Microsoft Timeline Storyteller One Pager **Dr. Leisl Carr Childers**

Microsoft Timeline Storyteller is a dynamic timeline creation platform that relies on structured datasets that use dates as their anchor point. It can accommodate hundreds of datapoints and is particularly effective at revealing patterns that occur over time. To use Microsoft Timeline Storyteller effectively, consider how events are dated (by date, month, and year) and be consistent with the use of dates. Also remember to be consistent with the names and descriptions that accompany those dates. Timelines can reveal patterns if they are generated with an understanding of these elements. Here is how to get started with Microsoft Timeline Storyteller.

1. Open Google Chrome. This is the most compatible web browser.
2. Install Microsoft Timeline Storyteller (see installation instructions that follow).
3. Open Microsoft Timeline Storyteller.
    a. On WindowsOS:
        i. Navigate in Finder to your User space and right click on the Microsoft Timeline Storyteller file. Select **New Terminal at Folder**
        ii. Type in **cd timelinestoryteller** and Enter
        iii. Type in **npm start** and Enter
        iv. The response in Terminal should be:

```
timeline_storyteller@0.1.1 start

node app.js

Server started on port 8000

…

webpack: Compiled successfully.
```

        v. To exit the program, in PowerShell, push **Command and C**, they type **Y** for yes to end and close PowerShell and the web browser.
    b. On MacOS:
        i. Open PowerShell.
        ii. Type in **cd timelinestoryteller** and Return
        iii. Type in **npm start** and Return
        iv. The response in PowerShell should be:

```
timeline_storyteller@0.1.1 start

node app.js

Server started on port 8000

…

webpack: Compiled successfully.
```

        v. To exit the program, in Terminal push **Command and C** and close the Terminal application and web browser.
4. The data for the platform must be structured in an Excel spreadsheet. Column headings are standardized and must be ordered and formatted as follows: start_date, end_date, category, facet, and content_text.

- The data entered in the start date and end date can be the same if the date is a single day or they can be different if the dates cover a time period.
- The data entered in the category column will be color coded in the legend.
- The data entered in the facet column can be the same as the category column or it can help differentiate subsets within the categories.
- The data entered in the content text column provides descriptive information on the dates.

5. Microsoft Timeline Storyteller can upload datasets as JSON, Google Sheets, and CSV (comma separated values) files. The best format to use is the CSV file. In the Excel file containing the chronological data, under File select Save As. Under File Format, select .CSV. Click save.
6. In Timeline Storyteller click "Use Timeline Storyteller in the Browser." Under Timeline Storyteller (Alpha) find the symbol for .CSV files and click it. Locate and upload the .CSV file.
7. Once the .CSV file is uploaded, a preliminary view of the dataset appears. If the categories look good, click the orange "Draw this Timeline" button. The dataset will populate the basic timeline.
8. There are a variety of ways to manipulate the timeline. The possibilities are enabled via the highlighted boxes in the tool bar. Experiment with the different layouts to determine the clearest way to visualize the dataset. Visualizations can be exported using the menu at the left.

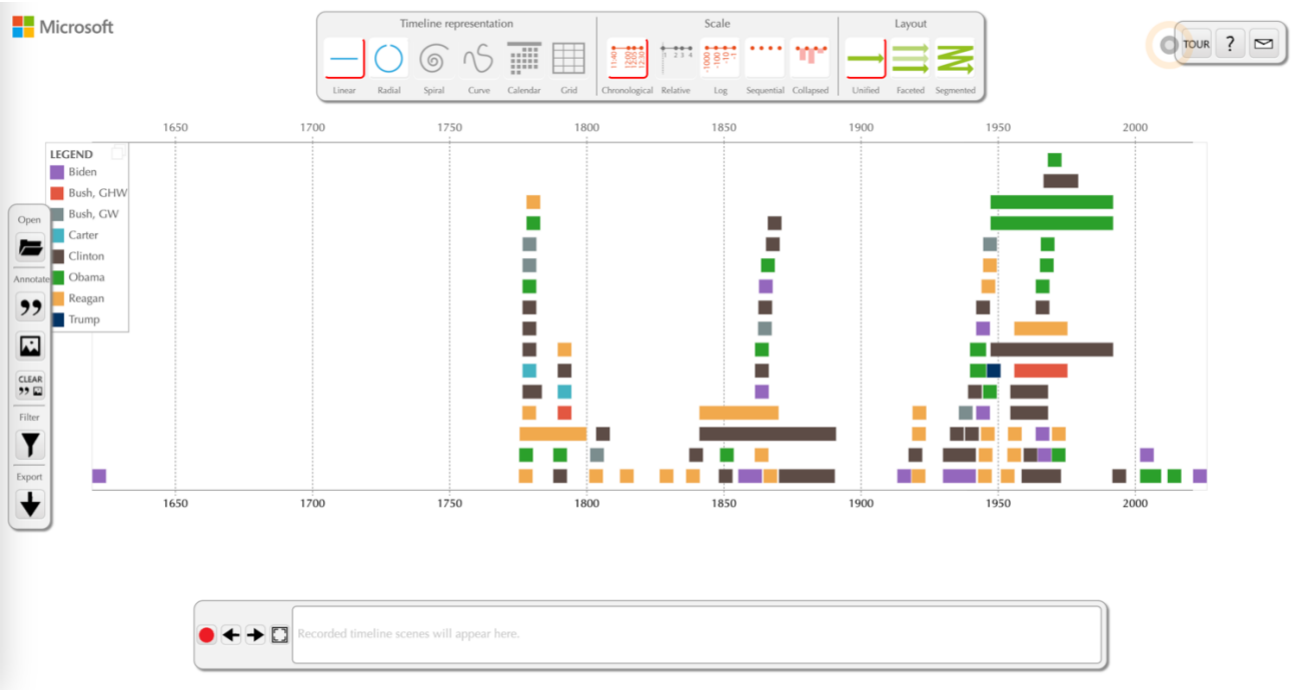


Figure 1: Example of a timeline generated by Microsoft Timeline Storyteller. The dataset that produced these visualizations is derived from the presidential inaugural addresses from 1977 to 2020 and depicts the historical references made in each speech.

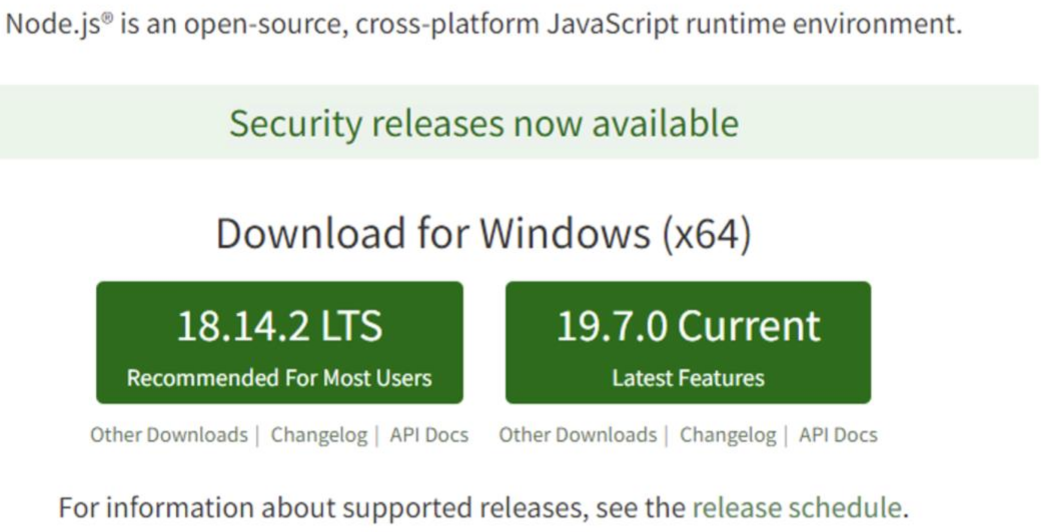
For more information on Microsoft Timeline Storyteller, see https://timelinesrevisited.github.io/ and http://timelinesrevisited.github.io/supplemental/gallery/.

Instructions for Installing Microsoft Timeline Storyteller

[To run in a web browser n- https://github.com/microsoft/timelinestoryteller.]

For PCs

1. Install Node.js - Navigate to https://nodejs.org/
    - Download the latest version of Node.js. Select the recommended version for most users



Node.js® is an open-source, cross-platform JavaScript runtime environment.

Security releases now available

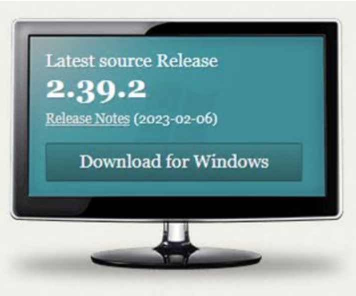
Download for Windows (x64)

18.14.2 LTS
Recommended For Most Users
19.7.0 Current
Latest Features
Other Downloads | Changelog | API Docs
Other Downloads | Changelog | API Docs

For information about supported releases, see the release schedule.

    - When the file downloads, open the file and follow the instructions for installing the application
    - Your PC will automatically install it to the correct location (OSDisk(C:) User or Program Files depending upon whether you have a user profile or not). Write down the location of the file
2. Open Finder and navigate to Windows PowerShell. Open the PowerShell application
    - The PowerShell application provides a command-line interface toWindows OS Type

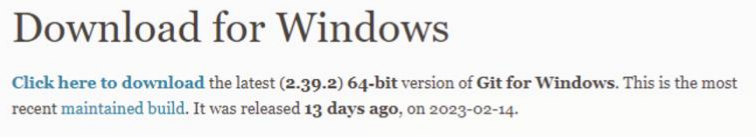
      node -v in the command line to verify that Node.js is installed correctly
    - The response in PowerShell should be **v18.14.2**
    - Close PowerShell
3. Install Git – Navigate to https://git-scm.com/
    - Download the latest version of Git



Latest source Release

2.39.2

Release Notes (2023-02-06)



Download for Windows

Click here to download the latest (2.39.2) 64-bit version of Git for Windows. This is the most recent maintained build. It was released 13 days ago, on 2023-02-14.

- Your PC will automatically install it to the correct location (OSDisk(C:) User or Program Files depending upon whether you have a user profile or not). Write down the location of the file. NOTE THAT BOTH NODE.JS AND GIT MUST BE INSTALLED IN THE SAME LOCATION
- Use the preset installation configuration provided by the Git Install Wizard. Do NOT select anything other than the presets
- In the PowerShell application type `git --version` to verify that Git is installed correctly
- The response in PowerShell should be `git version 2.39.2.windows.1`
- Close PowerShell and reopen for step 4

4. Clone the Timeline Storyteller repository to your computer
    - In the PowerShell application type:

      ```
      git clone https://github.com/microsoft/timelinestoryteller.git
      ```

    - The response in PowerShell should begin `Cloning into 'timelinestoryteller'...`
5. Install the dependencies to build the software. NPM stands for Node Package Manager
    - Type in the PowerShell application `cd timelinestoryteller`.
    - Type in the PowerShell application `npm install`.
    - The response in PowerShell should be

      ```
      > timeline_storyteller@0.1.1 test

      > npm run lint && npm run karma && npm run package

      > timeline_storyteller@0.1.1 lint

      > eslint src ...

        webpack: Compiled successfully.
      ```

6. Run the project.
    1. Type in the PowerShell application `npm start`
    2. This will start a local web server at `http://localhost:8000/`
    3. Open a web browser and navigate to `http://localhost:8000/` to use the Timeline Storyteller
    4. To exit the program, in PowerShell, push **Command and C**, they type `Y` for yes to end and close PowerShell and the web browser

Every time you want to run Microsoft Timeline Storyteller, open the PowerShell application and follow the protocol to start the file.

1. Open PowerShell.
2. Type in `cd timelinestoryteller` and Enter
3. Type in `npm start` and Enter
4. The response in PowerShell should be:

   ```
   > timeline_storyteller@0.1.1 start

   > node app.js
   ```

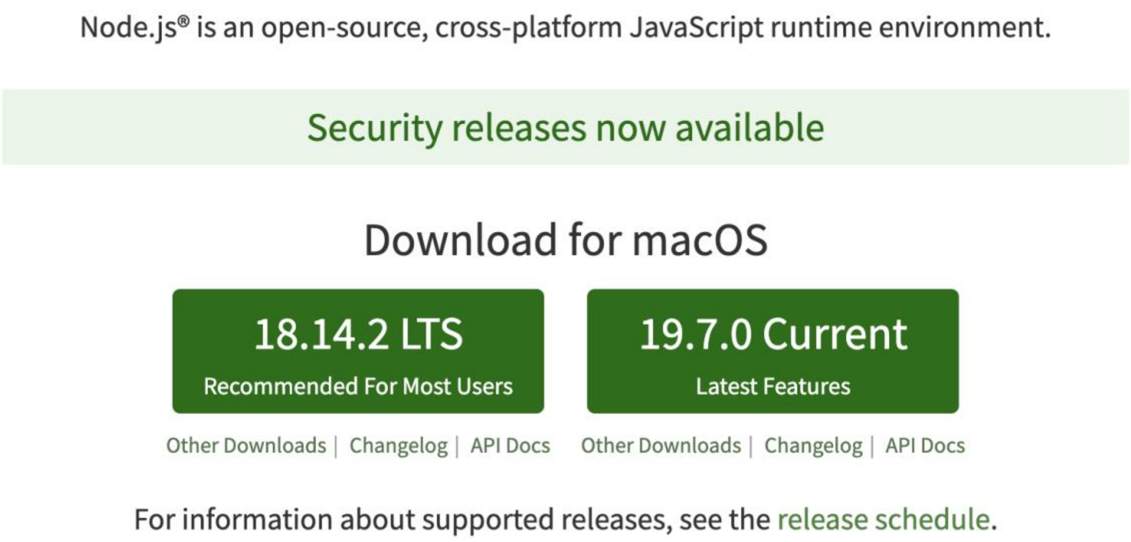
```
Server started on port 8000
…
webpack: Compiled successfully.
```

5. To exit the program, in PowerShell, push **Command and C**, they type Y for yes to end and close PowerShell and the web browser.

<u>Instructions for Installing Microsoft Timeline Storyteller</u>
[To run in a web browser - https://github.com/microsoft/timelinestoryteller.]

<u>For Macs</u>

1. Install Node.js - Navigate to https://nodejs.org/
    - Download the latest version of Node.js. Select the recommended version for most users



Node.js® is an open-source, cross-platform JavaScript runtime environment.

Security releases now available

Download for macOS

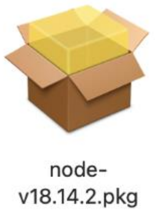
18.14.2 LTS
Recommended For Most Users

19.7.0 Current
Latest Features

Other Downloads | Changelog | API Docs   Other Downloads | Changelog | API Docs

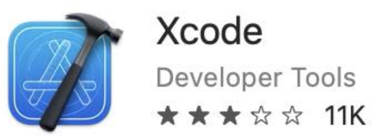
For information about supported releases, see the release schedule.

    - When the file downloads, open the file and follow the instructions for installing the application



node-v18.14.2.pkg

    - Your Mac will automatically install it to the correct location (Macintosh HD within the User file structure). Write down the location of the file
2. Open Finder and navigate to Applications. Under "Utilities," open the Terminal application
    - The Terminal application provides a command-line interface to macOS. Type **node -v** in the command line to verify that Node.js is installed correctly
    - The response in Terminal should be **v18.14.2**
3. Install Xcode – Navigate to the App Store and search for Xcode

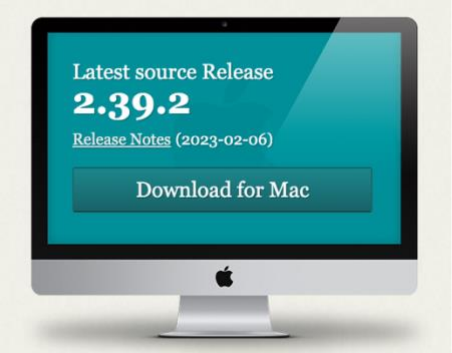


Xcode
Developer Tools
★★★☆☆ 11K

    - The download may take a few minutes
    - When the download is complete, navigate to Applications and open Xcode. Agree to the terms and conditions of use

- In the Terminal application, type **xcode-select -p**.
- The response in Terminal should be **/Applications/Xcode.app/Contents/Developer**

4. Install Git – Navigate to https://git-scm.com/
    - Download the latest version of Git

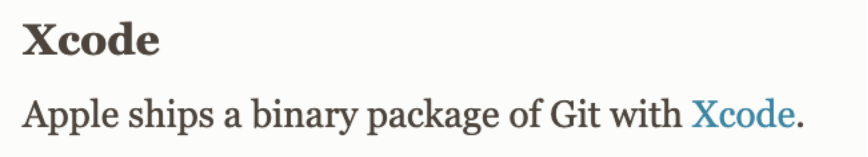


Latest source Release

2.39.2

Release Notes (2023-02-06)

- You will be using the version compatible with Xcode



Xcode

Apple ships a binary package of Git with Xcode.

- The download may take a few minutes
- Your Mac will automatically install it to the correct location (Macintosh HD within the User file structure). Write down the location of the file
- In the Terminal application type **git --version** to verify that Git is installed correctly
- The response in Terminal should be **git version 2.37.1 (Apple Git-137.1)**
- If the version of Git does not appear, run **sudo xcodebuild -license** and enter your password. Agreeing to the Xcode/iOS license requires admin privileges Once entered, retry the command.

5. Build the file directory
    - Navigate to Macintosh HD and click on the hard drive file structure
    - Select the Users folder and identify and click into your user profile
    - Create a **new folder labeled Microsoft Timeline Storyteller**. This is where the application you are building will live
    - In the Terminal application type **pwd**. The response should be **/Users/** succeeded by your user profile
    - In the Terminal application type **ls** (library structure). The response should be the same file structure as under your user profile and include the new folder **Microsoft Timeline Storyteller**
    - Close the Terminal application
6. Clone the Timeline Storyteller repository to your computer
    - Navigate in Finder to your User space and right click on the Microsoft Timeline Storyteller file. Select **New Terminal at Folder**
    - In the Terminal application type:

```
git clone https://github.com/microsoft/timelinestoryteller.git
```

- The response in Terminal should begin Cloning into 'timelinestoryteller'...

7. Install the dependencies to build the software. NPM stands for Node Package Manager
   - Type in the Terminal application cd timelinestoryteller
   - Type in the Terminal application sudo npm install
   - Enter your computer password when prompted
   - The response in Terminal should be

```
> timeline_storyteller@0.1.1 test

> npm run lint && npm run karma && npm run package

> timeline_storyteller@0.1.1 lint

> eslint src ...

  webpack: Compiled successfully.
```

   - Type in the Terminal application npm audit fix
   - Close the Terminal application

8. Run the project
   5. Navigate in Finder to your User space and right click on the Microsoft Timeline Storyteller file. Select **New Terminal at Folder**
   6. Type in the Terminal application cd timelinestoryteller
   7. Type in the Terminal application npm start
   8. This will start a local web server at http://localhost:8000/
   9. Open a web browser and navigate to http://localhost:8000/ to use the Timeline Storyteller
   10. To exit the program, in Terminal push **Command and C** and close the Terminal application and web browser.

Every time you want to run Microsoft Timeline Storyteller, follow the protocol to start the file.

1. Navigate in Finder to your User space and right click on the Microsoft Timeline Storyteller file. Select **New Terminal at Folder**
2. Type in cd timelinestoryteller and Return
3. Type in npm start and Return
4. The response in Terminal should be:

```
> timeline_storyteller@0.1.1 start

> node app.js

Server started on port 8000

…

webpack: Compiled successfully.
```

5. To exit the program, in Terminal push **Command and C** and close the Terminal application and web browser.