

Adobe® SVG Viewer for Windows

Release Notes

Version 2.0 (Build 55, 03/14/01)

Contents

- [About the SVG Viewer](#)
- [System requirements](#)
- [How to install the SVG Viewer](#)
- [How to copy Netscape plug-ins into other Browsers](#)
- [How to view SVG files in your Web browser](#)
- [How to turn artwork into SVG](#)
- [Tips and Hints](#)
- [Known Problems](#)
- [Revision History](#)
- [Contact Information](#)
- [Copyright Notice](#)
- [More Information on Adobe's support for SVG](#)

About the SVG Viewer

The Adobe SVG Viewer supports much of the [Candidate Recommendation](#) draft specification of SVG published on 11/2/00. Please keep in mind that the specification is under development, and will continue to evolve until it is made a recommended specification.

System Requirements

- Windows 95, 98, SE, 2000, ME, or NT 4.0 Service Pack 4 and up
- Netscape Navigator or Communicator versions 4.0 through 4.75, or Internet Explorer 4.0 or

higher. Netscape 4.77 is not supported. Netscape 6 is not supported.

- 13 MB of hard disk space
 - 32 MB of RAM recommended
-

How to install the SVG Viewer

1. Before installing, please close any application which you have used to view SVG with an old version of the viewer.
2. Run the installer. The latest released installers can be downloaded from <http://www.adobe.com/svg/viewer/install>.

How to copy Netscape plug-ins into other Browsers

If you install another copy of Netscape which is supported by the Netscape plug-in, you can copy the following files into your plug-ins folder to view SVG from within that browser:

- NPSVGVw.dll
- SVGView.dll
- SVGViewer.zip

How to view SVG files in your Web browser

Once you install the Adobe SVG Viewer, you should be able to view supported SVG files in any supported Web browser.

How to turn artwork into SVG

You can export SVG from Illustrator 9. If you wish to incorporate JavaScript into your SVG you can create your artwork in Illustrator and link to JavaScript functions with the JavaScript Interactivity Pallet and then export it as SVG, or export from Illustrator and add any JavaScript interactivity animation or filter effects by hand.

Tips and Hints

- Style attributes (or style attributes that use entity references) render more quickly than embedded stylesheets.
- Use tightly-bound percentages on the `x`, `y`, `width`, and `height` attributes on `filter` elements in order to minimize the area over which the filter effect needs to be calculated. This will result in graphics which display much more quickly.
- You can retrieve the SVG document object from within SVG event handlers by calling `getTarget` on the `evt` object which is passed in. For example:

```
// Retrieve the SVG document object:
var directTarget = evt.getTarget();
var svgDocument;
if( directTarget.getNodeType() != 9 ) // if not DOCUMENT_NODE
    svgDocument = directTarget.getOwnerDocument();
else
    svgDocument = directTarget;
```

Known Problems with the SVG Viewer

All Browsers

1. If you install the Adobe SVG Viewer as an administrator of a Windows 2000 system, you will need to view an SVG file and accept the EULA before non-administrators can view SVG on that machine.
2. It's possible for elements near the edge of an SVG graphic to not receive a mouseout event.
3. The SVG specification's support for masks has changed significantly since Illustrator 9 shipped, so Adobe SVG Viewer 2.0 might not display SVG masks exported from Illustrator 9.
4. If you have multiple Windows operating systems (such as Windows 98 and Windows NT) installed on the same partition (for example, drive C), you must install the Adobe SVG Viewer on both operating systems. If you install it on one operating system, the Viewer will not work correctly when you reboot under the other operating system.

Internet Explorer

1. You must have Internet Explorer 4 or higher installed in order to use the SVG Viewer as an ActiveX control in other applications (like Microsoft Office).
2. If you open more than one window containing a link to the same SVG file, and the SVG file is embedded using the HTML `<embed>` tag, then when you print one of the windows it may print based on changes to the DOM in the other window. This is due to limitations of Internet Explorer.
3. Due to Internet Explorer limitations, it's not possible to see properties of other windows from a script defined inside an SVG file.
4. Using the Adobe SVG Netscape plug-in under Internet Explorer on Windows is **not** supported.

Netscape

1. Netscape 4.x plug-ins cannot execute `onunload` scripts because at the time the plug-in receives notification from Netscape that the plug-in is being unloaded, the plug-in's script environment has already been closed down.
2. Netscape 6.0 reports errors when printing pages with plug-ins, and then fails to print embedded contents. Netscape will report two plugin error alerts. Dismissing the alerts presented will then result in Netscape printing the HTML page, but with blank area for the SVG content.
3. When you leave a Web page, Netscape can often unload the Adobe SVG Viewer plug-in before Netscape terminates all scripts running in the HTML. Therefore you should be careful when writing scripts to check to see if the plug-in is still loaded before you attempt to access the SVG DOM from `setTimeout` or `setInterval` callbacks in your HTML JavaScript.
4. Due to design limitations of Netscape, avoid using HTML with the following form:

```
<a href="#" onclick="myScriptWhichModifiesSVG()">
```

The problem is that JavaScript responds to the `onclick` event and begins executing the script, but then Netscape also responds to the fact that you clicked on a link, and interrupts JavaScript to re-load the current page. This conflict can be avoided by re-writing the code as follows:

```
<a href="javascript:myScriptWhichModifiesSVG()">
```

5. If a JavaScript event handler contains an error, any subsequent JavaScript will fail to execute. This is due to a bug in Netscape's handling of scripts executed by plug-ins.
6. Due to Netscape script limitations, you can't change the URL of another frame from an SVG

event handler.

7. Due to Netscape script engine limitations, calling the `alert()` function from within an SVG event handler might not work, and it might cause subsequent JavaScript to no longer execute. Similar problems occur for any Netscape modal window, including the security privileges window. This bug appears to have been partially fixed by Netscape in Netscape 4.75, but not on Windows 98.
8. There is a bug in Netscape involving plug-ins that use JavaScript. If you load a page with a plug-in that uses JavaScript and then disable Java and quit the browser, Netscape will crash.
9. Netscape does not always allow you to access the `document.embeds[]` array from within an SVG script. To work around this, use `document.mySVG`, where `mySVG` is the name of your embed object.
10. Netscape 4.x will crash if a plug-in invokes a link to a JPEG file. To work around this problem, embed your JPEG file in HTML or SVG before you link to it from your SVG file.
11. Errors in scripts executed by Netscape 4.x plug-ins do not show up in the Javascript console window.
12. Netscape 4.x leaks a small amount of memory every time a plug-in executes a script.
13. Netscape 4.x will crash when you close it if you launched it by double-clicking to load a Web page containing LiveConnect plug-in content.
14. If you use "Print Preview" in Netscape on a page which contains plug-ins (such as the Adobe SVG Viewer), Netscape will crash. For a similar reason Netscape will crash when you try to print a page containing plug-ins using Acrobat Distiller.
15. If you use Netscape 4.0x to view the same SVG file in two windows, and that file uses a global JavaScript function on the window object, sometimes JavaScript will not initialize in the second window. The work-around is to upgrade to a more recent version of Netscape.
16. Due to a bug in Netscape 6, attempting to right-click within the SVG area to bring up context menu fails in Netscape 6. Instead, you get the Netscape 6 context menu.
17. Netscape 4.x on Swedish systems has a bug where it does not identify its plug-in folder location correctly in the system registry, thereby making it impossible for installers to install plug-ins. You can work around this problem by copying the files by hand, as described [above](#).

Revision History

Changes since version 1.0 of the Adobe SVG Viewer

- Updated to support the [Candidate Recommendation](#) draft specification of SVG published on 11/2/00. The 2.0 Viewer retains backward compatibility with the 1.0 Viewer where possible, and therefore supports much of the 3/3/00 draft specification of SVG. However, some features of the specification (such as the handling of CSS units) have changed so significantly that it was not possible to retain backward compatibility.
- Elements and Attributes:
 - Added support for the following elements: `altGlyph`, `altGlyphDef`, `font`, `font-face`, `glyph`, `glyphRef`, `hkern`, `missing-glyph`, `mpath`, `pattern`, `switch`, and `vkern`.
 - Added support for the SVG Presentation Attributes.
 - Finished support for `use` element use of `symbol` elements.
 - Added support for the `spreadMethod` attribute on gradients.
 - Added support for the new attribute `primitiveUnits` on filter effects.
 - Added support for "discrete" mode for `feComponentTransfer`.
- CSS properties:
 - Added support for the following CSS properties: `alignment-baseline`, `color-interpolation`, `direction`, `dominant-baseline`, `flood-color`, `flood-opacity`, `kerning`, `lighting-color`, `pointer-events`, and `unicode-bidi`. Support for the `pointer-events` property does not include support for transparency on raster elements.
 - Added support for the `font-variant` property, but only when applied to SVG fonts.
 - Removed support for the `baseline-identifier` property, which has been replaced with the `alignment-baseline` property in the most recently supported specification.
 - Changed the default values for the properties `clip-rule` and `fill-rule` from `even-odd` to `non-zero` to reflect the changes in the most recently supported specification.
 - The latest Candidate Recommendation draft specification for SVG changes the way CSS units are handled. They are now converted to user space on parse, rather than being converted at display time. This significantly changes the way the "Zoom" command works.
- DOM:
 - The Viewer's DOM Level 2 support now conforms to the Candidate Release specification for DOM 2, except for the `KeyEvent` interface, which is not specified in the DOM 2 Candidate Release specification. Support for the `KeyEvent` interface conforms to the last DOM 2 published version of that interface, in the 9/99 draft specification.

- Added support for the `SVGRect` interface.
- Added support for the `SVGPoint` interface, except for the method `matrixTransform`.
- Added support for the `SVGTextContentElement` interface, except for the `getTextLength` and `getLengthAdjust` properties.
- Added support for `rootElement`, `getRootElement`, `currentScale`, `getCurrentScale`, `setCurrentScale`, `currentTranslate`, `getCurrentTranslate`, and `setCurrentTranslate` on the `SVGDocument` interface.
- Added support for the DOM2 methods `hasAttribute` and `hasAttributeNS` on the `Element` interface.
- Added support for a new, non-standard, method on the `Document` object called `garbageCollect`. This method takes no parameters and returns no results. Calling this method will free up any `Node` objects that have previously been removed from the tree. If a script has performed an operation that results in the removal of a node from the tree, once the script has no further use for the removed (and returned) node, the script should call `garbageCollect` on the SVG document. Once called, any reference a script has to a node which was not part of the document when `garbageCollect` was called will always fail when its methods are accessed. This is only of benefit for Netscape browsers, since Netscape's `LiveConnect` does not offer weak reference support. For easy cross-platform support, this call can be made under Internet Explorer, but it will do nothing on that platform.
- Other Changes:
 - To conform to the latest Candidate Recommendation draft specification for SVG, support for `userSpace` units has been dropped.
 - Added support for user stylesheets. The SVG Viewer looks for a file named `SVG-User-Styles.css`, according to the following search strategy:

Operating System	Paths Searched
------------------	----------------

Windows 2000	<ol style="list-style-type: none"> 1. My Documents 2. \Documents And Settings\<user>\Application Data</user> 3. \Documents And Settings\<user>\Local Settings\Application Data</user> 4. \Documents And Settings\All Users\Application Data
Windows 98	<ol style="list-style-type: none"> 1. My Documents
Windows NT	<ol style="list-style-type: none"> 1. \WINNT\Profiles\<user>\Personal</user> 2. \WINNT\Profiles\<user>\Application Data</user>
Macintosh OS 8.x	<ol style="list-style-type: none"> 1. System Folder:Preferences 2. Startup Volume:Documents
Macintosh OS 9.x (single- user)	<ol style="list-style-type: none"> 1. System Folder:Preferences 2. System Folder:Users:<user> 3. Startup Volume:Documents
Macintosh OS 9.x (multi- user)	<ol style="list-style-type: none"> 1. Startup Volume:Users:<user>:Preferences 2. Startup Volume:Documents

- The Adobe SVG Viewer now has its own color management support built in. This support will attempt to locate a suitable device profile for your display. If your computer has color management software installed and a monitor profile has been selected, then this monitor profile will also be used by the Viewer. If you do not have color management installed, or you have disabled it, the Viewer will look for an Adobe Monitor Settings profile on your system. This profile is created by the Adobe Gamma tool (installed with other Adobe products), and is usually stored in the same directories as other color profiles on the host system. Failing to find any suitable device profile, the Viewer will fall back on the standard sRGB profile and use that for all rendering.
- The Netscape plug-in now supports DOM access from Java clients.
- The ActiveX control now supports non-JavaScript (e.g. Visual Basic) event handlers.
- Modifications for Windows 2000 compatibility.

- Added an alternate way to pan SVG graphics with the keyboard for accessibility. If the scroll-lock is on, the arrow keys will pan the SVG graphic that has the focus.
 - Performance enhancements involving multiple simultaneous animations within the same SVG file. SVG files with small, widely-spaced simultaneous animations should generally display faster in the current version than in Version 1.0.
 - Mac plug-in performance enhancements.
 - Improved support for vertical text.
 - Added support for bi-directional text.
 - General performance optimizations.
 - Modified `feImage` to work like the `use` element.
 - Updated support for PNG images.
 - Memory usage optimizations.
 - Added support for animating paths.
-

Contact Information

To report bugs and/or provide feedback, please go to the [SVG Zone](#) on Adobe.com.

Copyright Notice

© Copyright 1998-2001 by Adobe Systems Incorporated. All rights reserved.

Adobe and Illustrator are trademarks of Adobe Systems Incorporated. Windows and Windows NT are registered trademarks of Microsoft Corporation in the United States and/or other countries. Mac OS and TrueType are trademarks of Apple Computer, Inc. in the United States and/or other countries. All other trademarks are property of their respective owners.