the hostname and secure port number (HTTPS) of your SAGE2 server
- Example;
  - *./sage\_shell.js myserver:443*
  - *./sage\_shell.js localhost:9090*
    - Starting> wss://localhost:9090
    - Client> connecting to wss://localhost:9090
    - Connect> 127.0.0.1:53873 (shell)
    - Initialize> uniqueID 127.0.0.1:53873
    - ...
    - > help
- All commands described above should work

## Command

- Used to evaluate a single command (and quits)
- Usage:
  - **sage\_command.js <url> <command>**
- Example:
  - *./sage\_command.js localhost:9090 clear*
  - Client> sending command: clear
  - Starting> wss://localhost:9090
  - Client> connecting to wss://localhost:9090
  - Connect> 127.0.0.1:53936 (commandline)
  - Initialize> uniqueID 127.0.0.1:53936
  - SAGE2> display configuration 1920 1080
- All commands described above should work

## Upload

- Uploads and opens a file from the local computer to the SAGE2 server
- Usage:
  - **sage\_upload.js <url> <path to file> [x y] [width height]**
    - position and size can be absolute values in pixels, if > 1.
    - if <= 1, values are treated as percent of the wall display size respectively to total width and height.
    - position of the application is its **center**
- Example:
  - *./sage\_upload.js localhost:9090 ~/Desktop/pluto.jpg*
    - Client> uploading /Users/luc/Desktop/pluto.jpg
    - Starting> wss://localhost:9090
    - Client> connecting to wss://localhost:9090
    - Initialize> uniqueID 127.0.0.1:53953
    - SAGE2> display configuration 1920 1080
    - Upload> success
- Works for supported SAGE2 assets: PDF, images, videos, ...

All the commands can be bundled in a script for automation:

- *./sage\_upload.js localhost:9090 ~/Desktop/img1.jpg*
- *./sage\_upload.js localhost:9090 ~/Desktop/img2.jpg 500 500 0.5 0.5*
- *./sage\_command.js localhost:9090 clear*
- *./sage\_command.js localhost:9090 open /user/images/img1.jpg*
- *./sage\_command.js localhost:9090 open /user/images/img2.jpg*

- `./sage_command.js localhost:9090 fullscreen app_0`
- `./sage_command.js localhost:9090 resize app_0 500`
- `./sage_command.js localhost:9090 moveby app_0 50 50`