



EECS E6893 Big Data Analytics

HW3: Data visualization

Tutorial II

Muyao Zi
mz3056@columbia.edu

Agenda

- Using D3.js to do data visualization and graph visualization
- Apache HTTP Server

Graphical Analysis using D3

- Network analysis and network visualization are more common now with the growth of online social networks like Twitter and Facebook, as well as social media and linked data, all of which are commonly represented with network structures.
- In general, when dealing with networks you refer to the things being connected (like people) as nodes and the connections between them (such as being a friend on Facebook) as edges or links.
- Networks may also be referred to as graphs, because that's what they're called in mathematics.



Graphical Analysis using D3

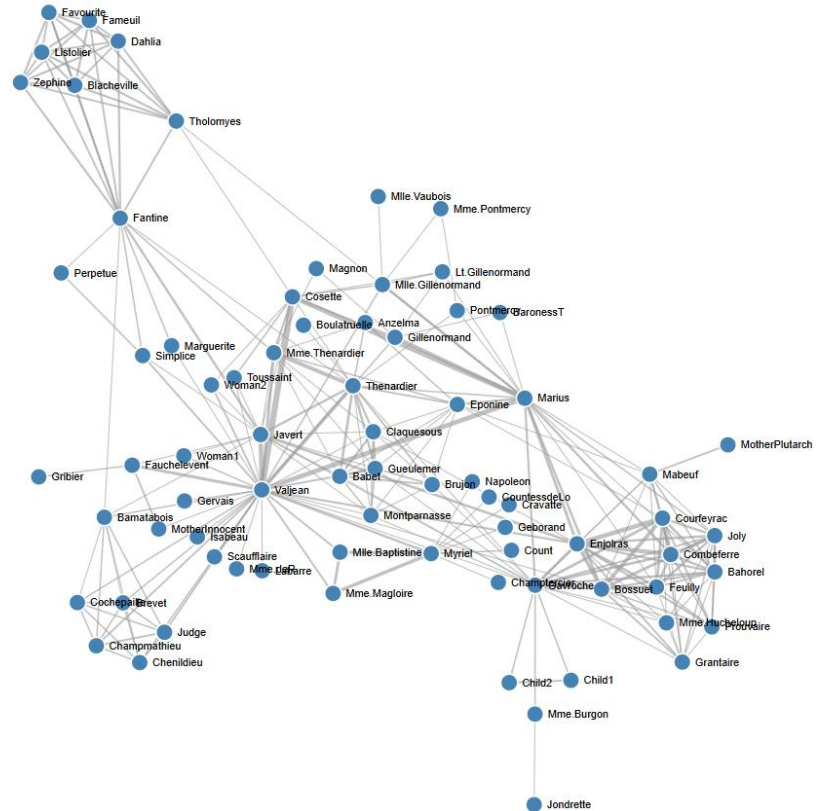
- D3 is particularly adept at creating Graphical visualizations due to its data-binding capability and built-in force simulations
- Real-world applications:
 - Social network analysis – friendships, interactions
 - Biological networks – protein interaction, genetic pathways
 - Infrastructure – internet routing, power grids

Force-directed Network diagrams and simulations in D3

- Automatically positions nodes
- Nodes repel each other, links act as springs
- Custom forces can be added
 - Centering, collision detection, etc.
- The force layout dynamically updates the positions of its elements to find the best fit, it does it continuously in real-time rather than as a preprocessing step before rendering

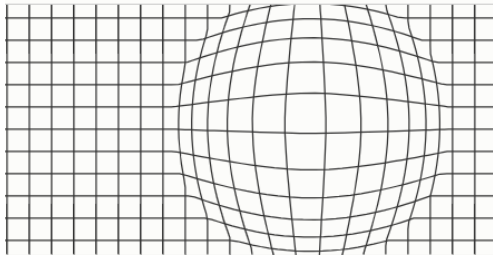
Example— Les Misérables network graph

- Each node represents a character in Victor Hugo's Les Misérables.
- Links represent co-occurrence in the same scene.



Fisheye Distortion in Network Visualization

- A fisheye lens **enlarges areas near your focus** while compressing the edges.
- In D3, this effect lets us zoom into dense graph clusters without losing the full view.
- Useful for exploring local connections in **large or crowded networks**.
- Compared to normal zoom, fisheye keeps the entire network visible.



More about [Fisheye Distortion](#)

Adding Fisheye Effect to a Force-Directed Graph

- radius: how far from the mouse the distortion spreads.
- distortion: how strongly the focus area expands.
- smoothing: how smooth the transition is between focus and edges.

```
const fisheye = d3Fisheye.radial()  
  .radius(200)  
  .distortion(6)  
  .smoothing(0.1);
```

Which D3 fisheye to use

Option A — Classic D3 Plugins fisheye

Repo: <https://github.com/d3/d3-plugins/tree/master/fisheye>
Model: `d3.fisheye.circular()` (Sarkar–Brown). No smoothing at the edge.
Works with: D3 v3–v7.

```
<script src="https://d3js.org/d3.v7.min.js"></script>  
<script src="https://cdn.jsdelivr.net/gh/d3/d3-plugins@master/fisheye/fisheye.js"></script>
```

Create & use:

```
const fisheye = d3.fisheye.circular().radius(200).distortion(4);  
svg.on("mousemove", (event) => {  
  fisheye.focus(d3.pointer(event, svg.node()));  
  // compute transformed positions for each node and redraw nodes+links  
});
```

Option C — Hard-coded fisheye

You implement the lens yourself: compute distance from the mouse to each node, push nearby nodes outward with a distortion function, and redraw using those transformed coordinates. No external plugin.

Option B — “Smoothed fisheye”

Repo: <https://github.com/duaneatat/d3-fisheye>
Model: `d3Fisheye.radial()` (adds `.smoothing()`) for a gradual, non-jumpy edge.
Works with: D3 v6/v7+.

Install/Include: `npm i d3-fisheye`

```
<script src="https://d3js.org/d3.v7.min.js"></script>  
<script src="https://cdn.jsdelivr.net/npm/d3-fisheye@1.0.1/dist/d3-fisheye.min.js"></script>
```

Create & use:

```
const fisheye = d3Fisheye.radial()  
  .radius(200) // lens size  
  .distortion(4) // magnification strength  
  .smoothing(0.3); // 0 = sharp cutoff, 1 = no effect  
  
svg.on("mousemove", (event) => {  
  const focus = d3.pointer(event, svg.node());  
  fisheye.focus(focus);  
  
  // 1) transform nodes (returns [x, y, z])  
  data.nodes.forEach(d => {  
    const [x, y, z] = fisheye([d.x, d.y]);  
    d.fx_ = x, d.fy_ = y, d.fz_ = z;  
  });  
  
  // 2) redraw links with transformed coords  
  link  
    .attr("x1", d => d.source.fx_ ?? d.source.x)  
    .attr("y1", d => d.source.fy_ ?? d.source.y)  
    .attr("x2", d => d.target.fx_ ?? d.target.x)  
    .attr("y2", d => d.target.fy_ ?? d.target.y);  
  
  // 3) redraw nodes/labels (optionally scale by z)  
  node.attr("cx", d => d.fx_ ?? d.x)  
    .attr("cy", d => d.fy_ ?? d.y)  
    .attr("r", d => 8 * (d.fz_ ?? 1));  
  
  labels.attr("x", d => (d.fx_ ?? d.x) + 10)  
    .attr("y", d => d.fy_ ?? d.y)  
    .attr("font-size", d => `${8 * (d.fz_ ?? 1)}px`);  
});
```

What is Apache Web Server

Apache HTTP Server is one of the oldest and most widely used open-source web servers in the world. A web server's job is to serve web pages to users' browsers when they request them. When you type a URL in your browser, your request goes to a server; Apache receives that request, finds the right HTML, CSS, or JavaScript file, and sends it back over the internet.

Key ideas:

- It uses the HTTP/HTTPS protocol to communicate with browsers.
- It can serve static content (like .html, .css, .js, .png) or integrate with dynamic backends (like Python, PHP, or Node.js apps).
- It's cross-platform, meaning you can run it on Linux, macOS, or Windows.



Hosting webpage on Apache web server using virtual host (MAC users)

Step 1: Install xcode

```
(base) srividyaianampudi@Srividyas-Air ~ % xcode-select --install
xcode-select: error: command line tools are already installed, use "Software Update" to install updates
(base) srividyaianampudi@Srividyas-Air ~ % █
```

Step 2: Install Homebrew

Go to brew.sh in your browser and copy the command there to your terminal-
`/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"`

```
(base) srividyaianampudi@Srividyas-Air ~ % /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
=> Checking for `sudo` access (which may request your password)...
=> This script will install:
/opt/homebrew/bin/brew
/opt/homebrew/share/doc/homebrew
/opt/homebrew/share/man/man1/brew.1
/opt/homebrew/share/zsh/site-functions/_brew
/opt/homebrew/etc/bash_completion.d/brew
/opt/homebrew
```

⇒ Installation successful!

⇒ Homebrew has enabled anonymous aggregate formulae and cask analytics.

Read the analytics documentation (and how to opt-out) here:

<https://docs.brew.sh/Analytics>

No analytics data has been sent yet (nor will any be during this `install` run).

⇒ Homebrew is run entirely by unpaid volunteers. Please consider donating:

<https://github.com/Homebrew/brew#donations>

⇒ Next steps:

- Run these three commands in your terminal to add Homebrew to your `PATH`:

```
echo '# Set PATH, MANPATH, etc., for Homebrew.' >> /Users/srividyanampudi/.zprofile  
echo 'eval "$(opt/homebrew/bin/brew shellenv)"' >> /Users/srividyanampudi/.zprofile  
eval "$(opt/homebrew/bin/brew shellenv)"
```

- Run `brew help` to get started

- Further documentation:

<https://docs.brew.sh>

Step 3: Add Homebrew to your PATH

Follow instructions after installation

```
(base) srividyanampudi@Srividyas-Air ~ % echo '# Set PATH, MANPATH, etc., for Homebrew.' >> /Users/srividyanampudi/.zprofile  
(base) srividyanampudi@Srividyas-Air ~ % echo 'eval "$(opt/homebrew/bin/brew shellenv)"' >> /Users/srividyanampudi/.zprofile  
(base) srividyanampudi@Srividyas-Air ~ % eval "$(opt/homebrew/bin/brew shellenv)"  
(base) srividyanampudi@Srividyas-Air ~ % █
```

Step 4: Install apache2

Command - brew install apache2

```
(base) srividyanampudi@Srividyas-Air ~ % brew install apache2
=> Downloading https://ghcr.io/v2/homebrew/core/apr/manifests/1.7.0_3
##### 100.0%
=> Downloading https://ghcr.io/v2/homebrew/core/apr/blobs/sha256:02e6b44b3284fa471cce15592a8666
=> Downloading from https://pkg-containers.githubusercontent.com/ghcr1/blobs/sha256:02e6b44b328
##### 100.0%
```

Step 5: Start apache server

```
(base) srividyanampudi@Srividyas-Air ~ % sudo apachectl start
(base) srividyanampudi@Srividyas-Air ~ % █
```

This will start Apache HTTP server which can be tested by visiting localhost on the browser. The localhost 8080 gives the response as shown below:

A screenshot of a browser's address bar. It shows navigation icons (back, forward, refresh) on the left, followed by an information icon and the text "localhost:8080".

It works!

Step 6: Open httpd config file

- If you're using Intel-based Mac: `vim /usr/local/etc/httpd/httpd.conf`
- If you're using Mac with Apple Silicon: `vim /opt/homebrew/etc/httpd/httpd.conf`

Step 7: Update these lines

Listen 8080 to Listen 80

DocumentRoot "/usr/local/var/www" to DocumentRoot "/Users/your_account/Sites"

<Directory "/usr/local/var/www"> to <Directory "/Users/your_account/Sites">

AllowOverride None to AllowOverride All

```
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
#Listen 12.34.56.78:80
Listen 80
```

```
#
DocumentRoot "/Users/srividyanampudi/Sites"
<Directory "/Users/srividyanampudi/Sites">
#
# Possible values for the Options directive are "None", "All",
# or any combination of:
#   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
#
# Note that "MultiViews" must be named *explicitly* --- "Options All"
# doesn't give it to you.
#
# The Options directive is both complicated and important. Please see
# http://httpd.apache.org/docs/2.4/mod/core.html#options
# for more information.
#
Options Indexes FollowSymLinks

#
# AllowOverride controls what directives may be placed in .htaccess files.
# It can be "All", "None", or any combination of the keywords:
#   AllowOverride FileInfo AuthConfig Limit
#
AllowOverride All

#
# Controls who can get stuff from this server.
#
```

Step 8: Uncomment this line

LoadModule rewrite_module lib/httpd/modules/mod_rewrite.so

Update these lines

User _www to User your_account

Group _www to Group staff

```
LoadModule alias_module lib/httpd/modules/mod_alias.so
LoadModule rewrite_module lib/httpd/modules/mod_rewrite.so

<IfModule unixd_module>
#
# If you wish httpd to run as a different user or group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the user/group to run httpd as.
# It is usually good practice to create a dedicated user and group for
# running httpd, as with most system services.
#
User srividyaianampudi
Group staff
```

Step 9: Update this line

ServerName www.example.com:8080 to ServerName localhost

```
# If your host doesn't have a registered DNS name, enter its IP address here.
#
ServerName localhost

#
# Deny access to the entirety of your server's filesystem. You must
# explicitly permit access to web content directories in other
# <Directory> blocks below.
#
```

Step 10: Create Sites folder and add your html file to the folder

```
(base) srividyanampudi@Srividyas-Air ~ % mkdir Sites
(base) srividyanampudi@Srividyas-Air ~ %
(base) srividyanampudi@Srividyas-Air ~ %
(base) srividyanampudi@Srividyas-Air ~ % cd Sites
(base) srividyanampudi@Srividyas-Air Sites % ls
jsexample.html
```

Step 11. Restart apache

```
(base) srividyanampudi@Srividyas-Air Sites % sudo apachectl stop
(base) srividyanampudi@Srividyas-Air Sites %
(base) srividyanampudi@Srividyas-Air Sites %
(base) srividyanampudi@Srividyas-Air Sites %
(base) srividyanampudi@Srividyas-Air Sites % sudo apachectl start
```

Step 12: Test localhost to see your hosted website

<http://localhost>

A screenshot of a web browser's address bar. It features navigation icons (back, forward, refresh) on the left, an information icon (i) on the right, and the text 'localhost' in the center. The address bar is highlighted with a blue border.

Sample HTML to host on apache webserver

- Car
 - Ford

Step 13: Stop apache

```
(base) srividyaianampudi@Srividyas-Air Sites % sudo apachectl stop
Password:
(base) srividyaianampudi@Srividyas-Air Sites %
```

Creating and Hosting a Virtual Host

Step 1: Locate the Apache config directory

- Intel: `/usr/local/etc/httpd/`
- Apple Silicon: `/opt/homebrew/etc/httpd/`

Step 2: Enable Virtual Hosts: Edit `httpd.conf`. Find this line and uncomment it:

```
Include /opt/homebrew/etc/httpd/extra/httpd-vhosts.conf
```

Step 3: Edit the Virtual Hosts file `httpd-vhosts.conf` (`vim /opt/homebrew/etc/httpd/extra/httpd-vhosts.conf`).

Add this block at the bottom:

```
<VirtualHost *:80>
  ServerName localhost
  DocumentRoot "/your/project/file/directory"
  <Directory "/your/project/file/directory">
    AllowOverride All
    Require all granted
  </Directory>
</VirtualHost>
```

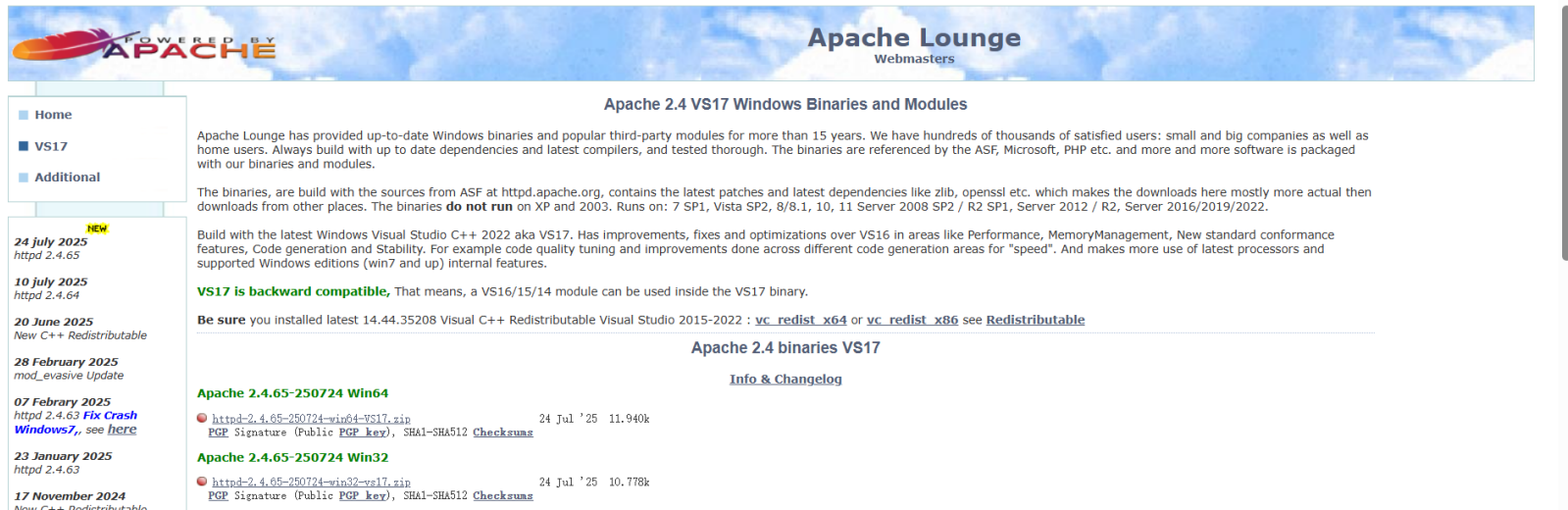
Step 4: Restart Apache: `sudo apachectl restart`

Visit `http://localhost`, your D3 visualization should appear.

Hosting webpage on Apache web server using virtual host (Windows users)

Step 1: Download Apache lounge zip file -

<https://www.apachelounge.com/download/>



The screenshot shows the Apache Lounge website. The header includes the Apache logo and the text "POWERED BY APACHE" and "Apache Lounge Webmasters". The main content area is titled "Apache 2.4 VS17 Windows Binaries and Modules". It contains several paragraphs of text explaining the binaries and their compatibility. There are also several news items listed on the left side of the page, including dates and descriptions of updates or fixes. At the bottom, there are links for downloading Apache 2.4 binaries for VS17, with specific links for Win64 and Win32 architectures.

Apache Lounge
Webmasters

Apache 2.4 VS17 Windows Binaries and Modules

Apache Lounge has provided up-to-date Windows binaries and popular third-party modules for more than 15 years. We have hundreds of thousands of satisfied users: small and big companies as well as home users. Always build with up to date dependencies and latest compilers, and tested thorough. The binaries are referenced by the ASF, Microsoft, PHP etc. and more and more software is packaged with our binaries and modules.

The binaries, are build with the sources from ASF at httpd.apache.org. contains the latest patches and latest dependencies like zlib, openssl etc. which makes the downloads here mostly more actual then downloads from other places. The binaries **do not run** on XP and 2003. Runs on: 7 SP1, Vista SP2, 8/8.1, 10, 11 Server 2008 SP2 / R2 SP1, Server 2012 / R2, Server 2016/2019/2022.

Build with the latest Windows Visual Studio C++ 2022 aka VS17. Has improvements, fixes and optimizations over VS16 in areas like Performance, MemoryManagement, New standard conformance features, Code generation and Stability. For example code quality tuning and improvements done across different code generation areas for "speed". And makes more use of latest processors and supported Windows editions (win7 and up) internal features.

VS17 is backward compatible, That means, a VS16/15/14 module can be used inside the VS17 binary.

Be sure you installed latest 14.44.35208 Visual C++ Redistributable Visual Studio 2015-2022 : [vc_redist_x64](#) or [vc_redist_x86](#) see [Redistributable](#)

Apache 2.4 binaries VS17

[Info & Changelog](#)

Apache 2.4.65-250724 Win64

- [httpd-2.4.65-250724-win64-vs17.zip](#) 24 Jul '25 11.940k
PGP Signature (Public [PGP key](#)), SHA1-SHA512 [Checksums](#)

Apache 2.4.65-250724 Win32

- [httpd-2.4.65-250724-win32-vs17.zip](#) 24 Jul '25 10.778k
PGP Signature (Public [PGP key](#)), SHA1-SHA512 [Checksums](#)

24 July 2025 **NEW**
[httpd 2.4.65](#)

10 July 2025
[httpd 2.4.64](#)

20 June 2025
New C++ Redistributable

28 February 2025
[mod_evasive](#) Update

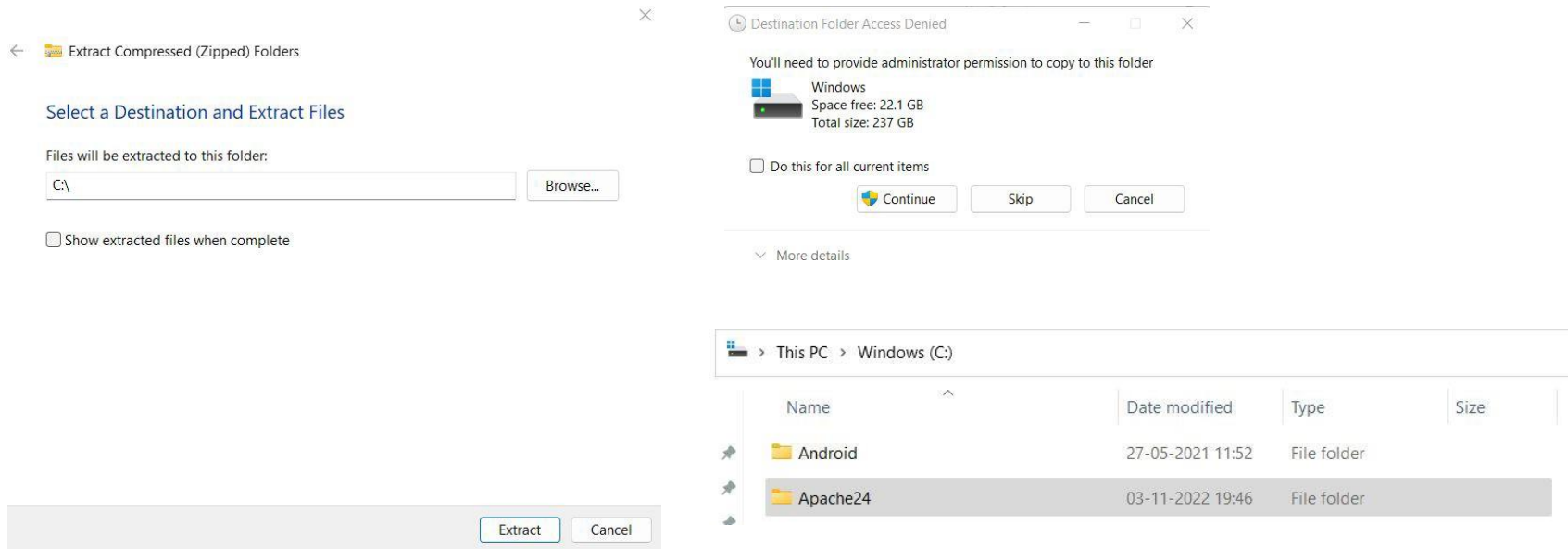
07 February 2025
[httpd 2.4.63](#) [Fix Crash Windows7](#), see [here](#)

23 January 2025
[httpd 2.4.63](#)

17 November 2024
[New C++ Redistributable](#)

Download 64bit if your system is 64bit else download 32bit zip file

Step 2: Once zip file is finished downloading unzip it to C drive. Click Yes for administrative privileges.



Step 3: Go to “C:\Apache24\bin\” and find **httpd** and we need to install this httpd. This is what is actually going to run apache. So we will go to start menu and type cmd to get command prompt. Right click as select **Run as administrator**. Click yes if it asks you for permission.

Step 4: Change directory to **C:\Apache24\bin** by typing **cd C:\Apache24\bin** in the terminal

Step 5: Now that we are in bin directory we will install httpd by typing the command **httpd -k install**. Allow access

> This PC > Windows (C:) > Apache24 > bin

| Name | Date modified | Type | Size |
|--------------------------|------------------|-----------------------|--------|
| iconv | 03-11-2022 19:46 | File folder | |
| ab | 03-11-2022 19:46 | Application | 97 KB |
| abs | 03-11-2022 19:46 | Application | 108 KB |
| ApacheMonitor | 03-11-2022 19:46 | Application | 42 KB |
| apr_crypto_openssl-1.dll | 03-11-2022 19:46 | Application extens... | 19 KB |
| apr_dbd_odbc-1.dll | 03-11-2022 19:46 | Application extens... | 31 KB |
| apr_idap-1.dll | 03-11-2022 19:46 | Application extens... | 15 KB |
| dbmmanage.pl | 03-11-2022 19:46 | PL File | 9 KB |
| htcacheclean | 03-11-2022 19:46 | Application | 100 KB |
| htdbm | 03-11-2022 19:46 | Application | 121 KB |
| htdigest | 03-11-2022 19:46 | Application | 84 KB |
| htpasswd | 03-11-2022 19:46 | Application | 117 KB |
| httpd | 03-11-2022 19:46 | Application | 30 KB |

Search: cmd

All Apps Documents Web More

Best match

Command Prompt App

Apps

- Git CMD (Deprecated)
- Anaconda Prompt
- mkTO.cmd
- Google Cloud SDK Shell

Search the web

cmd - See web results

Command Prompt App

- Open
- Run as administrator**
- Open file location
- Pin to Start
- Pin to taskbar

Administrator: Command Prompt

```
Microsoft Windows [Version 10.0.22000.1098]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd C:\Apache24\bin

C:\Apache24\bin>
```

Administrator: Command Prompt

```
Microsoft Windows [Version 10.0.22000.1098]
(c) Microsoft Corporation. All rights reserved.

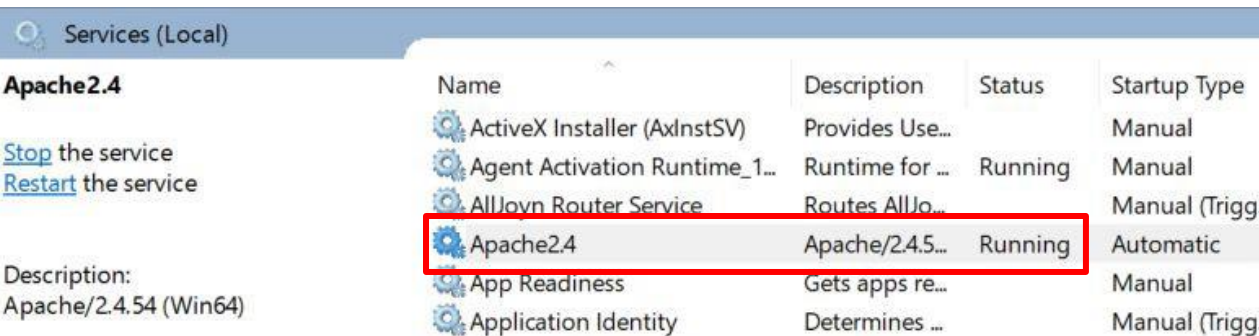
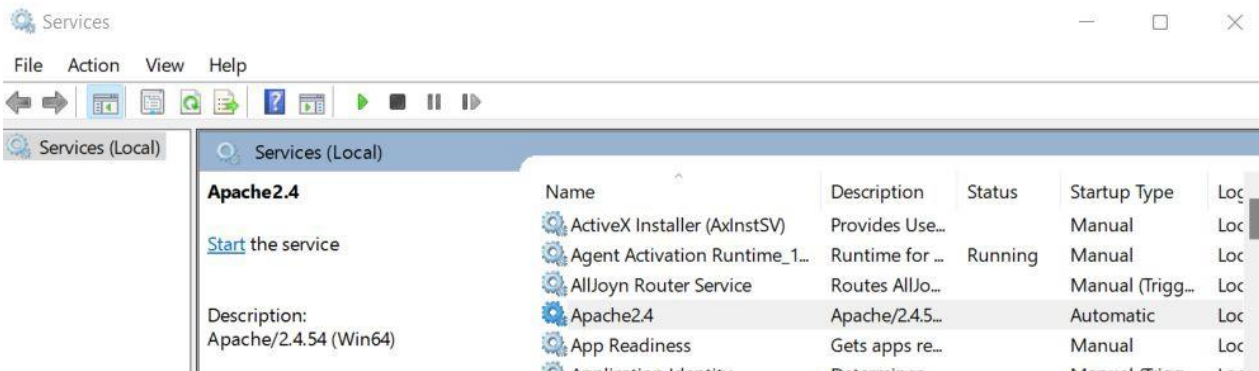
C:\WINDOWS\system32>cd C:\Apache24\bin

C:\Apache24\bin>httpd -k install
Installing the 'Apache2.4' service
The 'Apache2.4' service is successfully installed.
Testing httpd.conf...
Errors reported here must be corrected before the service can be started.

C:\Apache24\bin>
```

Step 6: Now Apache is installed as a service. If we go to Windows and type in Services in search bar and open Services we can see that Apache2.4 is installed.

Step 7: Now we will start the APache service by clicking **start** on left side. And then it will say that it is running. We can stop or restart the service on the left side if we need to.



Step 8: Now to check if it is running. We will go to 127.0.0.1 in your browser it shows that it works!

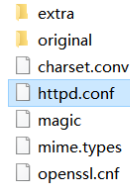


Step 9: If you have any errors you can check them in the logs folder in **C:\Apache24\logs** in errors.txt file

Step 10: Now to host our webpage we will go to **C:\Apache24\htdocs** and change the index.html to be your required html file and then again open 127.0.0.1 to see our webpage.

Creating and Hosting a Virtual Host

Step 1: Locate and open the file `C:\Apache24\conf\httpd.conf` in Notepad (Run as Administrator).



Step 2: Find and uncomment: configuration file

```
# Virtual hosts
Include conf/extra/httpd-vhosts.conf
```

Step 3: Edit the virtual hosts file: `C:\Apache24\conf\extra\httpd-vhosts.conf`
Add a block like this at the bottom:

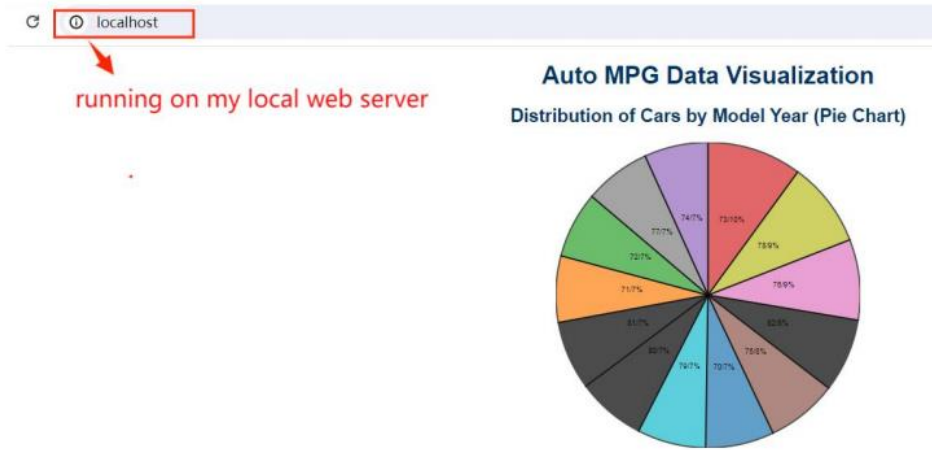
```
<VirtualHost *:80>
  ServerName localhost
  DocumentRoot "your_project_folder_directory"
  <Directory "your_project_folder_directory">
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
  </Directory>
</VirtualHost>
```

Note: You can use any path outside htdocs (like `D:/bigdata/hw3_tutorial2`). Apache will serve from there as long as it has read access.

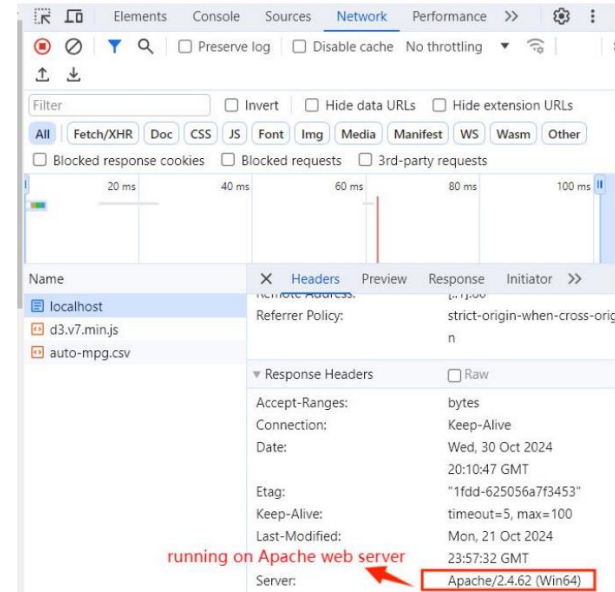
Or you can place your project files (HTML, JS, CSV) inside: `C:\Apache24\htdocs\your_project`

Step 4: Restart Apache service. Visit `http://localhost`, you should see your hosted visualization page.

We can see that our html file is hosted on Apache web server when we go to 127.0.0.1 on browser



Open browser's Developer Tools → Network tab.
can see that page is running on Apache web server



More about D3

- <https://d3js.org/>
- <https://observablehq.com/@d3/gallery>
- <https://github.com/d3/d3>
- https://www.tutorialspoint.com/d3js/d3js_working_example.htm

THANK YOU