Librsvg

An analysis of features of the current OT-SVG specs, SVG Native and SVG 1.1

The current situation of OT-SVG specs is that it says it supports all of the stuff from SVG 1.1 except a list of items which are banned, not supposed to be rendered at all, and a list that's optional, the remaining stuff is of course supported.

I'll give a list of the banned items and describe their status in the current OT-SVG specs, SVG Native and their support in `librsvg':

1. `<text>` and `<font>`: Banned in OT-SVG. Banned in SVG Native. `librsvg` supports `<text>` with `css` styling but doesn't support `<font>` tag!
2. `<foreignObject>`: Banned in OT-SVG. Banned in SVG Native. Not supported in `librsvg`.
3. `<switch>`: Banned in OT-SVG. Banned in SVG Native. Supported by `librsvg`.
4. `<a>`: Banned in OT-SVG. Banned in SVG Native. `librsvg` supports the node but of course it carries no significance in `PNG` files.
5. `<view>`: Banned in OT-SVG. Banned in SVG Native. Not supported by `librsvg`.
6. `XSL processing` is banned in OT-SVG. Banned in SVG Native. Not supported by `librsvg` as far as I can see!
7. `relative units like em`: OT-SVG bans them. SVG Native bans them too. `librsvg` does support these.
8. `SVG data within <image>`: OT-SVG says that use of `svg` images inside an `<image>` tag is banned. I assume the other types of image embeddings are legal then. SVG Native however sets the strict restriction that only `base64` encoded url based images are supported, that too of only `JPEG`, `PNG` and `APNG` frozen in time. `Librsvg` supports everything in its `<image>` tags. It can render external urls, url based images of `svg` (which OT-SVG bans), `png` and `jpeg` about which SVG Native and OT-SVG are fine!
9. `Color Profiles`: OT-SVG bans them. SVG Native bans `color profiles`. `librsvg` doesn't support these either.
10. `Content Style Type`: OT-SVG bans it. SVG Native bans it since it bans `<style>` attribute all together, so this attribute no longer makes any sense. `librsvg` doesn't support it I guess. Only mentions it at a single place in the source code that too in a comment saying `FIXME`. So I guess not much support.
11. `Use of CSS2 system color keyword`: It's a depreciated feature. Not really sure for now about it at the moment.

The problematic parts!

The following list lists only the parts which are not supposed to be rendered in `OT-SVG` fonts but `Librsvg` will render those properly! These will be the problematic parts if we choose to use `Librsvg` for rendering.

1. `<text>`
2. `<switch>`
3. `<a>` Librsvg renders the text inside it for example. No linking occurs though in PNGs.
4. `relative units`. Librsvg supports these! Problematic for us!

5. `svg data in <image>`. Librsvg supports these! Problematic for us!

Some Interesting Points

I think that SVG Native is a far stricter spec then OT-SVG currently is. It bans many things that OT-SVG allows. An easy example is `<style>` attribute. OT-SVG optionally supports it (not required) while SVG Native strictly bans it. There are others like this too! Filters would be another example. An interesting task would be to figure out if there is anything that `SVG Native specs` ban while OT-SVG require! Since OT-SVG just derives from SVG 1.1 and only specifies what's banned and what's optional one would have to see `SVG Native specs` and for each item that is banned see if OT-SVG bans it or not. Looking forward to doing this soon!

Availability

Very good in this regard. Can be installed easily in:

1. Ubuntu
2. Debain
3. Fedora
4. Arch Linux
5. CentOS
6. FreeBSD
7. Slackware

Dependencies

Well, I haven't found a way to print the whole tree so far! But it looks like there are heavy dependencies. `Cairo` has its own list. But `Libxml`, `libexpat` are some notable ones I saw. The library itself is light. But the dependency list looks long. A better comparison would be to compare the dependencies with say `resvg` or `svgnative`. Dependencies are tolerable if the library is easily distributed which is the case here! Installing it on my Ubuntu worked in seconds.

Maturity

It's associated with the GNOME project. What can I say more? It's quite old so I expect it to be very mature.

A big plus point!

It allows us to render a specific element that has a particular `id`. This is a huge plus! It's something very important that Behdad recently pointed out in the list.

Conclusion

Good in:

1. Distribution. Very easy to install on major systems.
2. Feature support is good.
3. Allows rendering specific element that has a particular ID.

Bad in:

1. Feature set is not restricted to OT-SVG or SVG Native. There are features which this library will just render which are actually supposed to be NOT rendered. That's a bad one! There doesn't seem to be any easy way to restrict it.