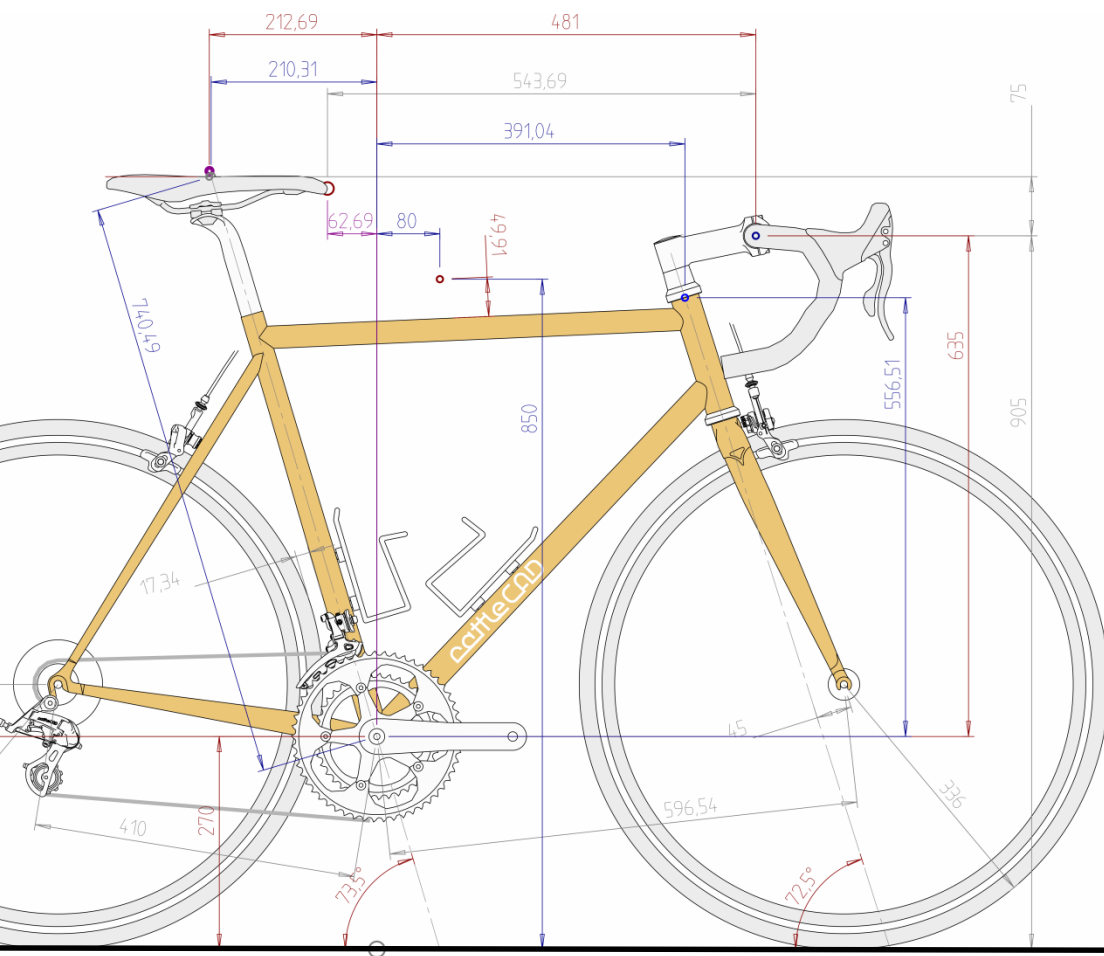




16th European Tcl/Tk User Meeting

July, 7th and 8th 2017, Munich, Germany



cattleCAD

<http://rattlecad.sourceforge.net/>
... design your custom Bike

Manfred Rosenberger

cad4tcl – a CAD package for Tcl

rattleCAD

- The Bicycle Design Process
- <http://rattlecad.sourceforge.net/>

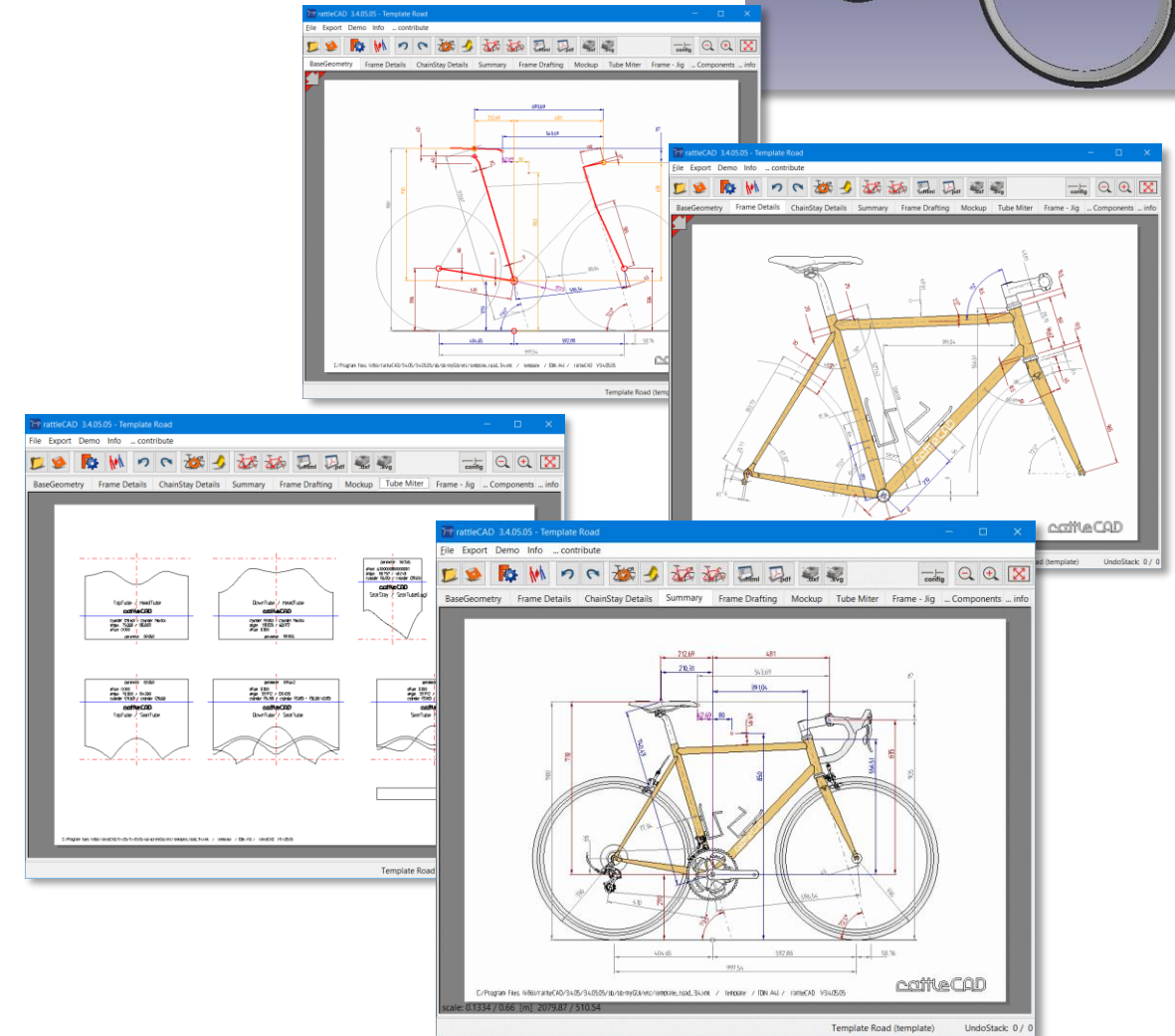
rattleCAD - written in tclTk

- Requirements to a CAD-Software

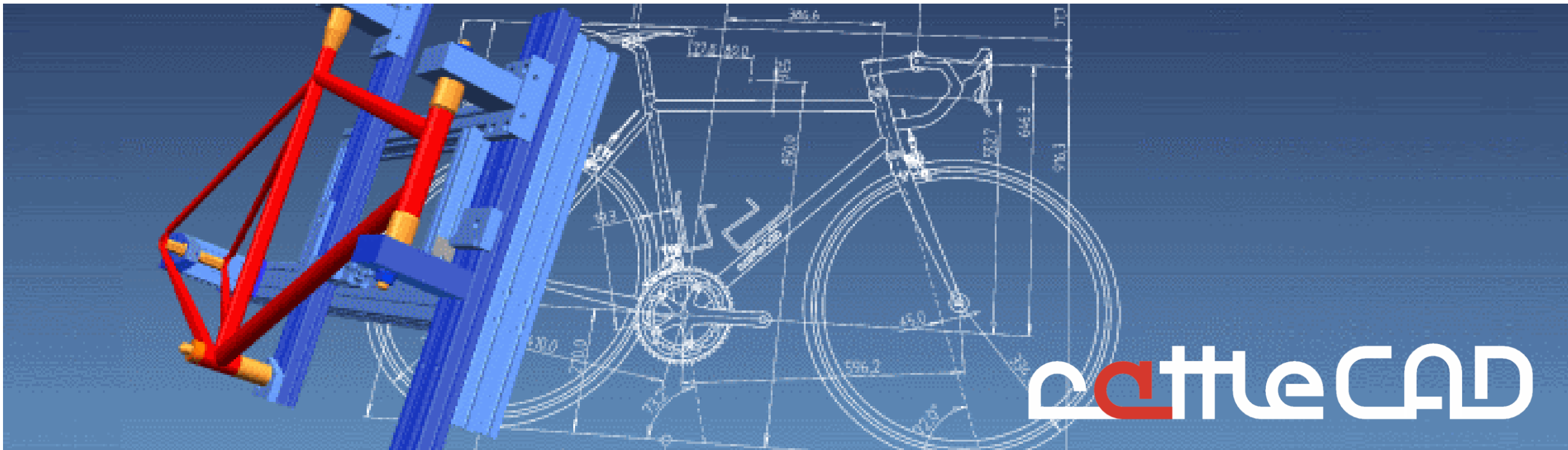
cad4tcl & svgDOM

- The CAD perspective of the canvas
- <https://cad4tcl.sourceforge.io/>
- <https://svgdom.sourceforge.io/>

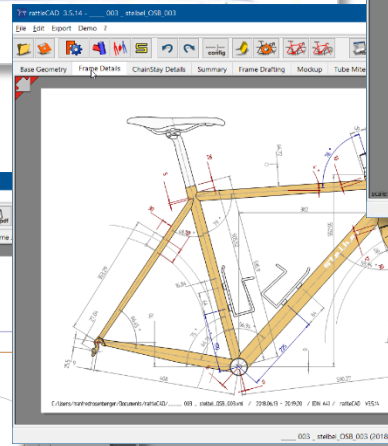
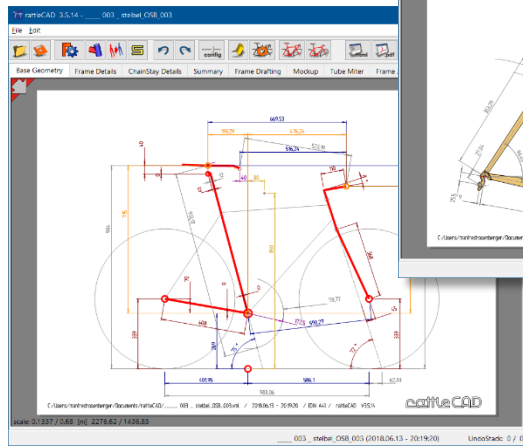
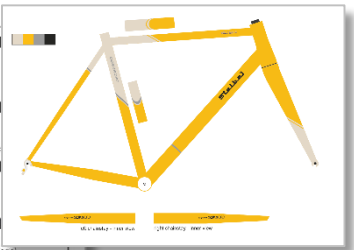
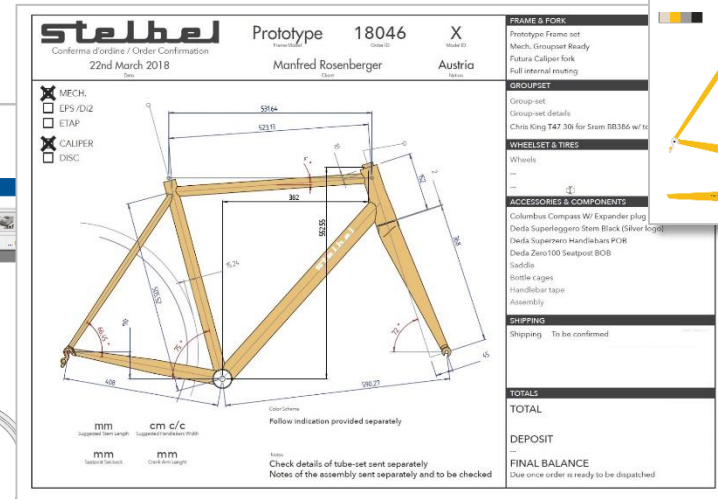
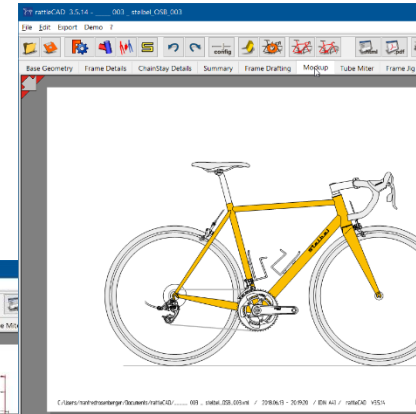
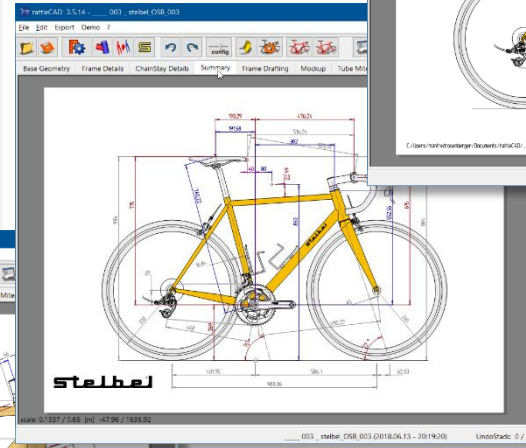
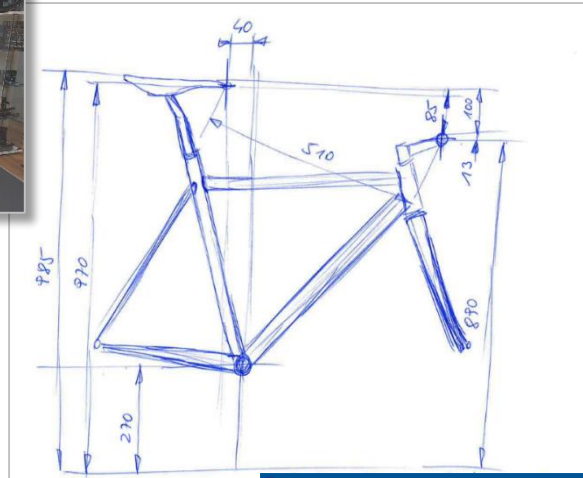
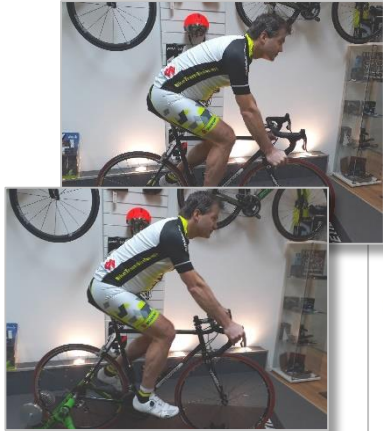
Let's start ...



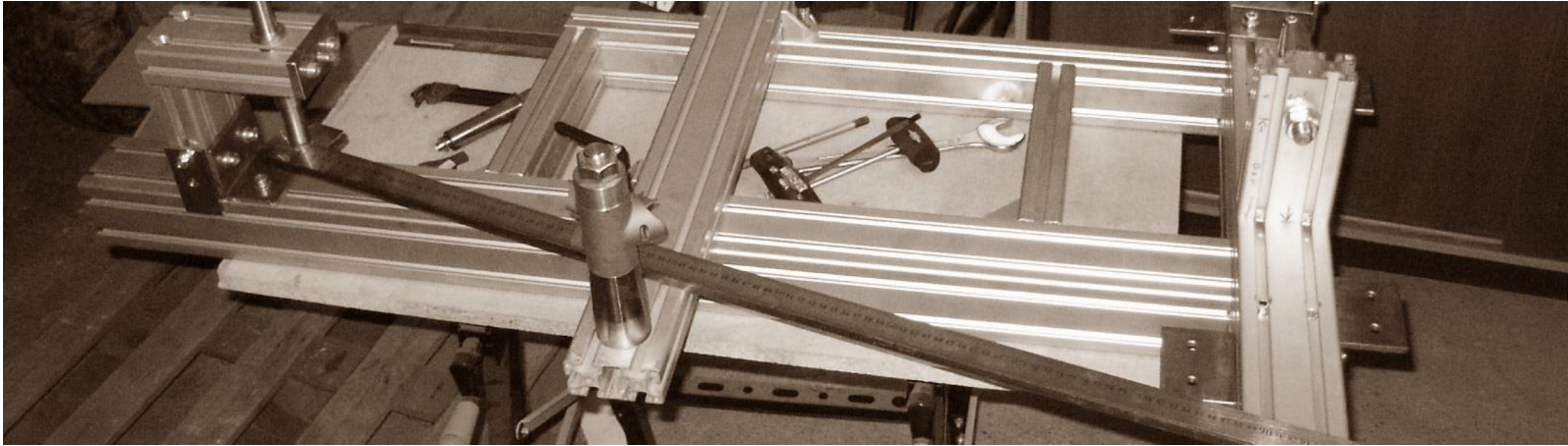
... may I introduce

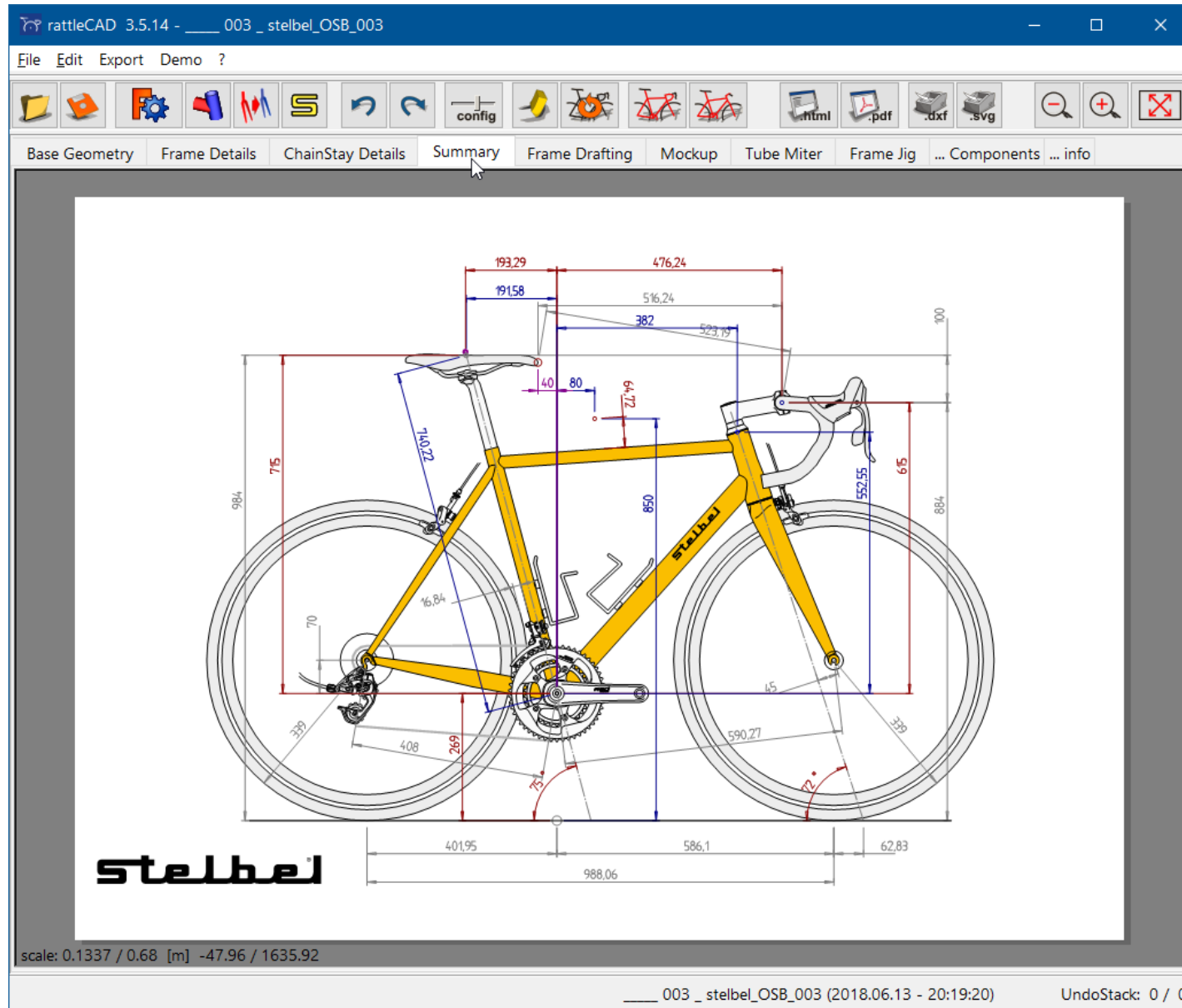


rattleCAD in the Bicycle Design-Process



rattleCAD – Requirements to a CAD-Software

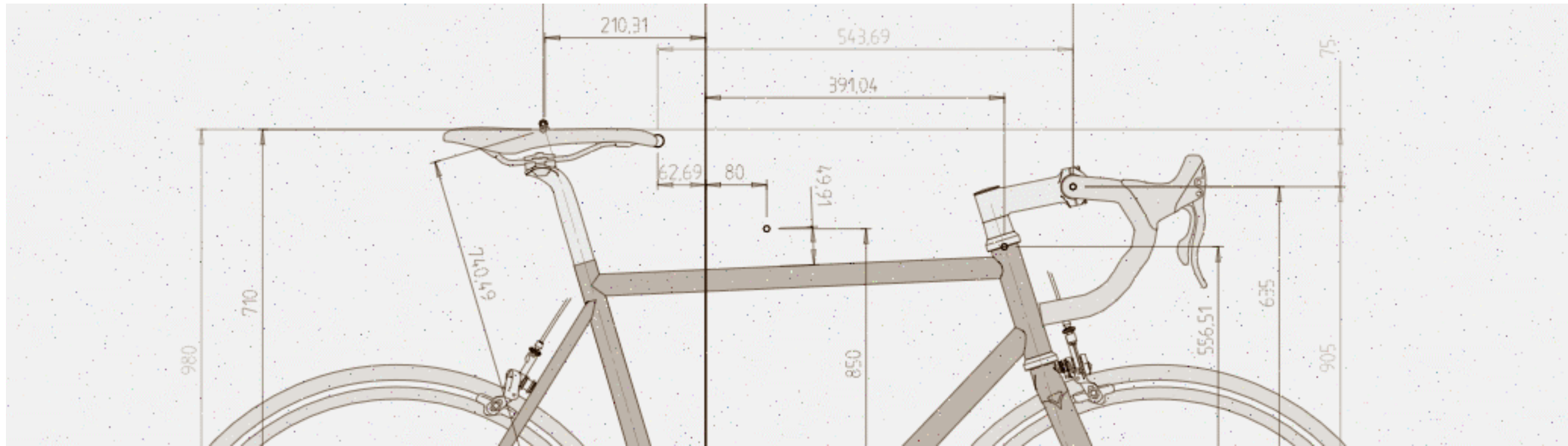




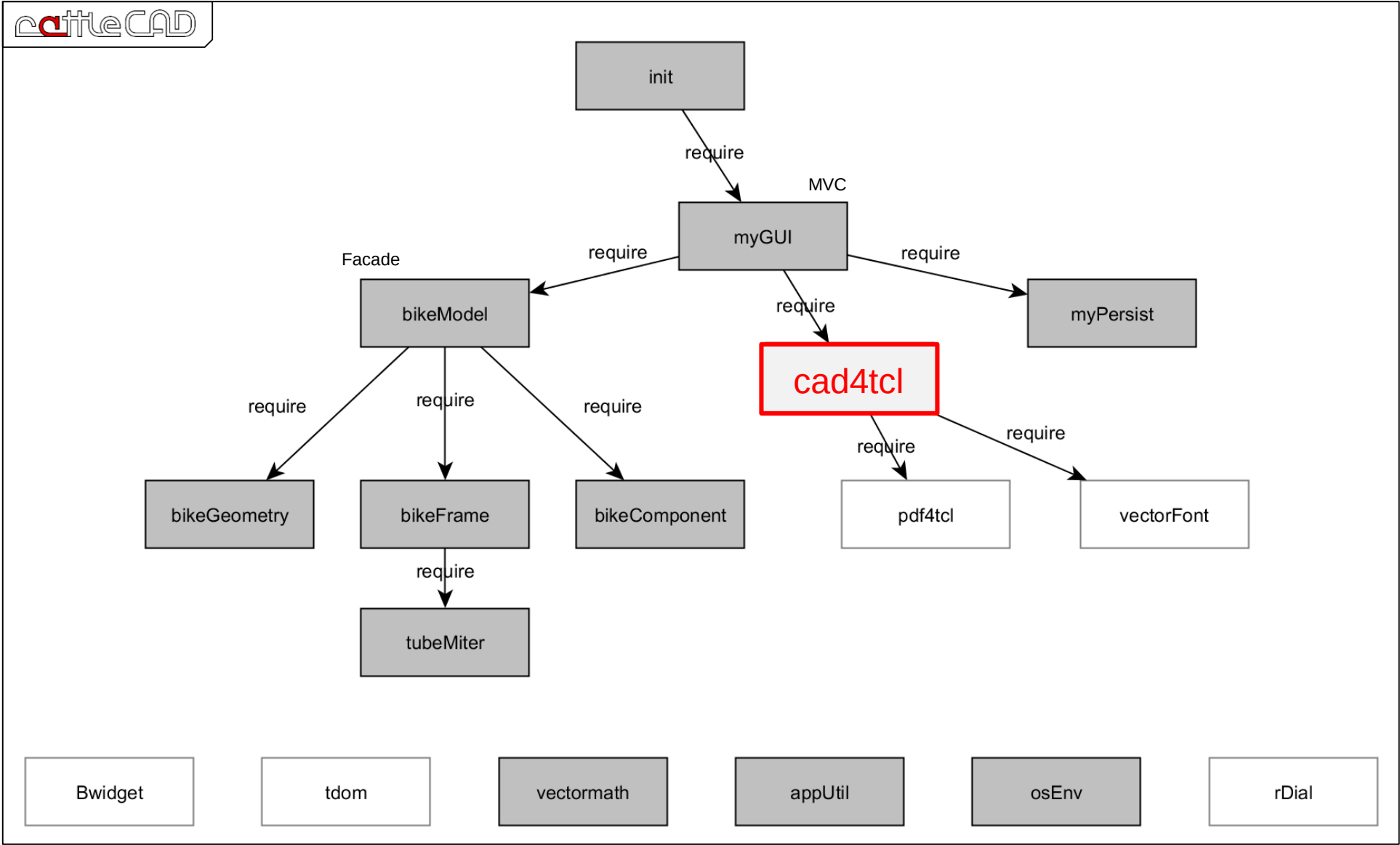
CAD-Features:

- zoom
- pan
- events
- components library
 - SVG
- export:
 - SVG (HTML)
 - PDF
 - (DXF)
- Plugin-Development

... cad4tcl - a CAD library extracted from rattleCAD ...

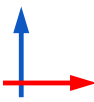


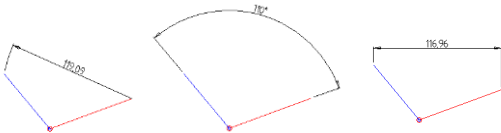
System Analysis: Packages (target Situation)





cad4tcl

- Origin: bottom-left 
- Default scale (eg. 1:5)
- pan & zoom
- Dimensioning



- File formats
 - SVG (import & export)
 - DXF (export)
 - PDF (export)

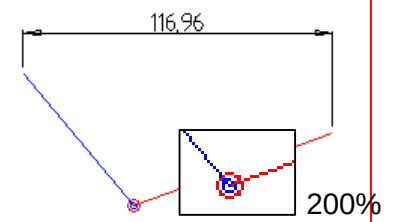
Tk

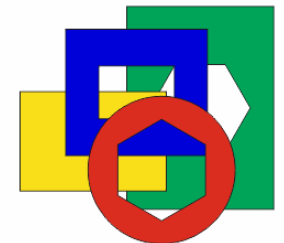
tk::canvas

- Origin: top-left 
- File formats
 - PS (export)

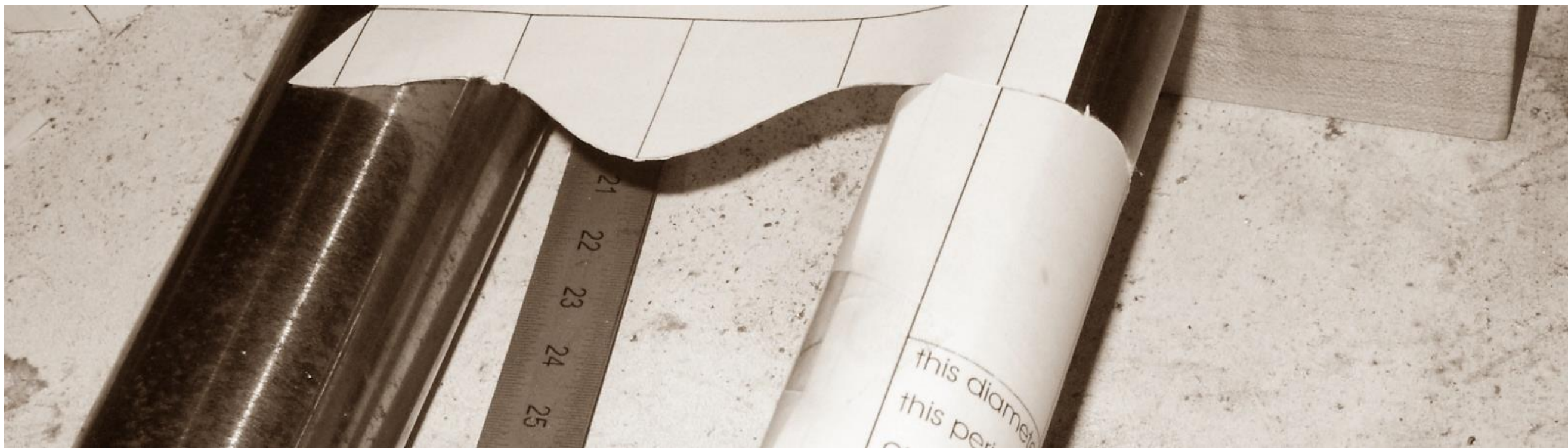
(Mats Bengtsson, 1959-2008)

tkpath

- Origin: top-left 
- Antialiasing 
- fully SVG-compatibility
 - e.g.: path-Element

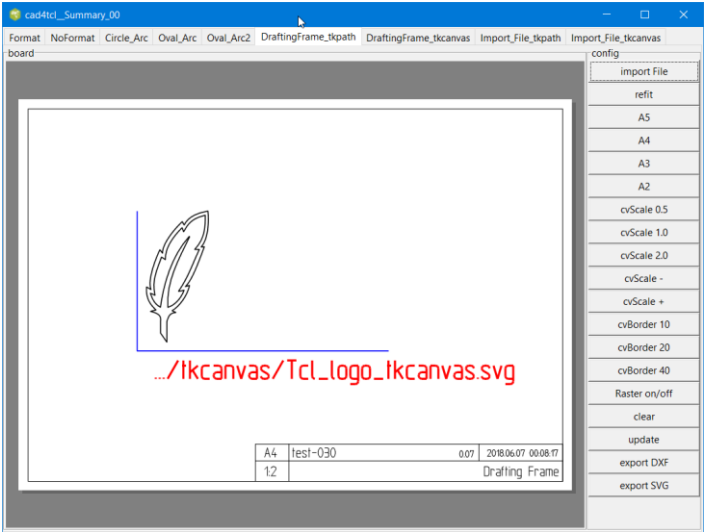


cad4tcl - live Demo (package) ...

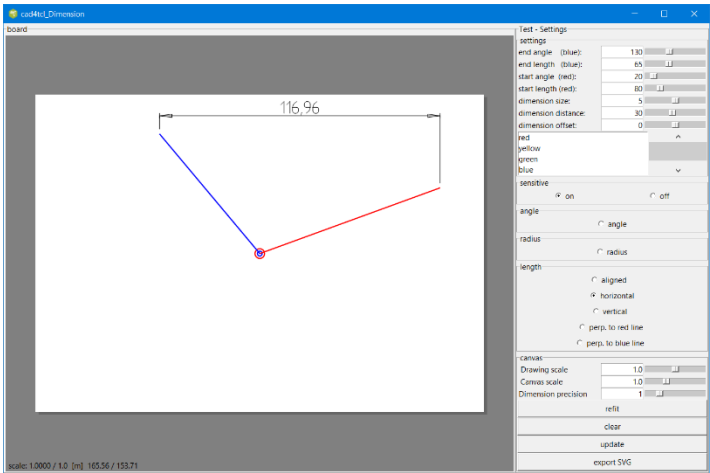


cad4tcl - Performance Show

<https://cad4tcl.sourceforge.io/examples.html>



.../cad4tcl/test/cad4tcl__Summary_00.tcl



.../cad4tcl/test/cad4tcl_Dimension.tcl

```
#####
#
# cad4tcl__test_00.tcl
# by Manfred ROSENBERGER
#
# (c) Manfred ROSENBERGER 2018/07/04
#
#
set TEST_ROOT_Dir [file normalize [file dirname [lindex $argv 0]]]
set TEST_Library_Dir [file dirname [file dirname $TEST_ROOT_Dir]]
set TEST_Sample_Dir [file join $TEST_ROOT_Dir _sample]
set TEST_Export_Dir [file join $TEST_ROOT_Dir _export]

puts " -> \${TEST_ROOT_Dir} ..... \${TEST_ROOT_Dir}"
puts " -> \${TEST_Library_Dir} ... \${TEST_Library_Dir}"
puts " -> \${TEST_Sample_Dir} ... \${TEST_Sample_Dir}"
puts " -> \${TEST_Export_Dir} ... \${TEST_Export_Dir}"

lappend auto_path [file dirname $TEST_ROOT_Dir]
lappend auto_path "$TEST_Library_Dir"
lappend auto_path [file join $TEST_Library_Dir _ext_Libraries]

foreach dir $tcl_library {
    puts " -> tcl_library $dir"
}

foreach dir $auto_path {
    puts " -> auto_path $dir"
}

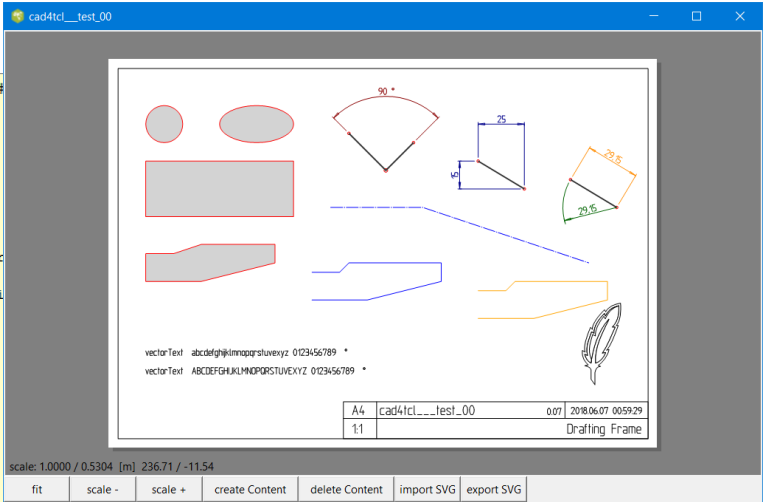
package require Tk
package require cad4tcl 0.07

#
# cad4tcl::canvasType
# ... 1 use tkpath if available
# ... 0 use tk::canvas, dont search for tkpath
set cad4tcl::canvasType 1

set cvObject [cad4tcl::new . 800 600 A4 1.0 30]
set cv_scale [$cvObject configure Canvas Scale]

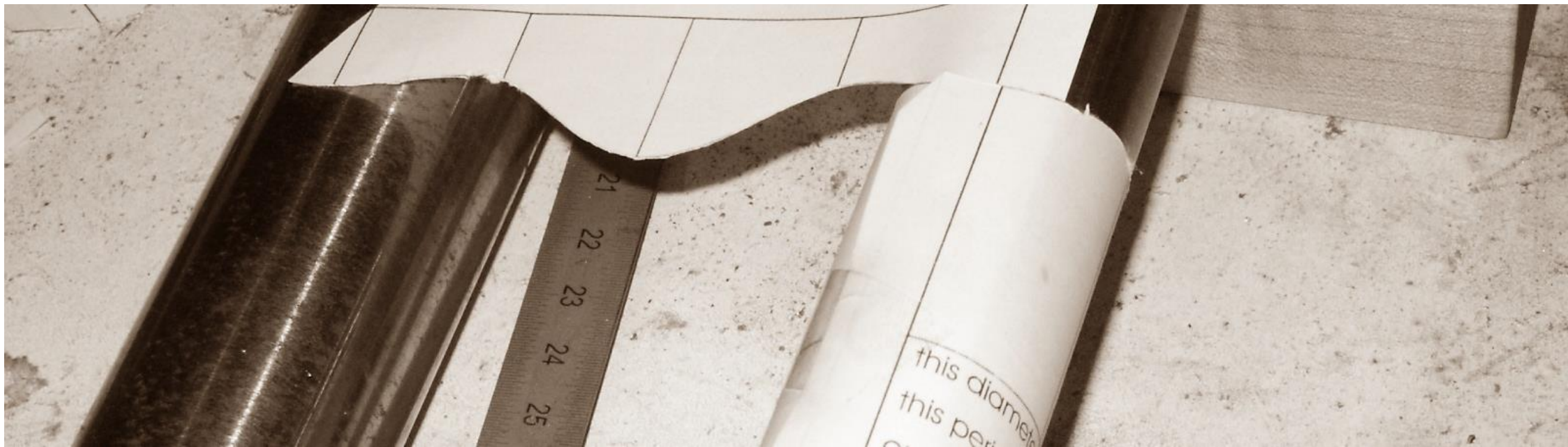
puts " -> \${cvObject} $cvObject"
puts " -> \${cv_scale} $cv_scale"
#
```

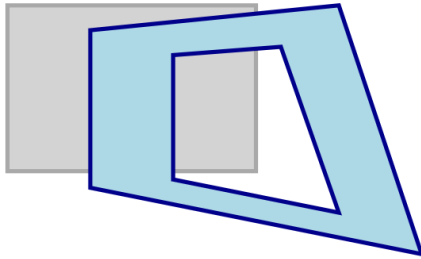
<https://cad4tcl.sourceforge.io/examples.html>



.../test/cad4tcl__test_00.tcl

svgDOM – simplify SVG

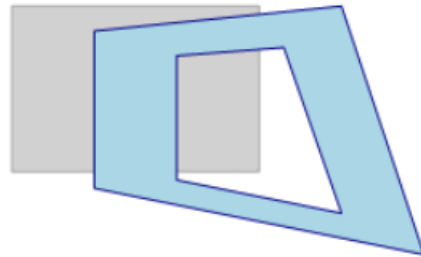




```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<svg id="svg2" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 100 100" version="1.1" >
  <rect id="rect001" fill="lightgray" stroke="darkgray" stroke-width="0.5"
    x="10" y="15" width="30" height="20"/>
  <path id="path002" fill="lightblue" stroke="darkblue" stroke-width="0.5"
    fill-rule="evenodd"
    d="M 20 18 L 50 15 60 45 20 37 Z
      M 30 21 L 43 20 50 40 30 36 Z"/>
</svg>
```

\$svgObj pathType **path**

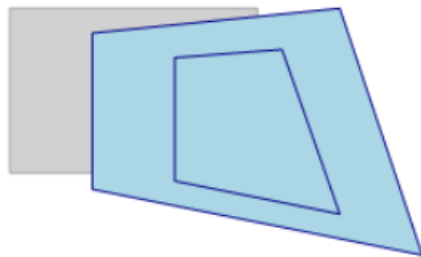
```
<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg" id="svg2" viewBox="0 0 100 100" version="1.1">
  <polygon id="rect001" fill="lightgray" stroke="darkgray" stroke-width="0.5"
    points="10.0,15.0 40.0,15.0 40.0,35.0 10.0,35.0"/>
  <path id="path002" fill="lightblue" stroke="darkblue" stroke-width="0.5"
    d="M 20.0,18.0 L 50.0,15.0 60.0,45.0 20.0,37.0 Z
      M 30.0,21.0 L 43.0,20.0 50.0,40.0 30.0,36.0 Z"/>
</svg>
```



\$svgObj pathType **fraction**

... limited svg-support in tk::canvas

```
<svg xmlns="http://www.w3.org/2000/svg" id="svg2" viewBox="0 0 100 100" version="1.1">
  <polygon id="rect001" fill="lightgray" stroke="darkgray" stroke-width="0.5"
    points="10.0,15.0 40.0,15.0 40.0,35.0 10.0,35.0"/>
  <g id="path002">
    <polygon fill="lightblue" stroke="darkblue" stroke-width="0.5"
      points="20.0,18.0 50.0,15.0 60.0,45.0 20.0,37.0" id="path002_0"/>
    <polygon fill="lightblue" stroke="darkblue" stroke-width="0.5"
      points="30.0,21.0 43.0,20.0 50.0,40.0 30.0,36.0" id="path002_1"/>
  </g>
</svg>
```



svgDOM – combine transform attributes

<https://svgdom.sourceforge.io/features.html>

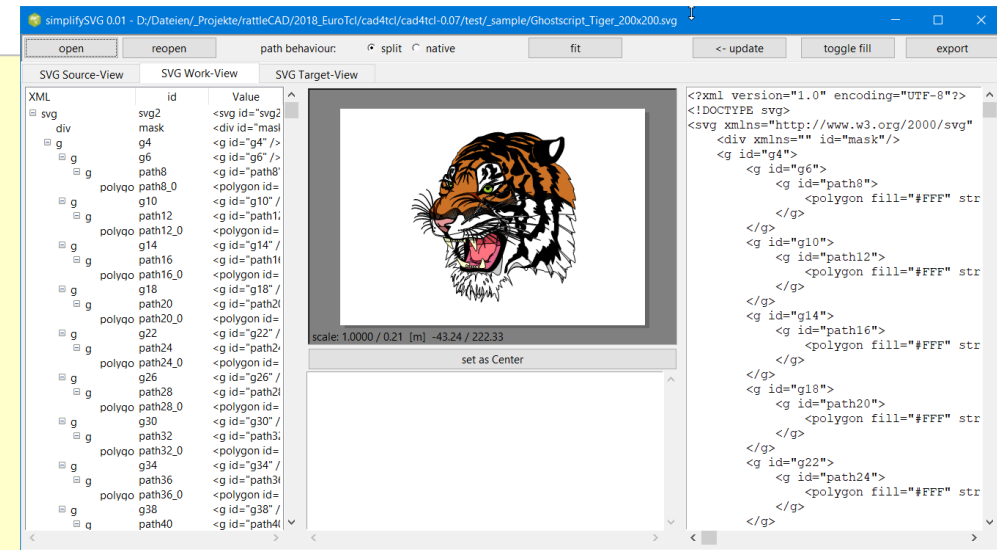
```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<svg id="svg2" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 150 150" version="1.1">
  <g transform="translate(50,20)">
    <g transform="scale(2)">
      <g transform="rotate(45)">
        <g transform="translate(5,10)">
          <rect x="10" y="15" width="30" height="20"
                fill="yellow" stroke="blue" stroke-width="1"/>
        </g>
      </g>
    </g>
  </g>
</svg>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg" id="svg2" viewBox="0 0 150 150" version="1.1">
  <g id="_uid_1">
    <g id="_uid_2">
      <g id="_uid_3">
        <g id="_uid_4">
          <polygon id="_uid_5" fill="yellow" stroke="blue" stroke-width="1"
                    points="
                      35.85786437626905,76.5685424949238
                      78.2842712474619,118.99494936611666
                      50.0,147.27922061357856
                      7.573593128807147,104.8528137423857"/>
        </g>
      </g>
    </g>
  </g>
</svg>
```

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<svg id="svg2" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 150 150" version="1.1">
  <g style="fill: red; stroke: blue; stroke-width: 3">
    <g stroke-width="0.5">
      <g stroke="red">
        <g fill="green" stroke-width="1">
          <rect fill="yellow"
            x="10" y="15" width="30" height="20"/>
        </g>
      </g>
    </g>
  </g>
</svg>
```

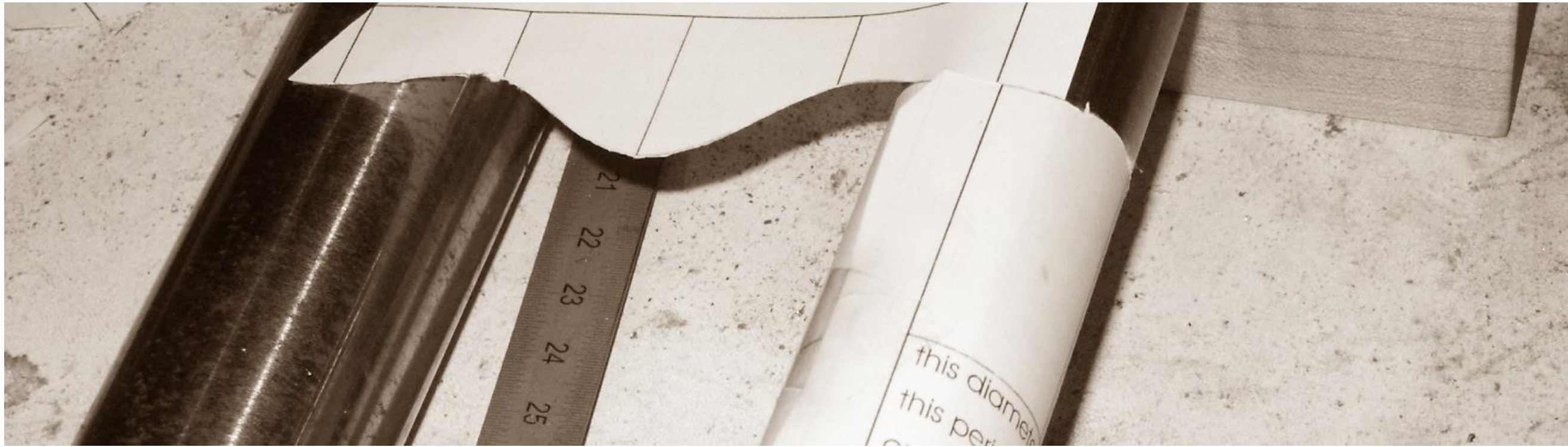
```
<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg" id="svg2" viewBox="0 0 150 150" version="1.1">
  <g id="_uid_1">
    <g id="_uid_2">
      <g id="_uid_3">
        <g id="_uid_4">
          <polygon id="_uid_5"
            fill="yellow"
            stroke="red"
            stroke-width="1"
            points="10.0,15.0 40.0,15.0 40.0,35.0 10.0,35.0"/>
        </g>
      </g>
    </g>
  </g>
</svg>
```

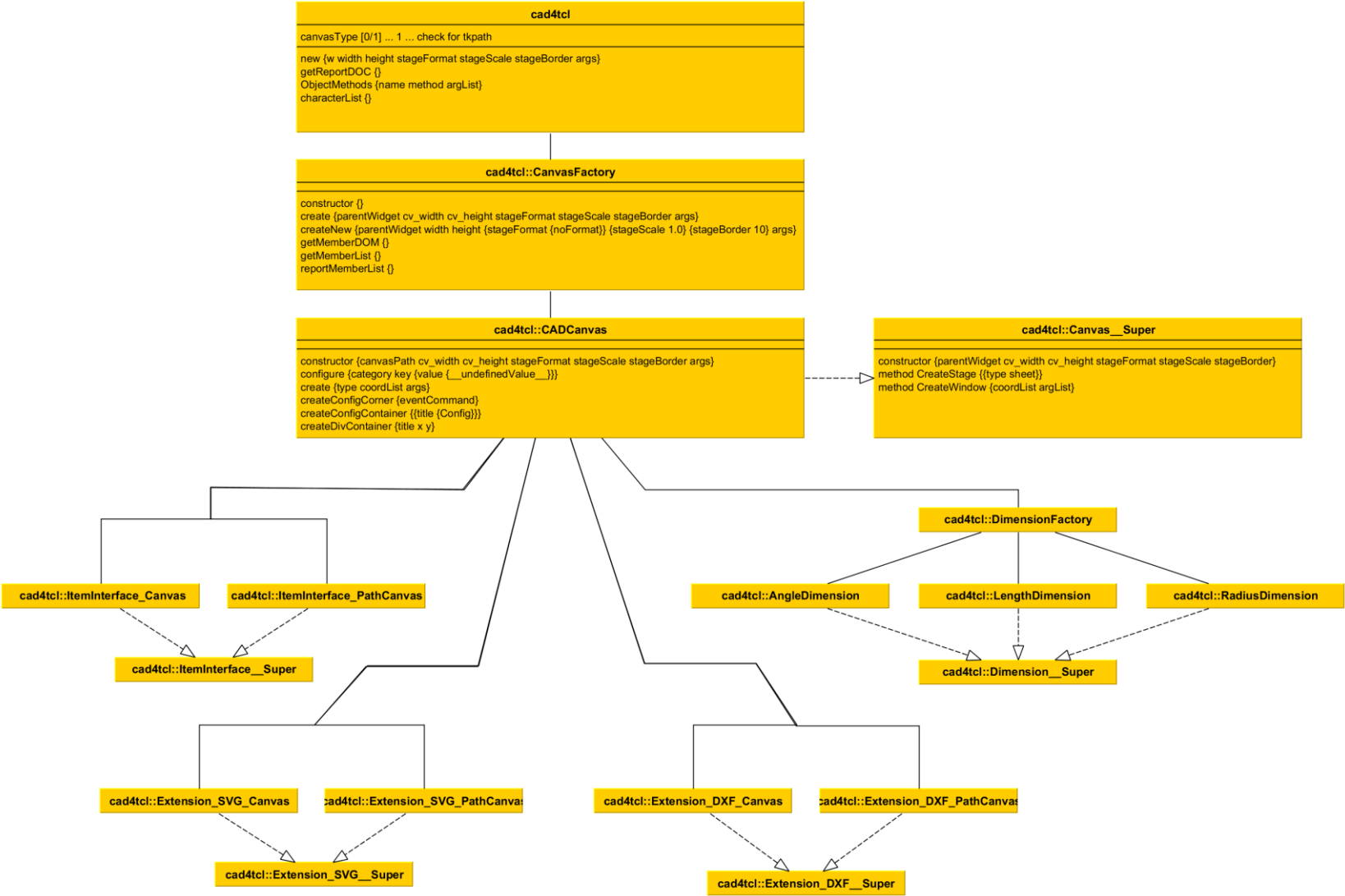
```
#
package require svgDOM 0.04
#
set fileInput .... ; # define your input fileName
set fileOutput ... ; # define your output fileName
#
#
set svgObject [::svgDOM::SVG new file $fileInput]
#
# default pathType is "fraction for use in tk::canvas
# to export optimized for tkpath set pathType to "path"
#
$svgObject pathType path
#
# get the converted svg-content
#
set svgXML [$svgObject getSVG]
#
# prepare output to write it to a svg-file
#
set outputContent {<?xml version="1.0" encoding="UTF-8" standalone="no"?>}
append outputContent "\n" [$svgXML asXML -doctypeDeclaration 1]
#
#
set fp [open $fileName w]
puts $fp $outputContent
close $fp
#
```



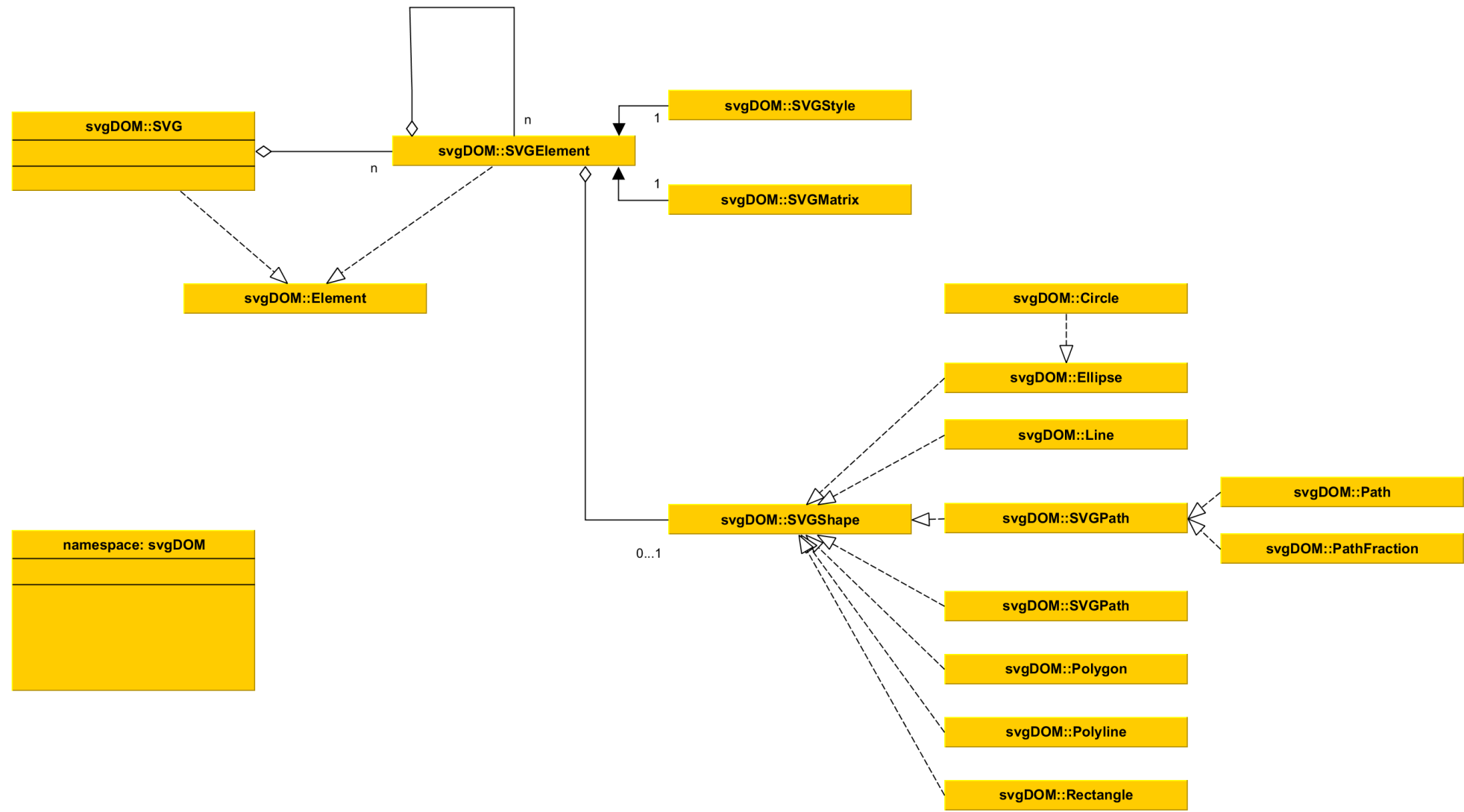
.../cad4tcl/test/app_simplifySVG.tcl

cad4tcl & svgDOM – Class Diagrams





<https://cad4tcl.sourceforge.io/architecture.html>



<https://svgdom.sourceforge.io/architecture.html>

Wish-list to the TCL/Tk-Community



- **canvas**
 - 74. Anti-aliased canvas
 - 20. Rotation of photos
 - 80. Oval rotation
 - 81. Zooming
 - ... including images on the canvas
 - 115. canvas
- **other**
 - 22. Transparent color
 - 58. Background Images And Transparency
 - 97. Combine ttk::treeview with a listvariable
 - 69. Improved bitmap editing

- **installer Framework, to deploy applications**
 - standard user are no computer experts (especially Mac-user)
 - starpack is not enough
 - MacOS requires an installation package
- **Documentation**
 - wiki.tcl.tk ...
 - link to the manuals (<https://www.tcl.tk/man>) on a more prominent position
 - description:
 - textual description is necessary, but often difficult to understand
 - an abstract to explain what a library is for
 - Context, Challenge, Approach and an Example of usage
 - examples:
 - one running script, showing the main purpose of a command or library (like Bwidget)
 - one running script per provided feature
 - what about the test-procedures of the provided libraries ... dont they exist?

Summary



Summary



- a CAD software written in TclTk

cad4tcl - a CAD library extracted from rattleCAD

- Introduction & Examples
- Free & Open & Usefull for others
- <https://cad4tcl.sourceforge.io/>

svgDOM - prepare SVG for use in cad4tcl

- Introduction & Examples
- Free & Open & Usefull for others
- <https://svgdom.sourceforge.io/>

let's talk ...



Manfred Rosenberger
manfred.rosenberger@gmx.net

