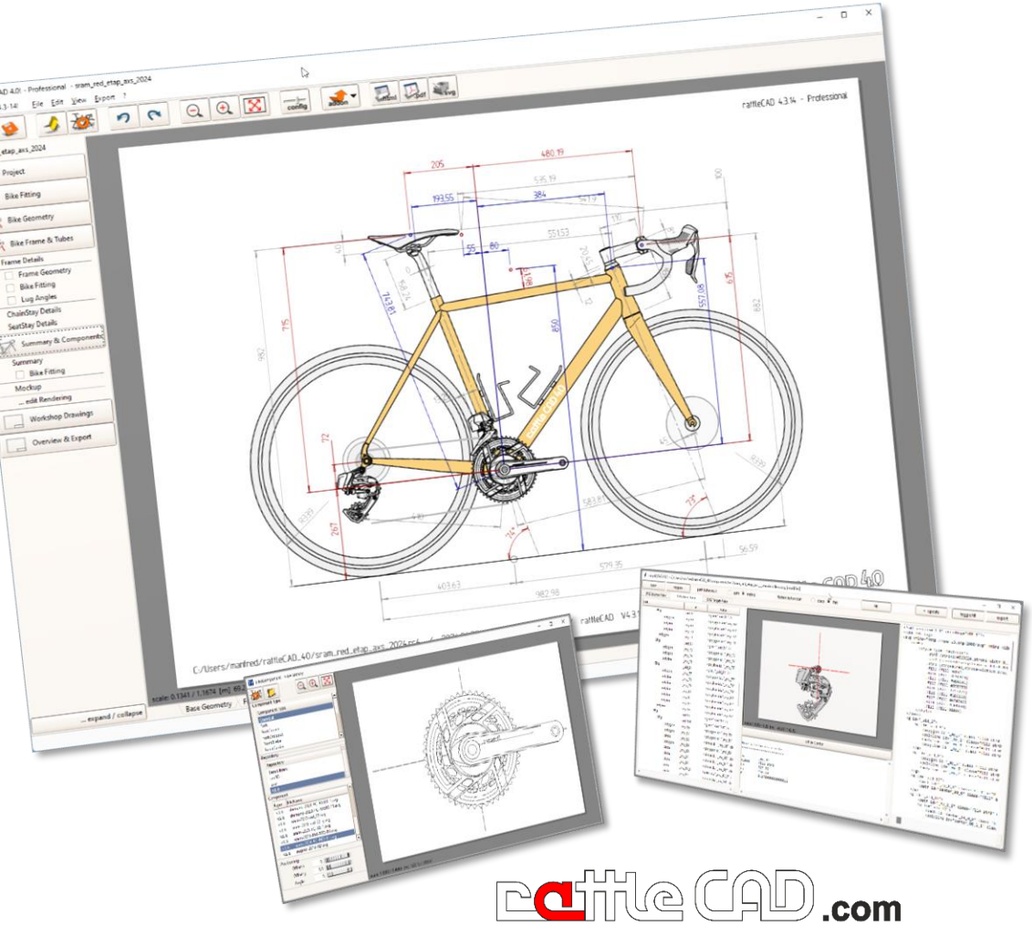




20th European Tcl/Tk Conference

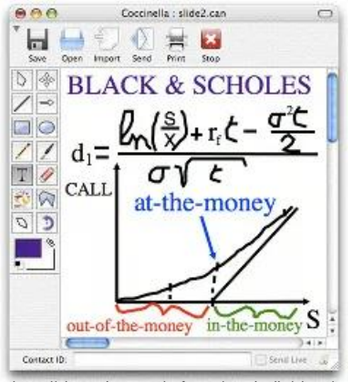
July 11th - 12st 2024, Vienna, Austria

# Evolution of Tk canvas

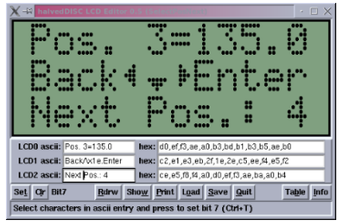
Manfred ROSENBERGER

# Use Cases

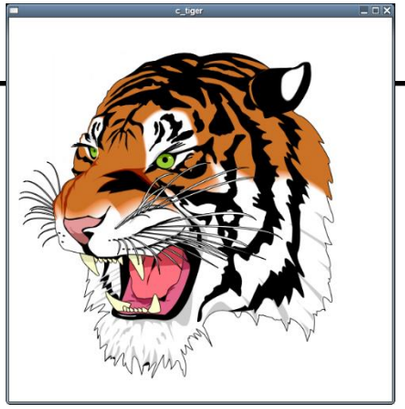
- Diagrams
- Visualize I/O datastreams
- Animations
- Arts / graphics
- Notes
- UI
- CAD
- …



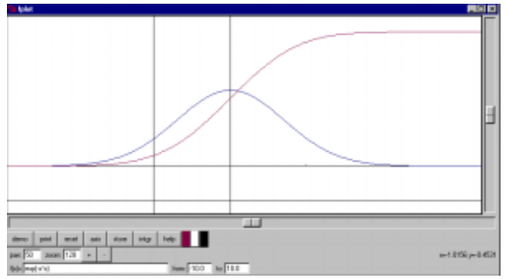

https://alternativeto.net/software/coccinella/about/



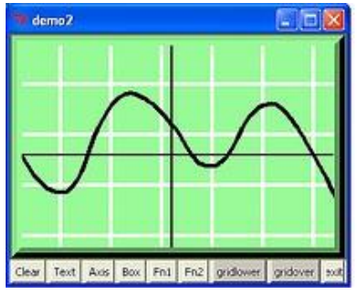

https://www.tech-edv.co.at/



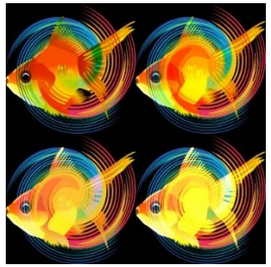

tclbitprint.sourceforge.net/tkpath



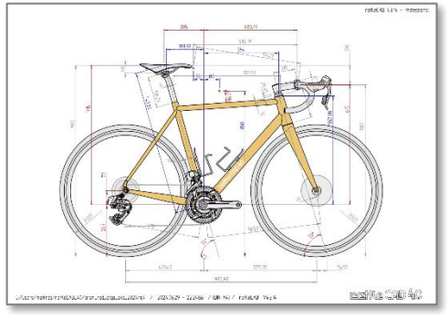
https://www.tech-edv.co.at/

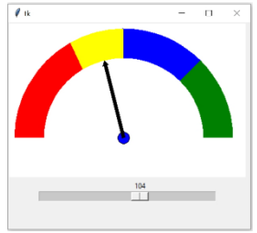



https://wiki.tcl-lang.org/page/Simple+Canvas+Demo

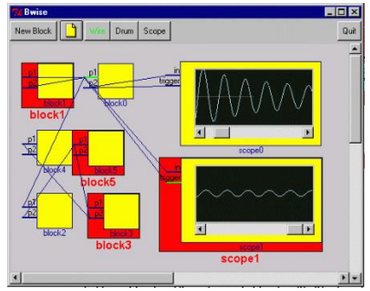


https://blend2d.com/

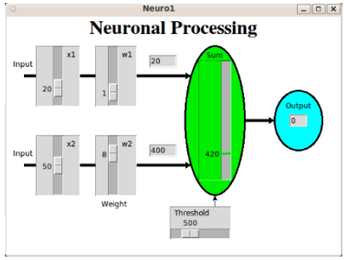




https://www.plus2net.com/python/tkinter-canvas-arc.php




https://theover.tripod.com/tcltk.html



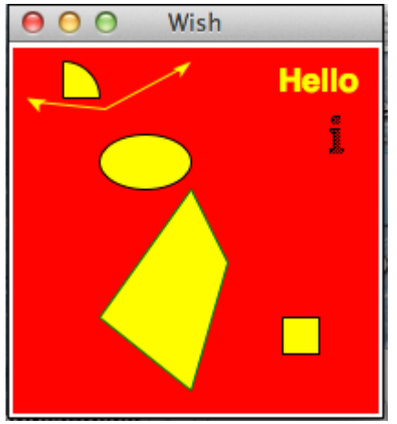

https://wiki.tcl-lang.org/page/Widgets+on+a+canvas

# Tk canvas - Libraries

- canvas
  - Part of Tk
- TkPath
  - https://tclbitprint.sourceforge.net/
- TkZinc
  - https://github.com/asb-capfan/TkZinc
- Blend2D
  - https://blend2d.com/
- …

# Tk canvas

- Part of Tk
- Description
  - The canvas widget is Tk's workhorse for 2D graphical display, and can handle both bitmap and vector graphics. It was inspired by Joel Bartlett's ezd program, which provides structured graphics in a Scheme environment.
- Features
  - A canvas is one of the most powerful concepts in Tk. It acts as a drawing plane for lines, rectangles, ovals, polygons, text, arcs (e.g. pieslices) as well as container widget to group other widgets, and it provides the ability to group elements together for creation, deletion, moving, etc
    - arc
    - bitmap
    - image
    - line
    - oval
    - polygon
    - rectangle
    - text
    - window




https://www.tutorialspoint.com/tcl-tk/tk_canvas_widgets.htm

# TkPath - advanced SVG canvas widget for Tk

- https://tclbitprint.sourceforge.net/
- written by Mats Bengtsson (1959 - 2008)
- Description
  - Add SVG support to Coccinella using Tkpath library
- Features:
  - reproduces all standard drawing canvas items
  - path drawing modelled after SVG
    - group
    - circle
    - ellipse
    - path
    - pimage
    - pline
    - polyline
    - ppolygon
    - prect
    - ptext
  - anti aliasing
- Implementation
  - MacOS: CoreGraphic
  - Windows: GDI, GDI+
  - X11: Cairo

https://tclbitprint.sourceforge.net/

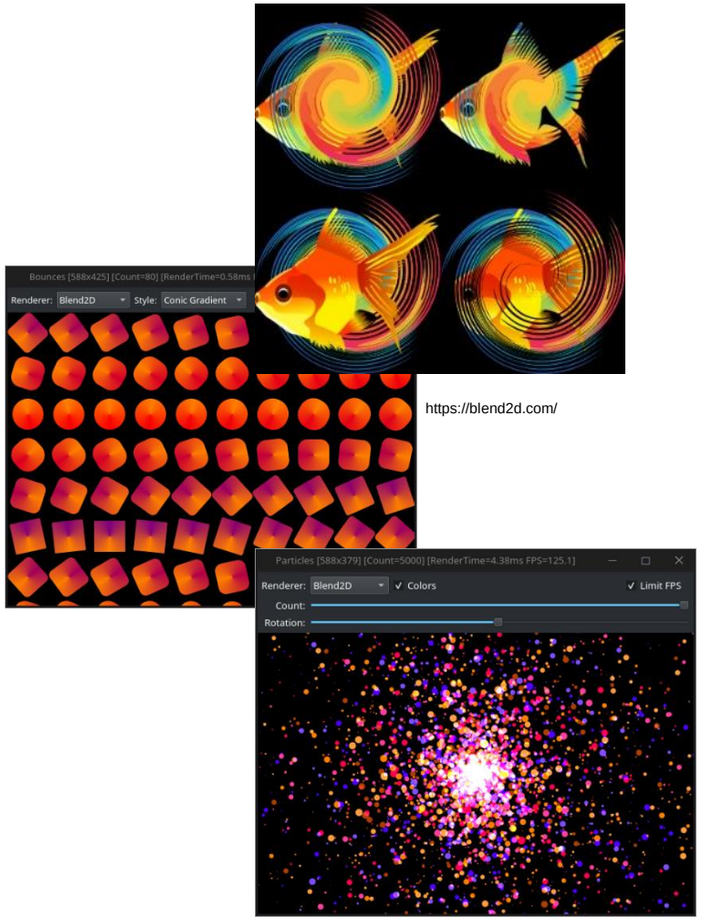
# TkZinc - canvas like widget extension to Tcl/Tk

- https://github.com/asb-capfan/TkZinc
- originally by Patrick Lecoanet
  - Christopher Chavez, Alfreod Tupone, Alex
- Description
  - Widget with Tcl, Perl, and Python Tk bindings.
  - Similar to canvas, but supports hierarchical structuring and affine 2d transforms.
  - Clippings can be set for subtrees
- Used in radar displays in two main French Air Traffic Control Centres
- Version:
  - https://github.com/asb-capfan/TkZinc/releases
  - 3.3.6 (Nov. 2020)
  - The 3.3.4 Reference Manual (Sep. 2006)
    - http://doro.poltava.ua/tkzinc/index.html
- Features
  - groups
  - display lists
  - clipping
  - transformation


http://doro.poltava.ua/tkzinc/index.html

# Blend2D

- https://blend2d.com/
- Description
  - high performance 2D vector graphics engine
  - written in C++
- started in 2017,
- developed by:
  - Petr Kobalicek
  - Fabian Yzerman
- Version
  - https://blend2d.com/download.html
    - https://github.com/blend2d/blend2d
  - 0.11.1 (Jul. 2024, started in 2017)
- License & Support
  - Zlib – License
  - Blend2D is an open-source rendering engine that is free to use by everyone including those who want to build commercial products with it. However, to make Blend2D more attractive for companies, commercial offering is also provided for customers that need more than a community support.



https://blend2d.com/

# Tk9 canvas - Improvements

**Improvements**

- canvas
- TkPath
- cad4tcl

# Tk9 canvas: canvas vs. Tkpath and cad4tcl (2018)

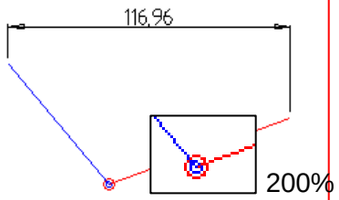
## Tk

(Mats Bengtsson, 1959-2008)

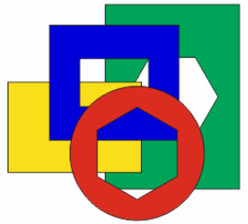
- Tk::canvas
  - Origin: top-left
  - File formats
    - PS (export)

## TkPath

- **Origin: top-left**
- **Anti aliasing**

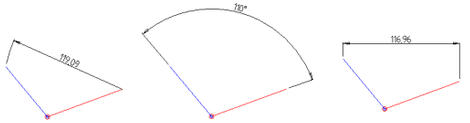



- **Allmost fully SVG-compatibility**
  - e.g.: path-Element



## cad4tcl

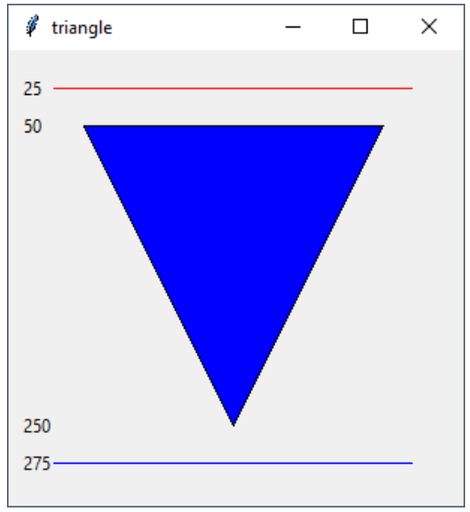
- **Tk-package based on**
  - TkPath, canvas
- **Origin: bottom-left**
- **Default scale (eg. 1:5)**
- **pan & zoom**
- **Dimensioning**



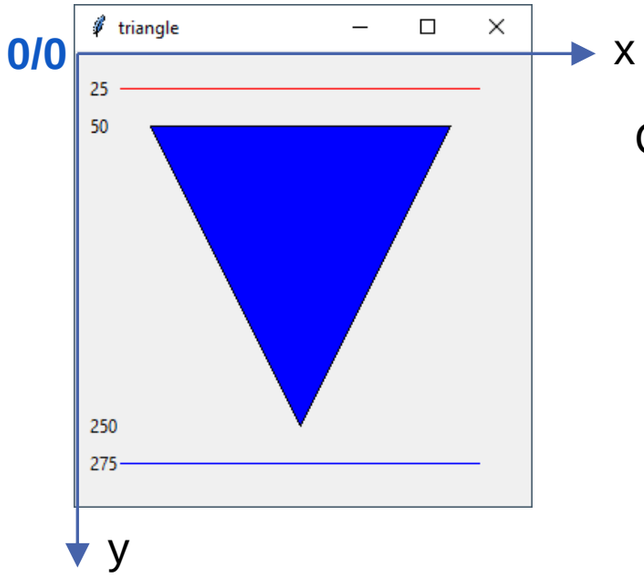

- **File formats**
  - SVG (import & export)
  - PDF (export)

# Tk9 canvas – top-left vs. bottom-left

```tcl
#!/usr/bin/env tclsh

# Load the Tk package
package require Tk

# Create a canvas widget
set canvas [canvas .canvas -width 300 -height 300]
pack $canvas

# Draw the triangle on the canvas
$canvas create polygon 50 50 150 250 250 50 -outline black -fill blue

# Draw a horizontal red line at y=250
$canvas create line 30 25 270  25 -fill red

# Draw a horizontal blue line at y=50
$canvas create line 30 275 270 275 -fill blue

# Create a text with value "50" at y=50
$canvas create text 10 25 -anchor w -text "25" -fill black
$canvas create text 10 50 -anchor w -text "50" -fill black

# Create a text with value "250" at y=50
$canvas create text 10 250 -anchor w -text "250" -fill black
$canvas create text 10 275 -anchor w -text "275" -fill black

```

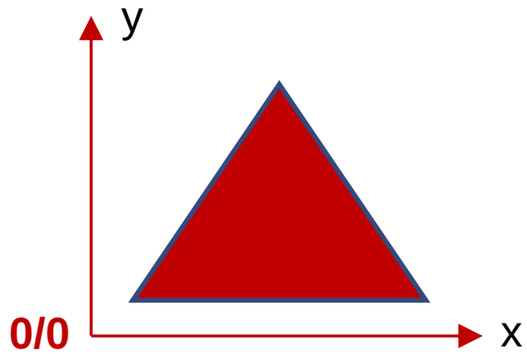



**top-left** (left hand - orientation)




Context:
- writing
- printing

**bottom-left** (right hand - orientation)




Context
- Function plots
- CAD

# Tk9 canvas - Wishlist

- Main features
  - anti aliasing
  - clipping
- SVG-Standard
  - item parameters like in SVG
  - styles
  - groups
- View management
  - Integrated zoom & pan events
  - Optional visible area
    - Like viewBox in SVG
  - Drawing areas
    - DIN & Letter -Formats
    - free size
  - Manipulation via SVG-DOM
    - SVG-Items
    - Tk-widgets?
- Origin
  - top-left
  - bottom-left (CAD)
- Import formats
  - .svg
- Export formats
  - .svg
  - .pdf
  - .wmf
  - Images (.png, .jpg, …)
- Boolean operation
  - Unite / Cut / Intersection
- Tklib
  - https://core.tcl-lang.org/tklib/doc/trunk/embedded/md/toc.md
  - canvas::sqmap
  - Canvas::zoom
  - crosshair
  - Diagrams
  - Plotchart
  - …

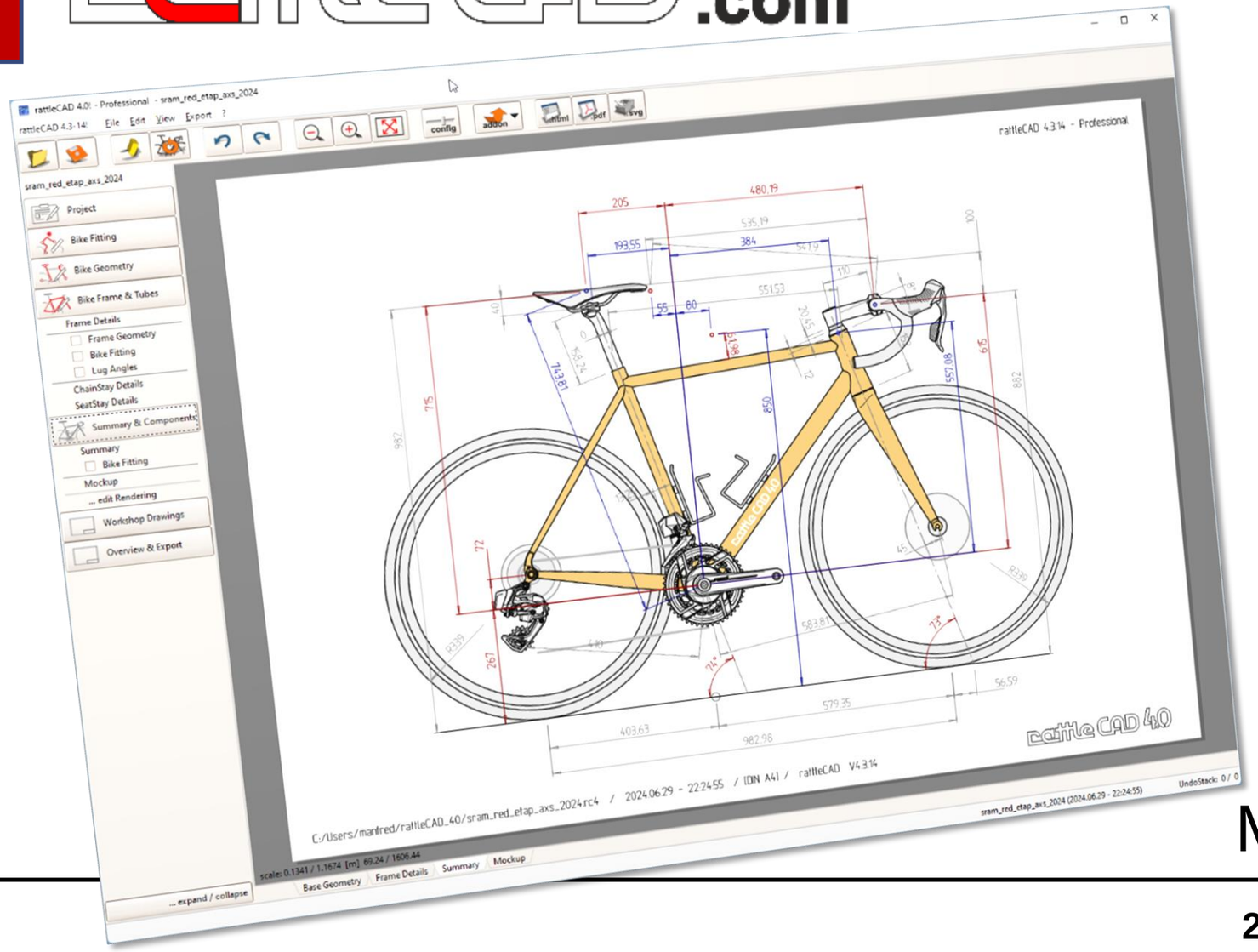
# Your requirements to Tk9 – canvas?

- SVG-Widgets in canvas
- HTML-Rendering
- …

- ????????

# Thanks!

rattleCAD.com




Manfred **ROSENBERGER**

**20th European Tcl/Tk Conference, July 2024**