



CADViewer JS

CADViewer JS – Generating Automated Hotspots in drawings

High level interaction with content is one of the cornerstones of CADViewer JS web viewer and AutoXchange conversion software.

To generate automated hotspots in CAD drawings, such as AutoCAD DWG or MicroStation DGN drawings, the drawing must contain two layers with content that through an automated process are matched to a set of interactive objects.

The CAD drawing, in addition to regular graphical content, must contain:

- 1: A layer with text object IDs. – The Text Layer
- 2: A layer with polygons that surrounds the text object IDs. – The Room Layer

An automated process in AutoXchange will then link the Text and Room layers into interactive objects.

The process can be done directly using the AutoXchange command line interface, or it can be done interactively using CADViewer JS.

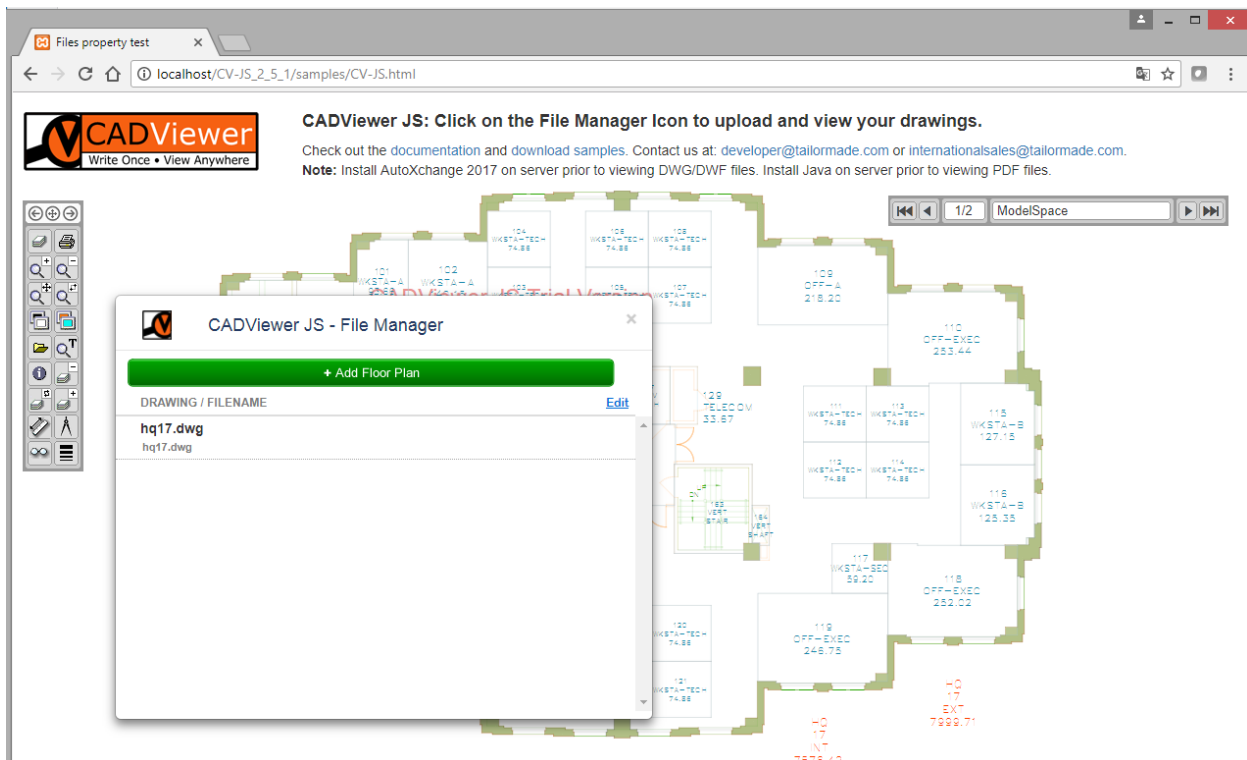
Below are described the two methods of creating hotspots.

Method 1:

Generating Hotspots in drawings interactively using CADViewer JS.

Creating hotspots directly in CADViewer JS, uses the CADViewer interface. It gives the admin user a graphical understanding of the layer structure and can be used if the layer organization of the file is not known beforehand.

1: Open the drawing in CADViewer JS.

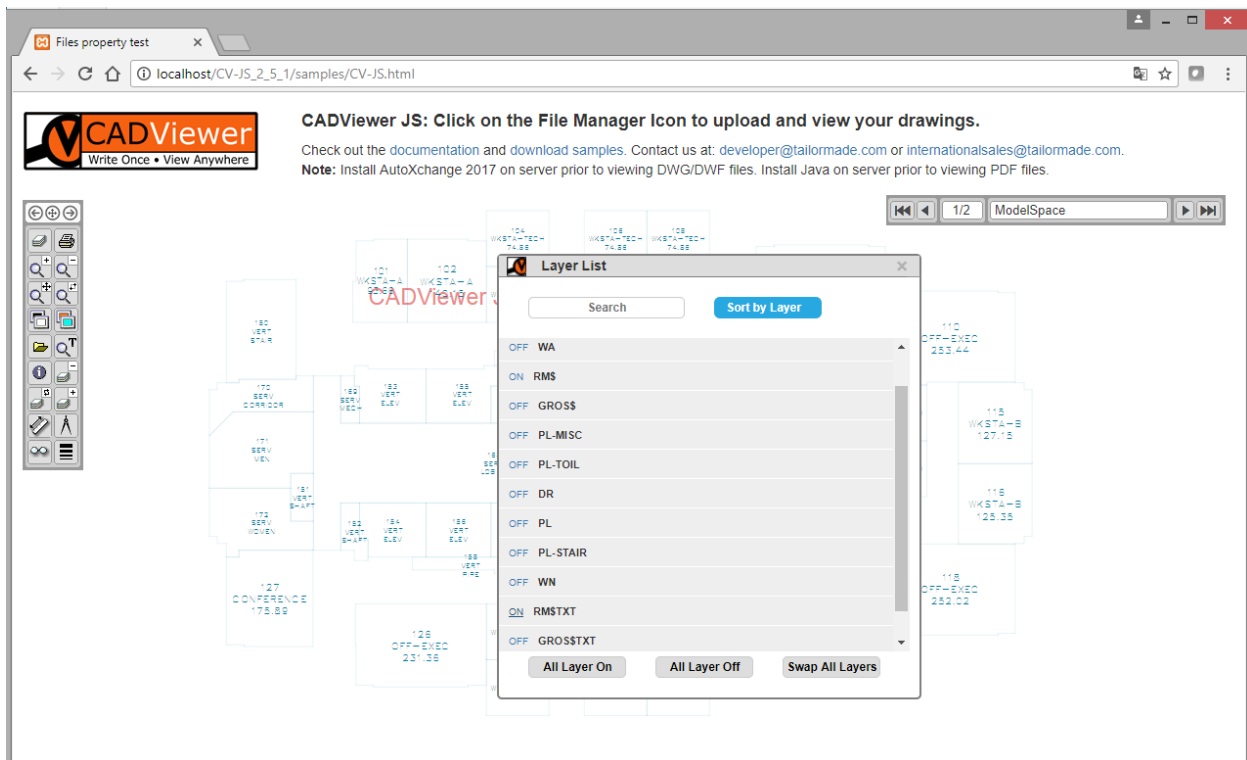




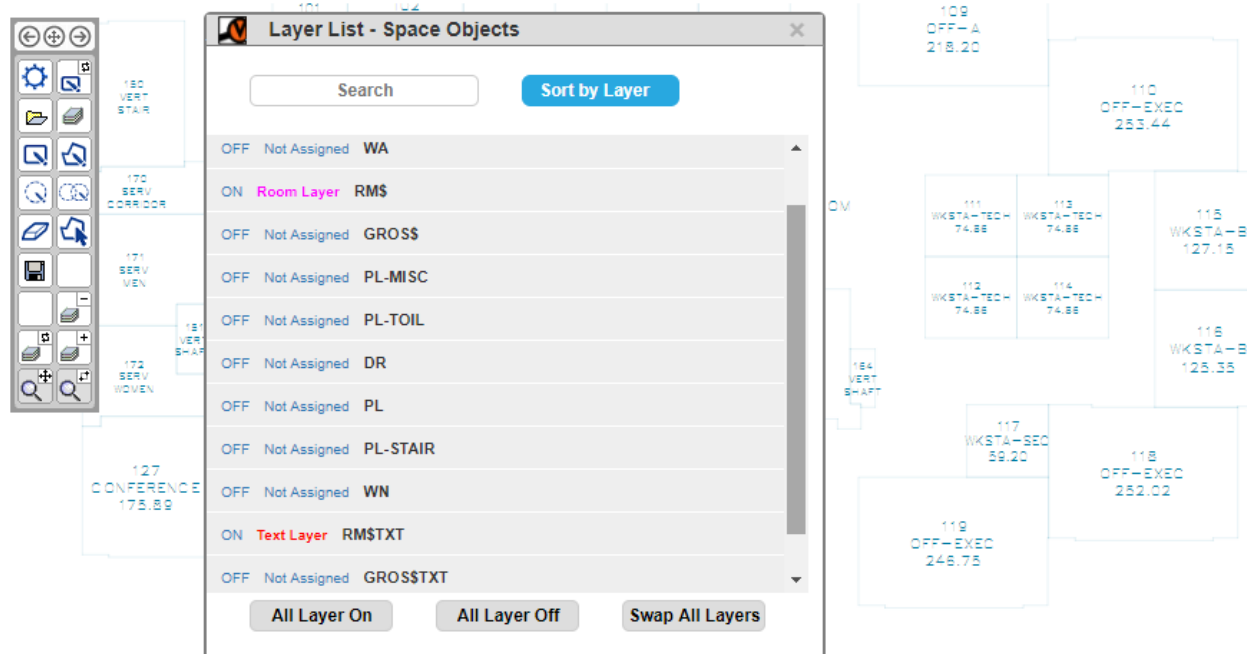
CADViewer JS – Generating Automated Hotspots in drawings

2: Use layer tool to find Room Layer and Text Layer.

There are several ways to identify layers with CADViewer JS. It can be done using the interactive layer tool to click on text and room layer to turn them off and then swap on/off layers to turn them on, alternatively use layer list to find layers. Here the Room Layer and Text layers have been identified.



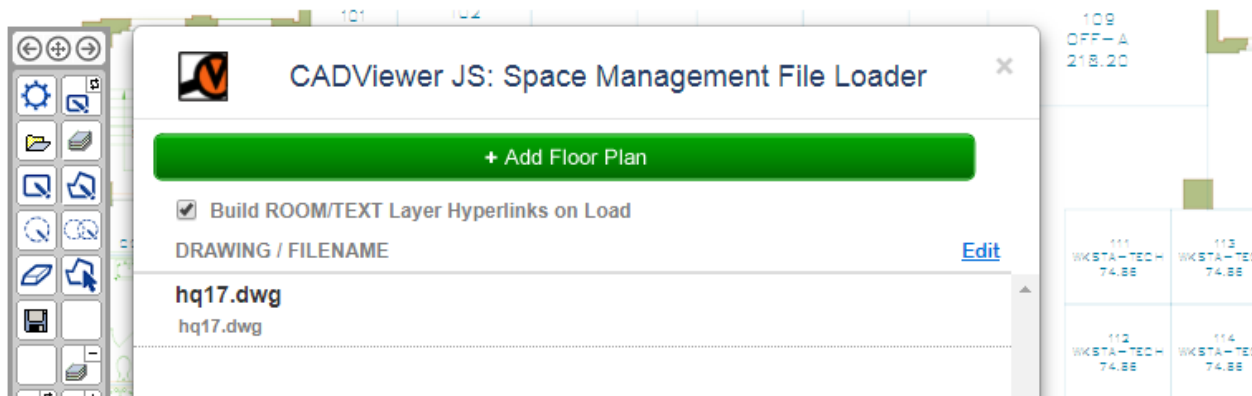
3: Use Space Object layer list command to specify the Room Layer and Text Layer name.



CADViewer JS – Generating Automated Hotspots in drawings

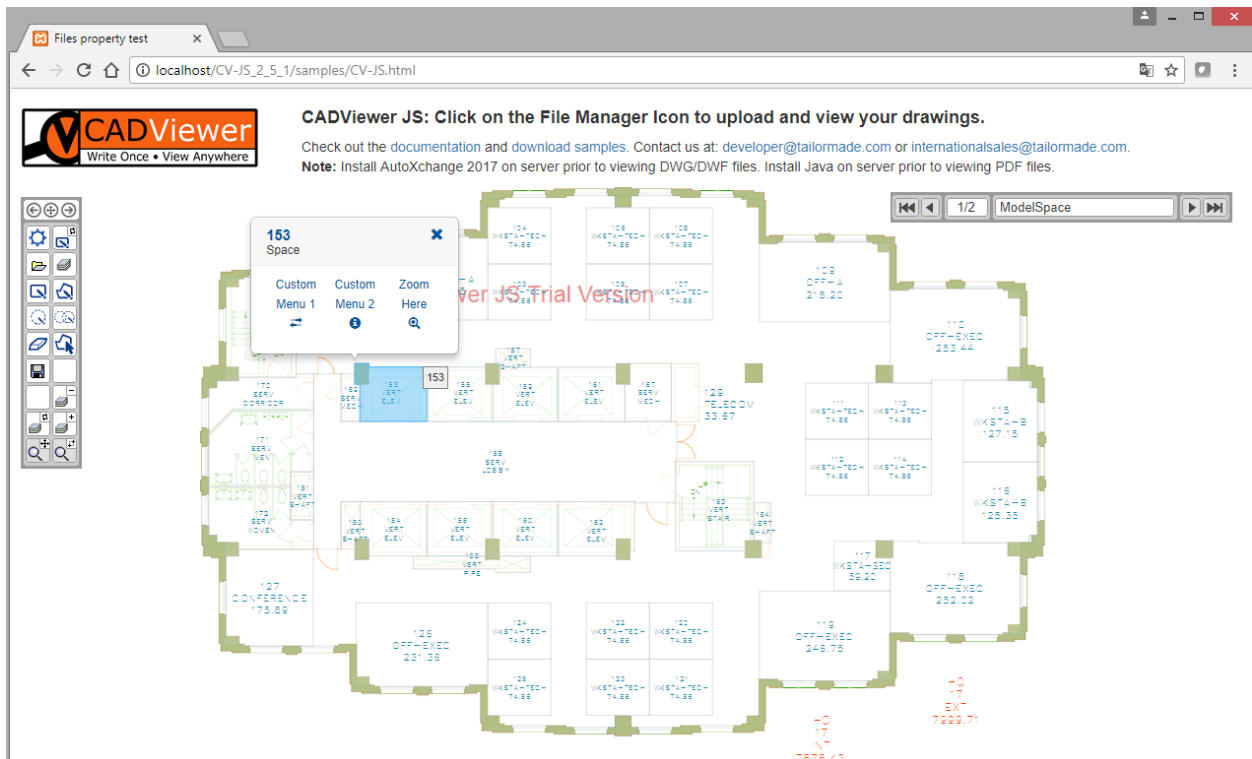
4: Use Space Management File Loader to load drawing with hotspots.

AutoXchange does the automated processing of space objects using the reference layer names.



5: Drawing with interactive hotspots.

Drawing now has interactive hotspots. The call-out modal is fully configurable. It can be set up to interact with database content.





CADViewer JS – Generating Automated Hotspots in drawings

Method 2:

Generating Hotspots in drawings using AutoXchange.

Creating hotspots server side with AutoXchange ahead of time instead of doing an on-the-fly conversion from CADViewer JS can save valuable time and computing resources, especially with very large files.

For small and medium sizes drawings the execution time does not differ much using on-the-fly conversion compared to conversion ahead of time.

The process to create an SVG file with hotspots ahead of time, can be done on both Windows and Linux.

It is illustrated with Windows below, *but the process is similar on Linux*, the only difference being that the executable on Windows is >ax2017 and on Linux it is \$./ax2017_L64_01.

Please follow general [CV-JS install instructions](#) on Linux on how to associate .so libraries needed.

1: Navigate to the install folder of AutoXchange 2017:

```
C:\xampp\htdocs\CV-JS_2_5_1\converters\ax2017\windows>
```

2: In AutoXchange 2017 folder, check that the AX2017 is running properly:

```
>ax2017 -?
```

3: Build the command line:

The following command that needs to be used are:

- i= : input file name, this can also include path
- o= : output file name, this can also include path
- f=svg : output format, this must be specified to SVG
- model : specify the converter to use the model space of the drawing
- rl= : specify the name of the room layer
- tl= : specify the name of the text layer

Optional:

-treatas= : some formats like Microstation DGN uses a different file name extension PCF to encapsulate DGN files, AutoXchange needs to be told that. In this case the parameter is: -treatas=dgn. This is not needed for AutoCAD DWG drawings.



CADViewer JS – Generating Automated Hotspots in drawings

4: Execute the command line:

```
C:\xampp\htdocs\CV-JS_2_5_1\converters\ax2017\windows>ax2017 -i="hq17.dwg" -o="hg17.svg" -f=svg -model  
-tl="RM$TXT" -rl="RM$"
```

*AutoXchange 2017 V1.01.08a
Copyright (c) 2017 Tailor Made Software, Ltd.
All Rights Reserved*

Portions Copyright (c) 1997-2016, Tailor Made Software, Ltd.

*Input File Name: hq17.dwg
Output File Name: hg17.svg
Elapsed Time: 0.459 seconds
AutoXChange 2017 Finished Successfully*

5: SVG file with hotspots is now generated:

Move the converted SVG drawing to the folder where CADViewer JS is running and associate it with CADViewer JS, or implement the steps above as part of a batch process.

If there are issues with the conversion step above, you can add the command **-trace** to the command line. This will give a full trace on how AutoXchange loads files, parameters, etc., and therefore can be used as debug information in setting paths up correctly.

Contact:

Please contact us for further product information:

Inquiries: internationalsales@tailormade.com

Developers: developer@tailormade.com

Web: www.tailormade.com, www.cadviewer.com, www.vizquery.com

Address: **Tailor Made Software, Ltd.**

443 Ramsay Way, Suite 430, Kent, WA 98032