

# Vom Rechteck zur Fabrikvisualisierung - Prozessvisualisierung in APEX mit HTML5 und SVG

**Frank Schubert**  
**AIS Automation Dresden GmbH**  
**Dresden**

## Schlüsselworte

HTML5, SVG, JavaScript, CSS, Fabrikvisualisierung, Anlagenvisualisierung

## Einleitung

Für die Entwicklung von datenbasierten Webanwendungen stellt APEX ein ausgereiftes Entwicklungswerkzeug dar. Es bietet umfassende Möglichkeiten zur Präsentation von Daten in tabellarischer und grafischer Form. Mit Hilfe von HTML-Tags und CSS lassen sich vorgegebene Layouts einfach anpassen. HTML5 mit seinen erweiterten Sprachelementen bietet ganz neue Möglichkeiten innovative und interaktive Webseiten zu gestalten.

## Die Herausforderung

Im Rahmen eines Projekts bestand die Aufgabe, die bestehende Applikation FabEagle® TFM auf eine Mobile Plattform zu portieren. Augenmerk sollte dabei auf Plattformunabhängigkeit und ein flexibles Layout zur Umsetzung eines „Responsive Webdesign“ gelegt werden. Die bestehende Visualisierung, welche einen schnellen Überblick über den Fertigungszustand bietet, sollte erhalten bleiben. Da die vorhandene Applikation auf einer Oracle Datenbank basiert, stand der Verwendung von APEX als Framework nichts im Wege. Aber wie können die Fabrik- und Prozessvisualisierungen migriert werden?

## SVG und seine Möglichkeiten

Mit der Einführung von HTML5 rendern die modernen Browser SVG-Grafik (Scalable Vector Graphics) ohne Plugin in HTML-Seiten. SVG stellt Sprachelemente für Rechtecke, Kreise, Linien, Polygone, Elemente für Pfade, Text und Animationen zur Verfügung. SVG basiert auf XML, lässt sich mit CSS formatieren und mit Javascript dynamisch ändern. Eine vollständige Abhandlung aller Sprachelemente würde den Rahmen eines Konferenz-Manuskriptes sprengen. Anbei nur ein kleines Beispiel:

```
<svg width="640" height="480">
  <g stroke="#000000" stroke-width="2">
    <rect height="50" width="100" y="50" x="50" fill="#7fff00"/>
    <circle r="30" cy="100" cx="150" fill="#aad4ff"/>
  </g>
</svg>
```

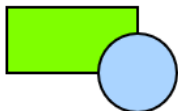


Abb. 1 Rechteck und Kreis - realisiert in SVG

Auf Grund der einfachen Umsetzbarkeit und der in der Datenbank bereits vorhandenen Layout- und Equipment-Koordinaten, lag es nahe, SVG selbst zu generieren. Beispielhaft werden hier 2 Objekte, eine Produktionslinie (line 1) und ein Equipment (equipment 1), in einem Fertigungsbereich (shop-floor) platziert.

Die folgenden Bilder zeigt das zugrundeliegende Koordinatensystem, die beispielhafte Umsetzung in SVG und die resultierende Darstellung.

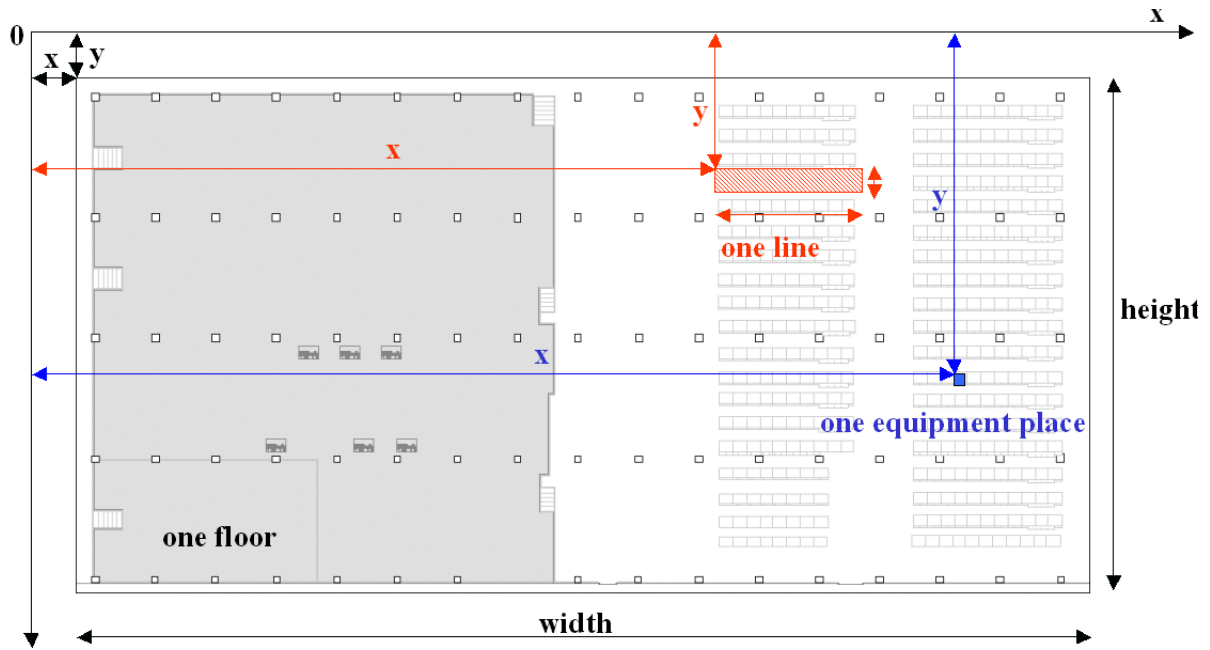


Abb. 2: Fab-Koordinatensystem

```
<svg width="1000" height="400">
  <g stroke="#000000" stroke-width="1">
    <!--shop-floor -->
    <rect x="0" y="0" width="1000" height="400" fill="#ffffff" />
    <!-- line 1 -->
    <rect x="600" y="100" width="100" height="10" fill="orange" />
    <!-- equipment 2 -->
    <rect x="800" y="230" width="20" height="10" fill="blue" />
  </g>
</svg>
```

Abb. 3: Implementierung mit SVG anhand der existierenden Datendatenpunkte



Abb. 4: Resultierende Darstellung

### Schöner geht immer ...

Die grundsätzlichen Anforderungen an die Fabrikvisualisierung sind nun erfüllt. Anlagen können positionsgenau im Fertigungsbereich platziert werden. Ein Farbschema dient zur Visualisierung der Anlagenzustände. Werden nun aber mehrere Anlagen dargestellt, wird es schnell unübersichtlich. Zusätzliche räumliche Merkmale fehlen. Natürlich könnte man die Gruppe „shop-floor“ erweitern und zusätzliche Information einzeichnen. Wir sind aber einen anderen Weg gegangen. Zu jeder Fertigung existieren CAD Zeichnungen, welche als SVG exportiert werden können. Werden diese SVG-Dateien unter den „Shared Components“ der Applikation abgespeichert, können sie über ein Image-Tag eingebunden werden.

```
<svg width="1010" height="710">
  <!--shop-floor -->
  <image x="0" y="0" height="710" width="1040"
        xlink:href="#APP_IMAGES#FAB_withoutTools.svg" />

  <!-- equipments -->
  <g stroke="#000000" stroke-width="1">
    <!-- equipment 1 -->
    <rect x="555" y="127" width="100" height="10" fill="orange" />
    <!-- equipment 2 -->
    <rect x="800" y="230" width="10" height="20" fill="blue" />
  </g>
</svg>
```

Abb. 5: Integration von externen SVG-Dateien über ein Image Tag

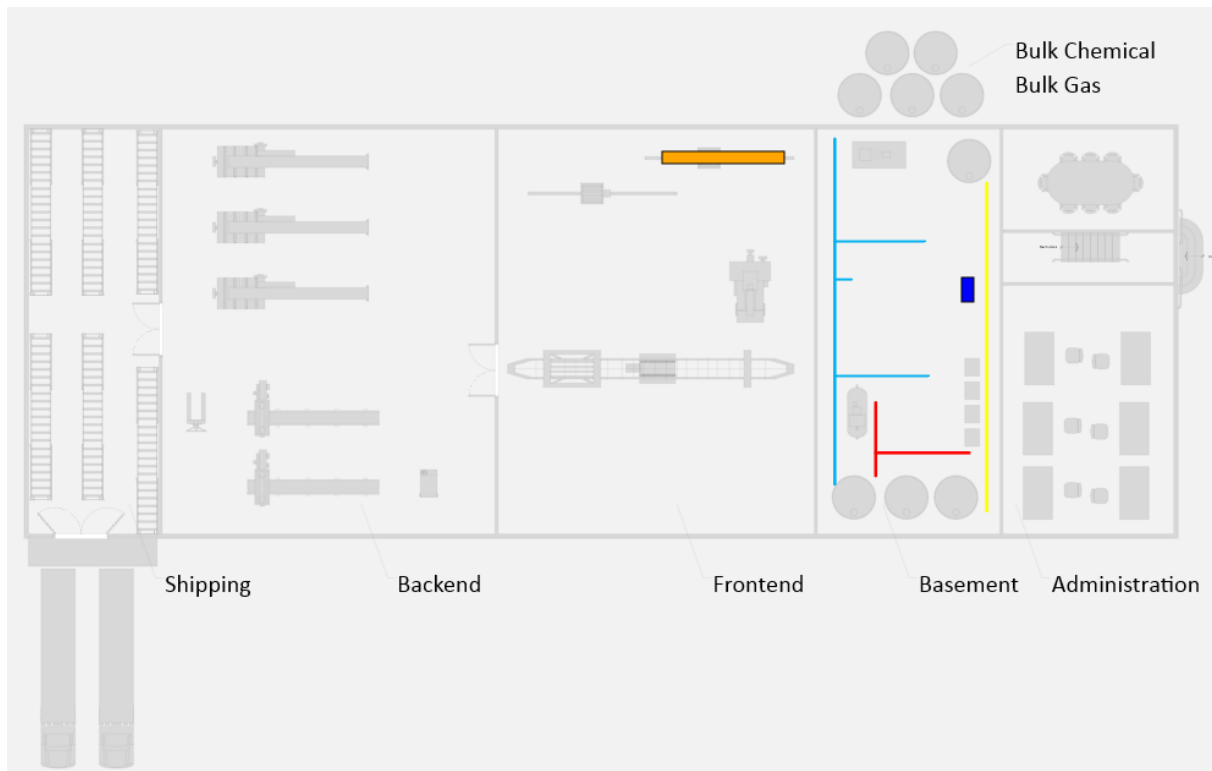


Abb. 6: Resultierende Darstellung – externes SVG als Fabriklayout und zwei Equipments

Somit haben wir mit wenigen Zeilen Code eine sehr ansprechende Visualisierung implementiert. Wenn die Equipments noch zu „rechteckig“ sind, kann auch deren Darstellung durch SVG-Dateien ersetzt werden.

### Benutzerinteraktionen

Von innovativen Benutzeroberflächen erwarten Anwender heutzutage dennoch mehr Funktionalität als nur eine statische Grafik. Tooltips, Links und Zoom gehören zu modernen Anwendung einfach mit dazu. Die Kombination von Javascript und SVG machen auch das möglich. Den SVG-Elementen können auch Maus-Events (onmouseout, onmousemove, onclick) hinzugefügt werden. So lassen sich mit Javascript-Funktionen Tooltips ein- bzw. ausblenden und Links öffnen.

```
<svg id="svg2" width="1010" height="710">
  <!-- Fab Floor -->
  <image x="0" y="0" height="710" width="1040"
        xlink:href="#APP_IMAGES#FAB_withoutTools.svg" />

  <!-- Equipments -->
  <g stroke="#000000" stroke-width="1">
    <!-- equipment 1 -->
    <rect x="555" y="127" width="100" height="10" fill="orange"
          onmouseout="tooltip_hide('tt_eq_1');"
          onmousemove="tooltip_show('tt_eq_1','svg2',evt);"
          onclick="window.location='f?p=400:100::::: ...' />
    <!-- equipment 2 -->
    <rect x="800" y="230" width="10" height="20" fill="blue"
          onmouseout="tooltip_hide('tt_eq_2');"
```

```

onmousemove="tooltip_show('tt_eq_2','svg2',evt);"
onclick="window.location='f?p=400:100::::: ...'/">

<!-- Tooltips -->
<div id="tt_eq_1" visibility="hidden" class="xstooltip">
  Name: FEWS01<br>
  Typename: Wire Saw DS271<hr>
  Comment: --
</div>

<div id="tt_eq_2" style="visibility: hidden;" class="xstooltip">
  Name: FEWS02<br>
  Typename: Wire Saw DS271<hr>
  Comment: --
</div>
</svg>

```

Abb. 7: Implementierung Maus-Events

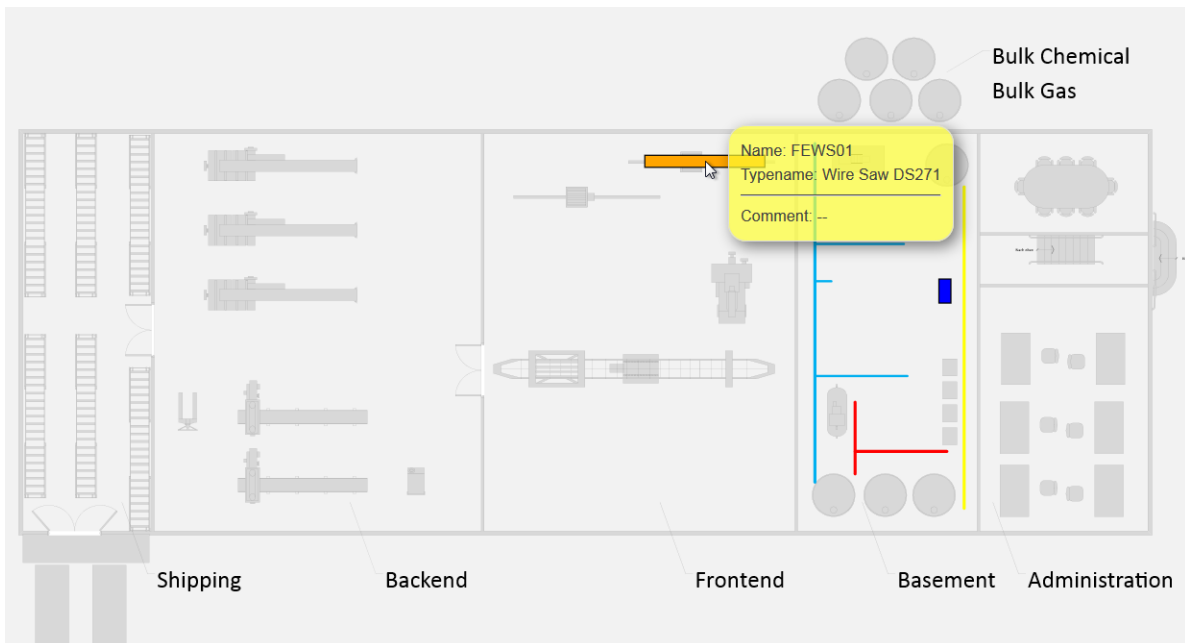


Abb. 8: Resultierende Darstellung – Equipment mit Tooltip

### Prozessvisualisierung

Nach dem gleichen Prinzip kann man auch Prozessvisualisierungen bzw. Anlagenfließbilder in eine HTML Seite integrieren. Die Aktualisierung der Prozesswerte und Zustände erfolgt über Javascript. Die Objekte werden über das „id“-Attribut der SVG-Elemente angesprochen und aktualisiert.

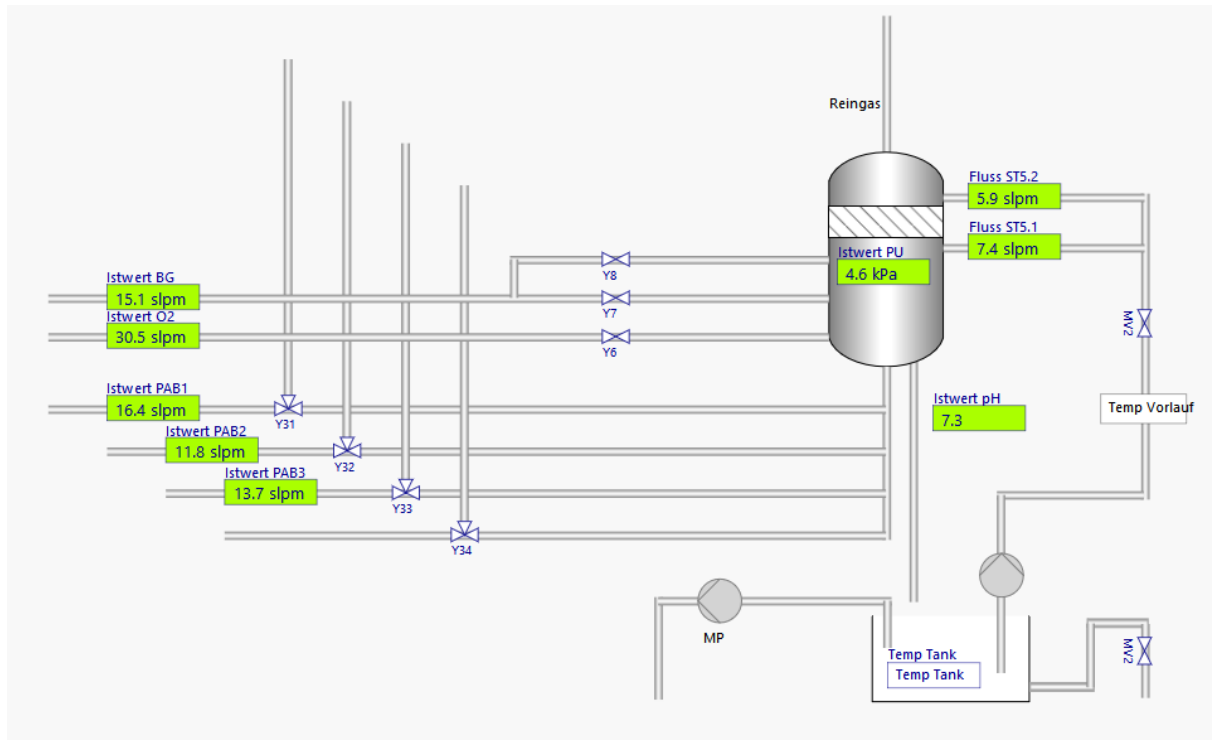


Abb. 8: Prozessvisualisierung mit SVG

### Fazit:

Mit HTML5 und SVG lassen sich komplexe Visualisierungen einfach in APEX integrieren. Die resultierenden Grafiken sind skalierbar und auch auf mobilen Endgeräten gut lesbar. Trotz komplexer Grafiken konnten keine Performanzprobleme bei der Darstellung im Browser festgestellt werden.

### Kontaktadresse:

Frank Schubert  
 AIS Automation Dresden GmbH  
 Otto-Mohr-Straße 6  
 D-01237 Dresden

Telefon: +49 (0) 351-2166 1240  
 Fax: +49 (0) 351-2166 3240  
 E-Mail: <mailto:Frank.Schubert@ais-automation.com>  
 Internet: <http://www.ais-automation.com>

TFM Mobile: <http://tfmportable.ais-automation.com>