image template
drawing links
rules – coloring links by position, part 1
rules – coloring links by position, part 2
bundling links
density histograms
scatter plots
rules – coloring links by size
A Original data set.
B Color of certain links is modified using rules.
C Geometry of nearby intra-chromosomal links has been adjusted to point the link outwards.
D Rules were used to change the thickness of links.
**Links as Bundles**

*Left* Links are drawn as lines (with adjustable, but constant, thickness). The lines start and end in the middle of the regions that define the link. *Right* Links are drawn as ribbons whose ends take on the thickness of the regions that define the link.

When links are drawn as ribbons, thickness is not necessarily constant across the link. Depending on the orientation of the start and end regions, and the relative orientation of the scales of the ideograms that the link connects, ribbons can twist. This twisting can be explicitly controlled (e.g. all ribbons can be made flat, regardless of orientation of scale and link regions).
LESSON 1
IDEOGRAM, TICK, GRID AND LABEL LAYOUT
The figure template showing mouse chromosomes 1-19 and human chromosome 1.

We will be rescaling the human chromosome so that it occupies ½ of the figure in order to reveal detail.

```plaintext
karyotype = 
  ../data/karyotype.human_mouse_labels.txt

chromosomes_units = 1000000
chromosomes_display_default = no

chromosomes =
  mm1;mm2;mm3;mm4;mm5;mm6;mm7;mm8;mm9;mm10;
  mm11;mm12;mm13;mm14;mm15;mm16;mm17;mm18;mm19;hs1

chromosomes_reverse =
  mm1;mm2;mm3;mm4;mm5;mm6;mm7;mm8;mm9;mm10;
  mm11;mm12;mm13;mm14;mm15;mm16;mm17;mm18;mm19

chromosomes_breaks = -hs1:120-140

#chromosomes_scale = hs1:11.8

<highlights>
  <highlight>
    show = yes
    file = ../data/highlight.txt
    r0 = 0.99r
    r1 = 0.999r
  </highlight>
</highlights>
```
chromosomes_scale = hs1:11.8

Mouse chromosomes 1-19 occupy $\frac{1}{2}$ of the figure and human chromosome 1 is shown in the other $\frac{1}{2}$.

The human chromosome has an axis break at 120-140 Mb to remove the centromere from the display (there is no data for this region).

Notice that the scale of mouse chromosomes runs counter-clockwise.
The links show 2,300 top alignments between human chr1 and mouse chr1-19. When transparency is used for link lines, it is possible to discern regions where the links are denser. The color for each link line here is `black_a5`.
When transparency is not used for link lines, dense links form a solid shape making it impossible to discern regions where the links are denser. The color for each link line here is black (note, no _aN suffix).
LESSON 3
RULES – COLORING LINKS BY POSITION, PART 1
<rule>
condition = _CHR2_ eq "mm11"
color = chr11_a3
z = 10
thickness = 2p
</rule>

Rules are used to color all links on mouse chr11.
A second rule is added to uniquely color all \textit{mm11} links that start at 20-50Mb of \textit{hs1}.
LESSON 4

RULES – COLORING LINKS BY POSITION, PART 2
COLORING BY CHROMOSOME AND POSITION

sessions/4/4/etc/circos.conf

```xml
<rules>
  <rule>
    condition  = _START1_ > 10e6 
                &&
                _END1_ < 20e6

    color      = eval( "chr" 
                     . substr(_CHR2_,rindex(_CHR2_,"m")+1) 
                     . "_a4" )

    z          = 10
    thickness  = 2p
  </rule>
</rules>
```

Links within 10-20Mb on *hs1* are colored by their destination chromosome.
Using a rule, all links are colored by the chromosome associated with their ends.

```xml
<rule>
  condition = 1
  color = eval( "chr"
    . substr(_CHR2_, rindex(_CHR2_, "m")+1)
    . "a4" )
  #z = 10
  #thickness = 2p
</rule>
```
LESSON 5

BUNDLING LINKS
The bundlelinks tools is used to logically group adjacent links together, forming larger links. Links are bundled based on their size and distance to each other.

Bundles are ideally drawn as ribbons, rather than lines, because bundle ends typically span a significant section of an ideogram.

The process of bundling links involves:

A. Logical grouping of adjacent links within the same or different chromosomes (chrA, chrB, chrC).

B. Checks for link bundling:
   - Bundle start and end points.
   - Conditions for bundling:
     - $x, y \leq \text{max\_gap}$
     - $x \leq \text{max\_gap\_start}$
     - $y \leq \text{max\_gap\_end}$
The result of bundling links from the previous session. Using `radius2`, the ends of the links are drawn closer to the mouse chromosomes.

```xml
<links>
<link_chain>
ribbon = yes
file = ../data/bundles.txt
bezier_radius = 0r
radius = 0.85r
thickness = 0p
color = black_a10

<rules>
<rule>
condition = 1
color =
  eval("chr"
        .
        substr(_CHR2_,rindex(_CHR2_,"m")+1)
        ."_a2")
radius2 = 0.99r
z = eval(_SIZE1_)
</rule>
</rules>
```
By setting the $z$ value to be inversely proportional to link size, small links are drawn on top.
Varying the minimum number of links per bundle changes the sensitivity of bundling.

When a large number of links is required (e.g. \( n=20 \)), only those regions that are connected by a large number of links are turned into bundles.

Decreasing \( n \) (e.g. \( n=5,10 \)), increases the number of bundles.

If the cutoff is small (e.g. \( n=2,3 \)), it is possible to create a large number of bundles, because fewer links are required to form a bundle.

\[
\text{max\_gap\_1 = 3Mb} \\
n = \text{minimum links per bundle}
\]
LESSON 6

DENSITY HISTOGRAMS
Recall that the input data to a stacked histogram tracks has a comma-delimited list of values.

```
hs1 81000000 81999999
  1.0000,0.0000,0.0000,1.0000,0.0000,0.0000,
  1.0000,0.0000,1.0000,0.0000,0.0000,0.0000,
  0.0000,0.0000,1.0000,2.0000,0.0000,0.0000,0.0000
```
Three density histograms summarize information about the synteny between human chromosome 1 and the mouse genome. The links in this figure are drawn as bundles, but the density histograms are calculated based on the individual links.
The normalized histogram (left) is further modified by sorting the stacked bins by value (right).
SESSION 4 / LINKS AND RULES

LESSON 7

SCATTER PLOTS
A scatter plot is added to the figure to show average conservation within 1Mb bins on *hs1*.

Rules are applied to the plot to color glyphs based on value.
Glyphs are not affected by any rules because the first rule matches all points but does not change formatting.

```
<rule>
importance = 110
condition  = 1
#
# no formatting parameters are modified by this rule
</rule>
```
Glyph size is made proportional to the deviation of the data point (distance to average).
Glyph size is made proportional to the deviation of the data point (distance to average).
By mapping value onto glyph size and then placing all the glyphs at the same radial position (by changing data values), a glyph track is created. Stacking such glyph tracks can create very interesting (and attractive) visualizations.

```
<rule>
importance = 60
condition = 1
value = 0.47
</rule>
```
RULES ARE POWERFUL

Unaltered scatter plot track.

Glyph color, size, shape and position have been altered with rules.
LESSON 8

RULES - COLORING LINKS BY SIZE
ENCODING BUNDLE SIZE AS GRAYSCALE

```xml
<rule>
  importance = 100
  condition = 1
  color = eval("black_a"
               .int(max(1,6-_SIZE1_/5e6))
  #flow = continue
</rule>
```

Bundles are shaded in proportion to their size on *hs1*. 
Rules help create three groups of links.

Links on *mm8* and *mm11* are drawn on top, in order of link size, and colored by mouse chromosome color. Links on *mm5* are drawn next, with a subtler red tint. All other links are drawn below and shaded in proportion to their size.

```
<rule>

importance = 90

condition = _CHR2_ eq "mm8"
    ||
    _CHR2_ eq "mm11"

color = eval("chr"
    . substr(_CHR2_, rindex(_CHR2_,"m")+1)
    . "a1")

z = eval(_SIZE1_)

</rule>
```
<rule>
importance = 80

condition = _CHR2_ eq "mm5"

color = eval("chr"
  . substr(_CHR2_,rindex(_CHR2_,"m")+1)
  . ":a3")

z = 10
</rule>

Bundles are shown in three layers. At the top are mm8 and mm11 bundles, below are lightly colored mm5 bundles, with all remaining bundles beneath.