QUICK START GUIDE

HAIviz: Healthcare Associated Visualization Tool

v.1.0.0

Budi Permana

v.2024.01.15

Using HAIviz

Use it online

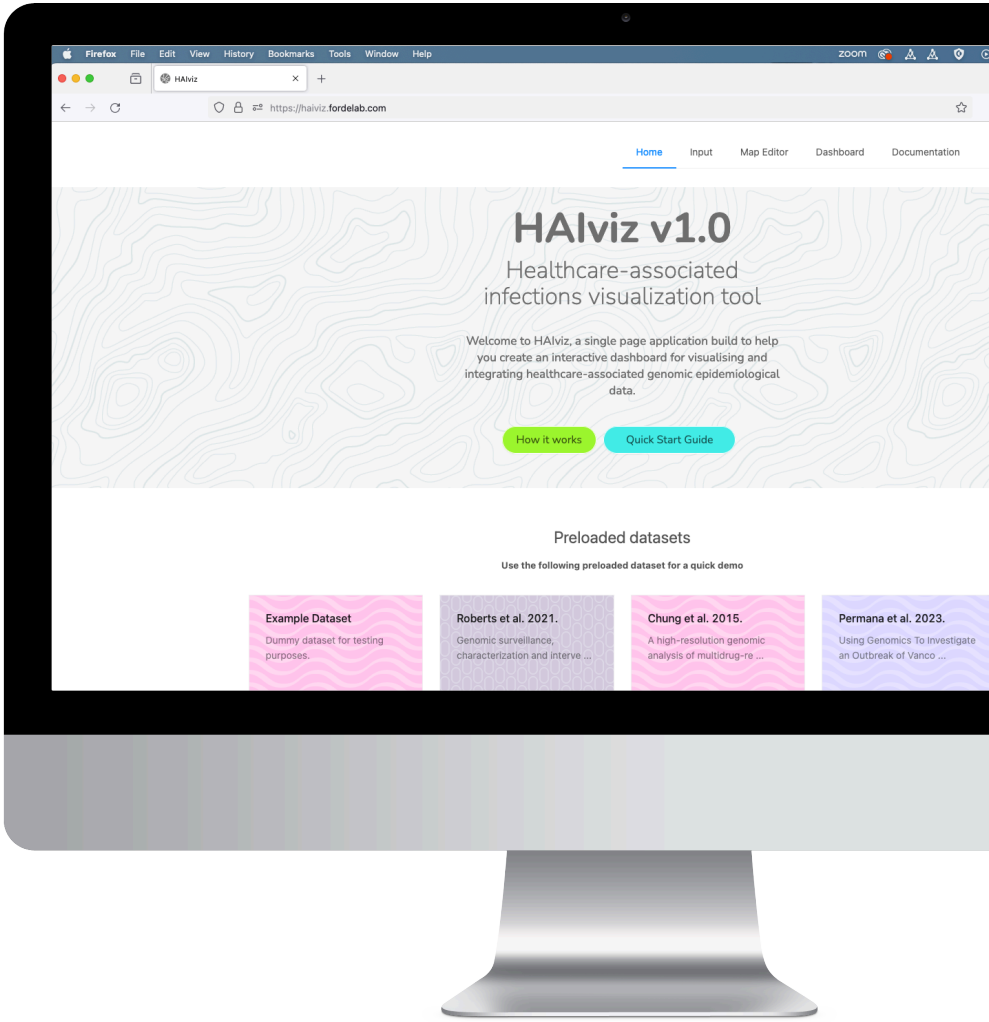
HAIviz is deployed at https://haiviz.fordelab.com for online use. Users can visit the web page using modern browsers (e.g., Google Chrome, Firefox, Microsoft Edge), drag and drop the input files, and instantly create visualisation dashboard.

All visualisation processes are performed locally in the user's browser with no data uploaded to the server, ensuring the safety of private data

https://**haiviz.fordelab.com/**

Tested and compatible with

Chrome provides the best experience.

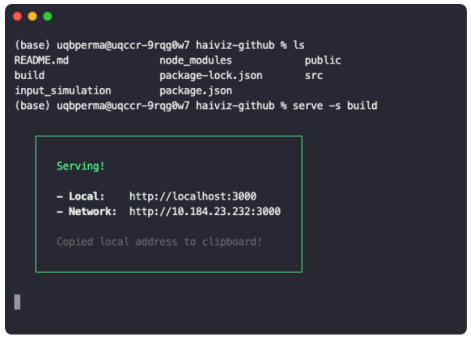


Self host or use it offline

Users also can use HAIviz offline by serving it through a static file server, such as, but not limited to `serve` Node.js package.

1. Install Node.js

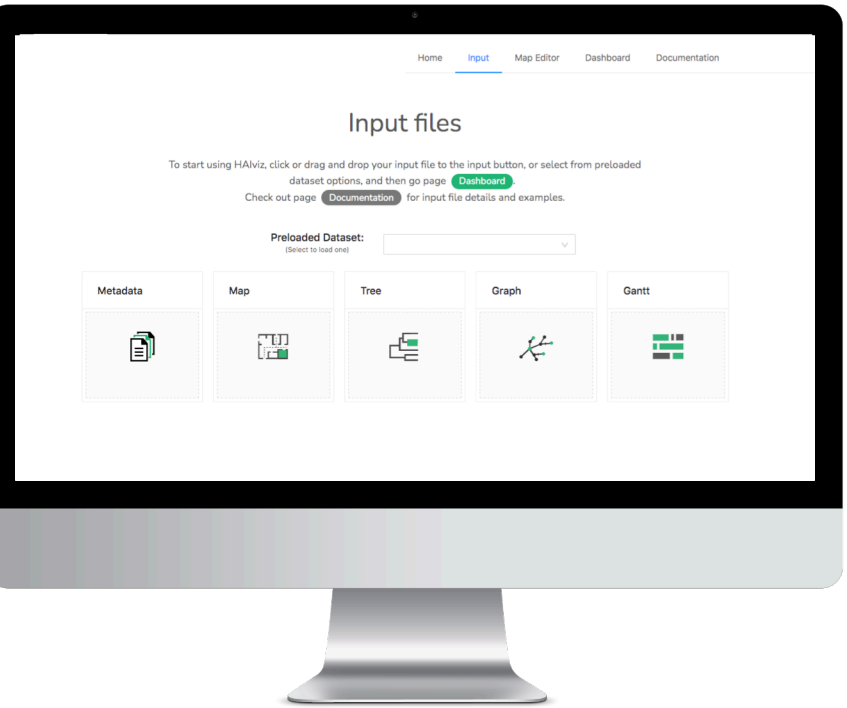
2. Install `serve` Node.js package



3. Serve the build directory

HAIviz is a single page application (SPA) visualisation tool that runs on the browser. Users can visualise and explore data by loading their input files or setting up preloaded datasets (can do it on the offline mode only).

Input Files



HAIviz is showing page *Input*. Users can click the file input loader or drag and drop the files into the input area. Input file will be parsed and validated, if file is valid, the window's icon in side-menu in page Dashboard will be **activated**.

Window's icon

Metadata

A table contains information about the isolates or samples, written in CSV format. No duplicated records in column **id** and all dates must written in **ISO 8601 format (YYYY-MM-DD)**.

Mandatory columns (fixed header) | **Other columns (optional, user-defined header)** | **Color columns (optional, [column]:color)**

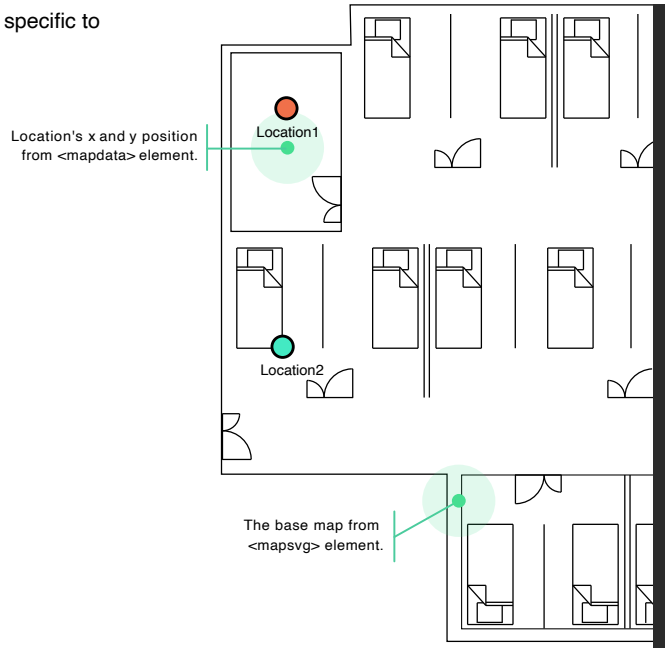
id	date	location	species	source	location:color	species:color
Isolate1	2019-12-20	Location1	Species1	Patient	blue	#9e0142
Isolate2	2019-12-21	Location2	Species2	Environment	lightgreen	#c12949

Local map

An XML file contains SVG element and location indexes. This file is specific to HAIviz and can be created in page *Map Editor*.

```
<?xml version="1.0" encoding="UTF-8"?>
<haivizmap>
    <mapsvg>
        <svg
        xmlns="http://www.w3.org/2000/svg"
        id="haiviz-localmap-svg"
        width="1000" height="1000"
        viewBox="0 0 1000 1000" >
        <!-- The JPEG/PNG is embedded as
        the SVG element here-->
        </svg>
    </mapsvg>
    <mapdata>
        <location name="Location1" x="100" y="150"/>
        <location name="Location2" x="100" y="350"/>
    </mapdata>
</haivizmap>
```

Example of SVG attributes from a JPGE base image of 1000x1000px.



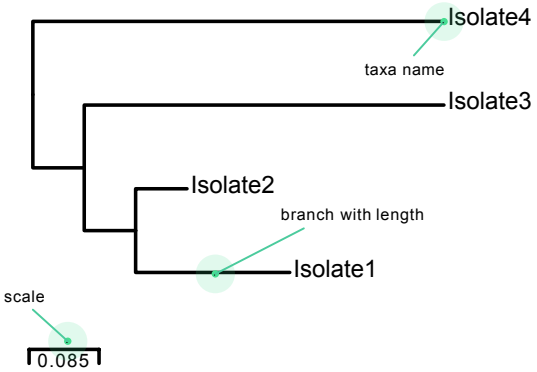
Input Files (cont.)

Phylogenetic tree

A Newick formatted phylogenetic tree file with taxa name (tip label) and branch length.

taxa name branch length

```
(Isolate4:0.5,(Isolate3:
0.4375,(Isolate2:0.062
5,Isolate1:0.1875):0.0
625):0.0625);
```



Network (graph)

A text file describing a graph object written in DOT language (**https://graphviz.org/doc/info/lang.html**). Using igraph R package you can export a network object into DOT format file with this following code: *write.graph(YourGraphObject, "myGraphInDOT.gv", format = "dot").*

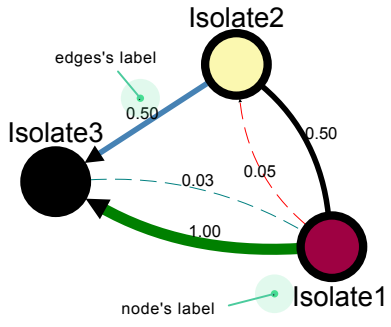
directed graph object

node node's attribute

```
digraph {
  0[name=Isolate1];
  1[name=Isolate2];
  2[name=Isolate3];
  0->1[color="red" weight=0.05 style="dashed"];
  1->0[color="black" weight=0.5 style="solid"];
  0->2[color="green" weight=1style="solid"];
  1->2[color="steelblue" weight=0.5 style="solid"];
  2->0[color="teal" weight=0.03 style="dashed"]
}
```

edge

edges's attributes

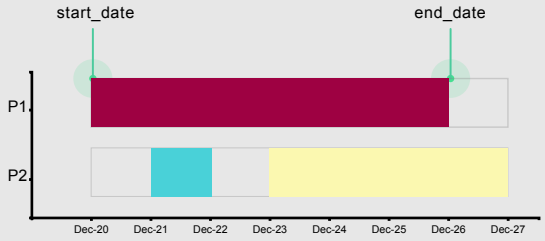


Movement timeline

A table file describing an individual movements (e.g. patient) from time to time at a single or multiple locations, written in CSV format. Column dates cannot empty and must written in **ISO 8601 format (YYYY-MM-DD),** and column *start_date* must be less than or equal to column *end_date*.

Mandatory headers and columns (Fixed header name) Optional

pid	start_date	end_date	location	location_color
P1	2019-12-20	2019-12-25	Location1	#9E0142
P2	2019-12-21	2019-12-22	Location2	#49D1D8
P2	2019-12-23	2019-12-26	Location3	#FBF8B0



Dashboard Interface

Visualisation menu

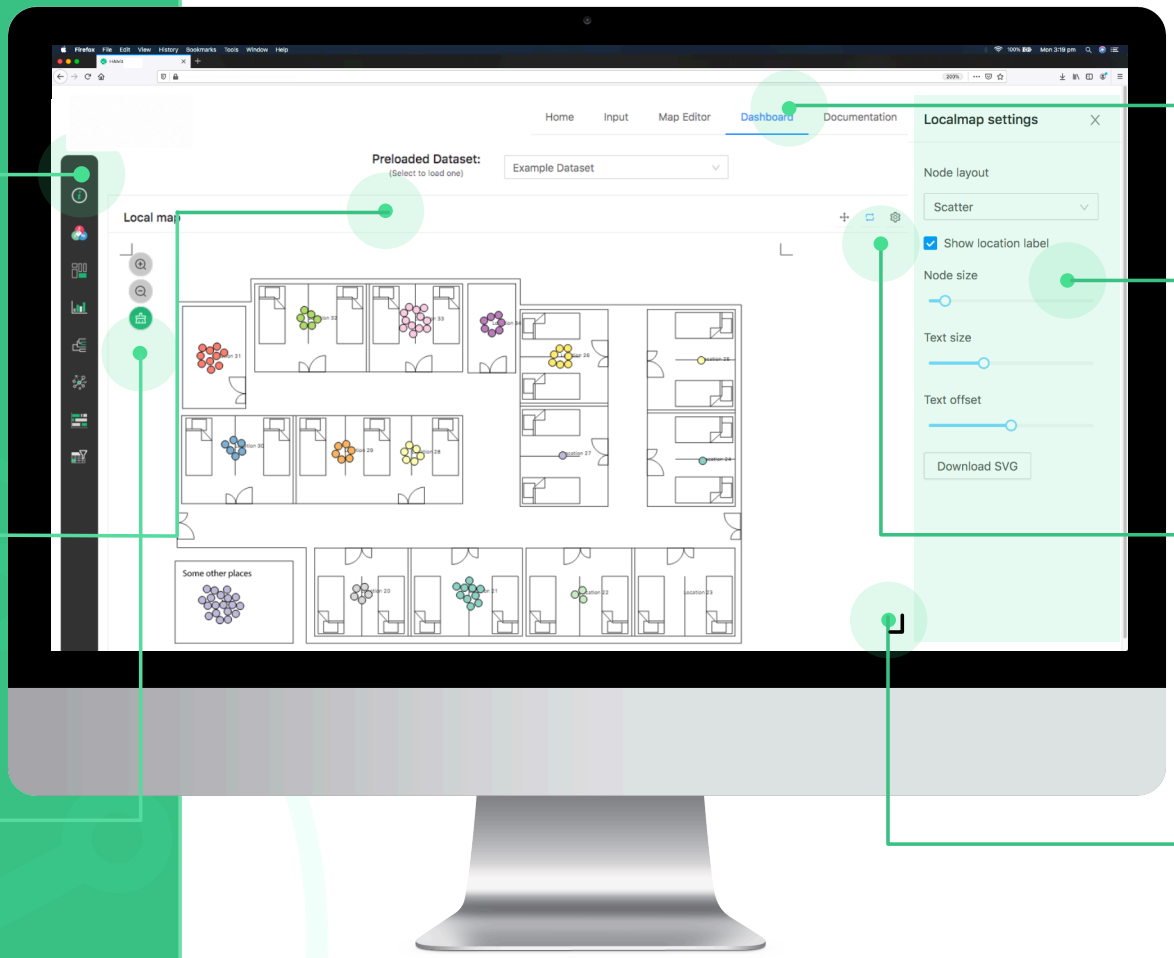
Host all visualisation windows. Each icon will be activated as soon as visualisation from your input file is ready. Click on the icon to display the window.

Visualisation window

A window container where the interactive graphics are being rendered. The window can be individually moved, resized and closed.

Visualisation controller

Let you change the current state of visualisation display, such as, zoom in, zoom out and clear data selection.



Navigation

The main application navigation to let you jump between pages.

Window settings

A side drawer provides you additional actions or settings related to the current window.

Window controller

(From left to right) Let you move the window, re-draw visualisation, open additional settings and close the window.

Resize controller

Let you change the current size of your visualisation window.

HAIviz is showing page Dashboard with map window displayed. Coloured nodes (circles) represent isolates clustered based in their location. Users can click, mouseover, zoom, pan, change the node layout, node size, location text size and download the the visualisation result.

Creating local map

HAIviz is showing page *Map Editor*. Users can start creating the map by loading an JPEG or PNG image. When the image file is loaded, users can add, rename, update or remove a location. A final map can be download for HAIviz use.

(!)
- For a map to be downloadable, at least one map location must be added.
- Try open the downloaded Map (*haivizMap.xml*) in a text editor, you even can change the coordinates manually.
- The x and y coordiates follow the input base map image (e.g. x=0 and y=0 will be the very top-left of the image).

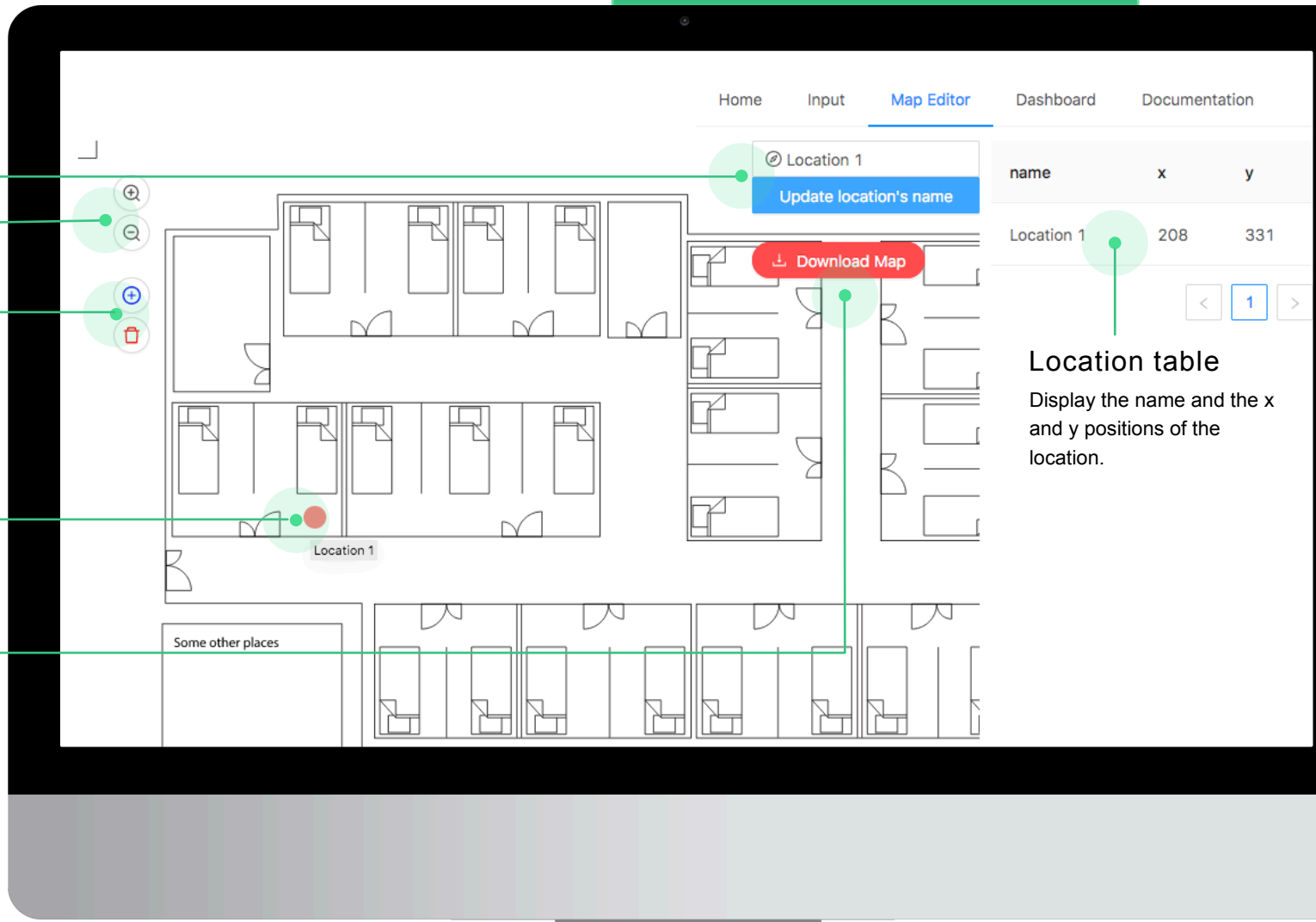
Location labeller
Let you enter and update the name of a location.

Zoom controller
Let you zoom in and zoom out the map.

Location logger
Let you add a new location and remove the selected location.

Location marker
To remove or update location name, users can click or drag to select the marker. Mouse over the circle will display the location label.

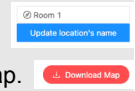
Map downloader
Let you download the map for later usage or load the map directly to HAIviz.

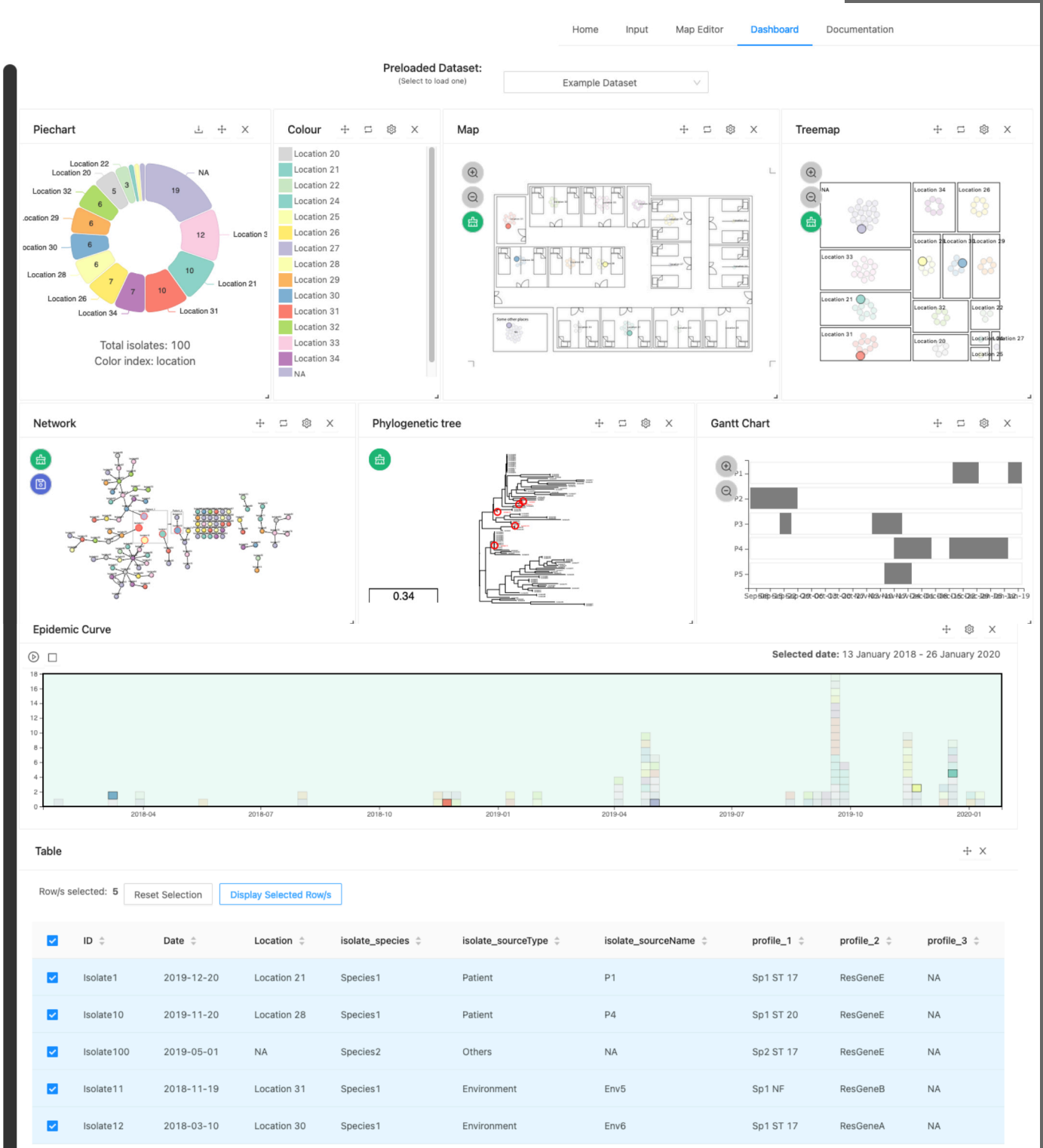


Location table
Display the name and the x and y positions of the location.

Five quick steps:

1. Add a location marker.
2. Move the marker to the desired coordinate.
3. Set the label for the marker.
4. Click the 'Update location's name' to update.
5. Repeat steps 1 to 4, then click Download Map.





Interaction And Integration

- HAIviz is showing the integrated and interactive visualisation windows in page *Dashboard* created from an example dataset.

- To demonstrate integration functionality, five isolates were selected from the table window and were highlighted in the other active windows.

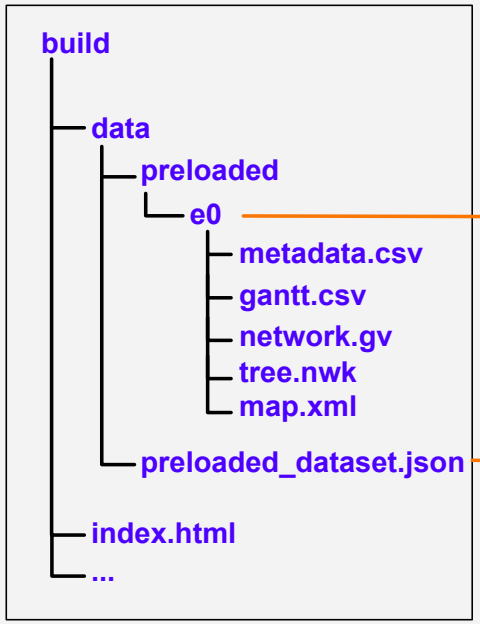
- Selection can also be performed in other windows, such as using an interactive brush on temporal distribution window to create animation.

- Image produced by HAIviz can be saved to an SVG format, enabling quick and flexible editing for report and publication.

Setting up preloaded dataset

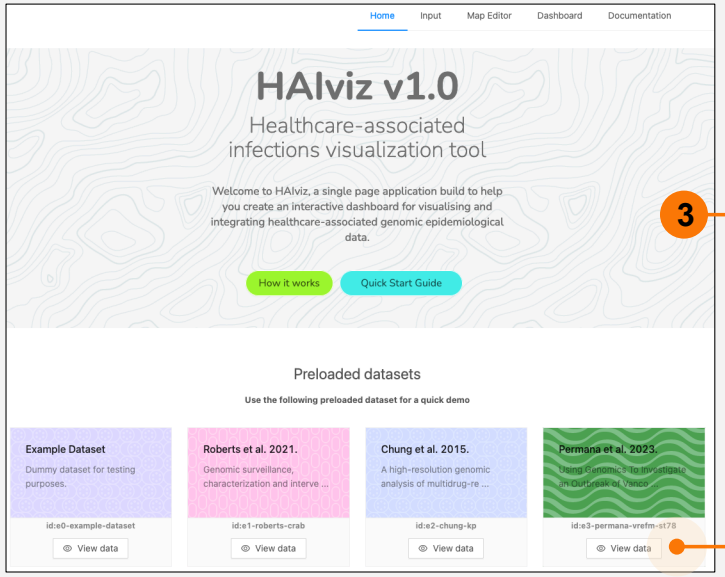
When users self-host or use HAIviz offline, they can set up multiple preloaded datasets. This feature allows users to ‘permanently’ link their input files to HAIviz, avoiding the need to manually re-inputting their input files.

Example of directory tree of HAIviz build directory



1 **Create a directory in the *preloaded* directory then add the input files.**

2 **Update the preloaded_ dataset.json file**



3 **Datasets is listed in HAIviz Home, Input, and Dashbord page.**

Click View data (in page Home) or select from the preloaded dataset dropdown button (in page Input or Dashboard) to load the all input files to HAIviz.

Example of preloaded_dataset.json file

```
{
  "data_list": [
    {
      "id": "e0-example-dataset",
      "name": "Example Dataset",
      "description": "Dummy dataset for testing purposes.",
      "metadata": "./data/preloaded/e0/metadata.csv",
      "map": "./data/preloaded/e0/map.xml",
      "tree": "./data/preloaded/e0/tree.nwk",
      "network": "./data/preloaded/e0/network.gv",
      "gantt": "./data/preloaded/e0/gantt.csv"
      }
    ],
  "description": ["This JSON file describes the preloaded datasets."]
}
```

An example R script to progamratically setup the preloaded datasets is given in the `input_simulation` directory in HAIviz repository : setup_preloaded_dataset.R.

THANK YOU

for reading this guide

HAIviz
Healthcare-Associated
Infections Visualization Tool

Thanks to all awesome web frameworks and libraries run on the background, HAIviz is now up and running and available worldwide. The following are the core libraries used by HAIviz.

"@nivo/pie": "^0.61.1",
"antd": "^4.2.0",
"babel-polyfill": "6.26.0",
"cytoscape": "^3.17.0",
"cytoscape-cose-bilkent": "^4.1.0",
"cytoscape-fcose": "^2.2.0",
"cytoscape-spread": "^3.0.0",
"cytoscape-svg": "^0.3.1",
"d3": "^5.16.0",
"d3-array": "^2.4.0",
"d3-color": "^1.4.1",
"d3-delaunay": "^5.2.1",
"d3-fetch": "1.1.0",
"d3-scale-chromatic": "1.2.0",
"dotparser": "^0.4.0",
"export-to-csv": "^0.2.1",
"immutable": "3.8.2",
"install": "^0.13.0",
"jscrambler": "^5.5.18",
"jspdf": "^2.5.1",
"lodash": "^4.17.15",
"moment": "^2.25.3",
"moment-range": "^4.0.2",
"npm": "^6.14.9",
"phylocanvas": "^2.8.1",
"phylocanvas-plugin-export-svg": "^1.0.0",
"phylocanvas-plugin-scalebar": "^1.1.1",
"prop-types": "15.6.1",
"react": "^16.13.1",
"react-app-polyfill": "^2.0.0",
"react-color": "^2.17.3",
"react-dom": "^16.13.1",
"react-faux-dom": "4.1.0",
"react-file-drop": "^0.2.8",
"react-grid-layout": "0.16.6",
"react-measure": "1.4.7",
"react-pdf": "^4.2.0",
"react-player": "^2.7.2",
"react-redux": "5.0.7",
"react-router": "^5.1.2",
"react-router-dom": "^5.1.2",
"react-scripts": "2.0.0",
"recompose": "0.26.0",
"redux": "3.7.2",
"redux-immutable": "4.0.0",
"redux-promise": "0.5.3",
"reselect": "3.0.1",
"resize-observer-polyfill": "^1.5.1",
"svgsaver": "0.9.0",
"uuid": "^8.0.0",
"xml-formatter": "^2.0.1",
"xml-js": "^1.6.11"

React
Data-Driven Documents
Cytoscape
Ant Design
React-Grid Layout
Phylocanvas