

How to start with 3DHOP

Package content, local setup, online deployment

<http://3dhop.net> 15/6/2020

The 3DHOP distribution

Where to find it, what's inside

The 3DHOP distribution package

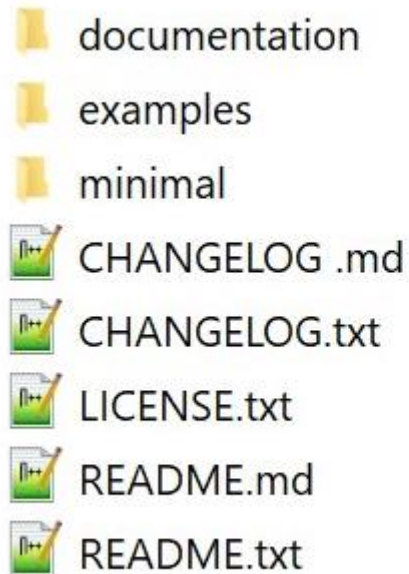
From the page 3dhop.net/download.php it is possible to download the **3DHOP package**.

The package contains the files needed by 3DHOP, plus all the HTML files, 3D models and images used in the tutorials shown in the resources section of the website. The package also contains the tool documentation and other useful resources.

In order to convert your 3D models in the multi-resolution format, you will need to get (from the same download page), also the **NEXUS package**.

Inside the distribution

Let us now take a look inside the 3DHOP distribution:



- **documentation:** the complete documentation of the JavaScript API
- **examples:** all the How-To shown on the 3DHOP website
- **minimal:** a ready-to-use viewer, in a self-contained folder

What is a 3DHOP viewer made of

Basically a set of “web files”:

- ◆ **.html file:** the actual web page, it contains the web page structure and formatting, the text and the other page elements. The 3DHOP components (HTML5 canvas, 3D scene description, loading and interaction scripts, interface elements) are declared here.
- ◆ **.js files:** the “brain” of the 3DHOP viewer, these files contain all the functions that make the viewer work.
- ◆ **.css file(s):** the web page appearance is controlled by these files.
- ◆ **multimedia files:** all the multimedia contents of the web page. 3D models, pictures, video, audio, and so on.

These files have to be accessible on a remote web server: the client computer, using a browser, will open the HTML page and access from there all the other files.

The MINIMAL folder

The MINIMAL folder contains **only** the essential files for a simple but usable 3DHOP viewer:

- /stylesheet a folder containing the 3DHOP css files
- /js a folder containing the JavaScript 3DHOP core
- /skins a folder containing the images used in the interface
- /models a folder containing the 3D model(s)
- 3DHOP_*.html the webpage(s) where the 3DHOP viewer is created

A novice developer should start from this examples and modify it. All the needed files are there, making easy the development and deployment (just load the whole folder on a web server).

Local deployment

Working on my PC

Looking for a web server

3DHOP has been designed to work **over the internet**, with the webpage and data on a remote web server, accessed from the browser through the web.

However, when developing web pages, it is much easier to work locally.

This can be done in two ways:

- ◆ *Install a local web server on your computer*
- ◆ *Force your web browser to load files from your local PC*

Local web server

... using Apache

Apache

Apache HTTP Server (<httpd.apache.org>) is the world's most widely used web server software, and it is a good choice to have a local web server installed on your device.

For an easier installation you can use a web server solution stack package like XAMPP (free, open source and cross-platform).

Just go to

www.apachefriends.org

download the XAMPP package and follow the installation instructions!



The screenshot shows the XAMPP website homepage. The navigation bar includes links for Apache Friends, Download, Add-ons, Hosting, Community, and About, along with a search bar and a language selector (EN). The main heading is "XAMPP Apache + MariaDB + PHP + Perl". Below this, there is a section titled "What is XAMPP?" which describes XAMPP as the most popular PHP development environment and a completely free, easy-to-install Apache distribution containing MariaDB, PHP, and Perl. To the right of the text is a video player titled "Introduction to XAMPP" featuring the XAMPP logo. At the bottom, there are three download buttons: "Download" (with a link to other versions), "XAMPP for Windows 7.4.6 (PHP 7.4.6)", "XAMPP for Linux 7.4.6 (PHP 7.4.6)", and "XAMPP for OS X 7.4.6 (PHP 7.4.6)".

Developing inside the local server

After the web server has been installed, just put the webpages and 3DHOP files you need to work on (e.g. the MINIMAL folder) in the web server folder of your local PC and work directly on those files with a text editor.

To see the page you are developing, just open a web browser and access the files at the local address exposed by your web server (generally <http://localhost>)

The web server way

The use of a local web server is not really difficult and provides the most reliable and comprehensive way to locally develop web pages and web applications (including 3DHOP).

However, it **does** require a certain amount of configuration, manual tweaking work, and time. For this reason, may be difficult for complete neophytes.

Local browser

... using Chromium-based browsers

Browsers and local files

The Chromium-based browsers (like for instance **Google Chrome**, or **Microsoft Edge**...) allows us to avoid installing a web server, loading web pages directly from our local PC.

However, due to security issues loading files from a local path, bypassing the server is an action usually forbidden to the browser.

We need to force our browser to accept local files; if a Chromium-based browser is launched using the parameter:

`--allow-file-access-from-files`

It will accept the local files and work correctly.

Chrome On Windows

- ◆ **Install Chrome on your PC**

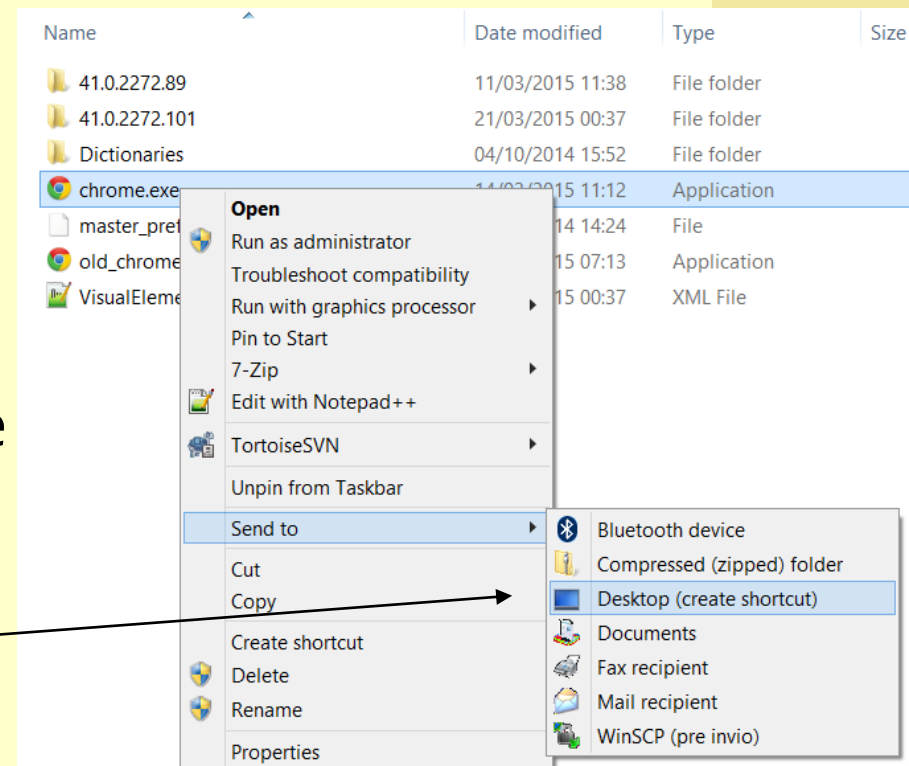
Go to www.google.com/chrome , download the browser version suitable for your platform, and install it.

- ◆ **Look for the file “chrome.exe”**

Search it on your hard disk and go in the folder that contains it (usually C:\Program Files\Google\Chrome).

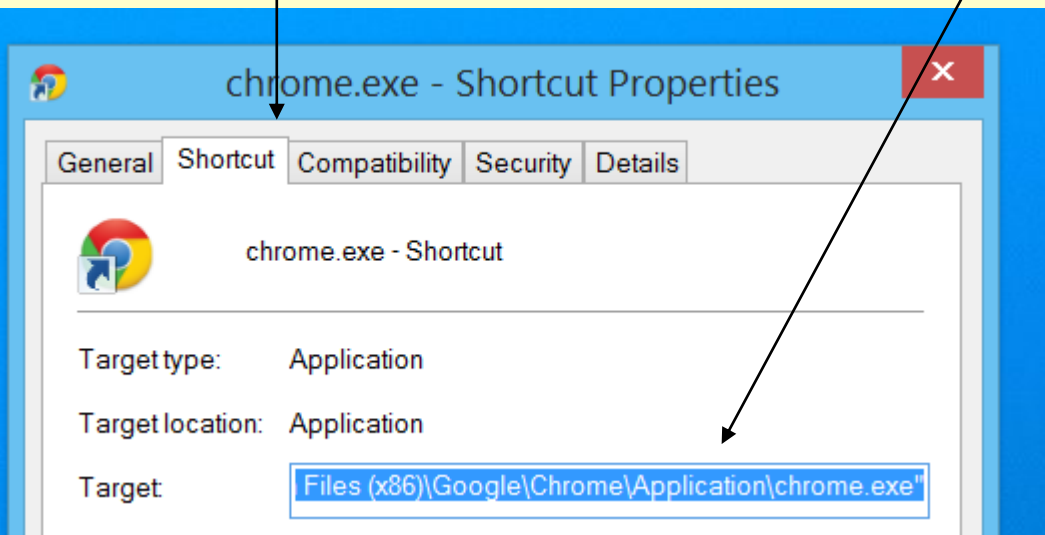
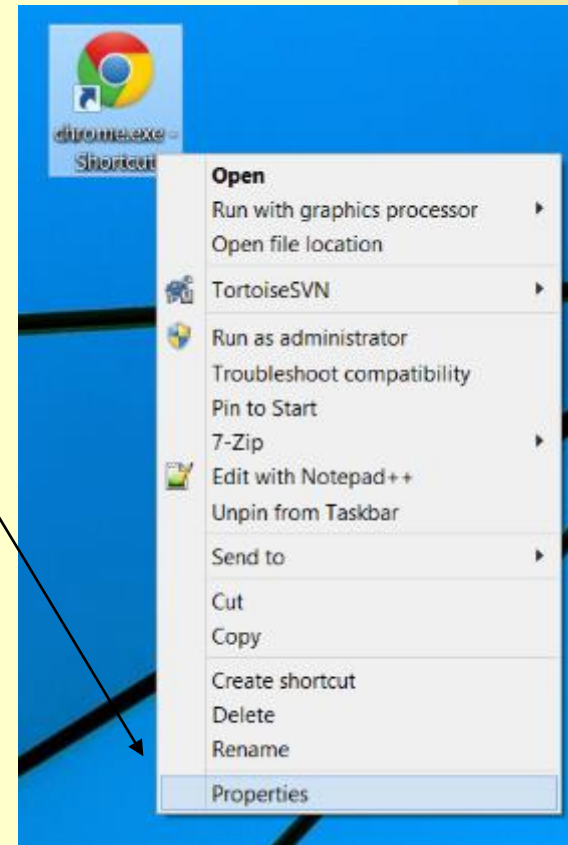
- ◆ **Create a desktop shortcut**

Create a link to chrome.exe and place it on your desktop (in Windows OS just right click on chrome.exe icon and then “Send to Desktop”).



Chrome On Windows

- ◆ **Open the shortcut properties window**
Go to your Desktop, right click on the just created Chrome shortcut icon, and then “Properties”.
- ◆ **Look for the shortcut “Target”**
In the shortcut properties window, go in the “Shortcut” tab and look for the “Target” string.



Chrome On Windows

◆ Edit the “Target” string

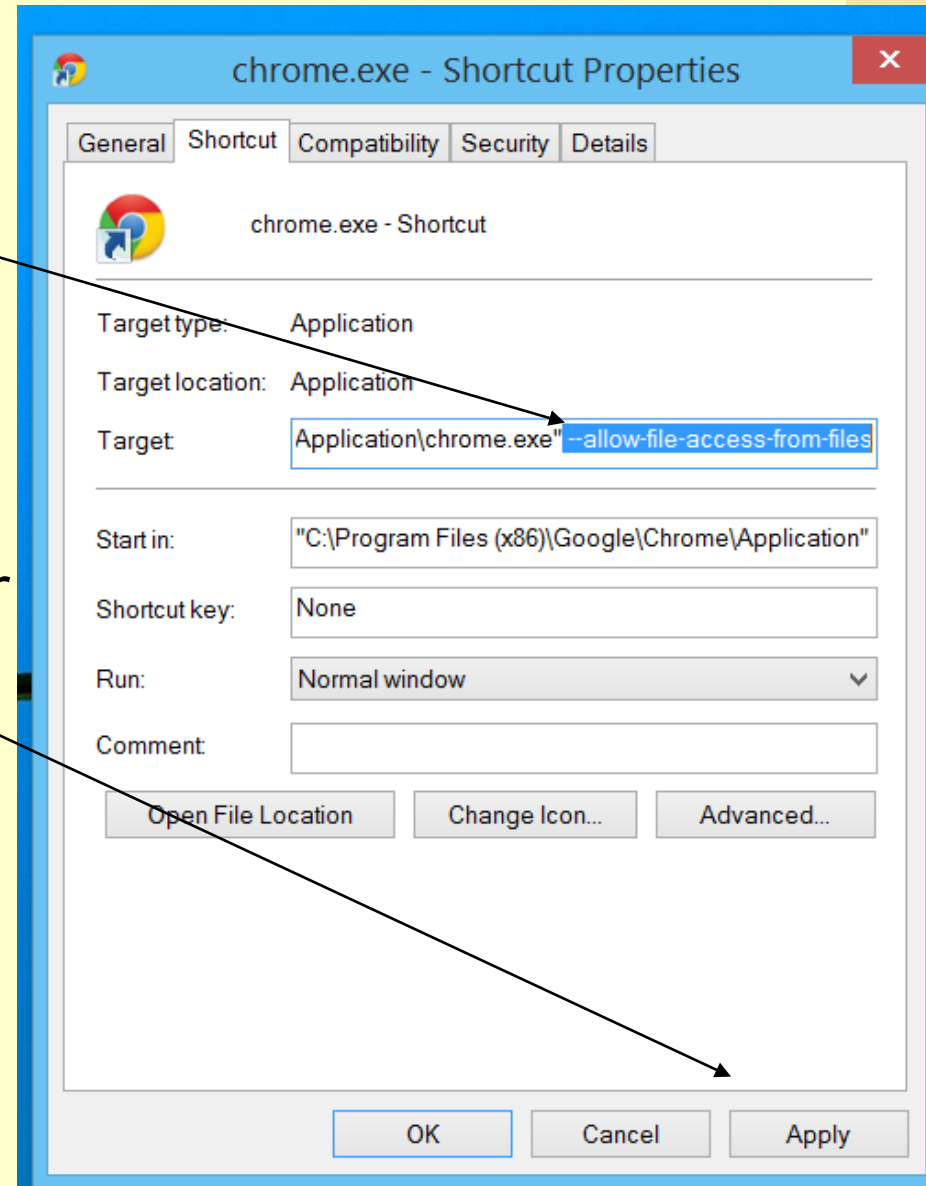
Change the target string just adding the parameter:

--allow-file-access-from-files

at the end of the string, then click “Apply”.

◆ Run Chrome

CLOSE ALL the active Chrome browser windows (if there are) and... that's it! Now if you launch the Chrome browser **from this shortcut** is able to access file directly on your PC.



Chrome On Windows

Always remember to launch Chrome using the modified shortcut! If you open Chrome from a different shortcut, from the start menu or by double-clicking on an HTML file, Chrome will **not** be able to work on local files.

You may open the HTML from the File->Open menu, or simply dragging the HTML file on the modified shortcut.

Now you can work on your local files, and immediately see the result on your browser.

Chrome On MacOS

- ◆ **Install Chrome on your Mac**

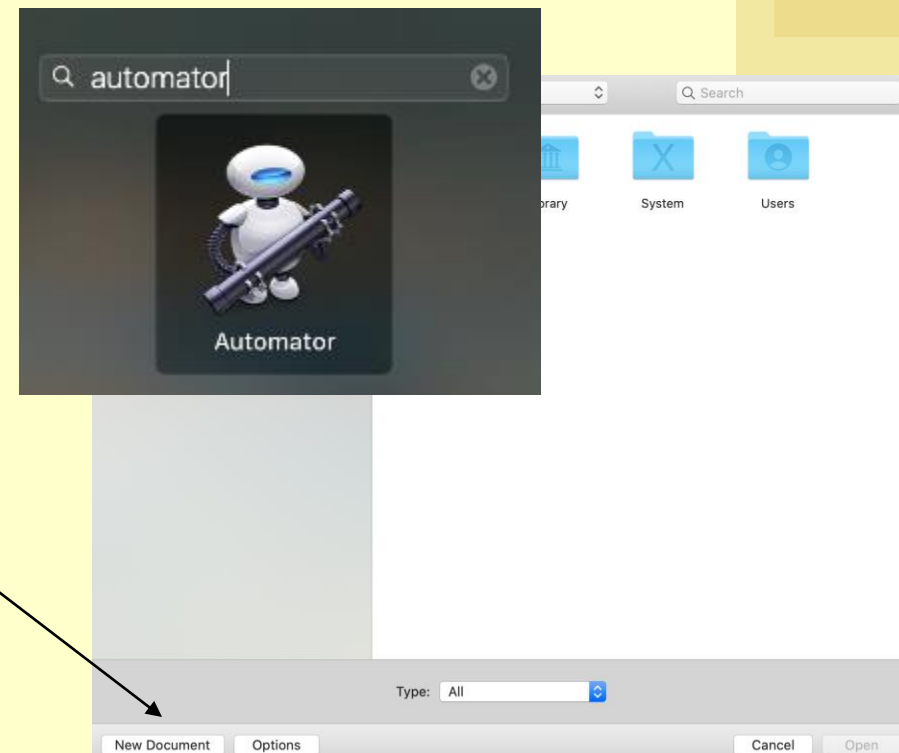
Go to www.google.com/chrome , download the browser version suitable for your platform, and install it.

- ◆ **Look for Automator and run it**

Open the Launchpad 🚀 type «automator» in the search box, then run it.

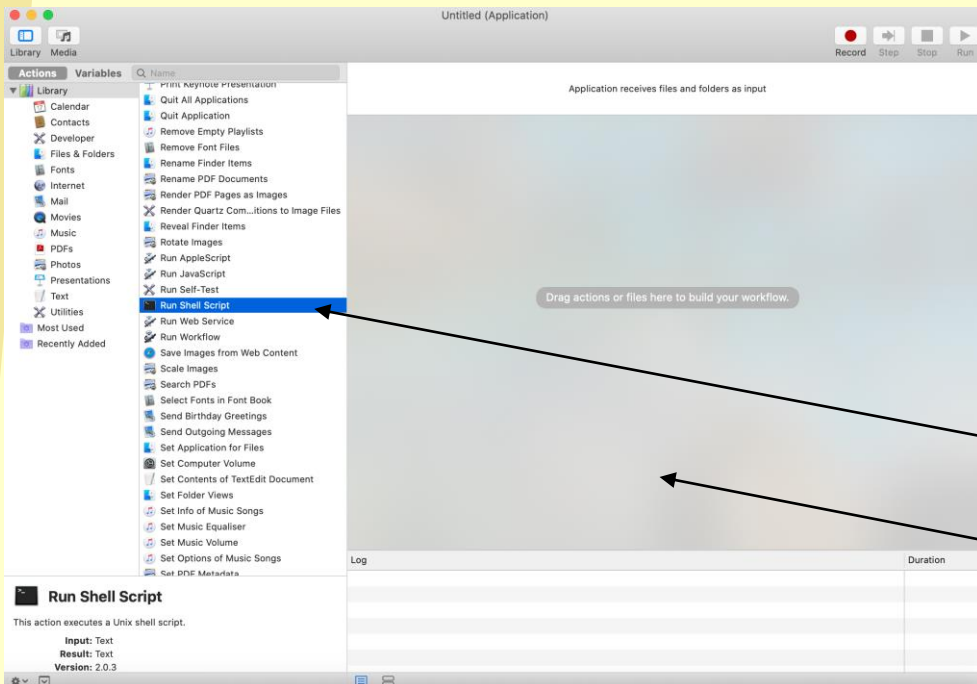
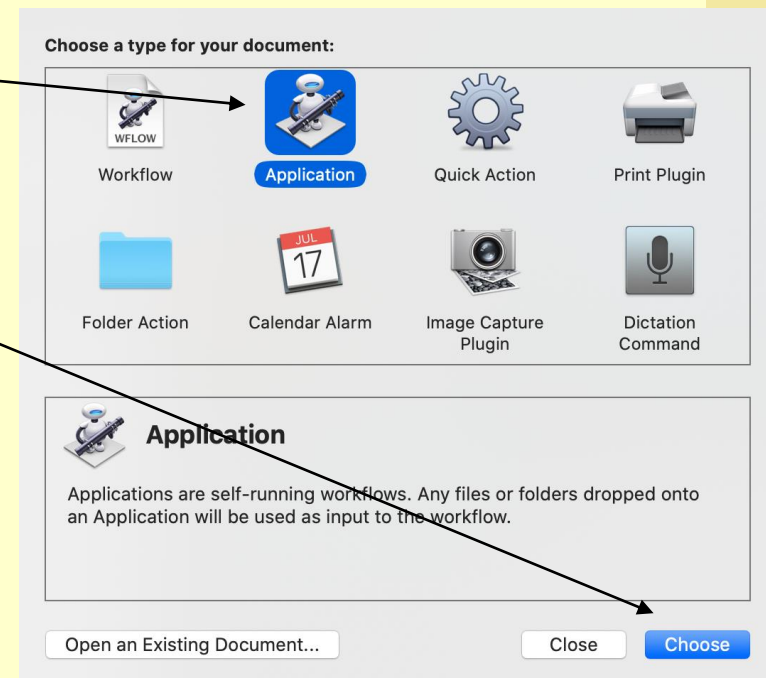
- ◆ **Create a new document**

Click on the button at the bottom left of the Automator window.



Chrome On MacOS

- ◆ **Choose the document type**
Select «Application» in the menu and then click on the button at the bottom right of the window.



- ◆ **Create a shell script action**
In the actions list look for «Run Shell Script» and drag & drop this action in the panel on the right.

Chrome On MacOS

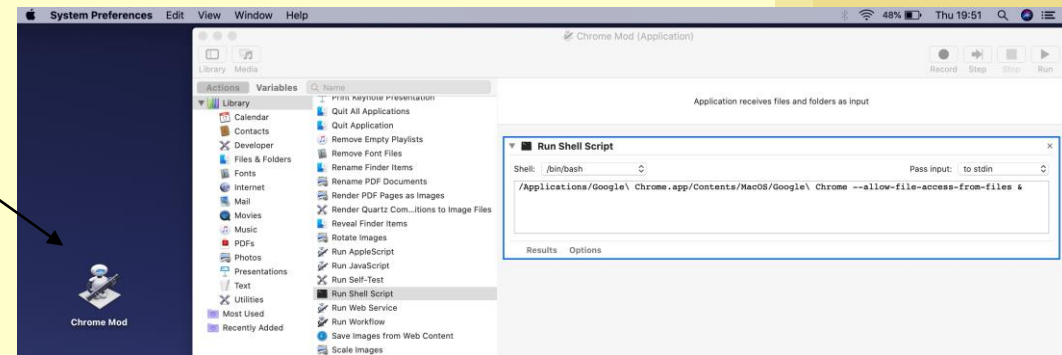
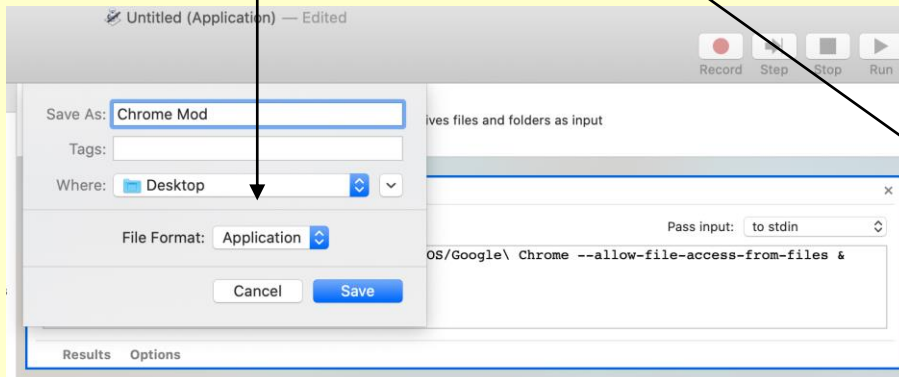
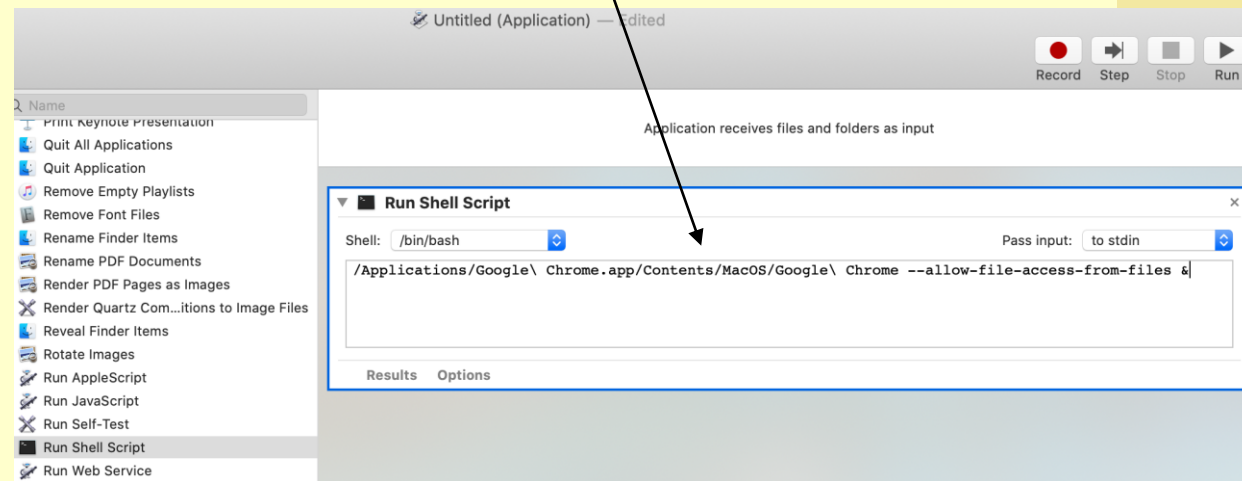
- ◆ **Add the usual string to the shell script text box**

Edit the shell script text box just adding this string:

```
/Applications/Google\ Chrome.app/Contents/MacOS/Google\ Chrome --allow-file-access-from-files &
```

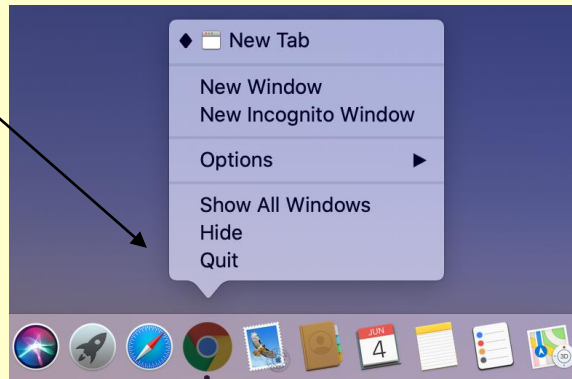
- ◆ **Save the script**

Save the shell script on your desktop, selecting «Application» as file format and choosing the name you prefer.



Chrome On MacOS

CLOSE ALL the active Chrome browser windows (if there are) and... that's it!



Now if you launch the browser using the Automator script, Chrome is able to access file directly on your PC.

Always remember to launch Chrome using the Automator script! If you open Chrome from a different shortcut, from the docked menu, or by double-clicking on an HTML file, Chrome will **not** be able to work on local files.

Chrome On MacOS

You may open the HTML from the File->Open menu, or simply dragging the HTML file on the browser window.

Now you can work on your local files, and immediately see the result on your browser.

Alternatively, you may run Chrome from terminal, using this command line:

```
open /Applications/Google\ Chrome.app --args --allow-file-access-from-files
```

Other browsers, other OSs

This routine works also with other Chromium-based browsers, like for instance **Microsoft Edge**...

The Chromium-based browsers, used with the launch parameter, will work on all operating systems...

Linux

- You may run Chrome from terminal, with the parameter as argument
- You may create an alias or a desktop shortcut (similarly to windows)

Remote deployment

Files on a web server

Over the Internet

The ideal use of 3DHOP is over the Internet, with the HTML and data files on a remote server, accessed from a client PC with a web browser.

3DHOP does not require a specialized web server, nor server-side resident software/daemon. In order to publish a 3DHOP visualization over the internet you just need some space on a web-accessible server.

To publish a 3DHOP viewer online, just move all the needed files in an accessible location of a web server.

Remote server

Looking at the `MINIMAL` folder is a good way to know which are the files needed by a 3DHOP visualization page.

Make sure all the files are accessible from the web.
Almost all web server are case-sensitive, so take care of upper- and lower-case in filenames.