

# Tree Editing & Visualization

Lisa Pokorny & Miguel  
Ángel Naranjo Ortiz

DE  
NOVO

Raw reads → Sanitize → Filter → Assemble → Translate

# WORKSHOP ON GENOMICS

DATABASES (Ensembl, UNIPROT, SRA...)

Download

Reduce  
redundancy

(Transcriptomics)

Orthology inference

Alignment

Trimming / Masking

1:1 orthologs

orthologs and/or paralogs

Concatenation

Analyses

Matrix construction, Model selection, ML, BI,  
molecular dating, ancestral reconstruction, etc.

**Evolutionary relationships**

**Gene family evolution**

**Prediction of function**

**HGT / ILS / DL**

Infer & compare  
Individual gene trees

Analyses

Gene tree/species tree reconciliation,  
gene duplic./loss, gene transfer, selection, etc.

DE NOVO Raw reads → Sanitize → Filter → Assemble → Translate

## WORKSHOP ON GENOMICS

DATABASES (Ensembl, UNIPROT, SRA...)

Download

Reduce  
redundancy

(Transcriptomics)

Orthology inference

Alignment

Trimming / Masking

1:1 orthologs

orthologs and/or paralogs

## WORKSHOP ON PHYLOGENOMICS

Concatenation

Analyses

Matrix construction, Model selection, ML, BI,  
molecular dating, ancestral reconstruction, etc.

Evolutionary relationships

Gene family evolution

Prediction of function

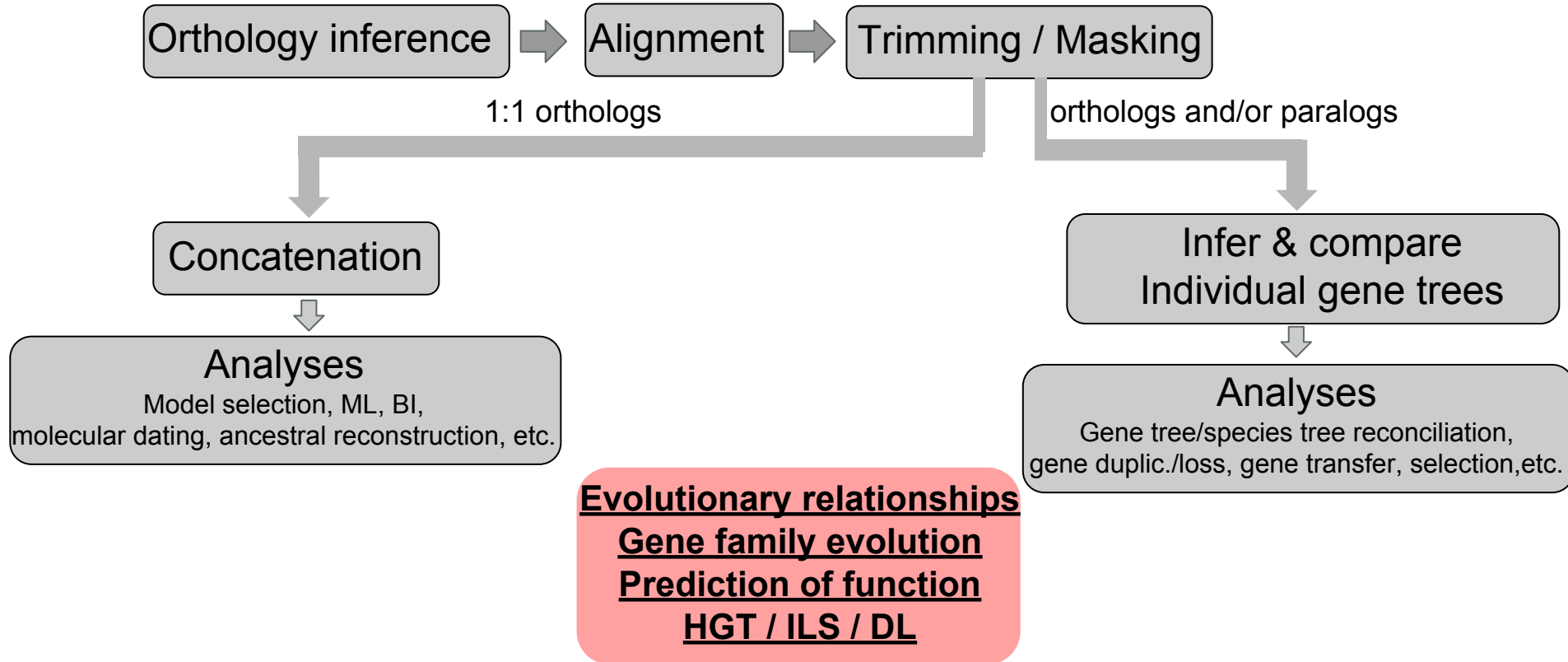
HGT / ILS / DL

Reconcile & Compare  
Individual gene trees

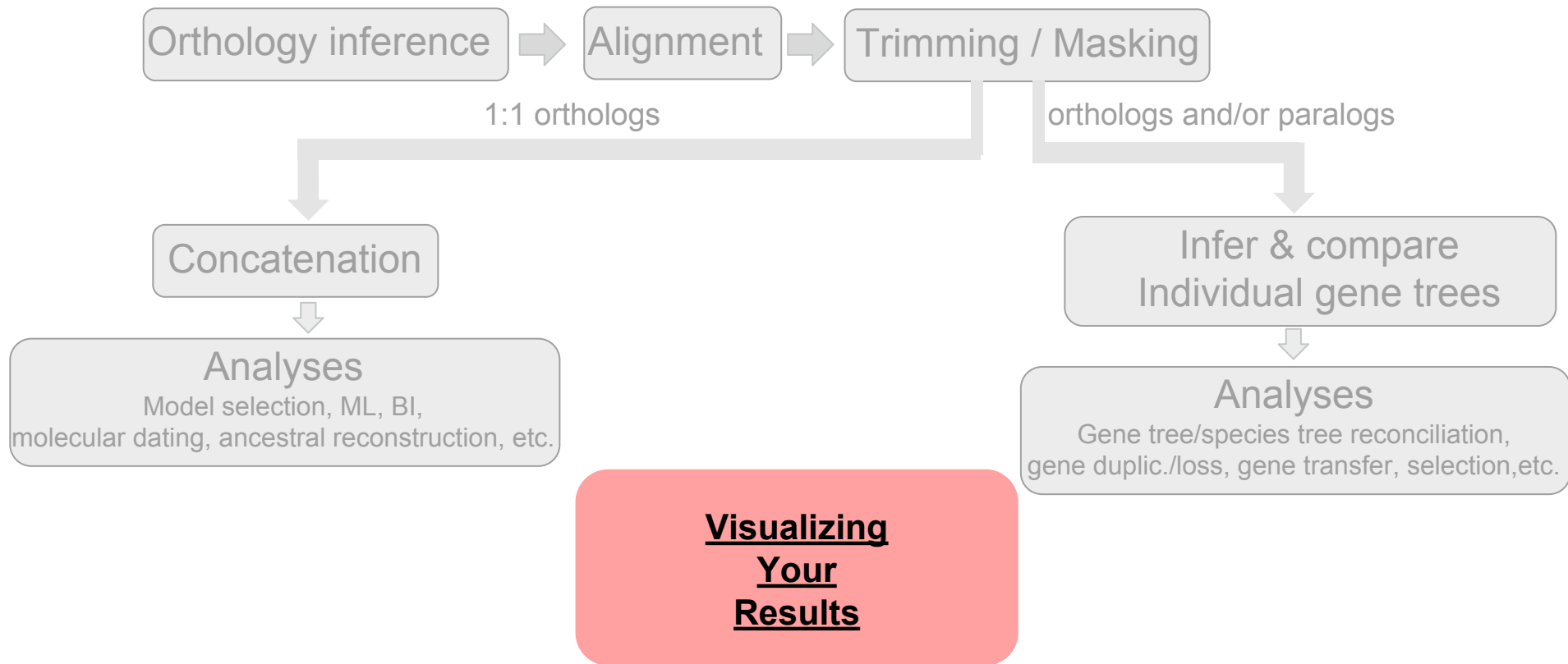
Analyses

Gene tree/species tree reconciliation,  
gene duplic./loss, gene transfer, selection, etc.

# WORKSHOP ON PHYLOGENOMICS



# WORKSHOP ON PHYLOGENOMICS





# **ESSENTIALS OF DATA VISUALIZATION**

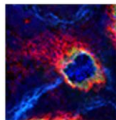
THINKING ABOUT DRAWING DATA + COMMUNICATING SCIENCE



SCIENTIFIC  
AMERICAN



GRAPHIC  
SCIENCE



PRESENTATIONS



DRINKS &  
SCIENCE WORKSHOP

VIZBI

KEYNOTE  
VISUAL DESIGN  
PRINCIPLES



# Insights into Data Visualization: 8-part mini-series

This video series focuses on relevant and practical concepts in scientific data visualization. Our aim is to make you think more clearly about visual presentation and to make you a better communicator.

Each video in the series presents fundamental ideas and is designed to provide constraints and guidance to your thoughts about communicating your data. The purpose of scientific data visualization is not merely to inform but also to answer and generate hypotheses.

Though few firm rules exist when it comes to how to achieve clarity—whatever the communication medium—we must meet core requirements such as consistency, redundancy and appropriate mapping between relevant and salience. We present these essential topics using biological data as examples. But if you're not a biologist, don't worry. Instead, think about the data structure rather than meaning and you'll be fine.

## Resources

Each video is about 15 minutes long and comes with a slide deck of the images used in the video, exercise and suggested solutions.

[Download all course materials.](#)

## Episodes

### 1 DATA ENCODING

MAKE IT EASY TO ANSWER RELEVANT QUESTIONS.

[watch](#) | [PDF](#) | When you think of data visualization, the first ideas that come to mind are a scatter plot, or a bar chart, a box plot or a network diagram. These are all data encodings—methods that relate data values to the positions, sizes and shapes of the lines or symbols that appear on the screen or in a figure. There are many data encodings—which do you choose?

### 2 SHAPES AND SYMBOLS

INTUITIVELY ENCODE ROLE AND RELEVANCE.

[watch](#) | [PDF](#) | Shapes and glyphs are really important. They make up the heart of a lot of data plots. Your default should be the circle. If you need different shapes, try to map the classes as intuitively as possible onto the shapes. Use less prominent symbols for data that are less relevant (such as reference data included for context).

### 3 COLOR

USE IT FOR EMPHASIS AND VISUAL SEPARATION.

[watch](#) | [PDF](#) | Color is one of the most exciting ways in which you can completely screw over your visualization. What can start off as a great diagram can be absolutely ruined by a lack of color judgment. When using color, ask yourself—do I need it? Try to work around it using grey tones from Brewer palettes. If you succeed, you're in a perfect place to use spot color, sparingly, for emphasis.

### 4 UNCERTAINTY

DON'T MAKE ERRORS IN ERROR BARS.

[watch](#) | [PDF](#) | Knowing the limits of your knowledge is very important. In biology, it's important to be able to sample the extent of biological variation. And so being able to show this and other forms of variation in measurements or any computed values in visualizations is very important—it addresses reproducibility and your capacity to make statistical inference. Often this is done with error bars. Ironically, there's a lot of error associated with the use of and interpretation of error bars.



#### LUNCH BREAK

Distractions and  
amusements, with a  
sandwich and coffee.



## 5 DESIGN

### ORGANIZE AND CLARIFY.

[watch](#) | [PDF](#) | Design plays a large role in data visualization. Think of design as choreography for the page. In our context it's not merely driven by aesthetic, but function. Although there's always room for aesthetic—gently applied—and I really encourage you to find your own and continue to refine it. But always remember, be understood before being articulate. Be legible before being attractive! Your goal here isn't to make inroads on the global stage of aesthetic studies. Become a good visual explainer. It's harder ... and more worth doing.

## 6 NOTHING

### NO DATA, NO INK.

[watch](#) | [PDF](#) | Data-to-ink ratio, taken to the extreme: if there is no data to show, no ink should be used. The idea of “no data to show” may correspond to a variety of scenarios. There may be sincerely no data to show—no values were collected. Or, there are no significant changes to see. Where possible, you should use empty space to indicate lack of data or lack of change in data. You should never be distracted by something that isn't relevant and empty space is not distracting—it really just provides contrast to adjacent elements, which presumably correspond to actual data or actionable data.

## 7 LABELS

### RESPECT TYPE AND USE IT TO ESTABLISH HIERARCHY.

[watch](#) | [PDF](#) | Open up a journal or your favourite text book. Find a figure. There's probably some labels in there. Maybe it's a multi-panel figure and the labels are the titles. Maybe there are some callouts that tell you what the parts are. If it's a plot there are probably axis labels and tick labels and maybe a legend with some labels. There's usually several informational layers in the image, each with their own labels. These labels should reflect that these layers are different. They should also reflect the relative importance of these layers.

## 8 PROCESS

### CREATING A VISUALIZATION FOR SCIENTIFIC AMERICAN GRAPHIC SCIENCE: FROM START TO FINISH.

[watch](#) | [PDF](#) | Let's now look at the process of designing a visualization from scratch—from the encoding all the way to design. This was a graphic I did for the [June 2015 issue of Scientific American](#). It appeared on the Graphic Science page.

# Essentials of Data Visualization

How do we get from data to visualization?



# Essentials of Data Visualization

How do we get from data to visualization?

|                 | Points | Lines                                  | Areas            | Best to Show                                      |
|-----------------|--------|----------------------------------------|------------------|---------------------------------------------------|
| Shape           |        | <i>Possible, but too Weird to Show</i> | <i>Cartogram</i> | <i>Qualitative Differences</i>                    |
| Size            |        |                                        | <i>Cartogram</i> | <i>Quantitative Differences</i>                   |
| Color Hue       |        |                                        |                  | <i>Qualitative Differences</i>                    |
| Color Value     |        |                                        |                  | <i>Quantitative Differences</i>                   |
| Color Intensity |        |                                        |                  | <i>Qualitative Differences</i>                    |
| Texture         |        |                                        |                  | <i>Qualitative &amp; Quantitative Differences</i> |

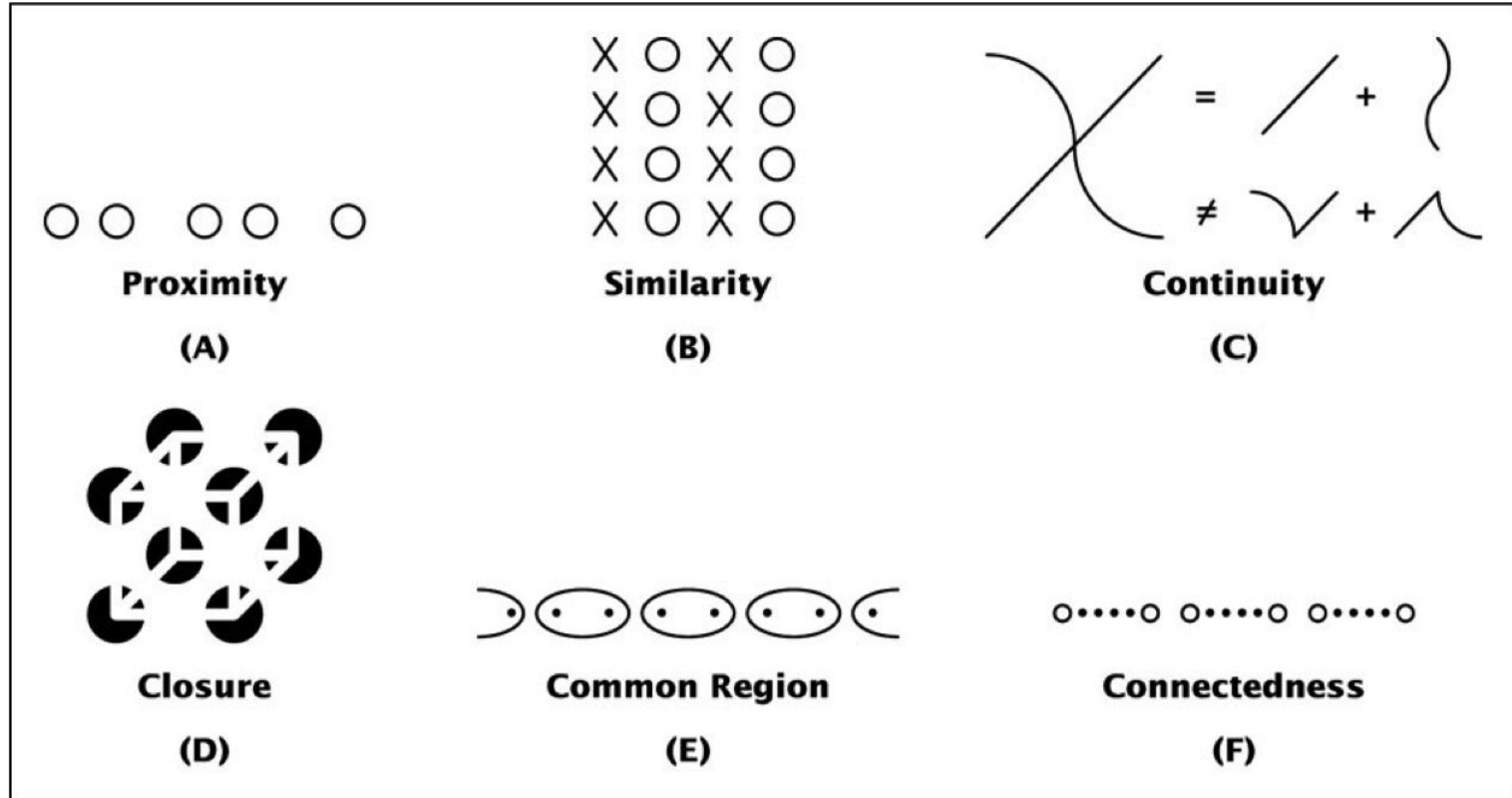
# Essentials of Data Visualization

How do we get from data to visualization?

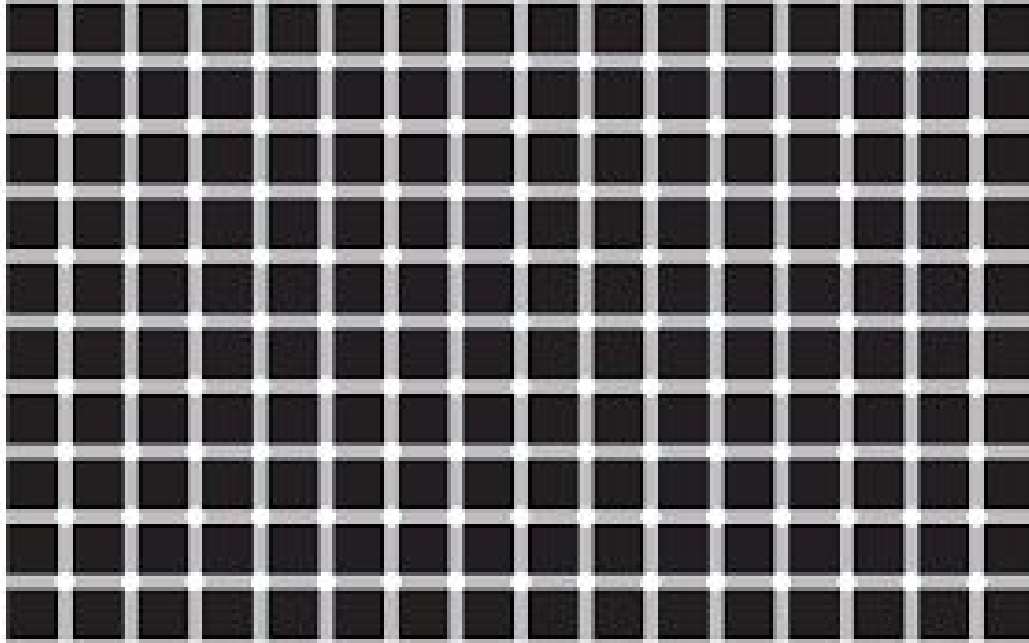
We need to understand:

- **properties of the data / data type**
  - Phylogenies (cladograms, phylograms, chronograms, cloudograms, etc.)
  - Networks (circos, hive plots, etc.)
- **properties of the image / visual encoding**
  - What? Points, lines, labels...
  - Where? 2D, 3D...
  - How? Size, shape, texture, color, hue...
- **the rules of mapping data to image**
  - Principles of grouping
  - etc.

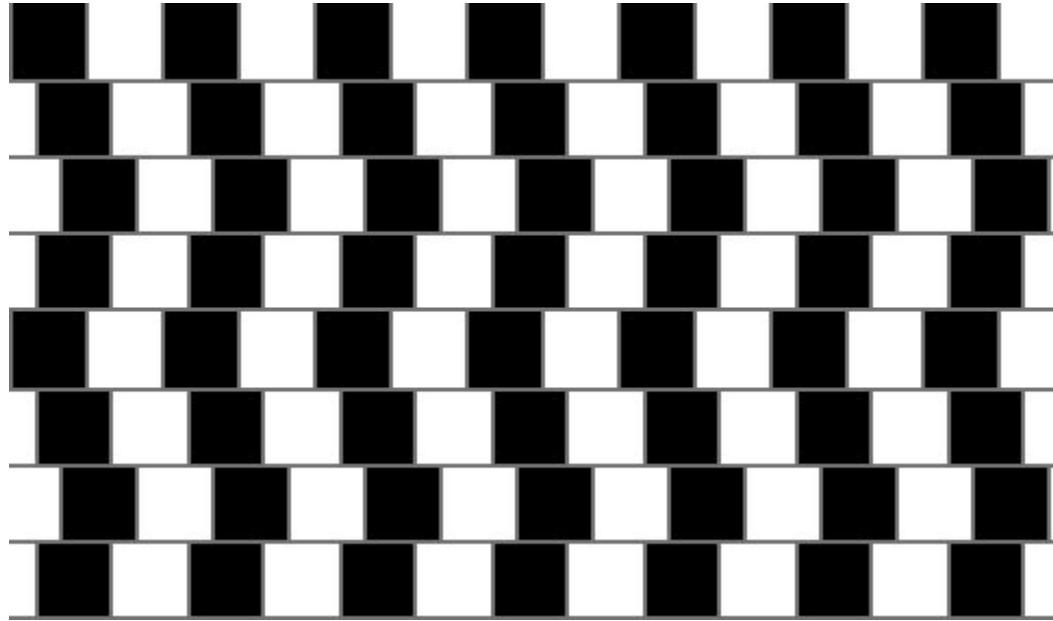
# Principles of Grouping



# Optical Illusions – Hermann Grid



# Optical Illusions – Café Wall

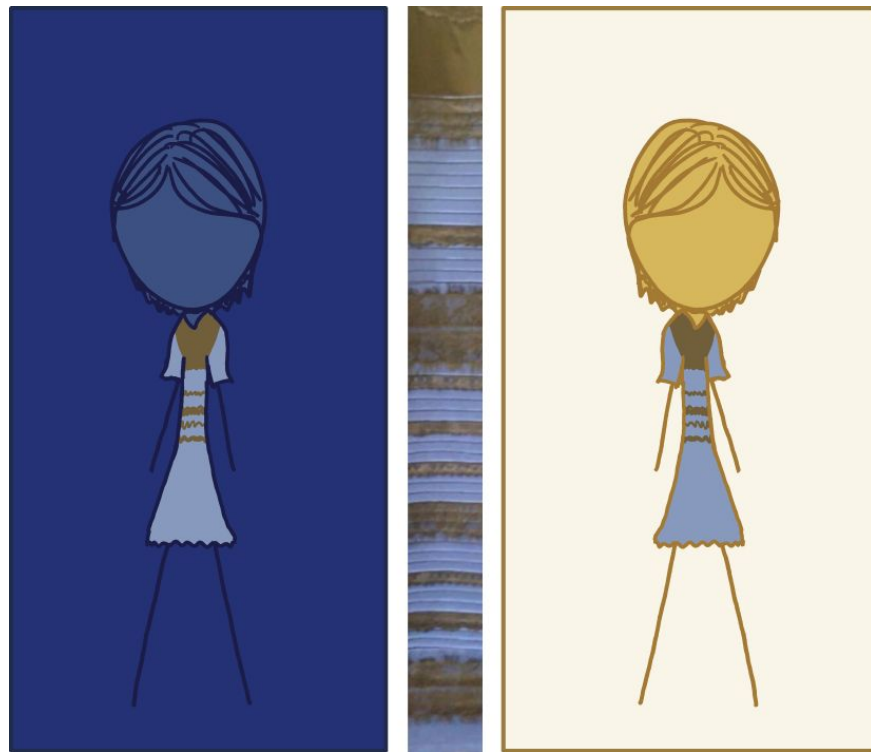


# Color



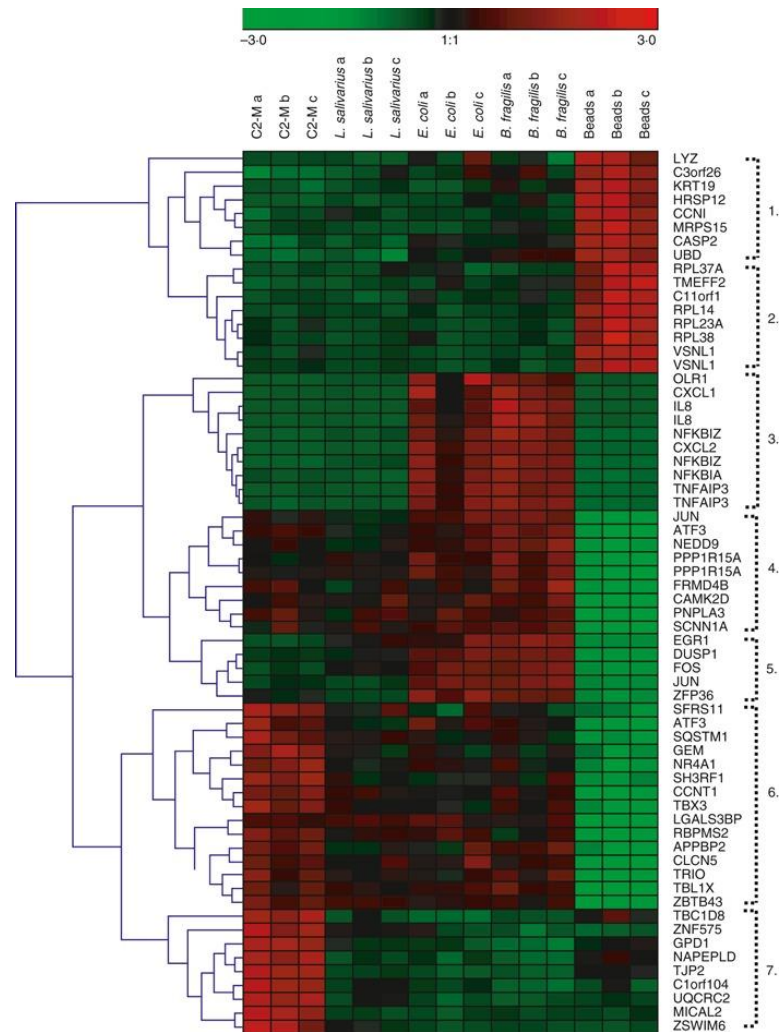
# Color Context

Context affects your color perception



# Color Context

Context affects your color perception



## Context affects your color perception



# Color Blindness



# Color Blindness

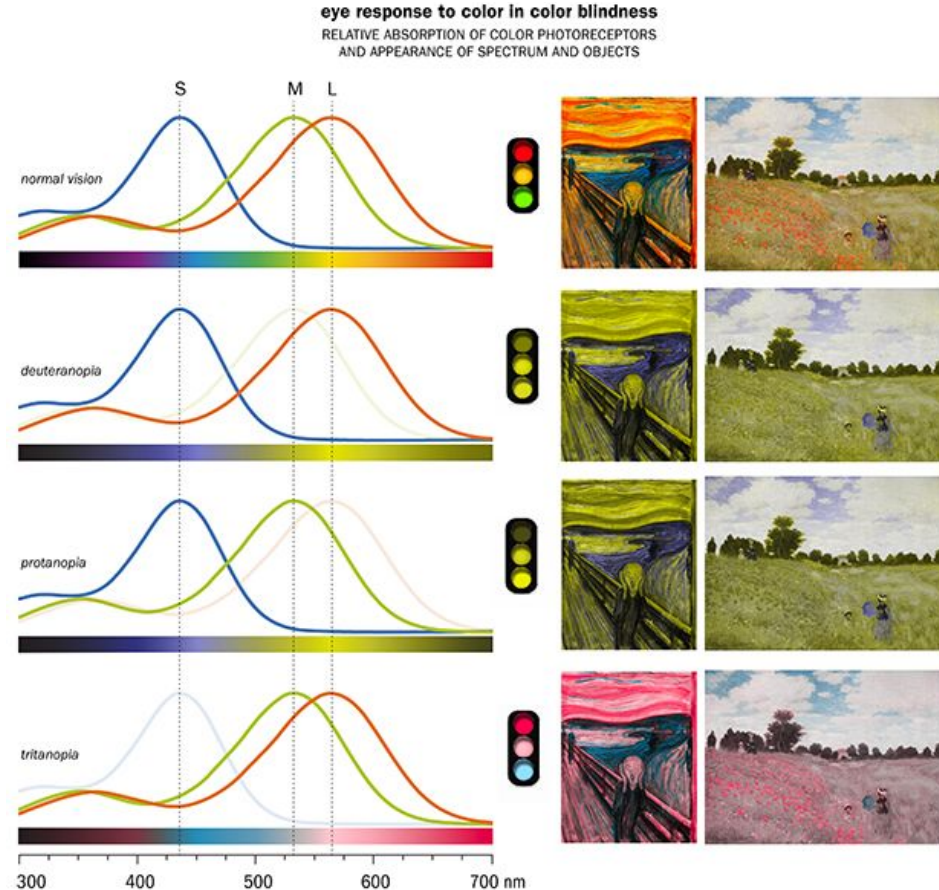
## COLOR PALETTES FOR COLOR BLINDNESS

In an audience of 8 men and 8 women, chances are 50% that at least one has some degree of color blindness<sup>1,2</sup>. When encoding information or designing content, use colors that is color-blind safe.

[Color Oracle](#) is a good and free color blindness simulator for Windows, Mac and Linux.

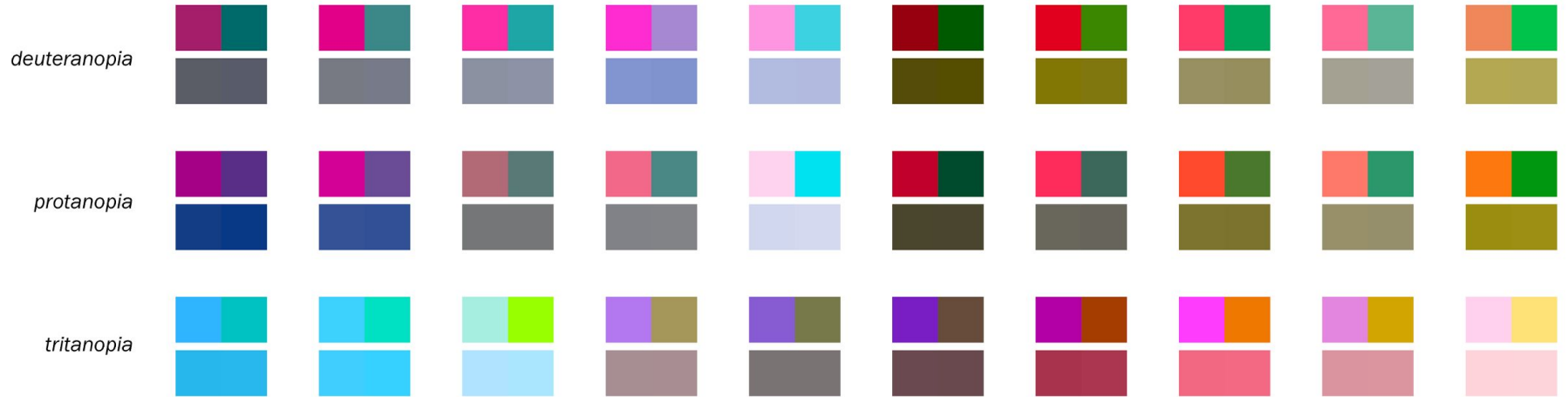
## COLOR RECEPTORS ARE REDUCED OR ABSENT IN COLOR BLINDNESS

The normal human eye is a 3-channel color detector<sup>3</sup>. There are three types of photoreceptors, each sensitive to a different part of the spectrum. Their combined response to a given wavelength produces a unique response that is the basis of the perception of color.



# Color Blindness

## indistinguishable colors in color blindness



<http://mkweb.bcgsc.ca/colorblind>

# Color Blindness – Palettes

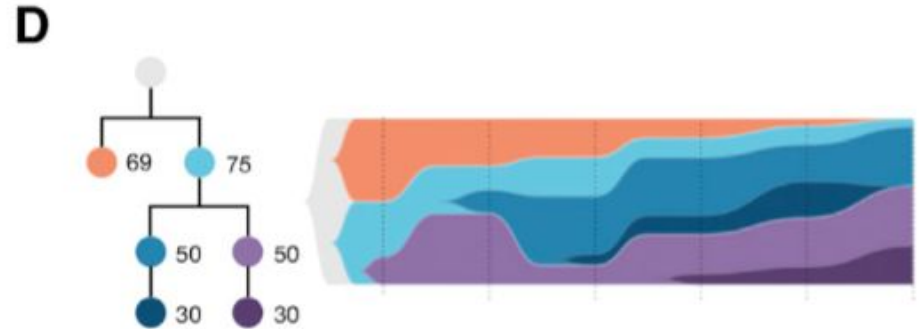
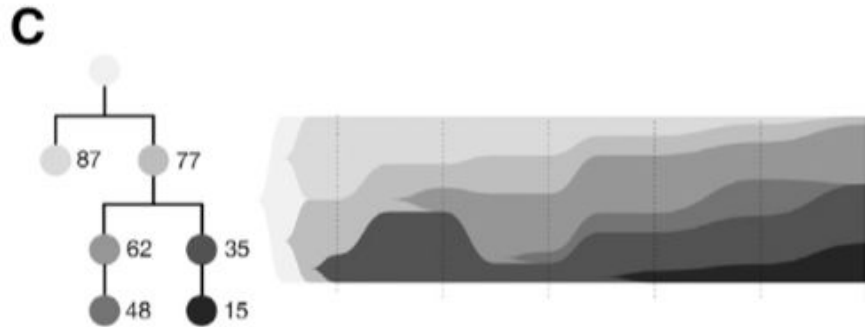
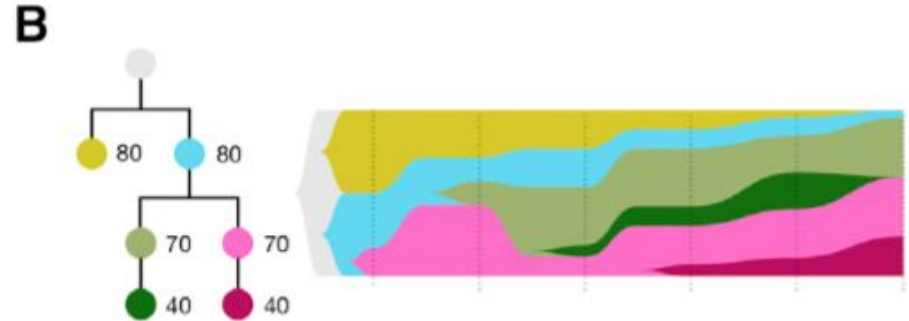
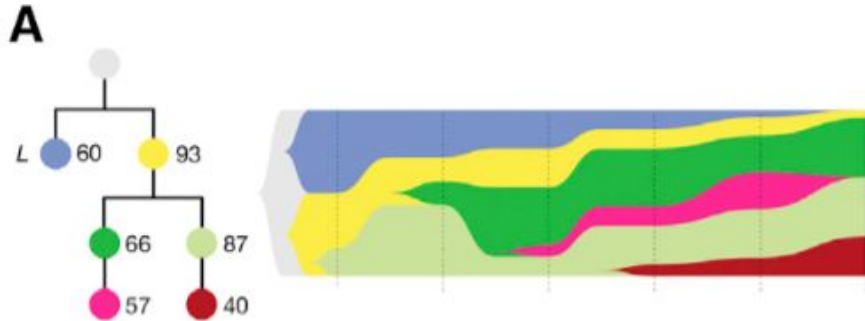
- General advice: <http://jfly.iam.u-tokyo.ac.jp/color/>
- Cartography: <http://colorbrewer2.org/>
- R plotting system (ggplot2):

[http://www.cookbook-r.com/Graphs/Colors\\_\(ggplot2\)/#a-colorblind-friendly-palette](http://www.cookbook-r.com/Graphs/Colors_(ggplot2)/#a-colorblind-friendly-palette)

|   | Original                                                                           | Simulation                                                                         |                                                                                    |                                                                                    |                  | Hue  | for Photoshop, Illustrator, Freehand, etc. |               |            |  |
|---|------------------------------------------------------------------------------------|------------------------------------------------------------------------------------|------------------------------------------------------------------------------------|------------------------------------------------------------------------------------|------------------|------|--------------------------------------------|---------------|------------|--|
|   |                                                                                    | Protan                                                                             | Deutan                                                                             | Tritan                                                                             |                  |      | C,M,Y,K (%)                                | R,G,B (0-255) | R,G,B (%)  |  |
| 1 |   |   |   |   | ■ Black          | – °  | (0,0,0,100)                                | (0,0,0)       | (0,0,0)    |  |
| 2 |   |   |   |   | ■ Orange         | 41°  | (0,50,100,0)                               | (230,159,0)   | (90,60,0)  |  |
| 3 |   |   |   |   | ■ Sky Blue       | 202° | (80,0,0,0)                                 | (86,180,233)  | (35,70,90) |  |
| 4 |   |   |   |   | ■ bluish Green   | 164° | (97,0,75,0)                                | (0,158,115)   | (0,60,50)  |  |
| 5 |   |   |   |   | ■ Yellow         | 56°  | (10,5,90,0)                                | (240,228,66)  | (95,90,25) |  |
| 6 |   |   |   |   | ■ Blue           | 202° | (100,50,0,0)                               | (0,114,178)   | (0,45,70)  |  |
| 7 |   |   |   |   | ■ Vermillion     | 27°  | (0,80,100,0)                               | (213,94,0)    | (80,40,0)  |  |
| 8 |  |  |  |  | ■ reddish Purple | 326° | (10,70,0,0)                                | (204,121,167) | (80,60,70) |  |

<http://jfly.iam.u-tokyo.ac.jp/color/>

# Color Blindness – Luminance



# Font Types

## Arial

Arabic Typesetting

Aparajita

AngsanaUPC

Angsana New

Andalus

**ALGERIAN**

**Aharoni**

Agency FB

AGaramondPro-Regular

*AGaramondPro-Italic*

***AGaramondPro-BoldItalic***

AGaramondPro-Bold

AdobeSongStd-Light

AdobeMyungjoStd-Medium

AdobeMingStd-Light

AdobeKaitiStd-Regular

**AdobeHeitiStd-Regular**

AdobeHebrew-Regular

*AdobeHebrew-Italic*

***AdobeHebrew-BoldItalic***

AdobeHebrew-Bold

**AdobeGothicStd-Bold**

**AdobeFanHeitiStd-Bold**

AdobeFangsongStd-Regular

AdobeArabic-Regular

*AdobeArabic-Italic*

*AdobeArabic-BoldItalic*

AdobeArabic-Bold

*ACaslonPro-SemiboldItalic*

ACaslonPro-Semibold

ACaslonPro-Regular

*ACaslonPro-Italic*

***ACaslonPro-BoldItalic***

**ACaslonPro-Bold**

## Cambria

Calisto MT

Californian FB

Calibri

*Brush Script Std*

*Brush Script MT*

BrowalliaUPC

Browallia New

**Broadway**

**Britannic**

Bradley Hand ITC

Bookshelf Symbol

Bookman Old Style

Bookman

Book Antiqua

Bodoni MT Poster Compressed

Bodoni MT Condensed

**Bodoni MT Black**

Bodoni MT

**BlackoakStd**

*Blackletter ITC*

BitChStd

**Bernard MT Condensed**

**Berlin Sans FB Demi**

Berlin Sans FB

Bell MT

Bauhaus

BatangChe

Batang

Baskerville Old Face

AvantGarde

Arial Unicode MS

Arial Rounded MT

Arial Narrow

**Arial Black**

## DotumChe

Dotum

**DokChampa**

DilemmaUPC

DFKai-SB

David

DaunPenh

Curly MT

Courier New

Courier

CordiaUPC

Cordia New

Corbel

COPPERPLATE GOTHIC LIGHT

Copperplate Gothic

***CooperBlackStd-Italic***

***CooperBlackStd***

***Cooper Black***

Constantia

Costolas

Comic Sans MS

Colonna MT

Chiller

**CHARLEMAGNESTD-BOLD**

ChaparralPro-Regular

*ChaparralPro-Italic*

***ChaparralPro-BoldIt***

***ChaparralPro-Bold***

Century Schoolbook

Century Gothic

Century

Centaur

**CASTELLAR**

Candara

Cambria Math

## Gautami

Garamond

Gabriola

*French Script MT*

*Freestyle Script*

FreesiaUPC

FrankRuehl

**Franklin Gothic Medium Cond**

**Franklin Gothic Medium**

**Franklin Gothic Heavy**

**Franklin Gothic Demi Cond**

**Franklin Gothic Demi**

Franklin Gothic Book

**Forté**

Footlight MT Light

FLUX

FELIX TITLING

FangSong

Euphemia

EucrosiaUPC

Ευχλιδ Σημβολ

Ε Μ

Ε Μ

Ε Μ

Ε Μ

Ε Μ

Ε Μ

Ε Μ

Ε Μ

Ε Μ

Ε Μ

Ε Μ

Ε Μ

Ε Μ

Ε Μ

## KodchiangUPC

Khmer UI

**Kartika**

Kalinga

Kaiti

Juice ITC

**Jokerman**

JasmineUPC

Iskoola Pota

IrisUPC

*Informal Roman*

Imprint MT Shadow

**Impact**

**HoboStd**

High Tower Text

Helvetica-Narrow

Helvetica

flarrington

Harlow Solid

Haettenschweiler

GungsuhChe

Gungsuh

GulimChe

Gulim

**GOUDY STOUT**

Goudy Old Style

Gloucester MT Extra Condensed

Gisha

Gill Sans Ultra

Gill Sans MT Ext Condensed

Gill Sans MT Condensed

Gill Sans MT

Gill Sans

Gill Sans

Gill Sans

## Lucida Fax Regular

**Lucida Fax Demibold**

Lucida Fax

Lucida Console

*Lucida Calligraphy*

**Lucida Bright Demibold**

Lucida Bright

LITHOSPRO-REGULAR

**LITHOSPRO-BLACK**

UlyUPC

Levenim MT

*LetterGothicStd-Slanted*

*LetterGothicStd-BoldSlanted*

LetterGothicStd-Bold

LetterGothicStd

Leelawadee

Latha

Lao UI

*Navarrothe Script*

**Kristen ITC**

KozMinPro-Regular

KozMinPro-Medium

KozMinPro-Light

**KozMinPro-Heavy**

KozMinPro-ExtraLight

KozMinPro-Bold

KozMinPr

KozGoPro-Regular

KozGoPro-Medium

KozGoPro-Light

**KozGoPro-Heavy**

KozGoPro-ExtraLight

**KozGoPro-Bold**

KozGoPr

Kokila

# Font Types However...

## IN TYPOGRAPHY

Serifs are the small lines tailing from the edges of letters and symbols, separated into distinct units for a typewriter or typesetter

Check out my sweet serifs!



**Serif fonts are easier to read in printed works**

This is because the serif make the individual letters more distinctive and easier for our brains to recognise quickly. Without the serif, the brain has to spend longer identifying the letter because the shape is less distinctive.

Sans-serif is a typeface that does not have the small projecting features called "serifs" at the end of strokes.

Modern, Minimal, Magnificent, I am Sans



**Sans serif fonts are better on the web**

An important exception must be made for the web. Printed works generally have a resolution of at least 1,000 dots per inch; whereas, computer monitors are typically around 100 dots per inch.

# Dyslexia

Dyslexia is a neurodevelopmental disorder (DRD) is the most common learning disability. Dyslexia is the most recognized of reading disorders, however not all reading disorders are linked to dyslexia.

Some see dyslexia as distinct from reading difficulties arising from other causes, such as a non-verbal cognitive deficit with vision or hearing, or poor or inadequate reading instruction. There are three proposed categories of dyslexia (auditory, visual and attentional), although individual cases of dyslexia are better explained by specific underlying neurological factors and co-occurring learning disabilities (e.g. attention-deficit/hyperactivity disorder, math disability, etc.). Although it is considered to be a specific language-based learning disability in the research literature, dyslexia also affects one's expressive language skills. Researchers at MIT found that people with dyslexia exhibit impaired voice-recognition abilities.

# Dyslexia Friendly Text

- Use a plain, evenly spaced sans serif font, e.g., Arial, ComicSans, Verdana, Trebuchet, Calibri, Century Gothic...
- Font size should be 12-14 point or larger
- Use dark coloured text on a light (not white) background
- Avoid underlining and *italics*: these tend to make the text appear to run together; use **bold** instead
- AVOID TEXT IN BLOCK CAPITALS!
- Use left-justified with ragged right edge
- Avoid narrow columns
- Line spacing of 1.5 is preferable
- Use bullet points and numbering rather than continuous prose

# Dyslexia Friendly Font Types – OpenDyslexic

## OpenDyslexic

Free, OpenSource Dyslexia Typeface

[Get it Free](#) [Browse Products](#) [Get it Free](#) [License](#) [News](#) [About](#) [Try it!](#) [FAQ](#)

### OpenDyslexic

OpenDyslexic is a new open source font created to increase readability for readers with dyslexia. The typeface includes regular, bold, italic, and bold-italic styles. It is being updated continually and improved based on input from dyslexic users. OpenDyslexic is free for Commercial and Personal use.

Download Now!

search here ...

Go

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| A | B | C | D | E | F | G | H | I |
| J | K | L | M | N | O | P | Q | R |
| S | T | U | V | W | X | Y | Z | a |
| b | c | d | e | f | g | h | i | j |
| k | l | m | n | o | p | q | r | s |
| t | u | v | w | x | y | z |   |   |

# Dyslexia Friendly Font Types – Lexie Readable



[Home](#) [Fonts](#) [Free Fonts](#) [Kernel](#) [My Account](#)

LEX  
LEXIE  
READABLE

A sample of the Lexie Readable font, showing the words 'LEX', 'LEXIE', and 'READABLE' in a large, bold, sans-serif typeface. The letters are thick and have a slightly irregular, hand-drawn quality, with some letters like 'X' and 'E' having unique, non-symmetrical shapes.

## Lexie Readable

Lexie Readable (formerly Lexia Readable) was designed with accessibility and legibility in mind, an attempt to capture the strength and clarity of Comic Sans without the comic book associations. Features like the non-symmetrical b and d, and the handwritten forms of a and g may help dyslexic readers. You can read more about the story behind Lexie Readable in the **Kernel**.

Lexie Readable now includes a full complement of Latin Extended-A characters and numerous subtle outline improvements.

The Regular and Bold weights can be downloaded and used freely without a licence by educational and charitable institutions as well as by individuals.

The **Basic Family** includes Regular, Italic, Bold and Bold Italic. The **Heavy** package includes a Heavy Outline version.

Download the Regular & Bold Free for Personal, Educational or Charity Use: [Download](#)

# Dyslexia Friendly Font Types – Dyslexie



Dyslexie font is patented

Like Share 15K

Size A A Background Language

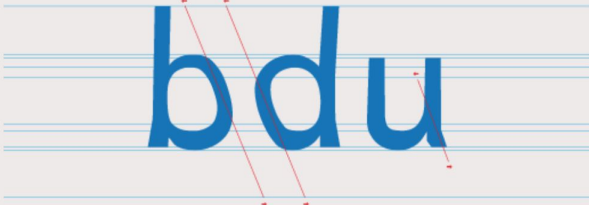
Home Dyslexie font Order Support Books/Apps Contact

## How the Dyslexie font works

To show you what makes the Dyslexie font different from other fonts, we will show you the 9 main features below.

### 02 Slanted parts

Characters which look quite similar have been adapted by changing the tails, to reduce the similarity and avoid the problem of mirror letters.



< 1 2 3 4 5 6 7 8 9 >

see all slides

# Phylogenetic Tree Visualization Software

- **Web Portals:**

- Phylo.IO @ <http://phylo.io>
- Interactive Tree of Life viewer (iTOL) @ <http://itol.embl.de>
- EVOLVIEW @ <http://www.evolgenius.info/evolview>

- **Software:**

- FigTree @ <http://tree.bio.ed.ac.uk/software/figtree/>
- DensiTree @ <https://www.cs.auckland.ac.nz/~remco/DensiTree/>
- TreeGraph2 @ <http://treegraph.bioinfweb.info>

- **Toolkits:**

- PYTHON library: ETE toolkit @ <http://etetoolkit.org>
- R repository: PhyTools @ <https://cran.r-project.org/web/packages/phytools/index.html>
- JavaScript library: jsPhyloSVG @ <http://www.jsphylosvg.com>

- [https://en.wikipedia.org/wiki/List\\_of\\_phylogenetic\\_tree\\_visualization\\_software](https://en.wikipedia.org/wiki/List_of_phylogenetic_tree_visualization_software)

# Phylogenetic Tree Visualization Software

- **Web Portals:**

- Phylo.IO @ <http://phylo.io>
- Interactive Tree of Life viewer (iTOL) @ <http://itol.embl.de>
- EVOLVIEW @ <http://www.evolgenius.info/evolview>

- **Software:**

- FigTree @ <http://tree.bio.ed.ac.uk/software/figtree/>
- DensiTree @ <https://www.cs.auckland.ac.nz/~remco/DensiTree/>
- TreeGraph2 @ <http://treegraph.bioinfweb.info>

- **Toolkits:**

- PYTHON library: ETE toolkit @ <http://etetoolkit.org>
- R repository: PhyTools @ <https://cran.r-project.org/web/packages/phytools/index.html>
- JavaScript library: jsPhyloSVG @ <http://www.jsphylosvg.com>

- [https://en.wikipedia.org/wiki/List\\_of\\_phylogenetic\\_tree\\_visualization\\_software](https://en.wikipedia.org/wiki/List_of_phylogenetic_tree_visualization_software)

# Web Portals

- **Phylo.IO** @ <http://phylo.io>
- **Interactive Tree of Life viewer (iTOL)** @ <http://itol.embl.de>
- **EVOLVIEW** @ <http://www.evolgenius.info/evolview>
- **Others:**
  - **PHYLOViZ Online** @ <https://online.phyloviz.net/index>, (10-min YouTube tutorial: <https://www.youtube.com/watch?v=hmbmurFV6ik>)
  - **Archaeopteryx-js** @ [http://www.phyloxml.org/archaeopteryx-js/bcl2\\_js.html](http://www.phyloxml.org/archaeopteryx-js/bcl2_js.html)
  - **TreeLink** @ <http://www.treelinkapp.com>
  - **T-REX** @ <http://www.trex.uqam.ca>
  - **AQUAPONY** @ <http://www.lirmm.fr/~cazaux/AQUAPONY/>
  - etc.

# Phylo.IO

Very simple and intuitive, fast

Allow to export tree figures as images (.svg)

Quite limited

1' demo @ <https://www.youtube.com/watch?v=IOQK3CP8GIA>

# Phylo.IO demo

**Phylo.io**  
Version: 1.0.0 (04/28/2016 15:06:36)

View

Compare

Tree: *Untitled* 

*Paste your tree or drag and drop your tree file here*

Small Example Tree

Large Example Tree

Render ▶

Clear

⚙ Settings

🔗 Share

📖 Help

© Dessimoz Lab



1. Choose view for a single tree, Compare for two trees
2. Add your trees into the text boxes in newick format
3. Click "Render"

# Phylo.IO demo

Phylo.io

Version: 1.0.0 (04/28/2016 15:06:36)

View Compare

Tree: *Untitled*

```
((sba:0.0398495207632660886
1,
(sca:0.2421230719967884248
4,
(cgl:0.29957965624763355228
,
(((kla:0.3282900962783522613
7,ago:0.352170853400062044
97):0.05741546733669361524
,
```

Small Example Tree **1**

Large Example Tree

**2** Render Clear

Settings

Share

Help

Zoom:

Phylogenetic tree visualization showing relationships between various taxa (sba, sca, cgl, kla, ago, ski, kwa, yli, lel, cal, ctr, pst, pgu, dha, sce, spa, smi, sku).

Download SVG

© Dessimoz Lab

2.02

# Phylo.IO demo

Phylo.io

Version: 1.0.0 (04/28/2016 15:06:36)

ViewCompare

Tree: Untitled

```
((sba:0.0398495207632660886
1,
(sca:0.2421230719967884248
4,
(cgl:0.29957965624763355228
,
(((kla:0.3282900962783522613
7,ago:0.352170853400062044
97):0.05741546733669361524
```

Small Example Tree  
Large Example Tree

RenderClear

Settings

Share

Help

Dessimoz Lab

Zoom:

↑

↓

←

→

1

2

reroot

sba

sca

cgl

kla

ago

skl

yli

lel

cal

ctr

pst

pgu

dha

sce

spa

smi

sku

SVG

2.53

Q

# Phylo.IO demo

**Phylo.io**  
Version: 1.0.0 (04/28/2016 15:06:36)

View

Compare

Tree: *Untitled*

```
((sba:0.0398495207632660886
1,
(sca:0.2421230719967884248
4,
(cgl:0.29957965624763355228
,
(((kla:0.3282900962783522613
7,ago:0.352170853400062044
97):0.05741546733669361524
,

```

Small Example Tree  
Large Example Tree

Render ▶

Clear

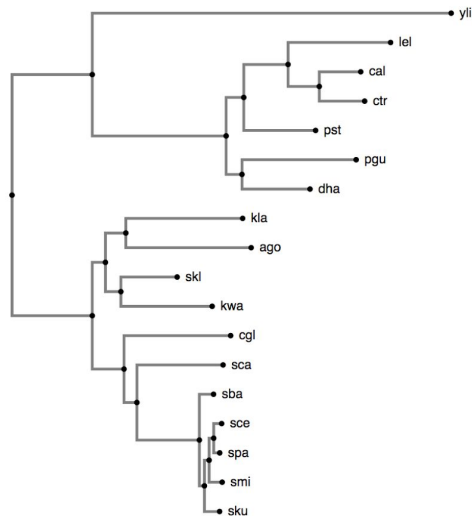
⚙ Settings

↩ Share

📖 Help

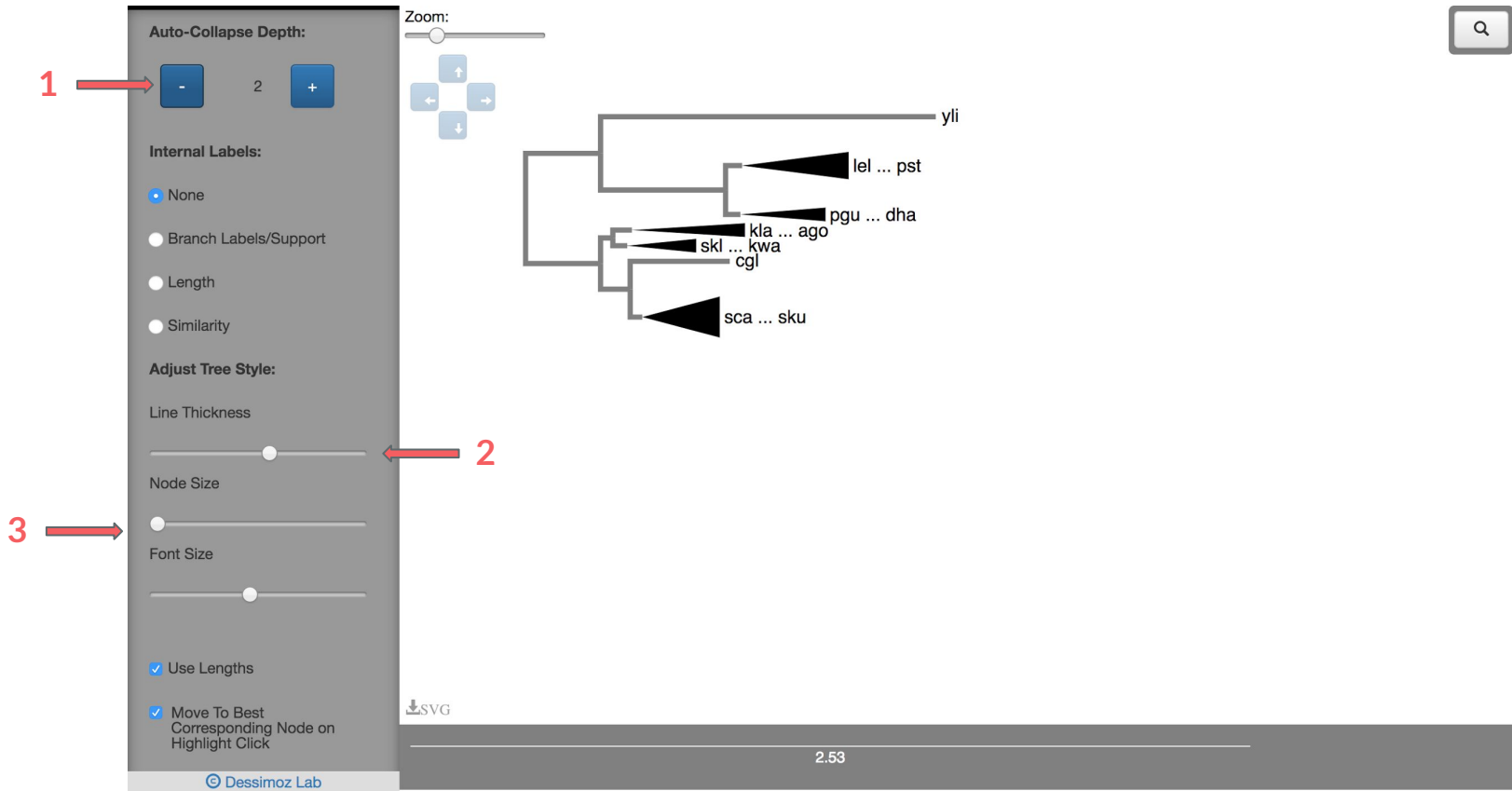
© Dessimoz Lab

Zoom:

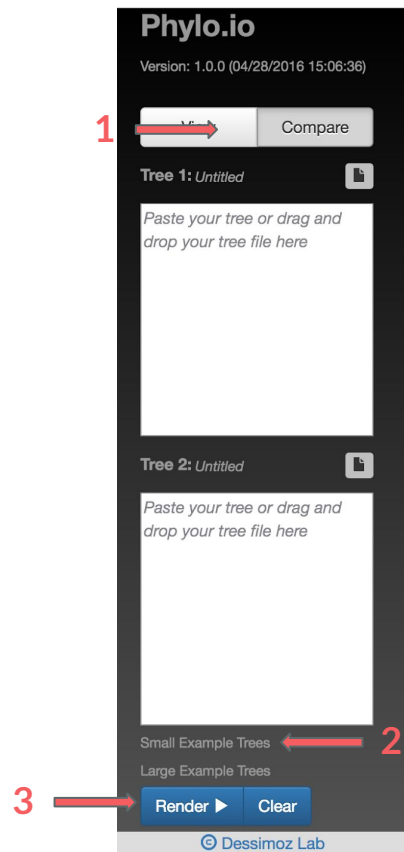


↓ SVG

# Phylo.IO demo



# Phylo.IO demo



1. Choose view for a single tree, Compare for two trees
2. Add your trees into the text boxes in newick format
3. Click "Render"

# Phylo.IO demo

**Phylo.io**  
Version: 1.0.0 (04/28/2016 15:06:36)

View Compare

Tree 1: *Untitled*

```
((yli:1.0085614391793067,
(((lel:0.2880678599618948,
(cal:0.11643570498932501,ctr
:0.12746420698315983):0.087
7413194006926):0.124498438
44102518,pst:0.20187483296
288805):0.0495118121082698
7,
(pgu:0.3208350693977453,dh
a:0.19121342958249643):0.04
```

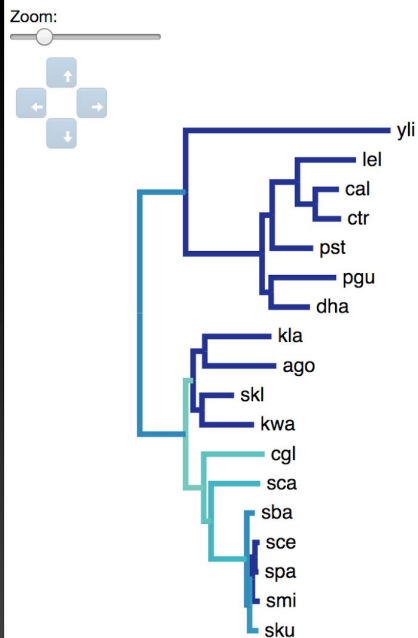
Tree 2: *Untitled*

```
((sba:0.0464593280587448156
9,
(smi:0.0331555022994616255
3,
(spa:0.0146690741917427595
2,sce:0.023837330693020668
95):0.01620853054082276729
):0.01834962617511207497):0
.06496501071301066799,
(sca:0.2192849714635205504
```

Small Example Trees  
Large Example Trees

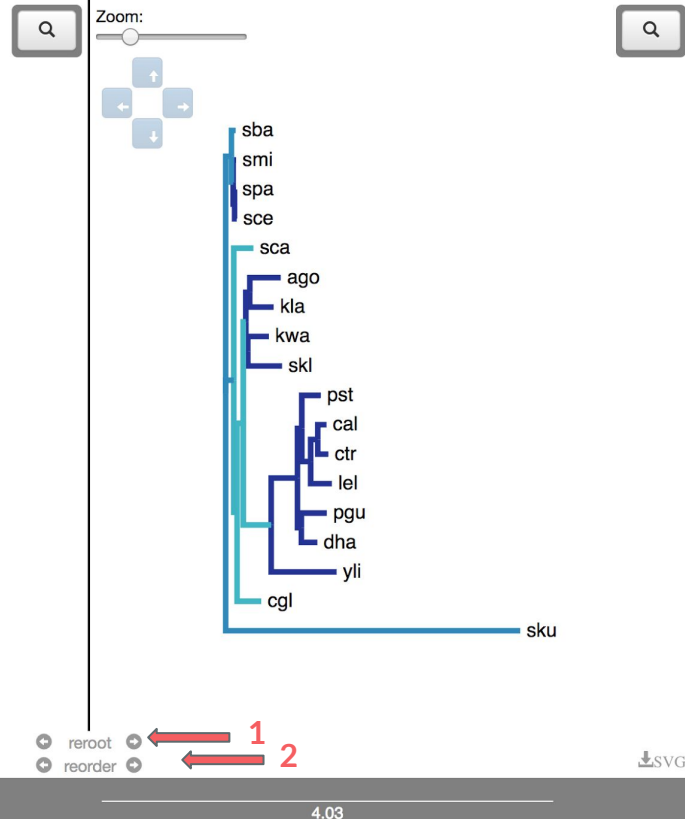
Render Clear

Dessimoz Lab

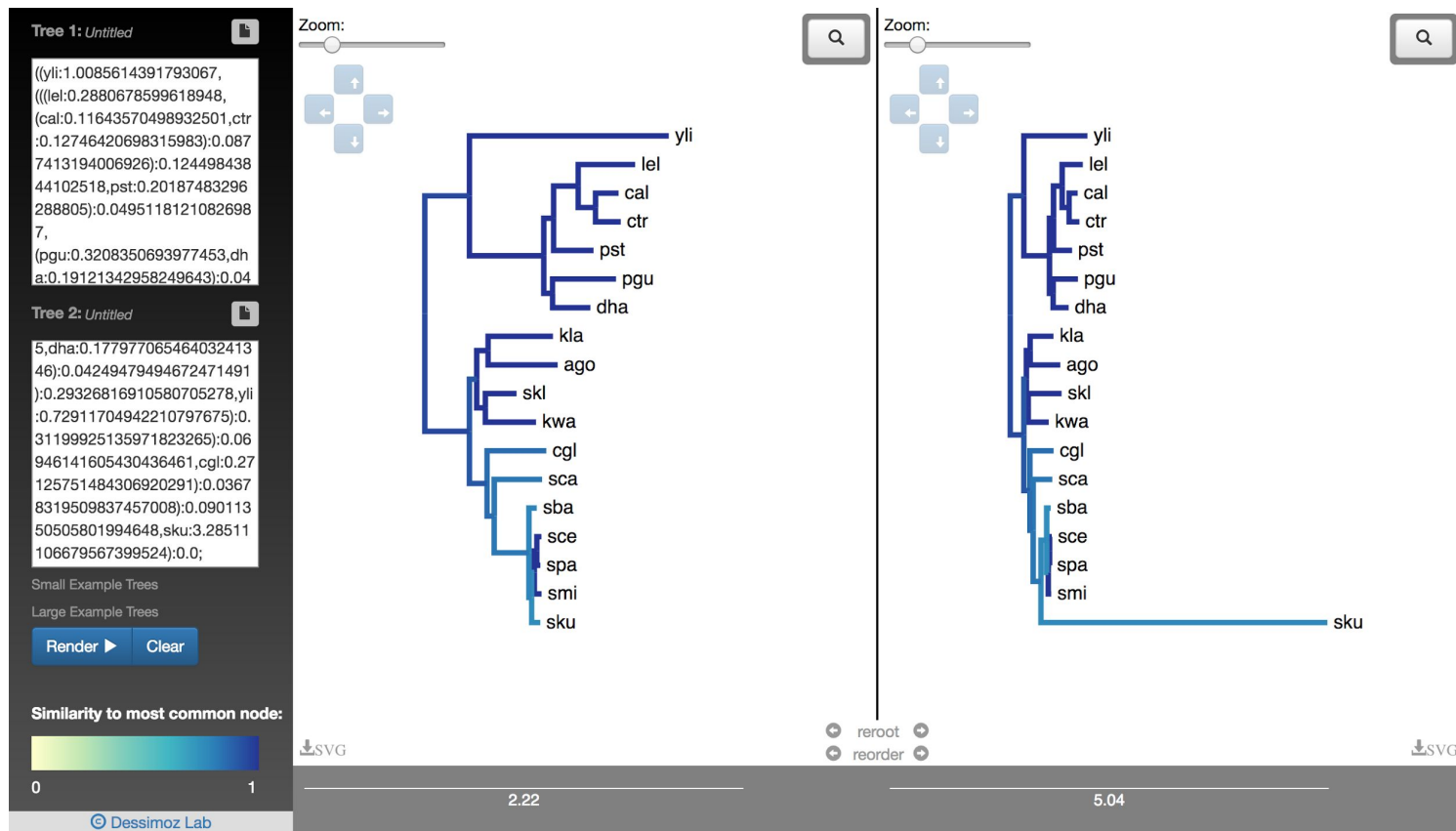


SVG

2.22



# Phylo.IO demo



# Phylo.IO demo

Phylo.IO demo interface showing two panels side-by-side, illustrating tree manipulation and visualization.

**Left Panel (Initial View):**

- Tree 2: Untitled**
- Sequence Data:**  
((sba:0.0464593280587448156  
9,  
(smi:0.0331555022994616255  
3,  
(spa:0.0146690741917427595  
2,sce:0.023837330693020668  
95):0.01620853054082276729  
):0.01834962617511207497):0  
.06496501071301066799,  
(sca:0.2192849714635205504
- Small Example Trees**
- Large Example Trees**
- Render** (blue button) **Clear** (blue button)
- Similarity to most common node:**  
0 1 (color scale bar)
- Settings** (gear icon)
- Share** (share icon)
- Dessimoz Lab**

**Right Panel (Manipulated View):**

- Tree 2: Untitled**
- Sequence Data:**  
((sba:0.0464593280587448156  
9,  
(smi:0.0331555022994616255  
3,  
(spa:0.0146690741917427595  
2,sce:0.023837330693020668  
95):0.01620853054082276729  
):0.01834962617511207497):0  
.06496501071301066799,  
(sca:0.2192849714635205504
- Small Example Trees**
- Large Example Trees**
- Render** (blue button) **Clear** (blue button)
- Similarity to most common node:**  
0 1 (color scale bar)
- Settings** (gear icon)
- Share** (share icon)
- Dessimoz Lab**

**Annotations:**

- 1** (red arrow) points to the **collapse >** button in the **Manipulated View**.
- 2** (red arrow) points to the **collapse >** button in the **Manipulated View**.
- 3** (red arrow) points to the **Share** button in the **Initial View**.

**Tree Structure (Initial View):**

- Root
- Branch 1: yli
- Branch 2: lel
- Branch 3: cal
- Branch 4: ctr
- Branch 5: pst
- Branch 6: pgu
- Branch 7: dha
- Branch 8: kla
- Branch 9: ago
- Branch 10: skl
- Branch 11: kwa
- Branch 12: cgl
- Branch 13: sca
- Branch 14: sba
- Branch 15: sce
- Branch 16: spa
- Branch 17: smi
- Branch 18: sku

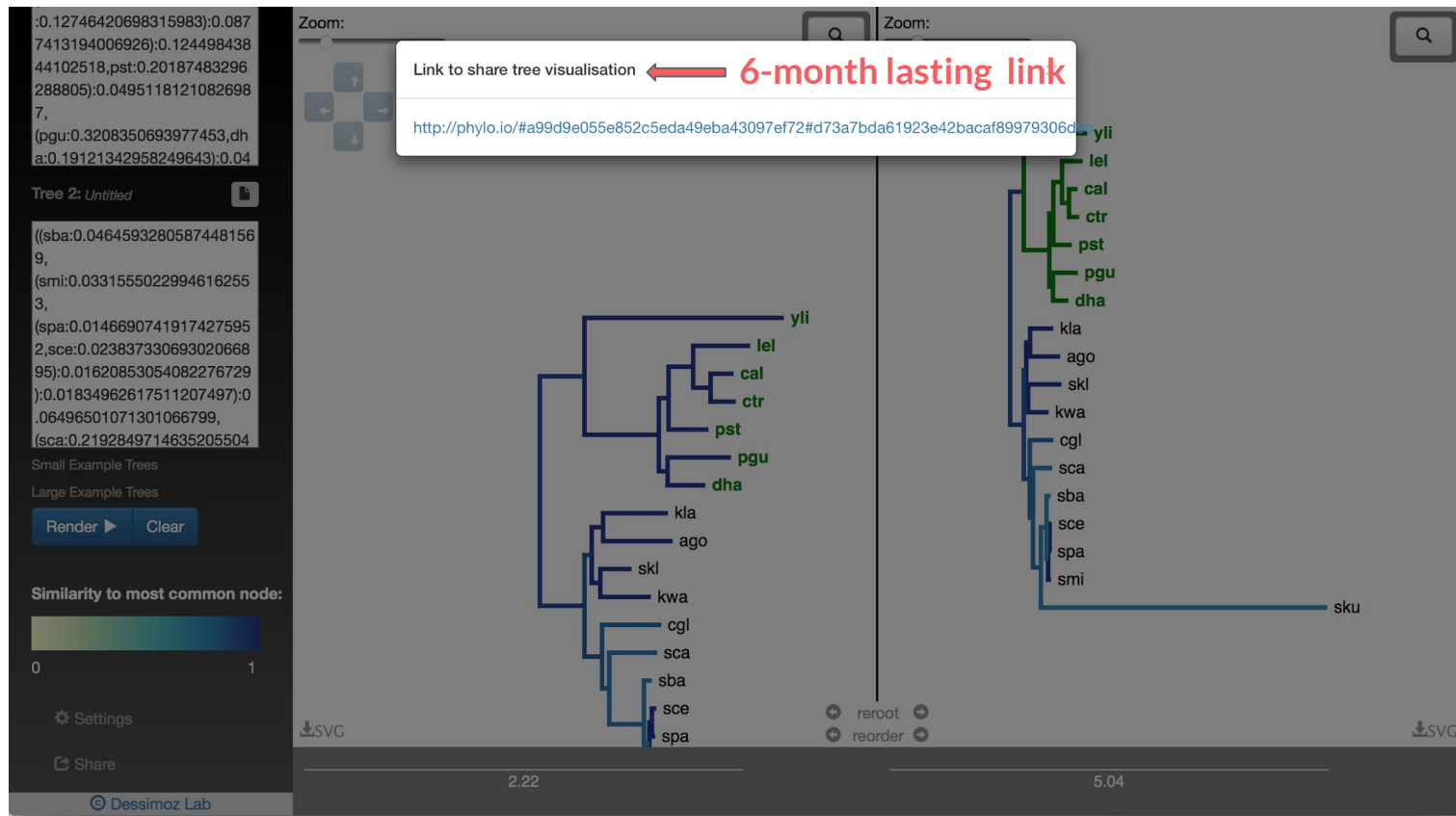
**Tree Structure (Manipulated View):**

- Root
- Branch 1: yli
- Branch 2: lel
- Branch 3: cal
- Branch 4: ctr
- Branch 5: pst
- Branch 6: pgu
- Branch 7: dha
- Branch 8: kla
- Branch 9: ago
- Branch 10: skl
- Branch 11: kwa
- Branch 12: cgl
- Branch 13: sca
- Branch 14: sba
- Branch 15: sce
- Branch 16: spa
- Branch 17: smi
- Branch 18: sku

**Buttons:**

- Zoom:** (slider)
- Search:** (magnifying glass icon)
- reroot** (blue button)
- reorder** (blue button)
- SVG** (download icon)

# Phylo.IO demo

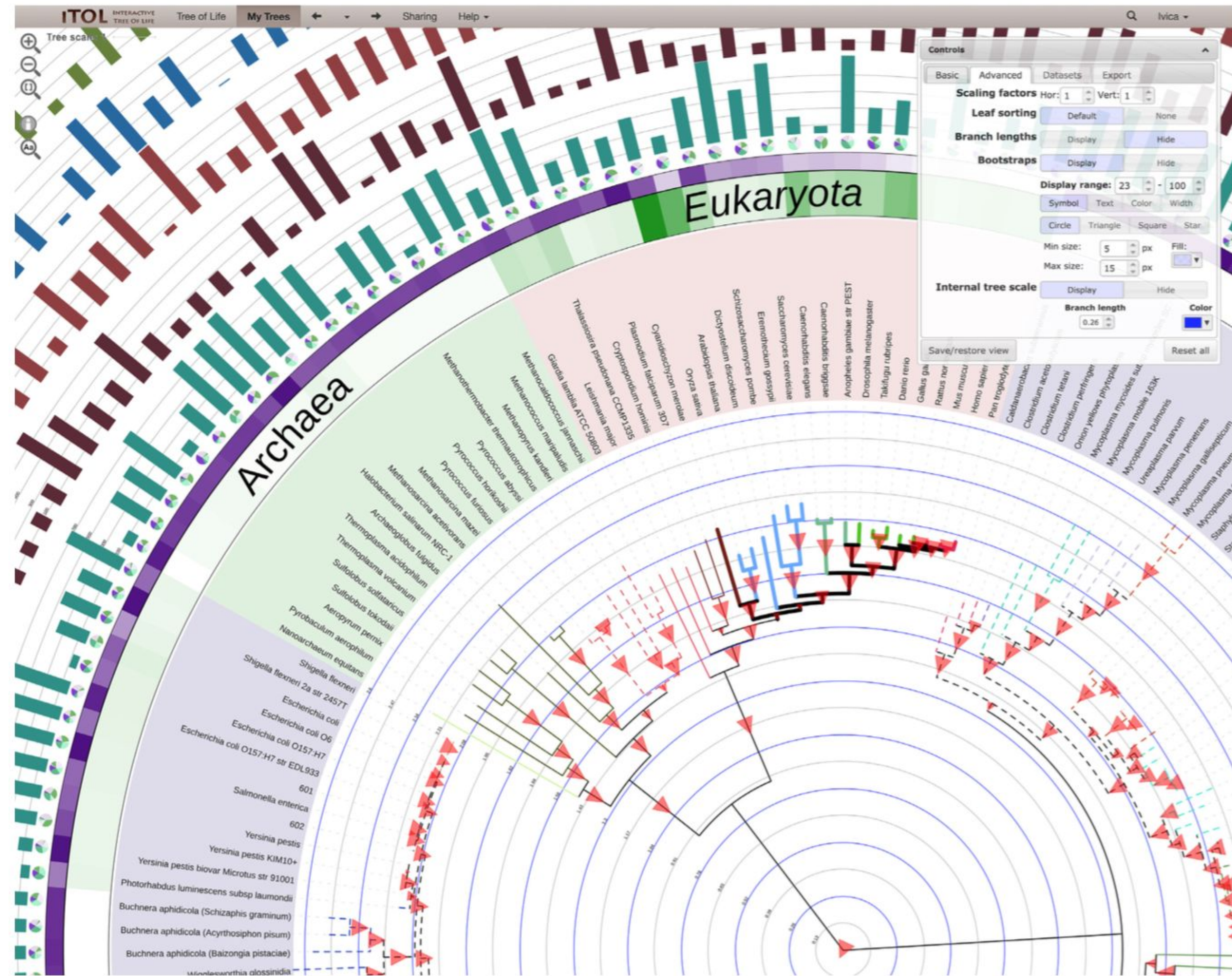


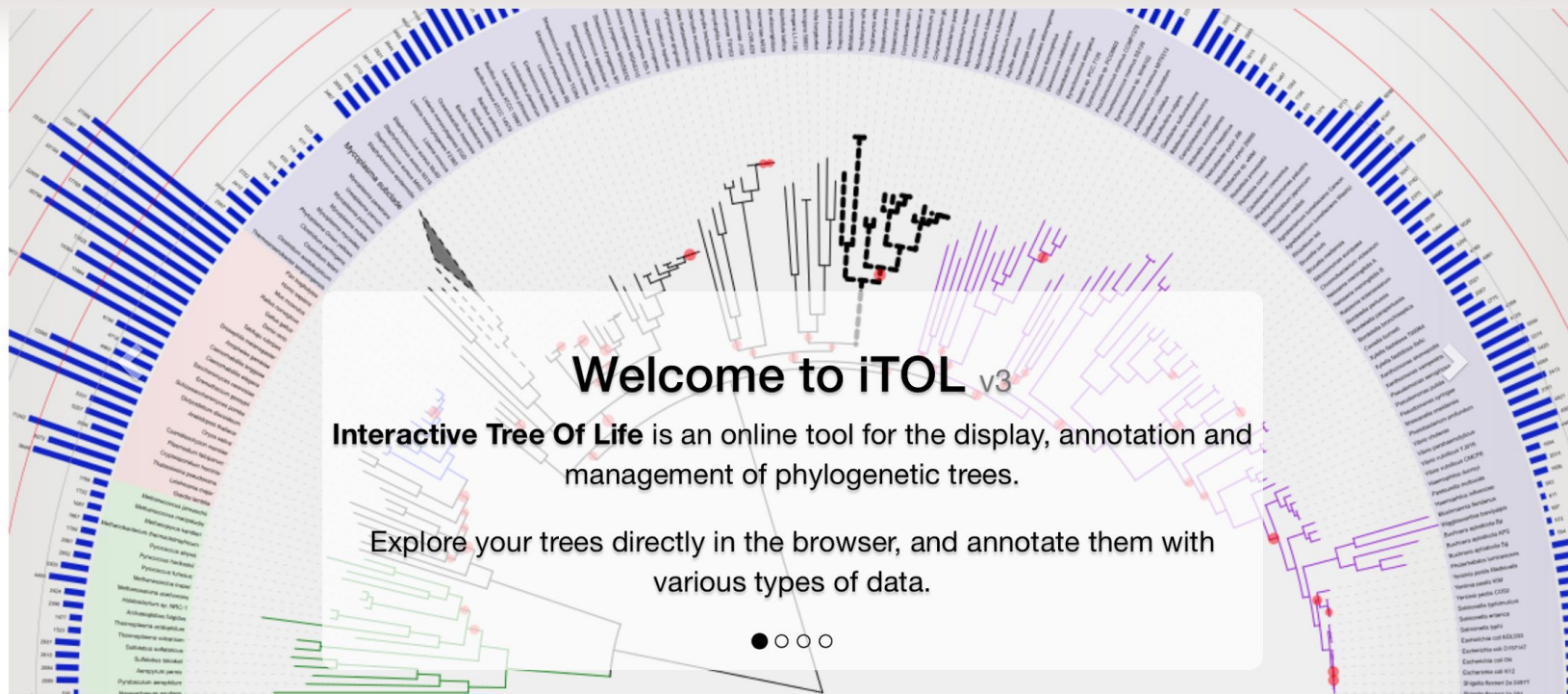
# Web Portals

- Phylo.IO @ <http://phylo.io>
- Interactive Tree of Life viewer (iTOL) @ <http://itol.embl.de>
- EVOLVIEW @ <http://www.evolgenius.info/evolview>
- Others:
  - PHYLOViZ Online @ <https://online.phyloviz.net/index>, (10-min YouTube tutorial: <https://www.youtube.com/watch?v=hmbmurFV6ik>)
  - Archaeopteryx-js @ [http://www.phyloxml.org/archaeopteryx-js/bcl2\\_js.html](http://www.phyloxml.org/archaeopteryx-js/bcl2_js.html)
  - TreeLink @ <http://www.treelinkapp.com>
  - T-REX @ <http://www.trex.uqam.ca>
  - AQUAPONY @ <http://www.lirmm.fr/~cazaux/AQUAPONY/>
  - etc.

# iTOL

- Web based
- Highly customizable/programmable
- Interactive interface
- Able to export trees in several formats
- Tree storing allowed
- Able to load really large trees





# iTOL demo – Uploading Data

1. Go to <http://itol.embl.de/>
2. If you wish to save your trees from session to session you may register
3. Go to Upload. There, you can copy our\_tree.nw or save it locally and upload it as Newick, Nexus or PhyloXML format. Then click upload!

# our\_tree.nw

```
(Centruroides:1.866817169516780e-01,Synsphyronus:3.982413562183681e-01,((Lip  
histius:1.586096567169307e-01,(Mastigoproctus:1.453966099439421e-01,Damon:1.  
332103864420345e-01)1.00:1.959232427470038e-02)1.00:2.980038140121848e-02,((  
Limulus:1.976959421750321e-01,(Peripatopsis_long_iso:3.261872681877993e-01,S  
cutigera:2.212767162268583e-01)1.00:1.057606285127356e-01)1.00:1.75486204668  
1610e-02,((Eremobates:1.924502405585649e-01,(Ixodes:3.038900301437783e-01,Te  
tranychus:7.026474543095570e-01)0.93:3.270977853676584e-02)0.93:2.1597666642  
09322e-02,(Metasiro:2.402845336869420e-01,((Ricinoides:1.810767667451328e-02  
,R_karschii:1.949305408859397e-02)1.00:2.919091302746515e-02,(Pseudocellus:4  
.098385269473430e-02,(Cryptocellus_Bocas:2.548915750691317e-02,Cryptocellus_  
becki:2.383929304335081e-02)1.00:2.194619857092223e-02)1.00:1.13747331795313  
1e-02)1.00:1.342660962530040e-01)1.00:1.510542954930570e-02)0.93:1.019147450  
482914e-02)0.93:1.829267541645866e-02)0.93:2.236446951190263e-02);
```

# iTOL demo – Uploading Data

Use this page to upload and visualize a new phylogenetic tree. It should be in a plain text file and in a supported format (Newick, Nexus or PhyloXML). You can also use *.jplace* files generated by RaxML or pplacer. Please check the [help pages](#) for detailed instructions.

Trees uploaded anonymously will be stored for 30 days, and are not protected from modifications by other users. If you want to keep them private and protected, or have multiple trees to visualize, we recommend creating an [iTOL personal account](#). If you already have an account, please [login first](#).

Datasets and other annotation should be dragged and dropped directly onto the interactive tree. Please check the [help pages](#) for detailed instructions and dataset template files.

## Upload a new tree

Tree name:

optional

Paste your tree into the box below, or select a file using the Tree file selector. You can also simply drag and drop the tree file onto the page.

Tree text:

Tree file:

Seleccionar archivo

Ningún archivo seleccionado

Upload

If you have trouble uploading your tree or datasets, please [contact us](#).

Upload  
your tree  
here

Paste your  
tree here

# iTOL demo – Control Panel

Once in the graphical interface we can go to control panel where we can change different representation values:

- Normal, circular or unrooted trees
- Branch length
- Support
- Tree scale
- Export image
- Datasets (More about this later)
- Others (Try toying around!)

# iTOL demo – Control Panel

The image displays the iTOL (Interactive Tree Of Life) web interface. The top navigation bar includes the iTOL logo, "Tree of Life", "Upload", "Sharing", "Help", and a "Login" link. On the left, a vertical toolbar contains icons for zooming in (+), zooming out (-), information (?), and text formatting (Aa). The main area shows a phylogenetic tree with a "Tree scale: 0.1" slider at the top left. The tree is rooted and has a horizontal orientation. To the right of the tree, a list of taxa is displayed, including: Centruroides, Synsphyronus, Liphistius, Mastigoproctus, Damon, Limulus, Peripatopsis long iso, Scutigera, Eremobates, Ixodes, Tetranychus, Metasiro, Ricinoides, R karschii, Pseudocellus, Cryptocellus Bocas, and Cryptocellus becki. On the right side, a "Controls" panel is open, showing various settings for the tree display. The "Basic" tab is selected, and the "Display mode" is set to "Normal". Other settings include "Parameters" (0° rotation), "Invert" (No), "Branch lengths" (Use), "Labels" (Aligned), "Label shift" (0), "Dashes" (On), "Label font" (Arial), "Font style" (20 px), "Branch lines" (1 px), and "Reset all" button.

**iTOL** INTERACTIVE TREE OF LIFE    Tree of Life    Upload    Sharing    Help    Login

Tree scale: 0.1

Centruroides  
Synsphyronus  
Liphistius  
Mastigoproctus  
Damon  
Limulus  
Peripatopsis long iso  
Scutigera  
Eremobates  
Ixodes  
Tetranychus  
Metasiro  
Ricinoides  
R karschii  
Pseudocellus  
Cryptocellus Bocas  
Cryptocellus becki

**Controls**

Basic    Advanced    Datasets    Export

**Display mode**    Circular    **Normal**    Unrooted

**Parameters**    0    ° rotation

**Invert**    Yes    **No**

**Branch lengths**    **Use**    Ignore

**Labels**    **Aligned**    At tips    Off

**Label shift**    0    **Dashes**    On    Off

**Label font**    Arial

**Font style**    20    px    B    I

**Branch lines**    1    px

Save/restore view    Reset all

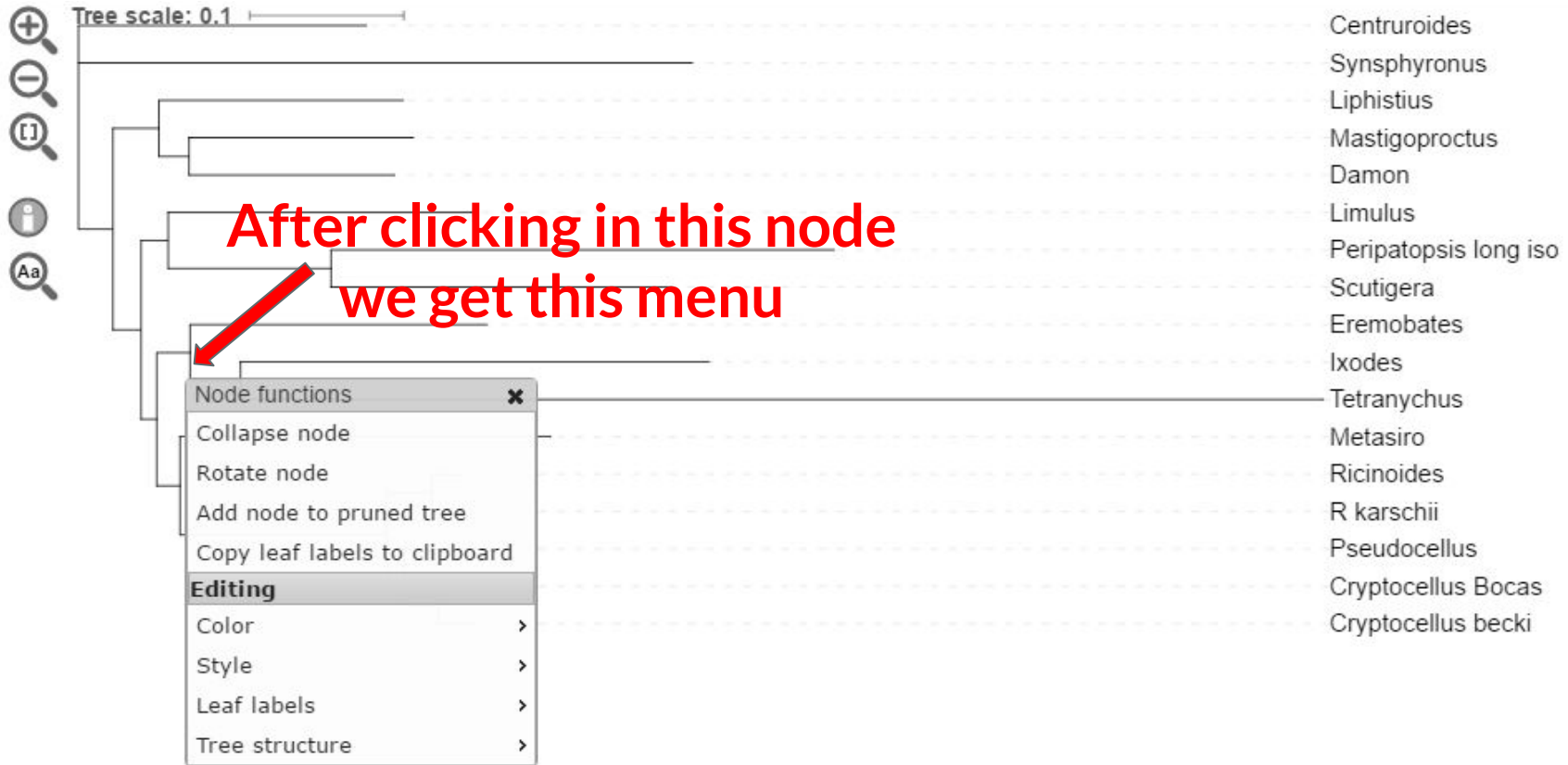
Look how many options!

# iTOL demo – Poking the Branches (& Nodes)

We can select any node or leaf by clicking on it. A new menu pops up that allows several functions. The menu is slightly different if the node is internal:

- Reroot the tree in that node
- Collapse or prune (remove) the node (only internal)
- Rotate node (only internal)
- Style:
  - Labels
  - Branch color, thickness, style...
  - Color ranges

# iTOL demo – Poking the Branches (& Nodes)



# iTOL demo – Datasets

iTOL allows a huge range of plots to compliment a tree:

<http://itol.embl.de/help.cgi#annot>

You can attach many types of data to your phylogenetic tree.

Examples:

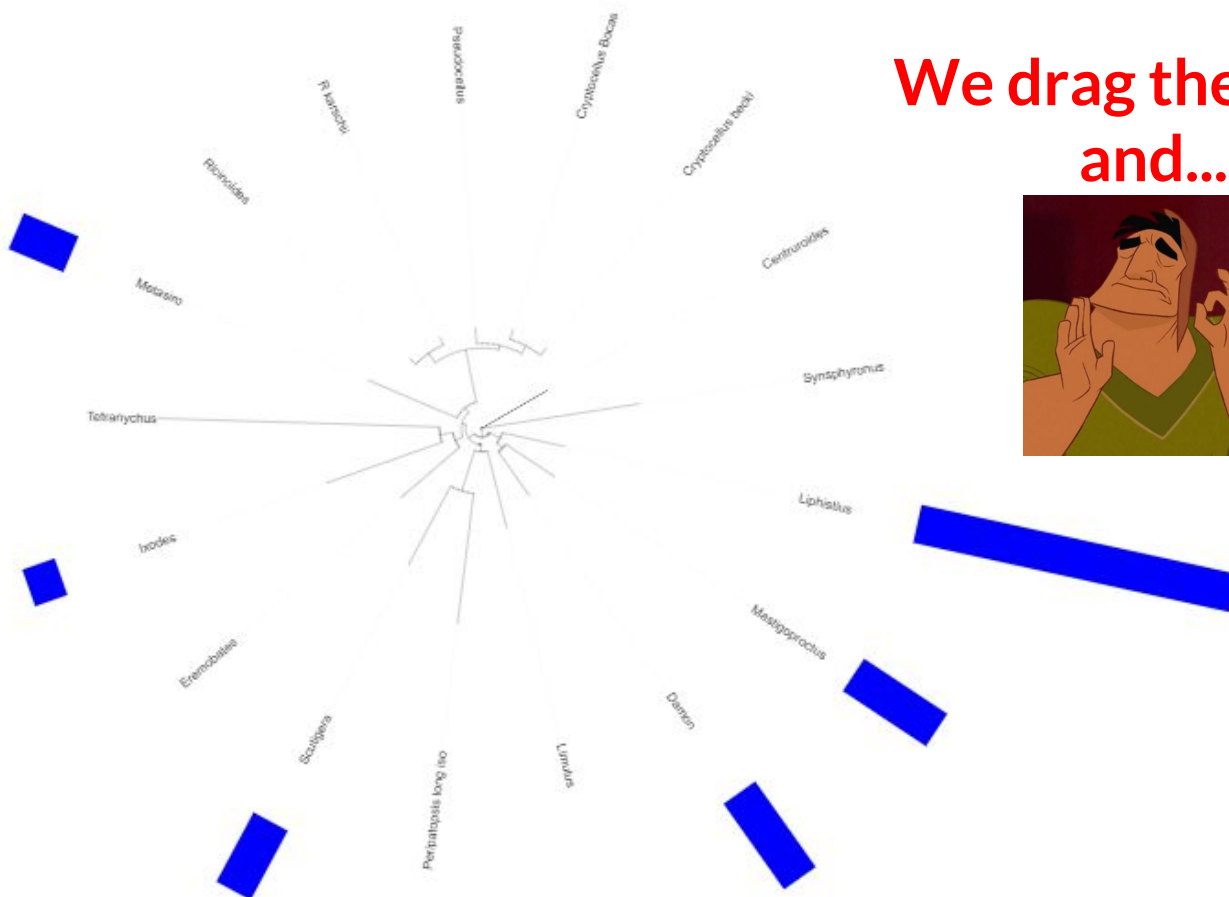
1. Histogram with the number of genes in a given family for each organism
2. Pie chart representing relative frequencies of transposon families
3. Horizontal connections between nodes to represent HGT
4. Phenotypic characters for each branch
5. Protein architecture for each branch

# iTOL demo – Datasets

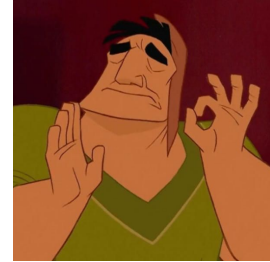
1. Dataset type ————— `DATASET_SIMPLEBAR`
2. Separator (Space, TAB or comma) ————— `SEPARATOR TAB`
3. Name of the dataset ————— `DATASET_LABEL simple_bar_testing`
4. Other characteristics ————— `COLOR #0000ff`  
— `WIDTH 1000`  
— `DATA`
5. Data (from now on we will indicate the node ID and plot values) —————  
`Ixodes 34`  
`Scutigera 78`  
`Metasiro 59`  
`Liphistius 321`  
`Damon 104`  
`Eremobates 12`  
`Mastigoproctus 99`



Tree scale: 0.1



We drag the file in  
and...



# iTOL demo – Datasets

You can add as many datasets as you want and visualize them

Once added, you can modify the global parameters through the interactive interface

A number of templates and examples are provided

Tree scale: 1



We can change color, width, etc. Now is thinner and red

### Controls

Basic

Advanced

Datasets

Export

Dataset simple\_bar\_testing

Legend color

Type Simple bar chart

Left margin 150 px

Maximum width 1400 px

Show internal values Yes No

Border style 3 px

Display value Yes No

Bar shift 0

Bar height factor 0.5 x

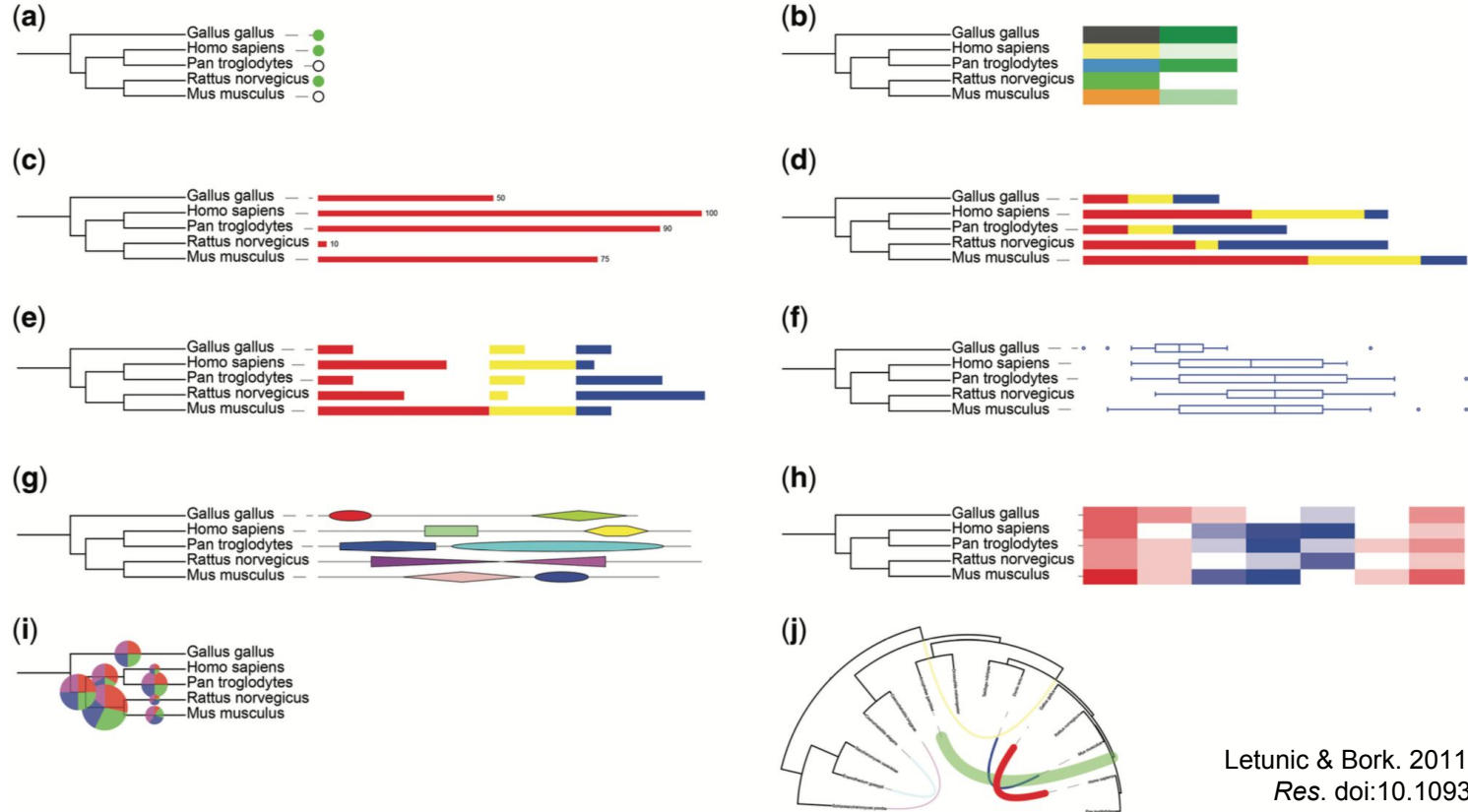
Bar zero at 0 dataset value

### Datasets

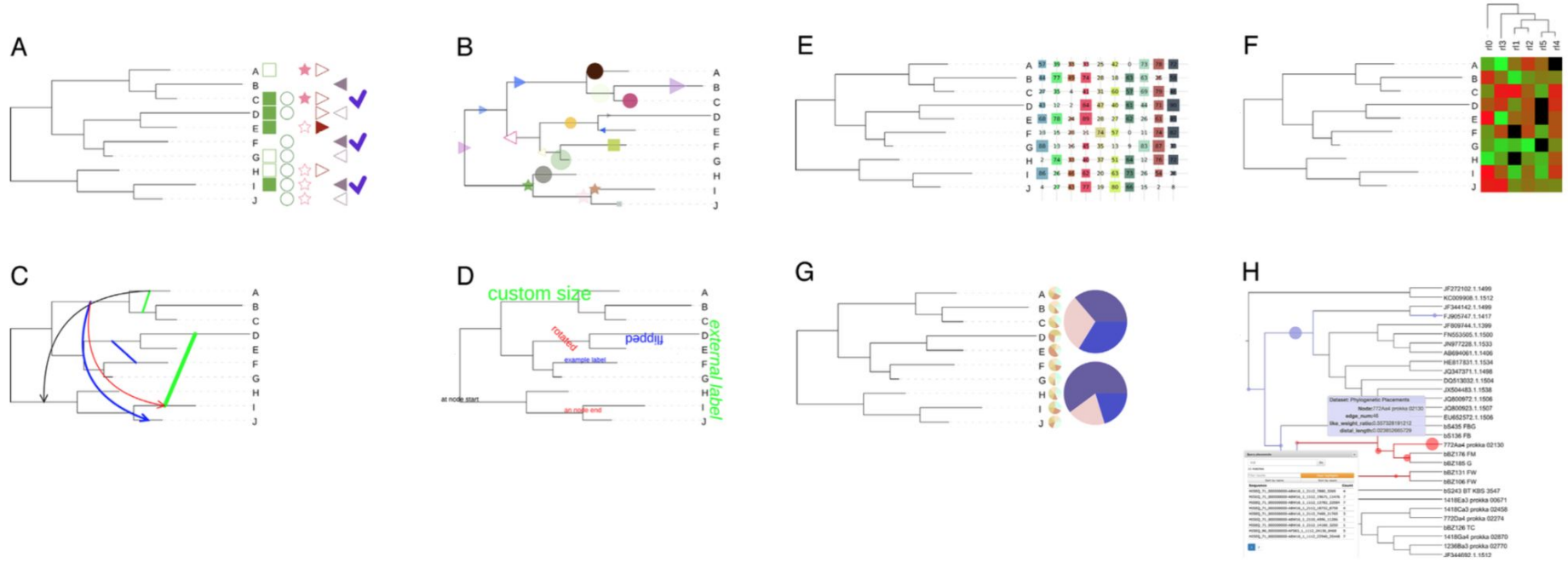
- simple\_bar\_testing ☒
- simple\_piechart\_testing ☒

Switch on-off datasets

# iTOL – Dataset Templates



# iTOL – Even More Templates



# Web Portals

- Phylo.IO @ <http://phylo.io>
- Interactive Tree of Life viewer (iTOL) @ <http://itol.embl.de>
- **EVOLVIEW** @ <http://www.evolgenius.info/evolview>
- Others:
  - PHYLOViZ Online @ <https://online.phyloviz.net/index>, (10-min YouTube tutorial: <https://www.youtube.com/watch?v=hmbmurFV6ik>)
  - Archaeopteryx-js @ [http://www.phyloxml.org/archaeopteryx-js/bcl2\\_js.html](http://www.phyloxml.org/archaeopteryx-js/bcl2_js.html)
  - TreeLink @ <http://www.treelinkapp.com>
  - T-REX @ <http://www.trex.uqam.ca>
  - AQUAPONY @ <http://www.lirmm.fr/~cazaux/AQUAPONY/>
  - etc.

# EVOLVIEW

Web based

Interactive interface

Exporting tree images in several formats allowed

Highly customizable, programmable

Trees can be stored

Claims to be able to operate with thousands of leaves/tips

# EVOLVIEW



|                     |      |
|---------------------|------|
| DEMOS (16 trees)    | +    |
| SHOWCASES (7 trees) | -    |
| showcase 01         | ⚙️ + |
| showcase 02         | ⚙️ + |
| showcase 03         | ⚙️ + |
| showcase 04         | ⚙️ + |
| showcase 05         | ⚙️ + |
| ss                  | ⚙️ + |
| shmbian             | ⚙️ - |

Basic Advanced Annotation upload Export

# EVOLVIEW



## DEMOS (16 trees)

yeast duplications



ensembl compara 2011



a demo of datasets



bars



leaf decoration and stroke



protein domains



group label



group label style 3



group label style 4



dotplots



Charts



☒ with strokes



☒ rect without strokes



heatmaps



column plots



bootstrap value styles



parentheses in leaf names



multiple bootstrap values



TimeLine



## SHOWCASES (7 trees)

Basic Advanced Annotation upload Export

pdf png jpeg tiff nwk nxs nhx xml

### Example of dot plot

dot 1

dot 2

dot 3

4

8

12

16

### rect dotplots without strokes

dataset 1

dataset 2

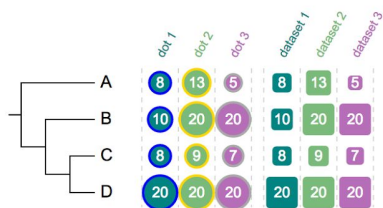
dataset 3

4

8

12

16



# EVOLVIEW demo

## A quick start

In this "quick start", we'll show you step-by-step how to use this tool to display and customise a tree:

- add a new tree (and a new project),
- change the branch and leaf colors of the tree by uploading some datasets,
- add two extra datasets to the tree,
- play around with the tree and its uploaded datasets, and
- export the customized tree as a pdf file as well as other formats

## Add a new tree to a new project

- at the "login" page, click "Use without an account"



Create an account Use without an account

Sign in with EvolGenius Account

Email:

Password:

Sign in ☒ Stay signed in

[Forgot your password?](#)

(beta) Sign in with

# EVOLVIEW demo

- at the "mytrees" page, click the folder icon on the "Basic" tab (the only icon that's clickable if no tree is active). in the popup window, enter the following contents and click the button "Submit" at the bottom:

Project: new project

Name: test tree

Data: (chicken, ((mouse, rat), (chimp, human)));



The screenshot shows the 'Upload a new tree' dialog box in the EVOLVIEW application. The dialog has a title bar with tabs: 'Basic', 'Advanced', 'Annotation upload', and 'Export'. Below the tabs is a toolbar with various icons. The main content area contains the following fields and controls:

- Project:** A text input field containing 'new project'. A red arrow labeled '2' points to this field.
- Name:** A text input field containing 'test tree'. A red arrow labeled '3' points to this field.
- Format:** A dropdown menu showing 'newick (also phylip)'.
- Data:** A text area containing the Newick string: '(chicken, ((mouse, rat), (chimp, human)));'. A red arrow labeled '4' points to this text area.

At the bottom of the dialog, there is a section labeled 'Or upload:' with a button 'Seleccionar archivos' and the text 'ningún archivo seleccionado'. At the very bottom right, there are two buttons: 'Cancel' and 'Submit'.

# EVOLVIEW demo

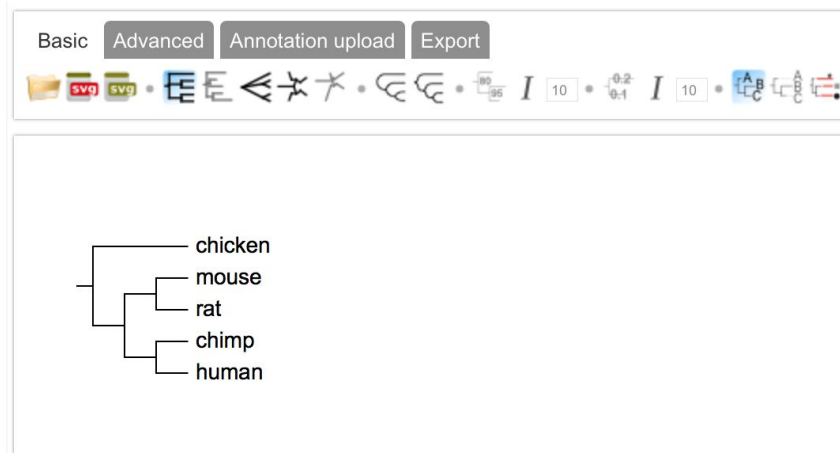
- at the "mytrees" page, click the folder icon on the "Basic" tab (the only icon that's clickable if no tree is active). in the popup window, enter the following contents and click the button "Submit" at the bottom:

Project: new project

Name: test tree

Data: (chicken, ((mouse, rat), (chimp, human)));

- the tree will be shown as the following if there is no error in your input:



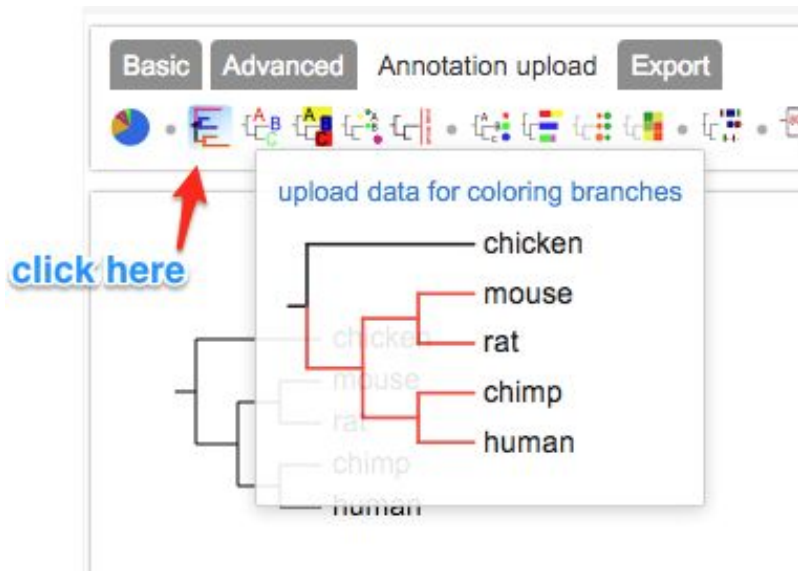
# EVOLVIEW demo

## Change the branch and leaf colors of the tree

in the following we'll add several datasets to change the appearances of the tree.  
Let's use the same tree again:

```
(chicken, ((mouse, rat), (chimp, human)))
```

- First of all add a dataset named "branch colors" to change the colors of the branches by clicking the "branch colors" icon on the "Annotation upload" tab:



# EVOLVIEW demo

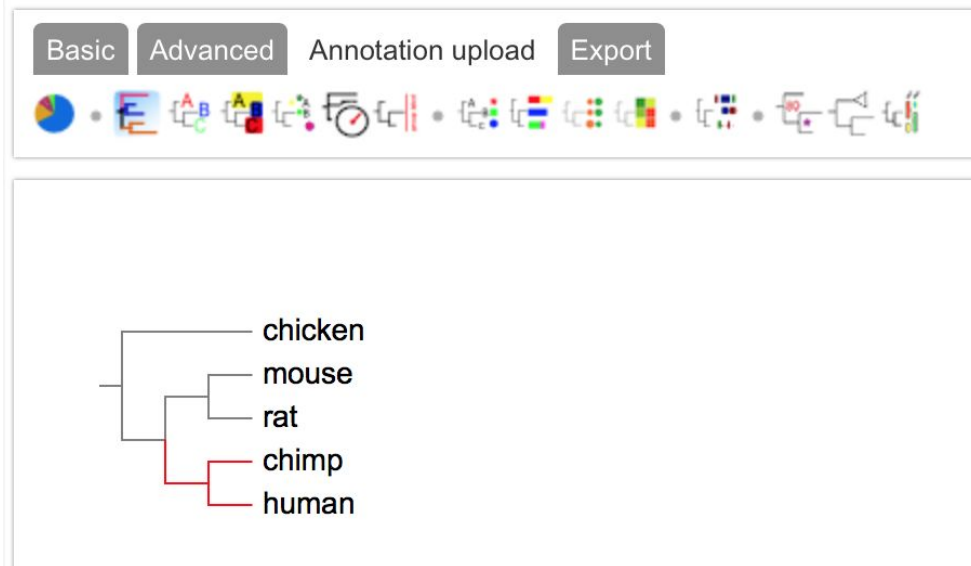
## lines start with # are annotations; you can put an annotation line anywhere in this dataset, provided that  
## the # is the first character of the line let the dataset begin: all fields of a line are separated by 'tab'  
## first all, color all branches with 'grey'

```
human,chicken    grey  ad
```

## and then highlight the branches connecting human and chimp

```
human,chimp red   ad
```

- After applying the dataset:



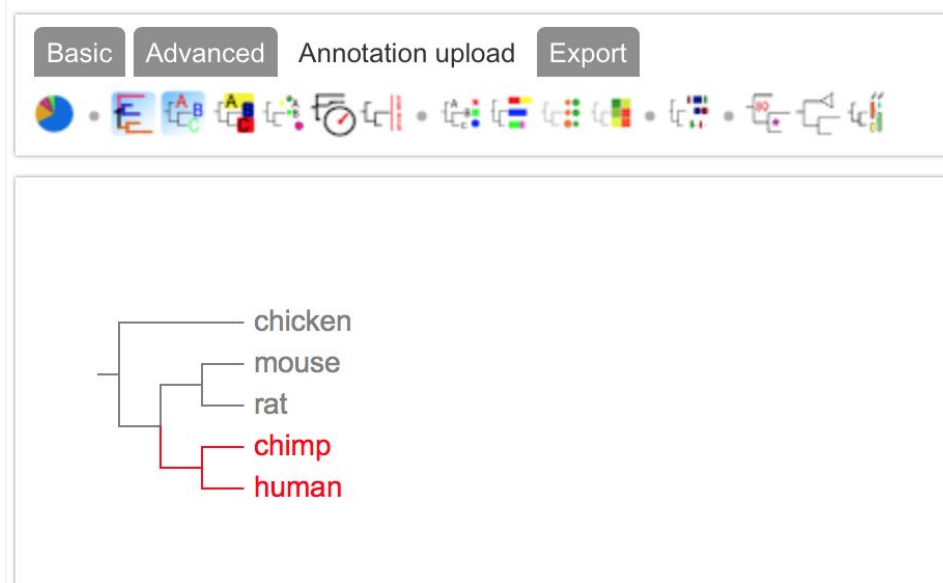
# EVOLVIEW demo

- then add another dataset to change colors of the tree leaves to match the colors of their corresponding branches by clicking the "tree leaf colors" icon on the "Annotation upload" tab:

```
## Let the dataset begin
## Each line of the 'data part' consists three fields sparated by a 'tab' character:
## - The first field specifies the location on the tree; for example 'human' indicates the leaf node
##   representing 'human' or the branch connecting to this leaf node, while 'human,mouse' indicates
##   the internal node representing the last common ancestor (LCA) of human and mouse
## - The second field specifies the color to be applied to the corresponding nodes / branches
## - The third field is optional; it can be one the following 'key-words':
##       ad: the color will be applied to all decendents of the node specified by the first field
##       prefix: the color will be applied to all leaf nodes (or connecting branches) whose names
##               start with the string in the first field
##       suffix: the color will be applied to all leaf nodes (or connecting branches) whose names
##               end with the string in the first field
##       anywhere: the color will be applied to all leaf nodes (or connecting branches) whose names
##               contain the string in the first field
## first all, color all leaves to 'grey'
human,chicken      grey  ad
## then, highlight human and chimp with 'red'
human,chimp red    ad
```

# EVOLVIEW demo

- then add another dataset to change colors of the tree leaves to match the colors of their corresponding branches by clicking the "tree leaf colors" icon on the "Annotation upload" tab:



# EVOLVIEW demo

## Add pie charts to the tree

From this section we'll add some more complicated datasets to the tree.

First of all, some pie charts; the pies will be displayed on the tree branches.

- click the "piechart" icon on the "Annotation upload" tab:
- copy and paste the following to the text box of the popup dialog box:

```
## Created on May 25, 2011
## Before the 'data section', now we have a new section that contains lines starting with '!'; this section
## is called the 'modifier section' because it's used to change / modify the default behaviors of the dataset
!Groups           A,B,C,D
!colors           #9F0251,#f88ef9,#effc00,#4D8963
!legendstyle      rect
!title            example pie chart
!maxradius        20
!minradius        8
!opacity          0.8
chicken,human     50,0,0,5
mouse,human       10,0,0,25
chimp,human       0,20,20,5
```

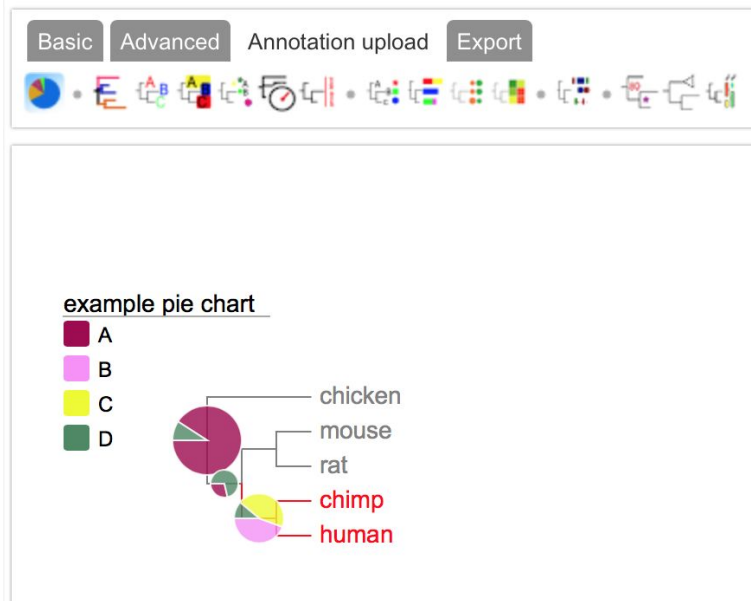
# EVOLVIEW demo

## Add pie charts to the tree

From this section we'll add some more complicated datasets to the tree.

First of all, some pie charts; the pies will be displayed on the tree branches.

- click the "piechart" icon on the "Annotation upload" tab:
- copy and paste the following to the text box of the popup dialog box:



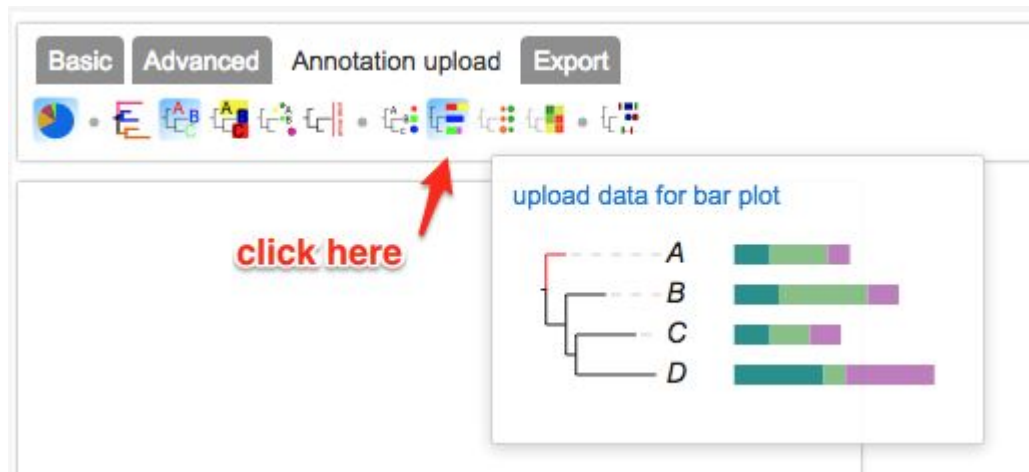
# EVOLVIEW demo

## Add bar plots to the tree (next to the leaf labels)

Here we'll add some bar plots to the tree; the bars will be plotted next to the leaf labels.

- first of all, click the "bar plots" icon on the "Annotation upload" tab:
- choose a name for the dataset and copy and paste the following to the text area in the popup dialog box:

```
##barplots
!groups      a,b,c
!colors      darkblue,darkgreen,darkred
!showLegends 0
!plotwidth   100
!align
!grid
chicken      2,3,1
mouse        8,9,2
human        20,3,4
chimp        10,20,1
rat          4,6,5
```

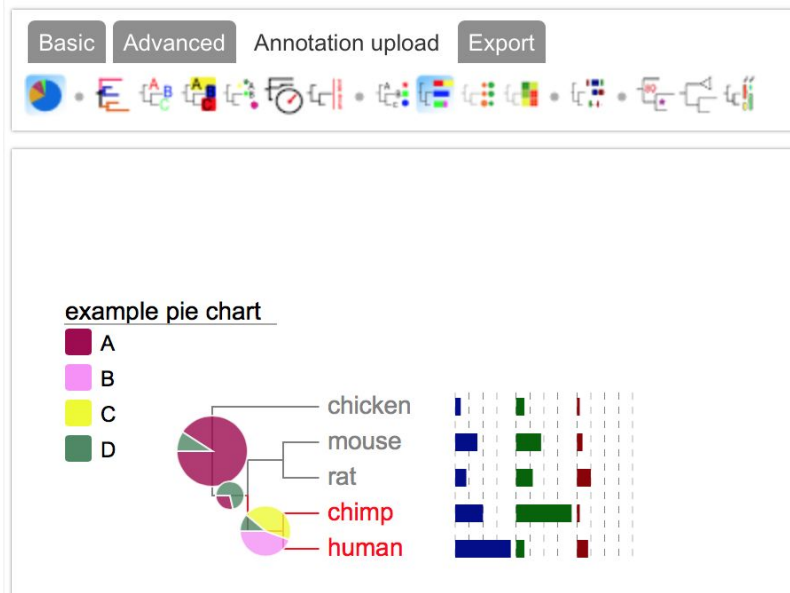


# EVOLVIEW demo

## Add bar plots to the tree (next to the leaf labels)

Here we'll add some bar plots to the tree; the bars will be plotted next to the leaf labels.

- first of all, click the "bar plots" icon on the "Annotation upload" tab:
- choose a name for the dataset and copy and paste the following to the text area in the popup dialog box:



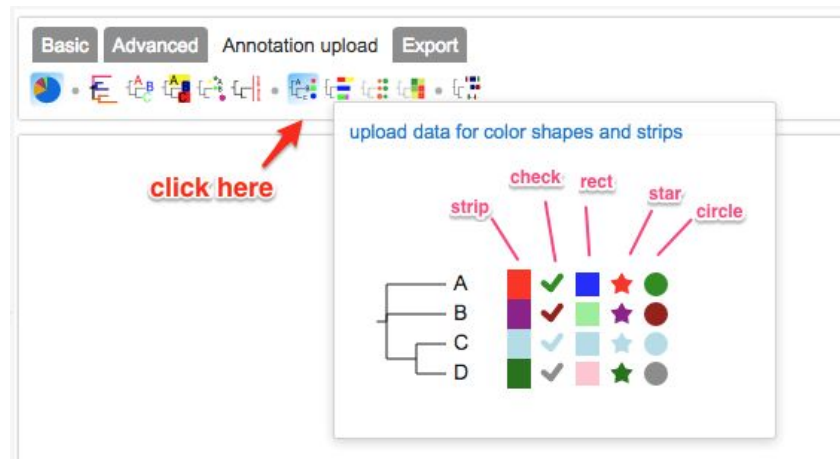
# EVOLVIEW demo

## Add colored objects/ shapes to the tree (next to the leaf labels)

Here we'll add a dataset called "color strips / colored shapes".

- click the "color strips" icon on the "Annotation upload" tab:
- copy and paste the following to the text area in the popup dialog box:

```
##color strips
!groups      a,b,c,d
!colors      blue,green,grey,red
!type        rect,circle,star,strip
!showlegends 1
human        red,green,blue,purple
chimp        purple,darkred,lightgreen,lightblue
mouse        lightblue,yellow
chicken      darkgreen,grey,pink,grey
rat          grey,orange,pink,orange
```



# EVOLVIEW demo

## Add colored objects/ shapes to the tree (next to the leaf labels)

Here we'll add a dataset called "color strips / colored shapes".

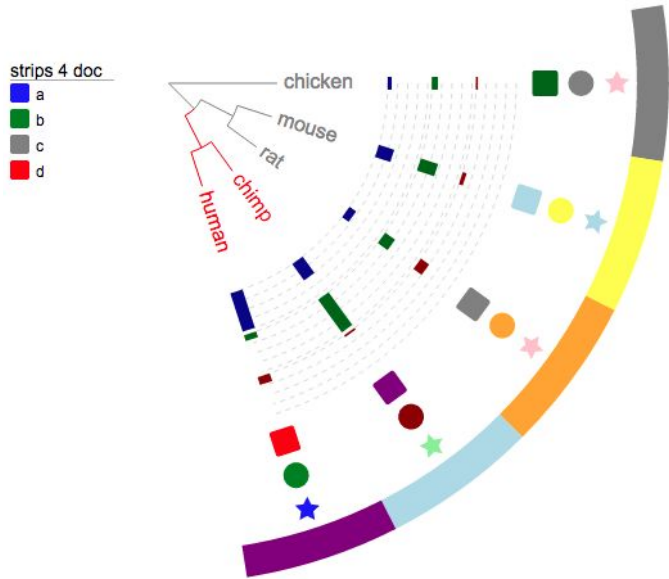
- click the "color strips" icon on the "Annotation upload" tab:
- copy and paste the following to the text area in the popup dialog box:

The screenshot shows the EVOLVIEW software interface. At the top, there are four tabs: 'Basic', 'Advanced', 'Annotation upload' (which is selected), and 'Export'. Below the tabs is a toolbar with various icons for file operations and data management. The main area of the interface displays a phylogenetic tree with leaf labels: 'chicken', 'mouse', 'rat', 'chimp', and 'human'. To the left of the tree, there are two legends. The first legend, titled 'example pie chart', shows four categories: A (purple), B (pink), C (yellow), and D (green). The second legend, titled 'color strips', shows four categories: a (blue), b (green), c (grey), and d (red). To the right of the tree, there is a grid of colored shapes (squares, circles, stars) corresponding to the categories in the legends. The shapes are arranged in a grid that matches the structure of the phylogenetic tree, with each shape representing a specific data point for a given species and category.

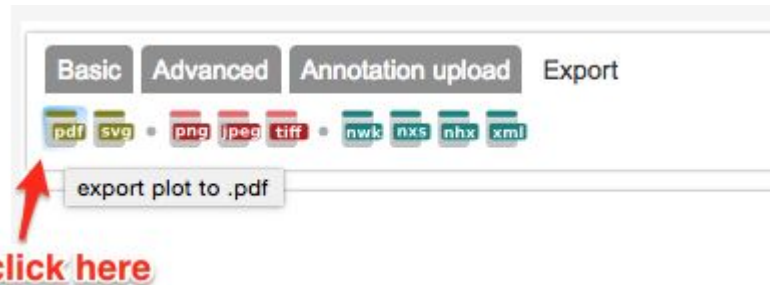
# EVOLVIEW demo

Play around and then export the tree to pdf file

- Try different display mode:



- Then download the plot as a PDF file:



# Phylogenetic Tree Visualization Software

- **Web Portals:**

- Phylo.IO @ <http://phylo.io>
- Interactive Tree of Life viewer (iTOL) @ <http://itol.embl.de>
- EVOLVIEW @ <http://www.evolgenius.info/evolview>

- **Software:**

- FigTree @ <http://tree.bio.ed.ac.uk/software/figtree/>
- DensiTree @ <https://www.cs.auckland.ac.nz/~remco/DensiTree/>
- TreeGraph2 @ <http://treegraph.bioinfweb.info>

- **Toolkits:**

- PYTHON library: ETE toolkit @ <http://etetoolkit.org>
- R repository: PhyTools @ <https://cran.r-project.org/web/packages/phytools/index.html>
- JavaScript library: jsPhyloSVG @ <http://www.jsphylosvg.com>

- [https://en.wikipedia.org/wiki/List\\_of\\_phylogenetic\\_tree\\_visualization\\_software](https://en.wikipedia.org/wiki/List_of_phylogenetic_tree_visualization_software)

# Software

- **FigTree** @ <http://tree.bio.ed.ac.uk/software/figtree/>
- **DensiTree** @ <https://www.cs.auckland.ac.nz/~remco/DensiTree/>
- **TreeGraph2** @ <http://treegraph.bioinfweb.info>
- Others:
  - **TreeView** @ <http://taxonomy.zoology.gla.ac.uk/rod/treeview.html>
  - **NJplot** @ <http://doua.prabi.fr/software/njplot>
  - **Dendroscope** @ <http://ab.inf.uni-tuebingen.de/software/dendroscope/>
  - **Spread3** @ <https://github.com/phylogeography/Spread3>
  - **SeaView** @ <http://doua.prabi.fr/software/seaview>
  - etc.

# FigTree



## FigTree

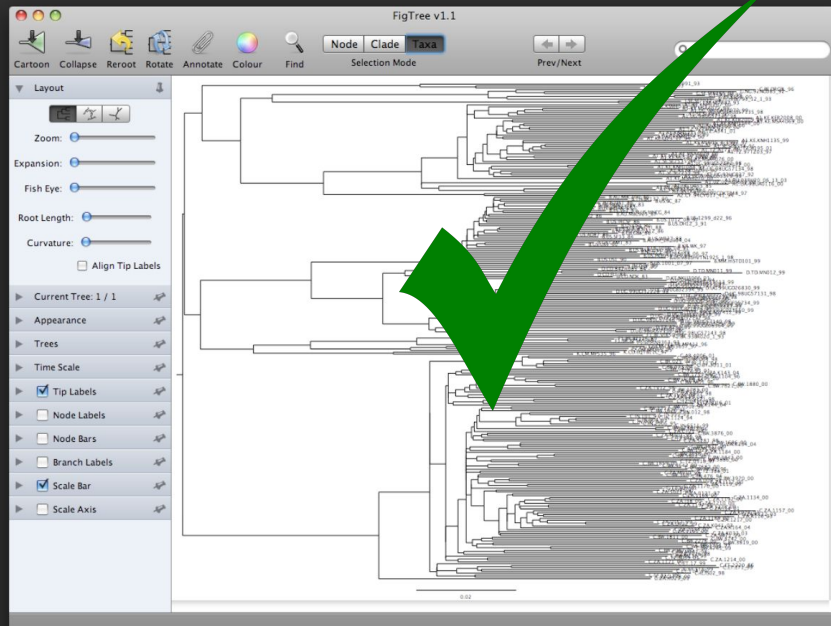
FigTree is designed as a graphical viewer of phylogenetic trees and as a program for producing publication-ready figures. As with most of my programs, it was written for my own needs so may not be as polished and feature-complete as a commercial program. In particular it is designed to display summarized and annotated trees produced by BEAST.

Subscribe to the Figtree Announcement mailing list:

Email:

[Visit this group](#)

This mailing list is for announcements of new versions of FigTree only.



## FigTree



Latest Version - v1.4.3

Bug fixes

Downloads

 **FigTree v1.4.3.dmg**  
Macintosh OS X executable version. This requires Java 1.6 or better which is installed by Mac OS X Snow Leopard (10.6).

Downloads: 1

 **FigTree v1.4.3.zip**  
Windows executable version. This is a ZIP archive.

Downloads: 1

 **FigTree\_v1.4.3.tgz**  
Java executable version. Will run on any system with Java 1.5 or better installed, including Linux and Mac OS X. This archive is tarred and gzipped.

Downloads: 1

New features:

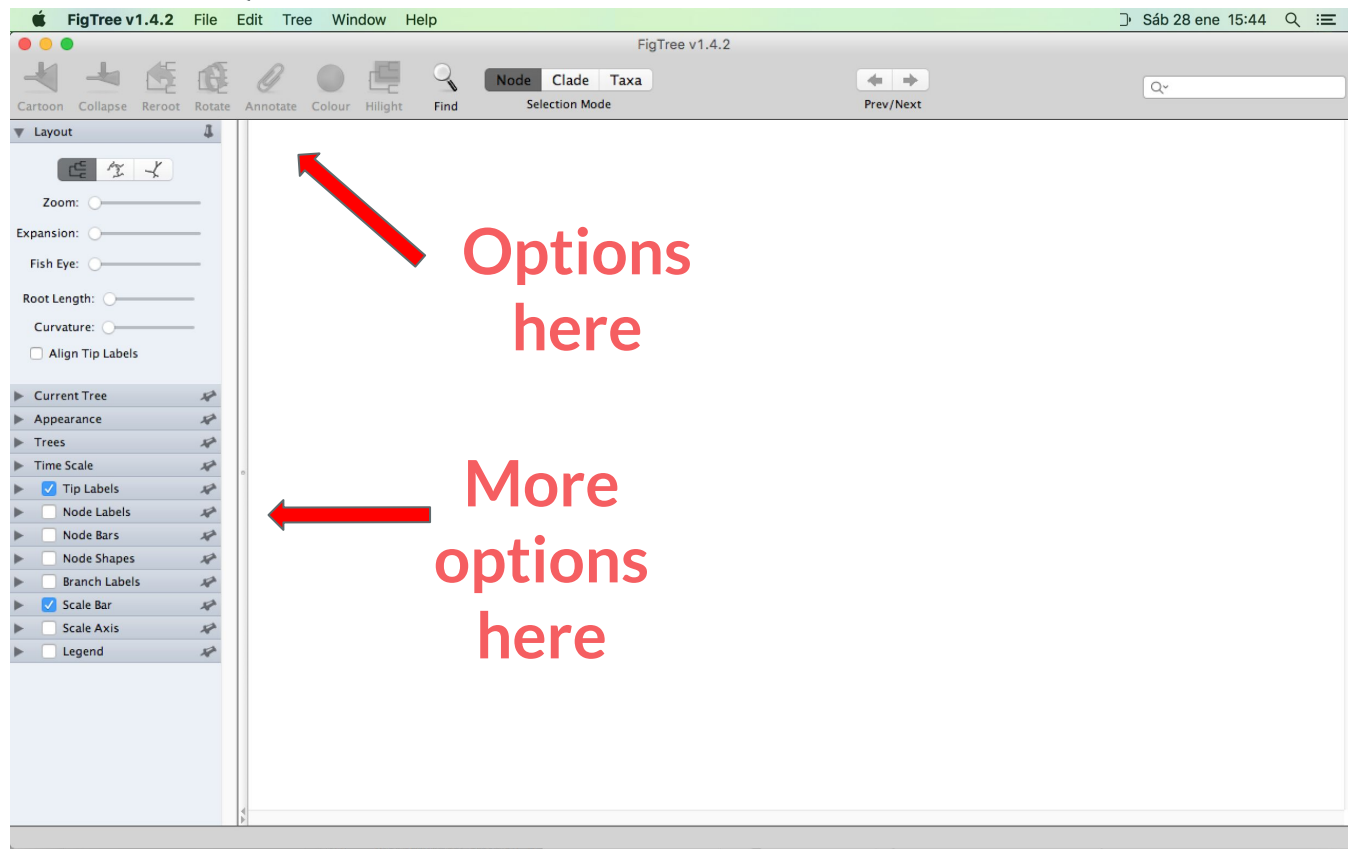
# FigTree

## Features:

- Cross-platform graphical tree display
- Three different tree styles: rectangular, polar and radial
- Display of node heights, branch lengths, support values and other annotations
- Node height range bars if available
- Collapse of clades into triangles
- Colouring of branches and tip labels
- Colouring by annotation (for example, support values)
- Quick search for tip labels or partial tip labels
- Printing and export as PDF graphics

# FigTree

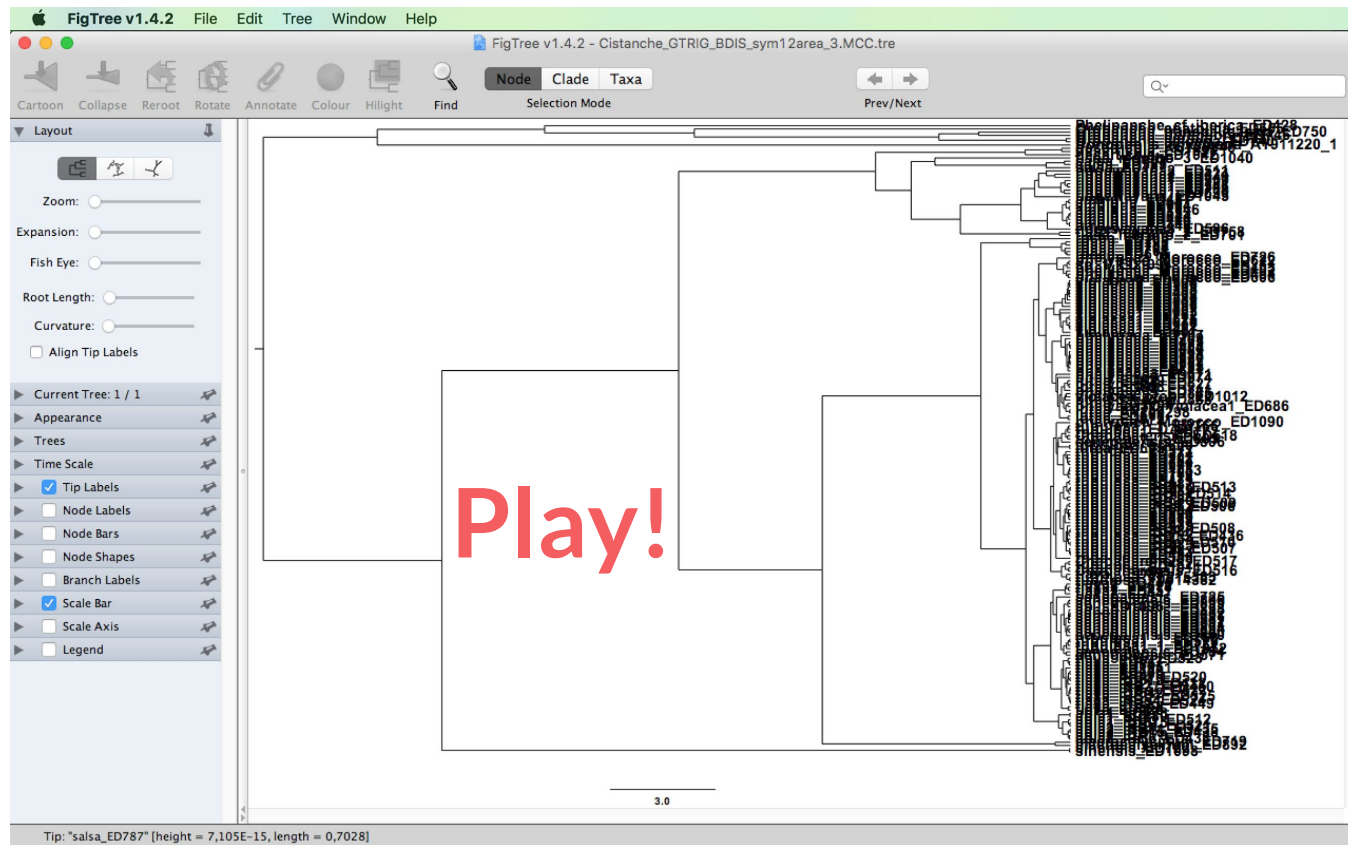
Open files  
here



Options  
here

More  
options  
here

# FigTree



# Software

- **FigTree** @ <http://tree.bio.ed.ac.uk/software/figtree/>
- **DensiTree** @ <https://www.cs.auckland.ac.nz/~remco/DensiTree/>
- **TreeGraph2** @ <http://treegraph.bioinfweb.info>
- Others:
  - **TreeView** @ <http://taxonomy.zoology.gla.ac.uk/rod/treeview.html>
  - **NJplot** @ <http://doua.prabi.fr/software/njplot>
  - **Dendroscope** @ <http://ab.inf.uni-tuebingen.de/software/dendroscope/>
  - **SeaView** @ <http://doua.prabi.fr/software/seaview>
  - etc.

# DensiTree

## Contents:

[INTRODUCTION](#)

[DOWNLOAD/INSTALLATION](#)

[META DATA](#)

[GEOGRAPHY](#)

[FAQ/How to's](#)

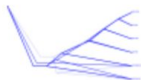
[SUPPORT & LINKS](#)

[WHAT IS NEW](#)

[ACKNOWLEDGMENTS](#)

## Introduction

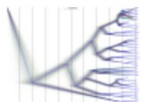
Bayesian hierarchical clustering methods provide a powerful tool for phylogenetic analysis, linguistic research and hierarchical clustering in general such as applied in marketing, political science, customer preference grouping etc. Bayesian methods use MCMC sampling which results in a large number of trees representing the distribution over all possible hierarchies. DensiTree is a program for qualitative analysis of sets of trees. To get an impression of the capabilities of DensiTree, have a look at the gallery below (click thumb nails to get larger image).



Show only consensus trees. This set shows that there is very little uncertainty in the topology of most of the tree, except for the few splits near the root.



Show only consensus trees. This highlights the uncertainty inside the clades, but shows that the split at the root into two groups is very certain (split into progressive and conservative politicians).



Show tree height by height grid and height bar. This tree set nicely demonstrates the increase in uncertainty of the node heights going from the leaves to the root.



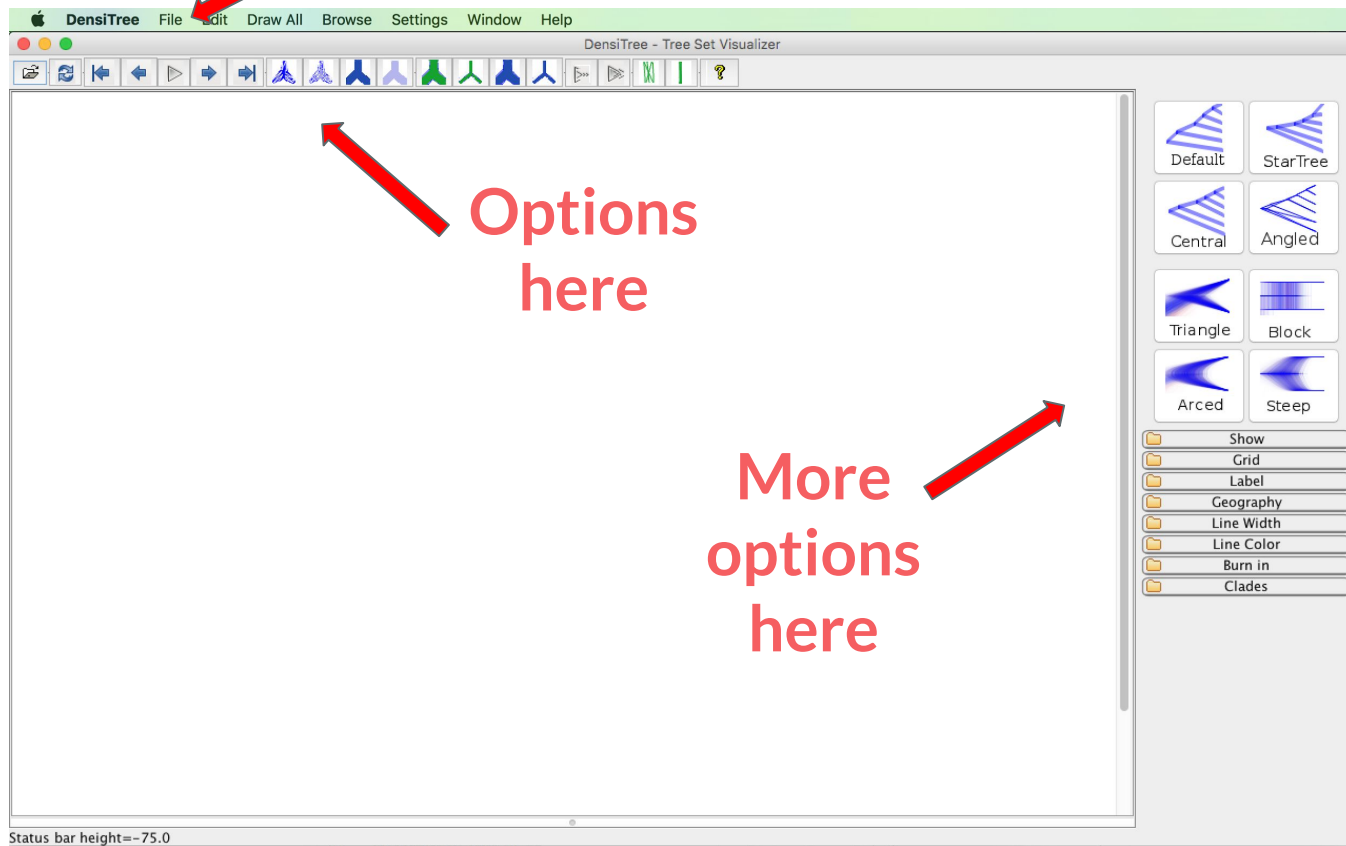
As Figure preceding but in block trees. This tree set was generated with calibration points, which show up as dense node heights, for example, the parent of Rrr and Bbb.



Decreased width of consensus trees, only consensus trees drawn. Intensity of consensus trees needed to be increased considerably. This is useful when there is large uncertainty in the topology and hence many consensus trees (over 900 in this example) with little overlap. Without intensity increase they would not show up and only a white image would be shown.

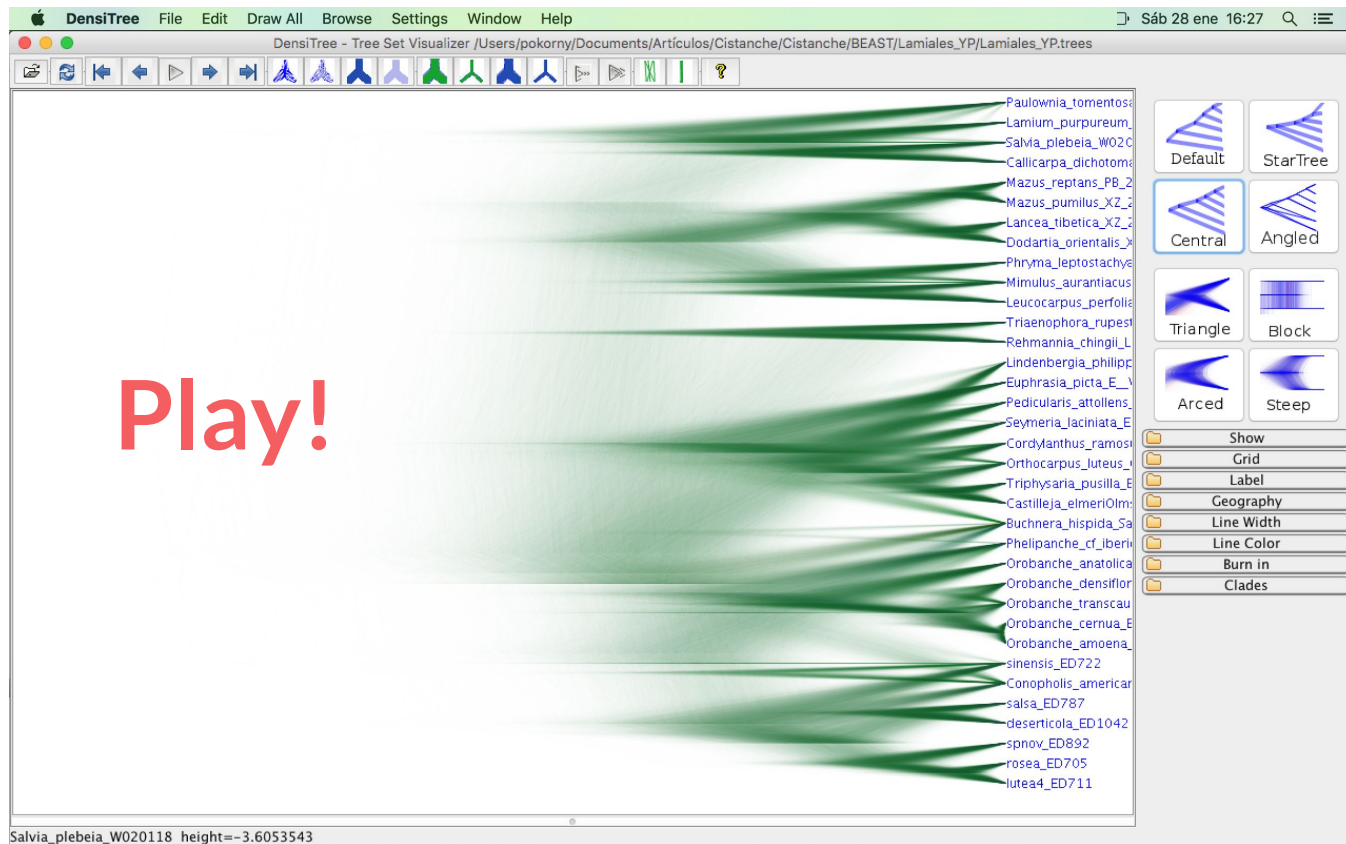
# DensiTree

Open files  
here

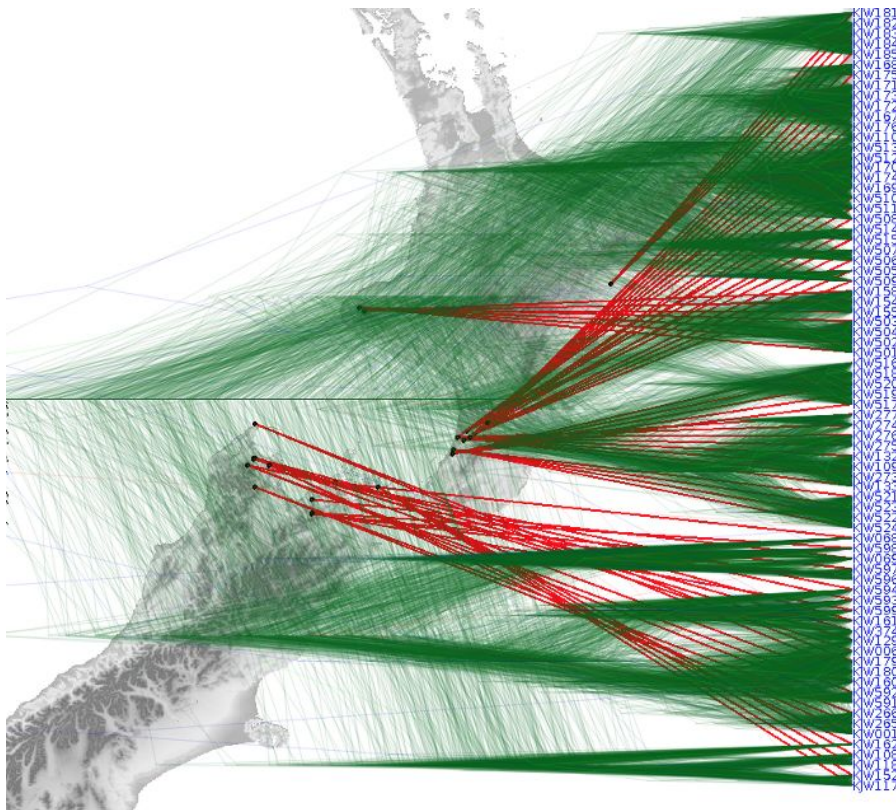


# DensiTree

Play!



# Cloudogram...



<https://www.cs.auckland.ac.nz/~remco/DensiTree/geography.html>

# Clouddogram Fail



<https://www.cs.auckland.ac.nz/~remco/DensiTree/geography.html>

# Cloudogram...

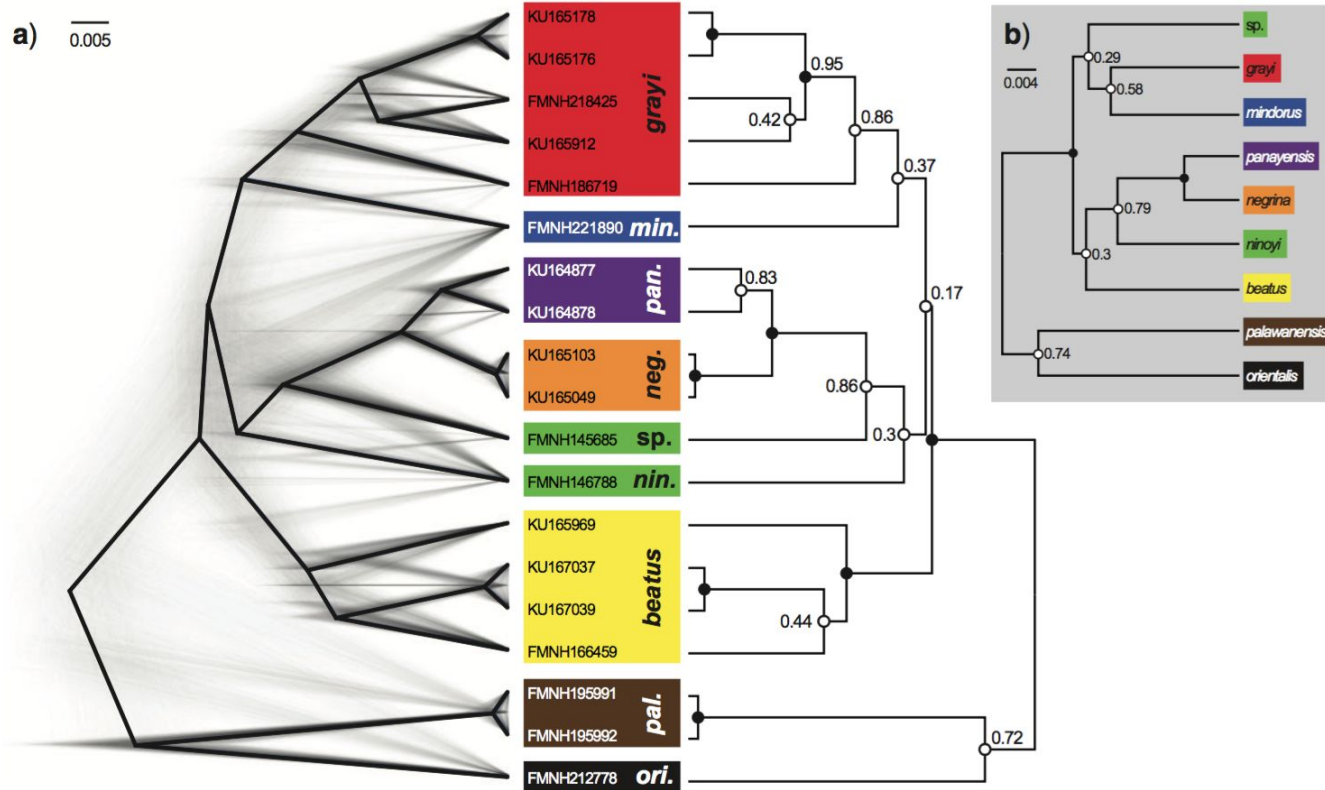


Fig. 3. Giarla & Esselstyn. 2015. *Syst. Biol.* 64(5):727–740.

# Cloudogram Success

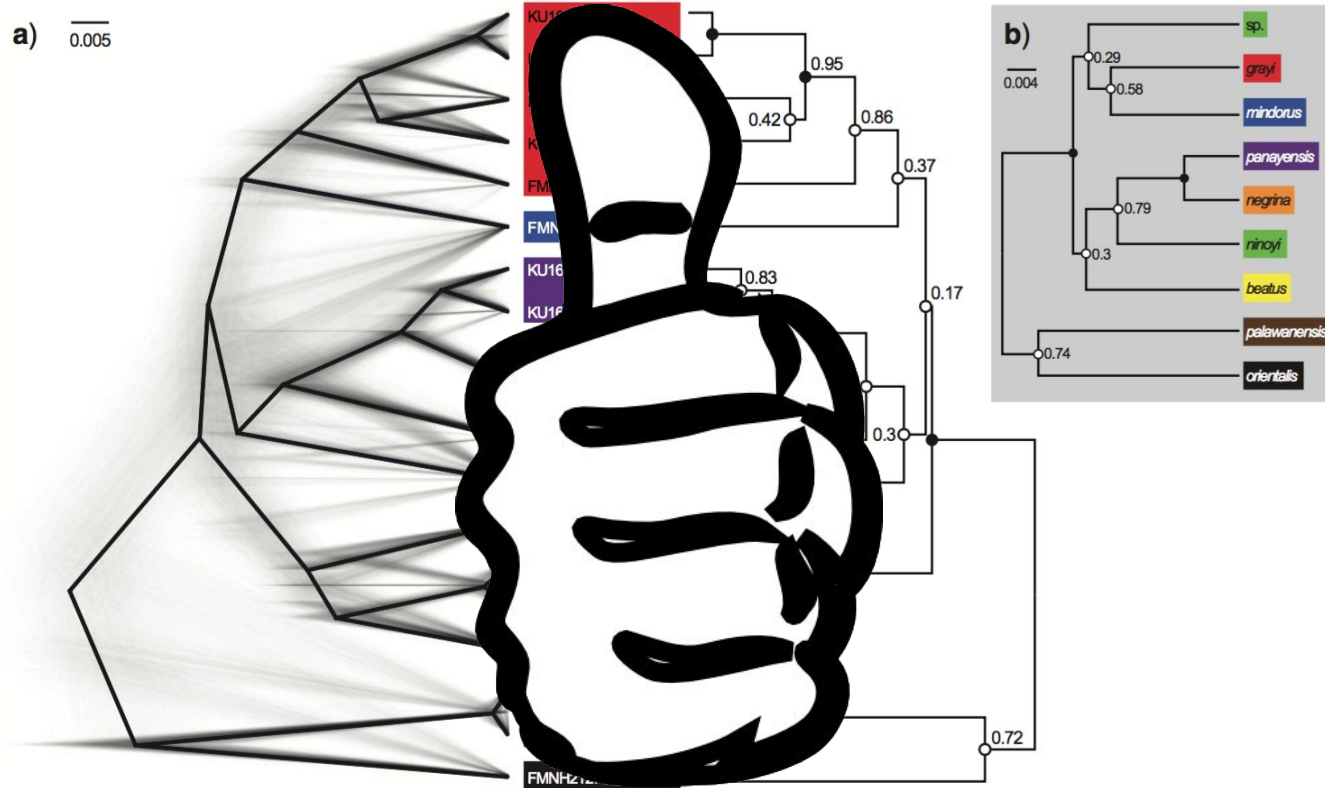


Fig. 3. Giarla & Esselstyn. 2015. *Syst. Biol.* 64(5):727–740.

# Software

- **FigTree** @ <http://tree.bio.ed.ac.uk/software/figtree/>
- **DensiTree** @ <https://www.cs.auckland.ac.nz/~remco/DensiTree/>
- **TreeGraph2** @ <http://treegraph.bioinfweb.info>
- Others:
  - **TreeView** @ <http://taxonomy.zoology.gla.ac.uk/rod/treeview.html>
  - **NJplot** @ <http://doua.prabi.fr/software/njplot>
  - **Dendroscope** @ <http://ab.inf.uni-tuebingen.de/software/dendroscope/>
  - **SeaView** @ <http://doua.prabi.fr/software/seaview>
  - etc.

## TreeGraph 2 - A feature rich and easy to use phylogenetic tree editor

## Download

## System requirements

## How to install

## License

**Help**

## Alphabetical index

## Video tutorials

## Categories

**Random page**

## Development

## Subversion repository

### Recent changes

## Documentation

Latest Javadoc

## XTG format

## Libraries and tools

## Publications

## Version 1

## Subversion repository

## About

**Publication:**

Stöver BC, Müller KF:  
TreeGraph 2: Combining  
and visualizing evidence  
from different  
phylogenetic analyses.

BMC Bioinformatics 2010,

11:7 **Open Access**

Highly accessed

**TreeGraph**  
2.13.0-748 beta



*TreeGraph 2* is a **graphical editor for phylogenetic trees** which allows you to apply lots of graphical formats to the elements of your tree. Moreover, it supports several (visible or invisible) **annotations** (e.g. support values) for every branch or node. These annotations can be imported from *Nexus* tree files or text files containing data in a table (e.g. exported from a spreadsheet program). *TreeGraph 2* is **licensed under GNU General Public License**.

*TreeGraph 2* is developed by **Ben Stöver**, **Sarah Wiechers**, and **Kai Müller**.

If you have any questions about *TreeGraph 2* feel free to contact [stoeve@bioinfweb.info](mailto:stoeve@bioinfweb.info). You can find more software that has been developed by the authors on <http://bioinfweb.info/Software>.

## Citation

*TreeGraph 2* has been published in BMC Bioinformatics:

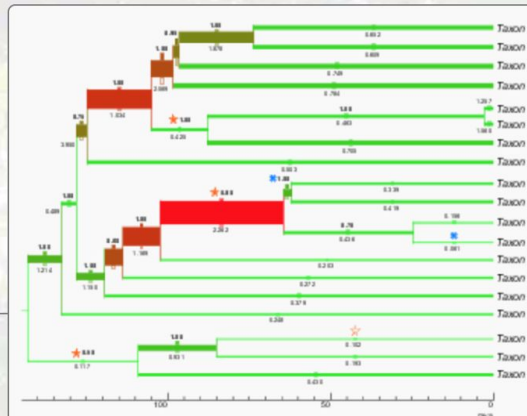
- Stöver B C, Müller K F: *TreeGraph 2: Combining and visualizing evidence from different phylogenetic analyses*. *BMC Bioinformatics* 2010, **11**:7
- DOI: 10.1186/1471-2105-11-7 [What is this?]

## Key features

- **Read trees** in *Newick*, *Nexus* format (including annotations in hot comments), *NeXML* or *PhyloXML*
- **Import annotations from text files** or **combine information** from different phylogenetic analyses
- An unlimited number of **numerical** or **textual annotations** on every branch
- **Export trees** to various vector and (anti-aliased) pixel graphic formats (e.g. *PDF*, *SVG*, *EMF* or *PNG*)
- Many **global** and **element specific formats** like line width or color and text formats
- Versatile editing and formatting options, such as **automatically setting branch widths** or **colors** according to the value of any **attached data**
- Editing operations like **rerooting**, **ladderizing** or **moving** and **collapsing** nodes or **copying** or **manually creating** whole clades

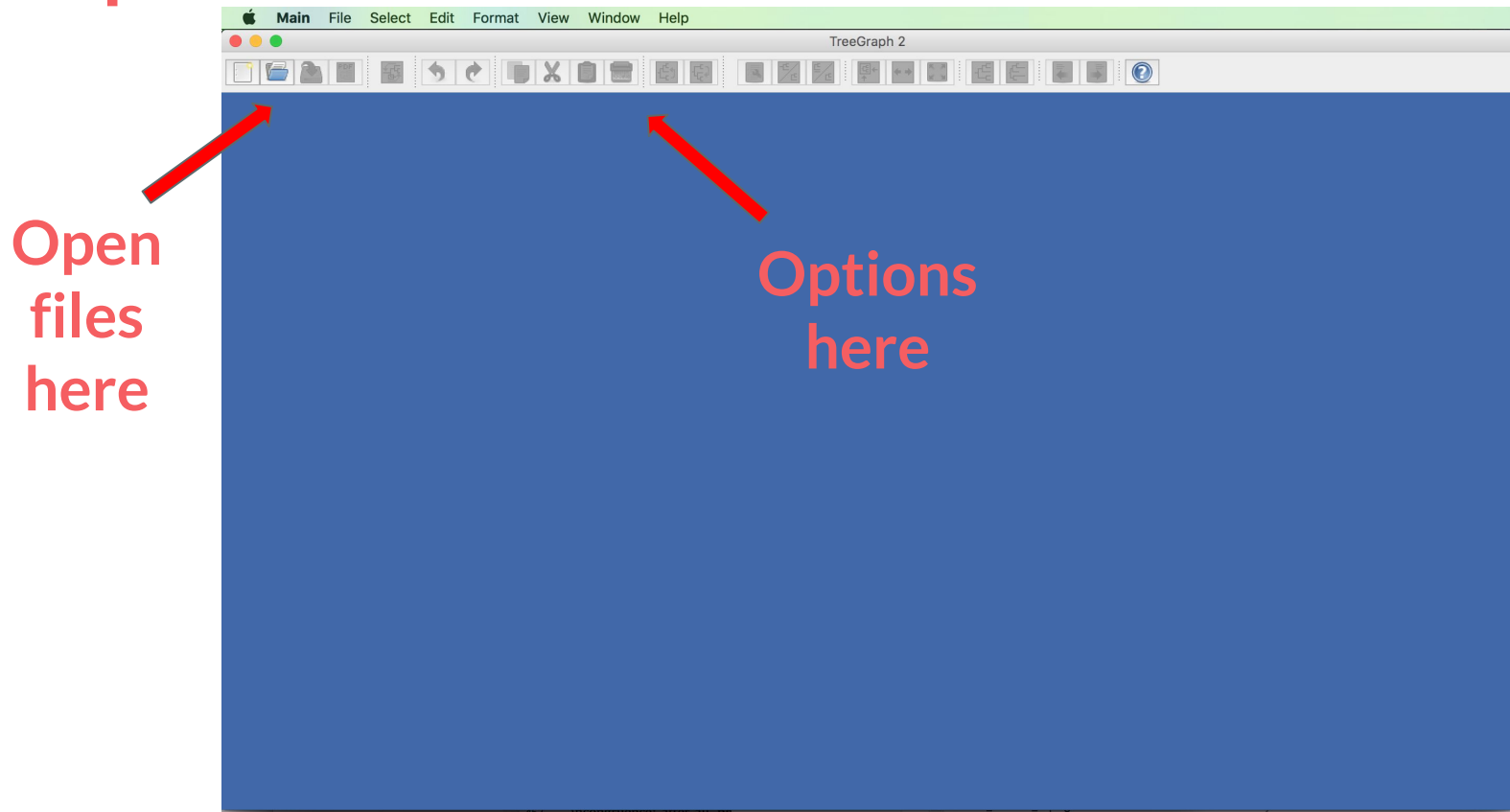
## Getting started

- Download *TreeGraph 2*
- Documentation (help system)
- Tutorials

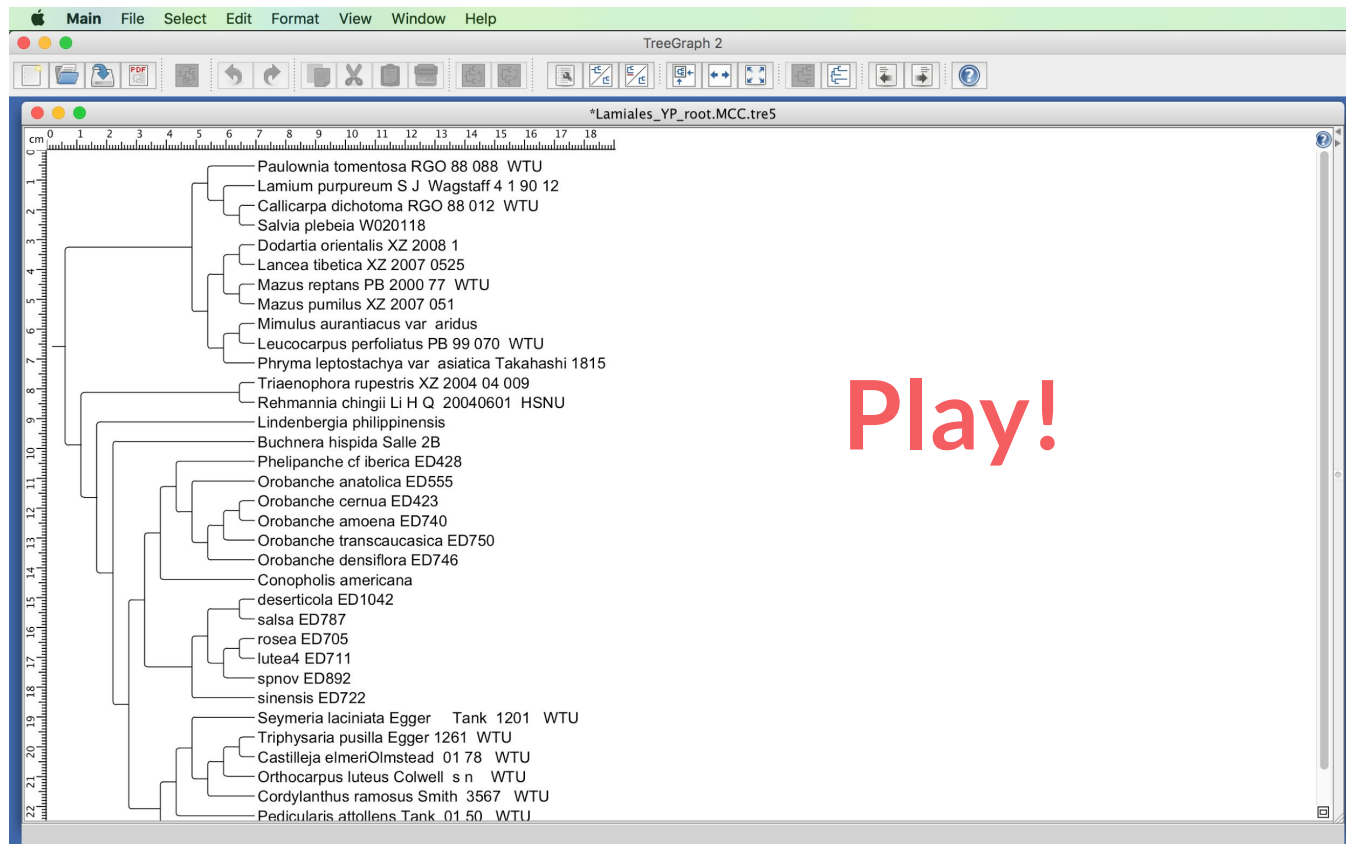


Example of some possible formats in *TreeGraph 2* [\[Enlarge\]](#)  
[\[Download\]](#)

# TreeGraph 2



# TreeGraph 2



Play!

# Tree Graph 2 – Documentation

## Feature overview

### The application

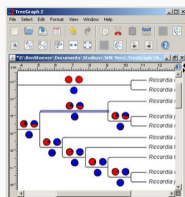
- Main menu
- Tool bar
- Document window
- Synchronizing tree selections

### Import/export

- Open files (Newick, Nexus (including hot comment annotations), NeXML, PhyloXML)
- Adding support values (Merging support values from different analyses)
- Importing node/branch data (Importing annotations from tables)
- Importing ancestral state probabilities
- Exporting commands for ancestral state analyses
- Exporting trees as graphics
- Exporting trees as Newick/Nexus files
- Exporting node/branch data (Exporting annotations to tables)
- Exporting pie chart label colors

### Parts of a TreeGraph 2 document

- Document element
- Node
- Branch
- Label
- Text label
- Icon label
- Pie chart label
- Legend
- Scale bar



## Formatting

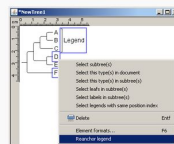
- Element formats
- Document formats
- Set colors by node/branch data (Allows you to display annotations as colors.)
- Set distance values by node/branch data (Allows you to display annotations e.g. as branch widths or text heights.)
- Scaling distance values
- Automatically position labels

## Annotations

- Node/branch data
- Text label
- Hidden node/branch data
- Adding support values
- Copying node/branch data
- Calculating node/branch data (Allows you to calculate annotations by mathematical expressions from other annotations.)
- Deleting node/branch data outside interval
- Data table of the document window

## Tree editing

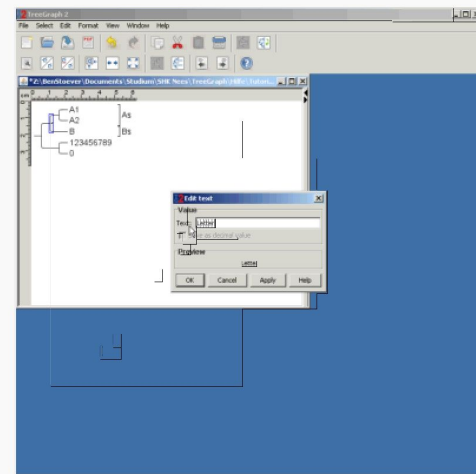
- Inserting nodes
- Collapsing nodes
- Collapsing nodes by support
- Anchoring legends
- Copying/cutting document elements
- Ladderizing
- Sorting terminal nodes
- Moving nodes
- Rerooting
- Editing text element values
- Changing branch lengths
- Replace text in node/branch data



## Video tutorials

In addition to the articles on single features a set of [screencasts](#) is provided that allow you to learn how to use TreeGraph 2 step by step. Currently the following screencasts are available:

- Combining support values from different analyses in one tree
- Creating trees by hand (e.g. helpful in teaching)
- Displaying taxon counts of the angiosperm orders as branch widths



Additional tutorials can be found at the [tutorial main page](#).

# Tree Graph 2 – Video Tutorials

## Video tutorials

### Adding support values

This tutorial shows how to merge support values from different analyses in one tree, which is one of the most important features of TreeGraph 2.

[\[Tutorial overview\]](#) [\[First step\]](#) [\[Download files\]](#)

### Creating trees

TreeGraph 2 is one of very few tree editors that allows creating whole new trees without importing any e.g. Nexus file as well as adding new **document elements** (e.g. **nodes**) to existing trees. This can be very helpful e.g. to rapidly generate a tree for a lecture.

[\[Tutorial overview\]](#) [\[First step\]](#) [\[Download files\]](#)

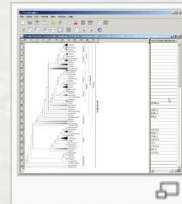
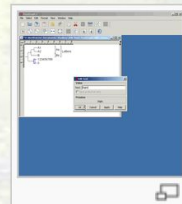
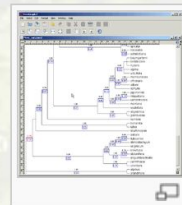
### Display taxon counts as branch widths

This tutorial demonstrates the ability of TreeGraph 2 to import **node/branch data** from tables and to display numeric values as formats (branch widths in this case). Precisely that means that we will import a text file which contains taxon counts of different angiosperm orders into an angiosperm tree and visualize the data as **branch widths**.

[\[Tutorial overview\]](#) [\[First step\]](#) [\[Download files\]](#)

## Other tutorials

- Calculating node/branch data: [Calculating node ages](#)
- Calculating node/branch data: [Converting ancestral character states into probability columns](#)



# Phylogenetic Tree Visualization Software

- **Web Portals:**

- Phylo.IO @ <http://phylo.io>
- Interactive Tree of Life viewer (iTOL) @ <http://itol.embl.de>
- EVOLVIEW @ <http://www.evolgenius.info/evolview>

- **Software:**

- FigTree @ <http://tree.bio.ed.ac.uk/software/figtree/>
- DensiTree @ <https://www.cs.auckland.ac.nz/~remco/DensiTree/>
- TreeGraph2 @ <http://treegraph.bioinfweb.info>

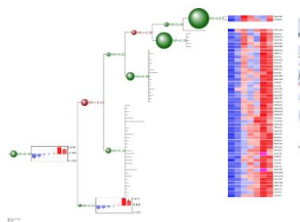
- **Toolkits:**

- PYTHON library: ETE toolkit @ <http://etetoolkit.org>
- R repository: PhyTools @ <https://cran.r-project.org/web/packages/phytools/index.html>
- JavaScript library: jsPhyloSVG @ <http://www.jsphylosvg.com>

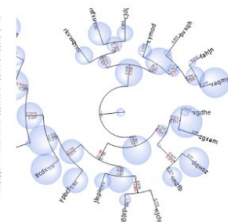
- [https://en.wikipedia.org/wiki/List\\_of\\_phylogenetic\\_tree\\_visualization\\_software](https://en.wikipedia.org/wiki/List_of_phylogenetic_tree_visualization_software)

# Toolkits

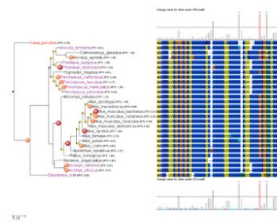
- **PYTHON** library: **ETE3 toolkit** @ <http://etetoolkit.org>
- **R** repository: **PhyTools** @ <https://cran.r-project.org/web/packages/phytools/index.html>
- **JavaScript** library: **jsPhyloSVG** @ <http://www.jsphylosvg.com>
- **Others:**
  - Archaeopteryx @ <https://sites.google.com/site/cmzmasek/home/software/archaeopteryx>
  - PhyD3 @ <https://phyd3.bits.vib.be/index.html>
  - GGTREE @ <https://guangchuangyu.github.io/ggtree/>
  - (More on phylogenetics in R here: <https://cran.r-project.org/web/views/Phylogenetics.html>)
  - etc.



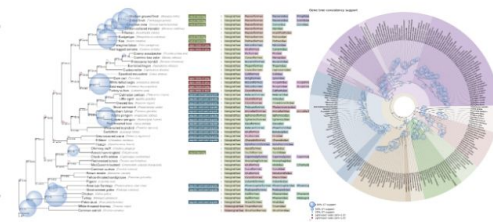
clustering with node validation and heatmap



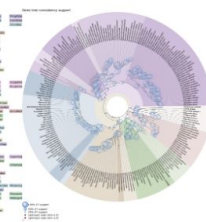
A bubble tree map



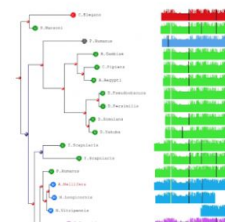
Tree showing the result of an adaptation test with CodeML



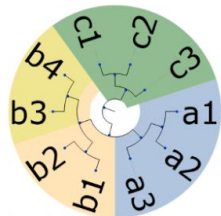
A birds tree with node labeling and NCBI taxonomy lineage



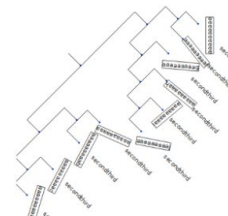
A tree of life representation



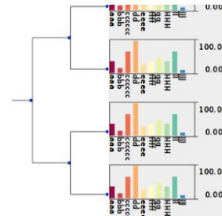
Custom tree showing intron information using aligned faces



standard circular tree representation



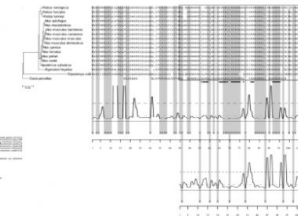
Tree using rotated faces



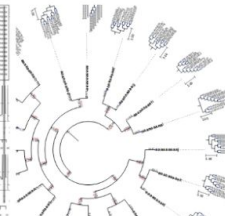
Tree using BarChart faces



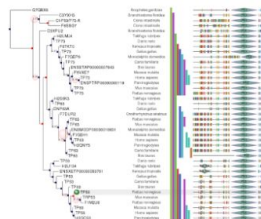
Per node analysis of a fungal tree of life



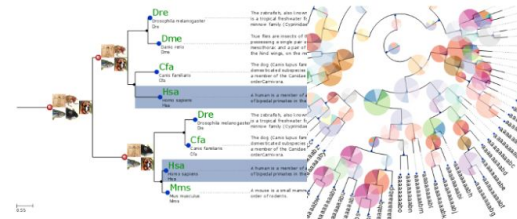
tree and site selection histogram



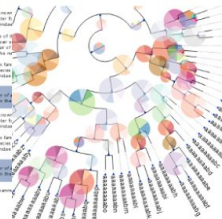
A tree of trees using TreeFace



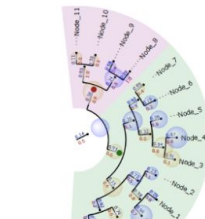
a tree showing domain structure and taxonomic classification



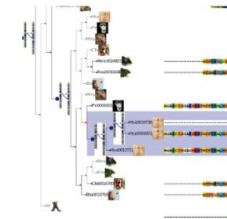
tree with custom annotations



PieChart tree example



A semicircular tree with dynamic node styles



Reconciled tree visualization with node images

# ETE3

Python package, not a Python-based program

You can combine it with all the power of Python (Ni!), but then you have to cut the mightiest tree in the forest with a herring (less user friendly)

Not only useful for annotations and visualization

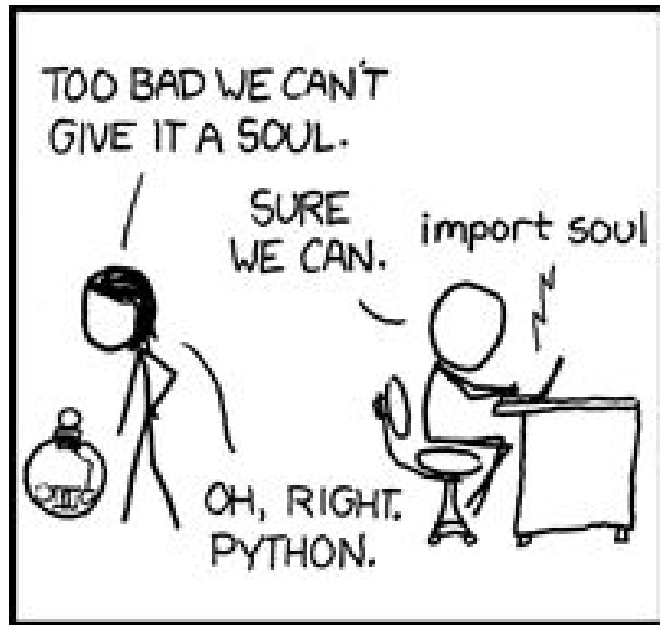
Nobody expects the Spanish Inquisition!



# ETE3

As a package, ETE3 is much more powerful when used programmatically. You make a script, you use it over as many trees you want.

Remember that ETE3 also has other functions. And that you can import more than a package at the same time. And that you can use ETE3 to make programs or scripts and place them into a pipeline.



<https://xkcd.com/413/>

# ETE3

You have already had your first contact with ETE3. You can refresh what you learned with the web tutorial for the programmable tree drawing engine.

[http://etetoolkit.org/docs/latest/tutorial/tutorial\\_drawing.html](http://etetoolkit.org/docs/latest/tutorial/tutorial_drawing.html)



# ETE3 – Annotating Trees

Four elements:

1. `TreeStyle`: Properties of the whole tree
2. `NodeStyle`: Properties of nodes. You create it and then attach to a node
3. `Faces`: Additional data to a node
4. `Layout`: Rules to place all together

They are Python objects and as such they can be combined with variables, loops, functions and all these sorcery

# Toolkits

- PYTHON library: ETE3 toolkit @ <http://etetoolkit.org>
- R repository: **PhyTools** @ <https://cran.r-project.org/web/packages/phytools/index.html>
- JavaScript library: jsPhyloSVG @ <http://www.jsphylosvg.com>
- Others:
  - Archaeopteryx @ <https://sites.google.com/site/cmzmasek/home/software/archaeopteryx>
  - PhyD3 @ <https://phyd3.bits.vib.be/index.html>
  - etc.

# Phytools

R based

This gives you nearly infinite “mathemagical” power, use it wisely

Absurdly powerful drawing tools

Just like ETE, it has many functions beyond tree representation

R is much more specialized than Python (also, not named after the Monty Python)



# Phytools demo

First we have to download Phytools. For this, go to R. You can do this by typing 'R' on the terminal or through any R graphic user interface (RStudio, R GUI, QuickR...).

Then we install the package. For this, type:

```
install.packages("phytools", repos="https://cloud.r-project.org")
```

Once finished, check the installation with:

```
packageVersion("phytools")
```

# Phytools demo

Now we load the package with:

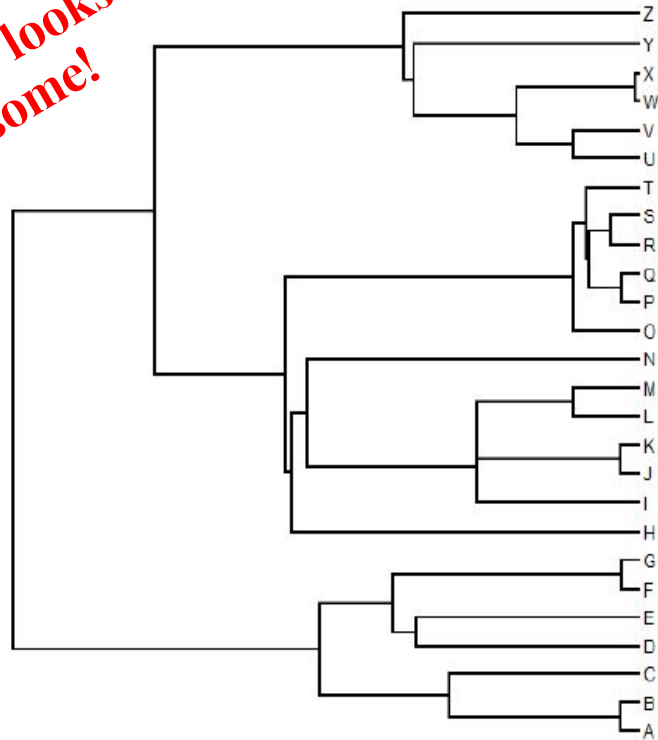
```
require(phytools)
```

Don't you feel all this power already? Let's start simulating a tree and plotting it:

```
tree<-pbtree(n=26,tip.label=LETTERS)
```

```
plotTree(tree)
```

*Our tree looks  
awesome!*





# Phylotools demo

Let's try now to represent the values in x as dots in the tree. The size indicates the value:

```
dotTree(tree,x,standardize=TRUE)
```

We are going to repeat it now with more data. First we create the data:

```
X<-fastBM(tree,nsim=10)
```

And now we plot again

```
dotTree(tree,X,standardize=TRUE)
```

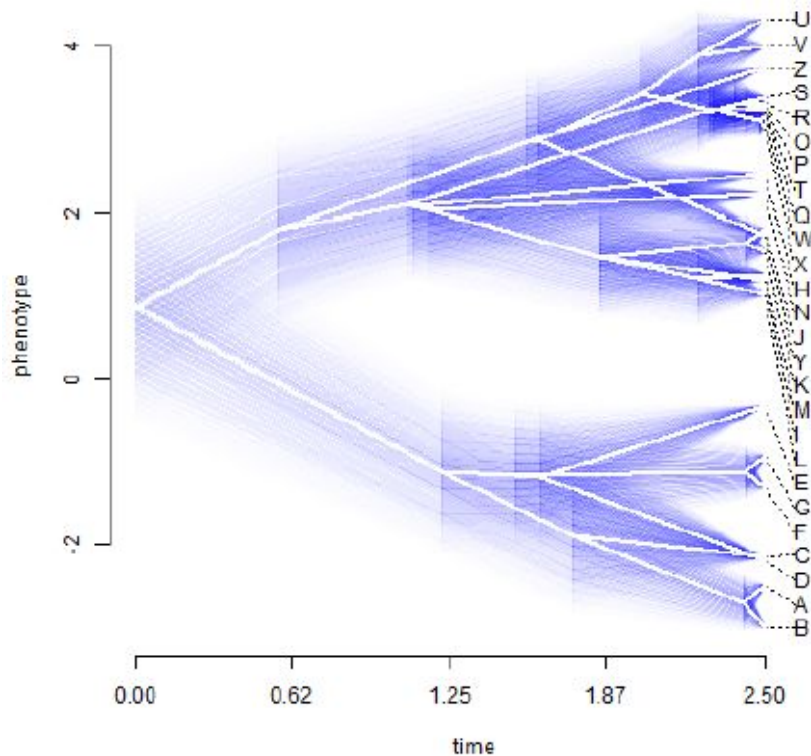
# Phylotools demo

Let's try a heat map with our data:

```
phylo.heatmap(tree,X,standardize=TRUE)
```

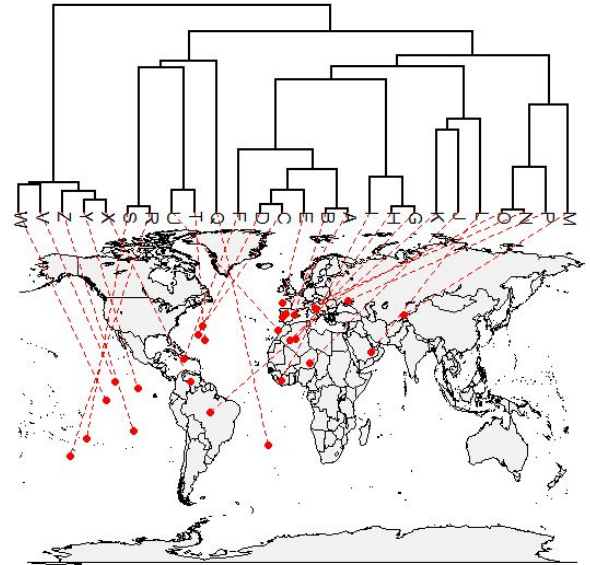
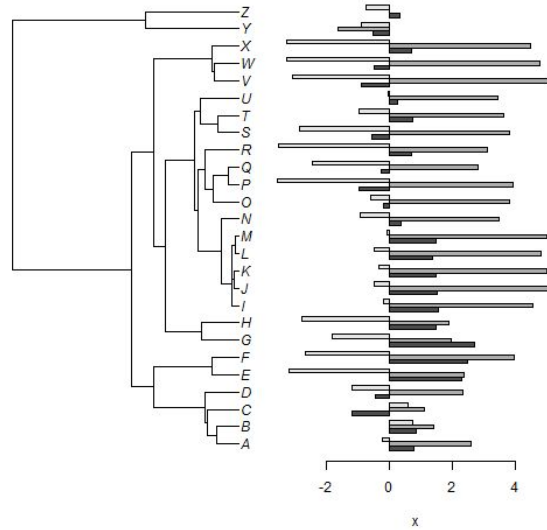
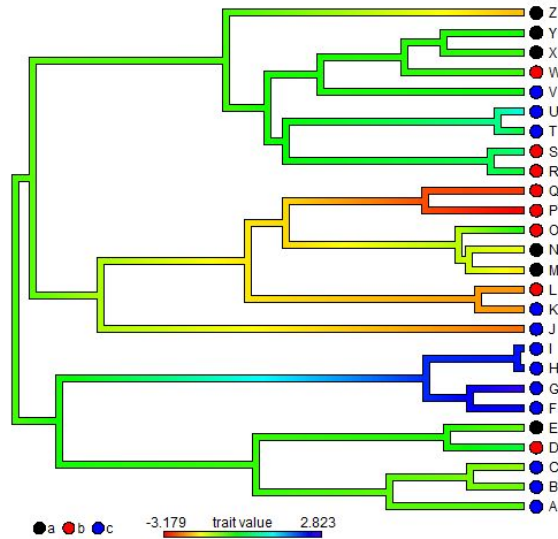
Now some hardcore stuff: Traitgram with the uncertainty of ancestral traits visualized using transparent probability density:

```
fancyTree(tree,type="phenogram95",x=x,s  
pread.cost=c(1,0))
```



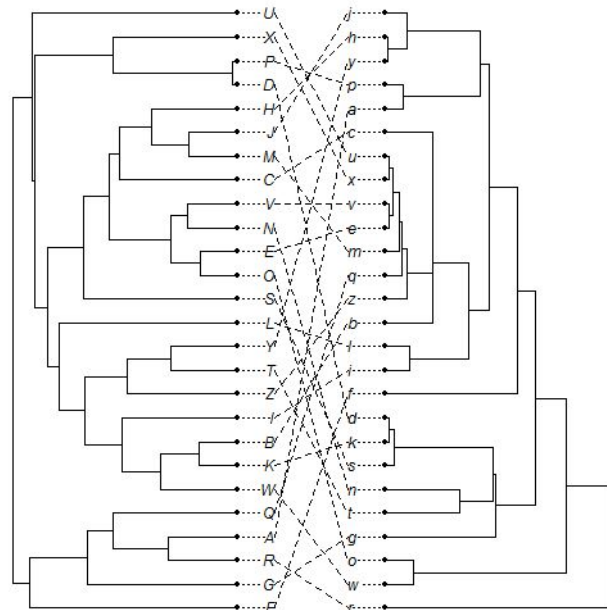
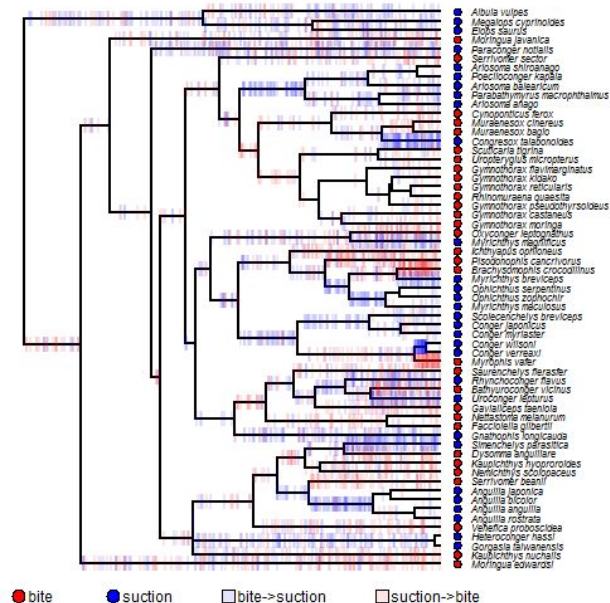
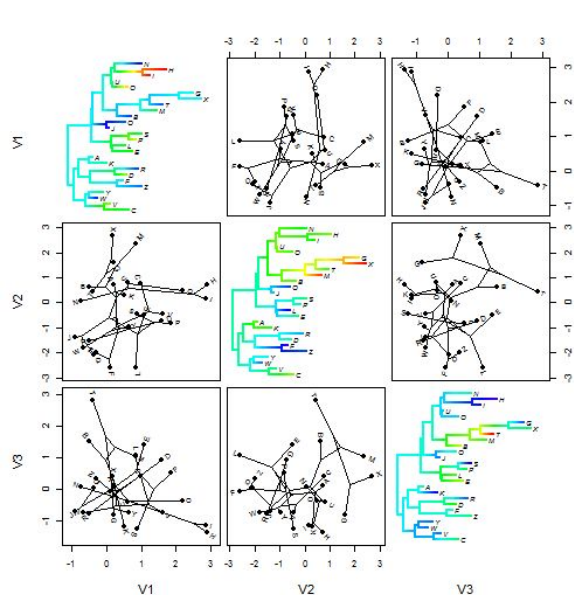
Now I write in comic sans because  
“matemagical” power corrupts anyone

# Phytools Examples



<http://blog.phytools.org/>

# Phytools Examples



<http://blog.phytools.org/>

# Toolkits

- PYTHON library: ETE3 toolkit @ <http://etetoolkit.org>
- R repository: PhyTools @ <https://cran.r-project.org/web/packages/phytools/index.html>
- JavaScript library: jsPhyloSVG @ <http://www.jsphylosvg.com>
- Others:
  - Archaeopteryx @ <https://sites.google.com/site/cmzmasek/home/software/archaeopteryx>
  - PhyD3 @ <https://phyd3.bits.vib.be/index.html>
  - etc.

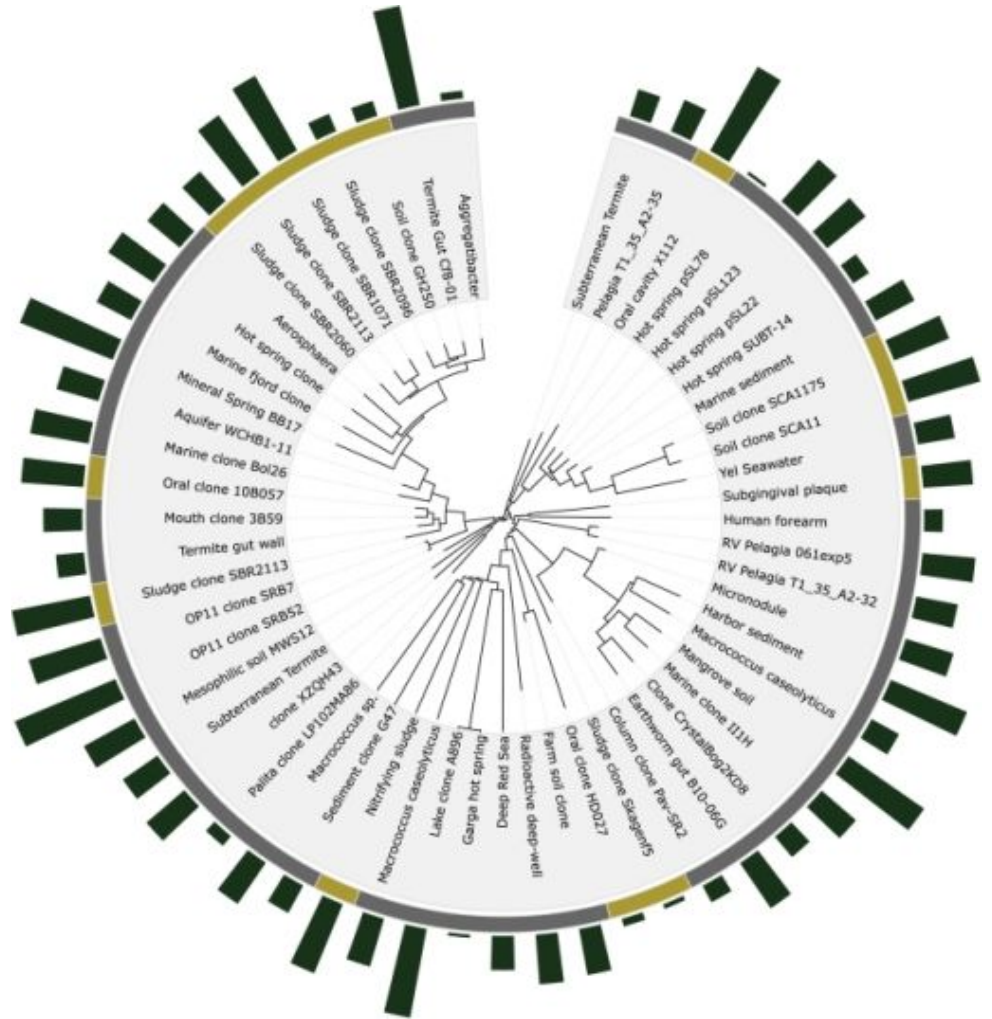
# jsPhyloSVG

JavaScript based

Much more focused on user  
side web visualization

Very simple syntax

A bit limited compared to other  
options

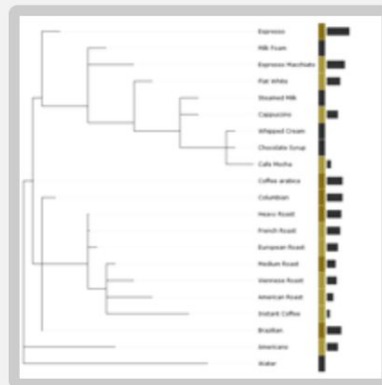


## How can jsPhyloSVG help me?

jsPhyloSVG is an open-source javascript library specifically built for rendering highly-extensible, customizable phylogenetic trees. jsPhyloSVG can render complex trees, yet offers a simple method to do so. It leverages the recent XML schema definition specified for phylogenetic trees, [phyloXML](#).

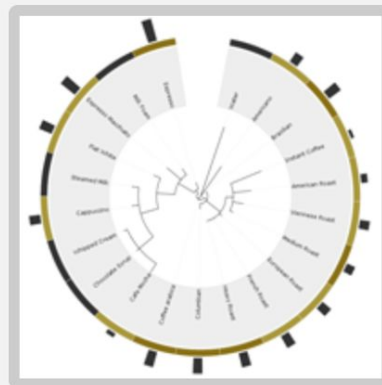
Here is a quick example of how to render a **rectangular** phylogenetic tree:

```
1 phylocanvas = new Smits.PhyloCanvas(  
2   dataObject,    // Newick or XML string  
3   'svgCanvas',   // Div Id where to render  
4   1000, 1000    // Height, Width in pixels  
5 );
```



The same tree, except this time a **circular** tree:

```
1 phylocanvas = new Smits.PhyloCanvas(  
2   dataObject,    // Newick or XML string  
3   'svgCanvas',   // Div Id where to render  
4   1000, 1000,    // Height, Width in pixels  
5   'circular'     // Type of tree  
6 );
```

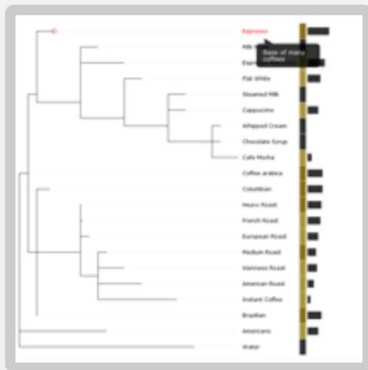


## What about interactive features?

The XML definition already includes the option of adding a hyperlink reference and description. jsPhyloSVG taps into this, and formats the text nodes with this information.

Here's an example of what the XML format looks like:

```
1 <clade>
2 <name>Subterranean Termite</name>
3 <branch_length>0.17793</branch_length>
4 <annotation>
5   <desc>GQ502663.1 Gut clone Cf8-01 </desc>
6   <uri>http://www.jsphylosvg.com/GQ502663 </uri>
7 </annotation>
8 </clade>
```



## Cool. Tell me more!

To get started, take a look at our [documentation](#). You will find plenty of code and examples.

We are always looking for ways to make our library better. So please, [let us know](#) what you like, and how we can make this better for you! If you create extensions, let us know so that the rest of the community can benefit.

## Documentation

### **1. Loading Trees**

- 1.1 Dependencies
- 1.2 Best Methods
- 1.3 Loading Simple Trees
- 1.4 Using AJAX

### **2. Working with Tree Data**

- 2.1 A Primer on passing data to the parser
- 2.2 Newick Format
- 2.3 PhyloXML Format
- 2.4 NeXML Format

### **3. Applying Visual, Interactive, and Charting Features**

- 3.1 Interactive Features
- 3.2 Parameters
  - 3.21 Overriding Styles in Javascript
- 3.3 Binary Arc Charts
- 3.4 Multiple Chart Tracks
  - 3.41 Internal Arc Charts
- 3.5 Bar Charts
- 3.6 Highlight Labels
- 3.7 Gradient Labels
- 3.8 Integrated Ribbons
  - 3.81 Integrated Ribbon Labels

### **4. Additional Manipulations**

- 4.1 Save tree as SVG

# Phylogenetic Tree Visualization Software

- **Web Portals:**

- Phylo.IO @ <http://phylo.io>
- Interactive Tree of Life viewer (iTOL) @ <http://itol.embl.de>
- EVOLVIEW @ <http://www.evolgenius.info/evolview>

- **Software:**

- FigTree @ <http://tree.bio.ed.ac.uk/software/figtree/>
- DensiTree @ <https://www.cs.auckland.ac.nz/~remco/DensiTree/>
- TreeGraph2 @ <http://treegraph.bioinfweb.info>

- **Toolkits:**

- PYTHON library: ETE toolkit @ <http://etetoolkit.org>
- R repository: PhyTools @ <https://cran.r-project.org/web/packages/phytools/index.html>
- JavaScript library: jsPhyloSVG @ <http://www.jsphylosvg.com>

- [https://en.wikipedia.org/wiki/List\\_of\\_phylogenetic\\_tree\\_visualization\\_software](https://en.wikipedia.org/wiki/List_of_phylogenetic_tree_visualization_software)



- Main page
- Contents
- Featured content
- Current events
- Random article
- Donate to Wikipedia
- Wikipedia store

- Interaction
  - Help
  - About Wikipedia
  - Community portal
  - Recent changes
  - Contact page

- Tools
  - What links here
  - Related changes
  - Upload file
  - Special pages
  - Permanent link
  - Page information
  - Wikidata item
  - Cite this page

- Print/export
  - Create a book
  - Download as PDF
  - Printable version

# List of phylogenetic tree visualization software

From Wikipedia, the free encyclopedia

This list of **phylogenetic tree viewing software** is a compilation of software tools and web portals used in visualising [phylogenetic trees](#).

**Contents** [hide]

1

Online software

2

Desktop Software

3

See also

4

References

5

External links

## Online software [ edit ]

| Name                                  | Description                                                                                                             | Site                | Citation            |
|---------------------------------------|-------------------------------------------------------------------------------------------------------------------------|---------------------|---------------------|
| <a href="#">Aquapony</a>              | Javascript tree viewer for <a href="#">Beast</a> .                                                                      |                     |                     |
| <a href="#">Archaeopteryx</a>         | Java tree viewer and editor (used to be ATV)                                                                            | <a href="#">[1]</a> | <a href="#">[1]</a> |
| EvolView,                             | an online tool for visualizing, annotating and managing phylogenetic trees.                                             | <a href="#">[2]</a> | <a href="#">[2]</a> |
| ETE toolkit                           | A Python Environment for Tree Exploration (online treeview)                                                             | <a href="#">[3]</a> | <a href="#">[3]</a> |
| Hypergeny                             | visualise large phylogenies with this hyperbolic tree browser                                                           | <a href="#">[4]</a> |                     |
| InfoViz Tree Tools                    | the generic Javascript InfoViz toolkit supports hyperbolic, space and icicle trees                                      | <a href="#">[5]</a> |                     |
| iTOL -<br>interactive<br>Tree Of Life | annotate trees with various types of data and export to various graphical formats; scriptable through a batch interface | <a href="#">[6]</a> | <a href="#">[4]</a> |

# Phylogenetic Tree Visualization Software

- **Web Portals:**

- Phylo.IO @ <http://phylo.io>
- Interactive Tree of Life viewer “ITOL” @ <http://itol.ezlab.net/>
- EVOLVIEW @ <http://www.evolview.ca/>

- **Software:**

- FigTree @ <http://tree.bio.ed.ac.uk/software/figtree/>
- DensiTree @ <https://www.nihs.gov/research/programs-services/contract-testing-services/densitree/>
- TreeGraph2 @ <http://treegraph2.bioinformatics.net/>

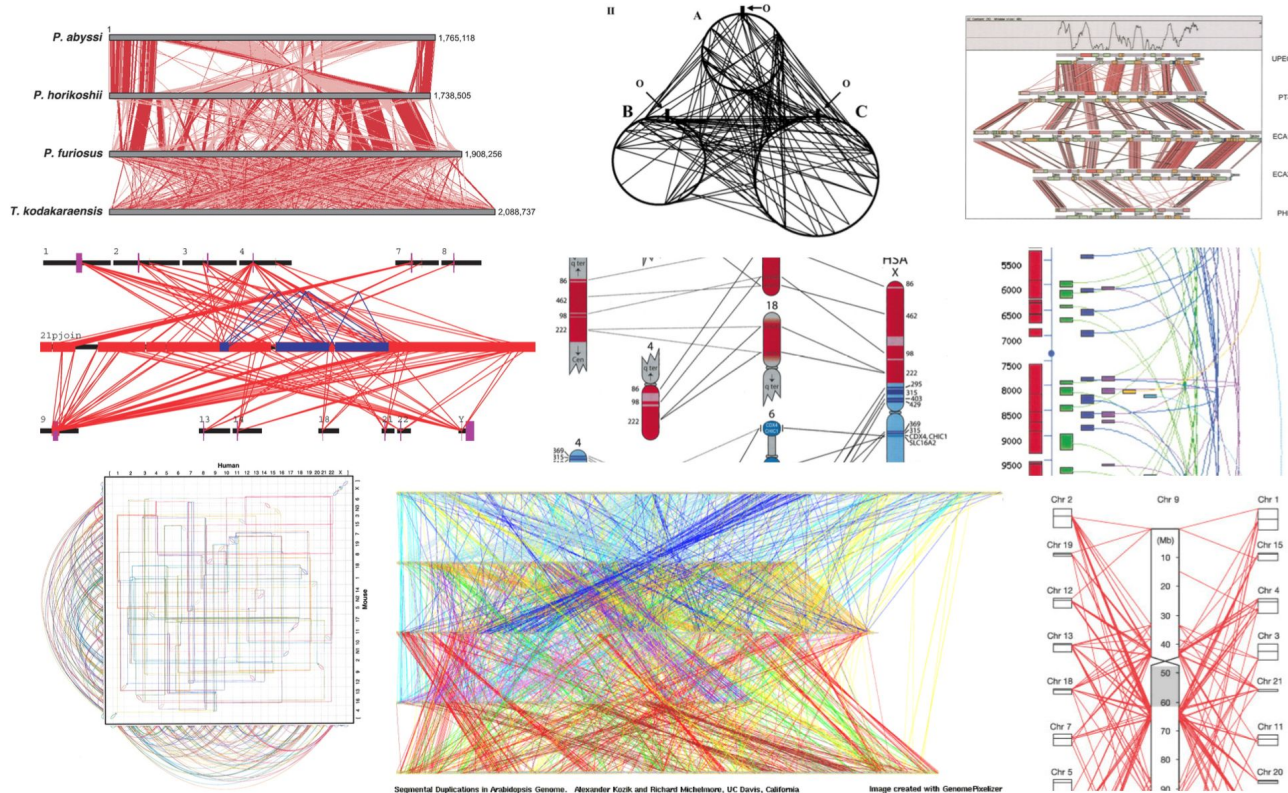
- **Toolkits:**

- PYTHON library: ET-Toolkit @ <https://github.com/et-toolkit/et-toolkit>
- R repository: PhyTools @ <https://github.com/PhyloTools/PhyTools/blob/master/packages/phytools/index.html>
- JavaScript library: jsPhyloSVG @ <http://www.jsphylosvg.com>

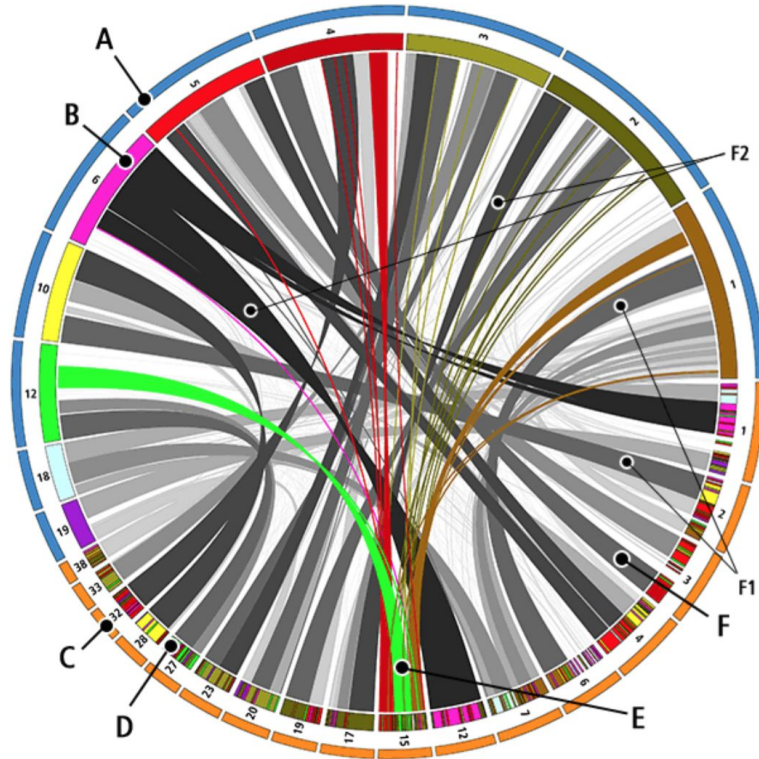
- [https://en.wikipedia.org/wiki/List\\_of\\_phylogenetic\\_tree\\_visualization\\_software](https://en.wikipedia.org/wiki/List_of_phylogenetic_tree_visualization_software)



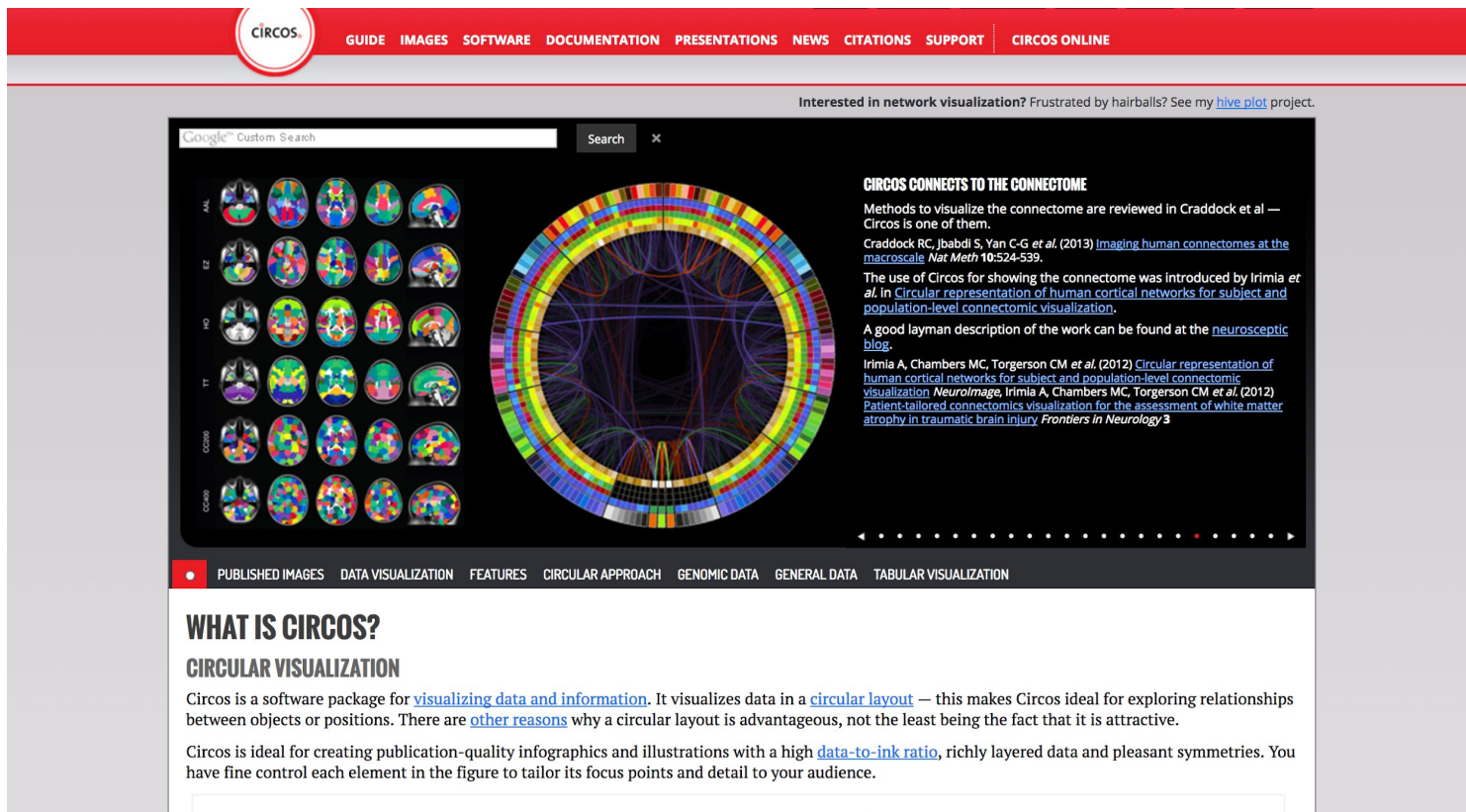
# Visualizing Networks – The Problem



# Visualizing Networks – Circos!



# Visualizing Networks – Circos



# Visualizing Networks – Circos

## TUTORIALS

### 0. [INSTALLATION AND CONFIGURATION](#)

1. [UNIX vs Windows](#)
2. [Installation](#)
3. [Configuration Files – Syntax, Colors, Fonts and Units](#)
4. [Colors](#)
5. [Data Files](#)
6. [Required Perl Modules](#)
7. [Runtime Parameters](#)
8. [Debugging](#)
9. [Common Errors](#)
10. [PNG Output](#)
11. [SVG Output](#)

### 1. [QUICK START](#)

1. [Hello World](#)
2. [Ticks & Labels](#)
3. [Ideogram Selection, Scale, Color & Orientation](#)
4. [Links & Rules](#)
5. [Histograms](#)
6. [Axes & Backgrounds](#)
7. [Heat Maps & Colors](#)
8. [Text](#)
9. [Automation & Macros](#)

### 2. [DRAWING IDEOGRAMS](#)

1. [Ideograms](#)
2. [Karyotypes](#)
3. [Filtering](#)
4. [Ordering](#)
5. [Cropping](#)
6. [Spacing and Axis Breaks](#)
7. [Tags](#)
8. [Variable Radius](#)
9. [Labels](#)
10. [Progression and Orientation](#)

### 3. [HIGHLIGHTS](#)

1. [Wedge Highlights](#)
2. [Highlight Parameters – Part I – Embedded in Data File](#)
3. [Highlight Parameters – Part II – Using z-depth](#)
4. [Highlight Parameters – Part III – Radial Position](#)
5. [Ideogram Highlights](#)
6. [Recipe 1 – Ideogram Highlights](#)
7. [Recipe 2 – Focusing on a Genome Region](#)
8. [Recipe 3 – Plot Axis Range Highlights](#)
9. [Drawing on Top of Data](#)

### 4. [TICK MARKS, GRIDS AND LABELS](#)

1. [Tick Marks – Basics](#)
2. [Tick Marks – Margins](#)
3. [Tick Marks – Label Margins](#)
4. [Tick Marks – Offsets](#)
5. [Grids](#)
6. [Label Formats](#)
7. [Relative Ticks](#)
8. [Ticks at Specific Positions](#)
9. [Tick Rings](#)

### 5. [LINKS AND RELATIONSHIPS](#)

1. [Drawing Basic Links](#)
2. [Link geometry](#)
3. [Link Formatting](#)
4. [Link Rules – Part I](#)
5. [Link Rules – Part II](#)
6. [Link Rules – Part III](#)
7. [Link Rules – Part IV](#)
8. [Link Rules – Part V](#)
9. [Ribbons](#)
10. [Ribbon Twists](#)

### 6. [2D DATA TRACKS](#)

1. [Scatter Plots](#)
2. [Line Plots](#)
3. [Histograms](#)
4. [Tiles](#)
5. [Heat Maps](#)
6. [Text—Basic](#)
7. [Text—Stacking](#)
8. [Text—Rules](#)
9. [Glyphs—Part I](#)
10. [Glyphs—Part II](#)
11. [Connectors](#)
12. [Putting It All Together](#)

### 7. [AXIS SCALING](#)

1. [Global Scale Adjustment](#)
2. [Global Relative Scale Adjustment](#)
3. [Adjusting Scale for Regions](#)
4. [Creating Zoomed Regions](#)
5. [Overlapping Zoomed Regions](#)
6. [Smoothing Scale](#)
7. [Combining Scales](#)
8. [Drawing Data with Scale Adjustment](#)

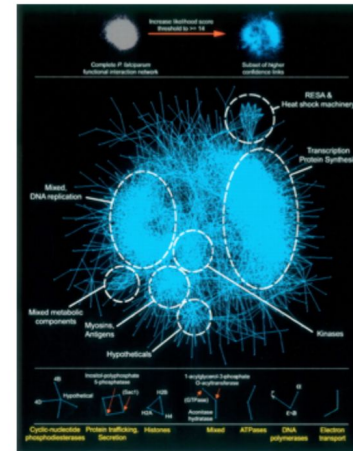
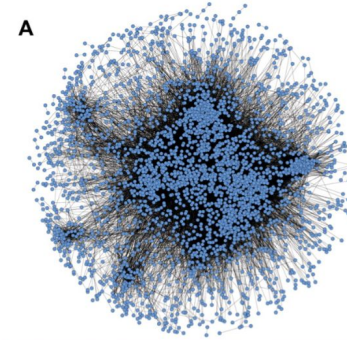
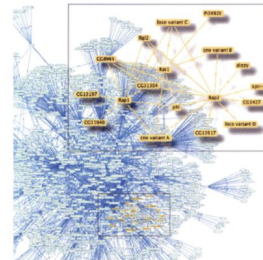
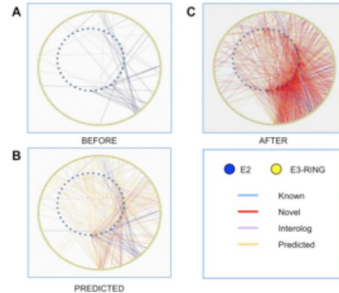
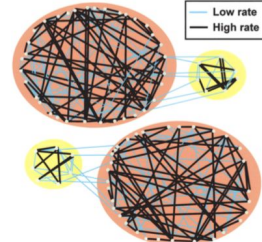
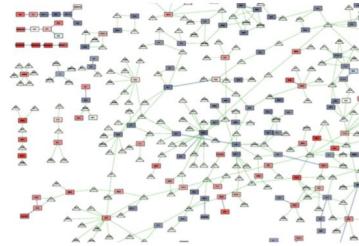
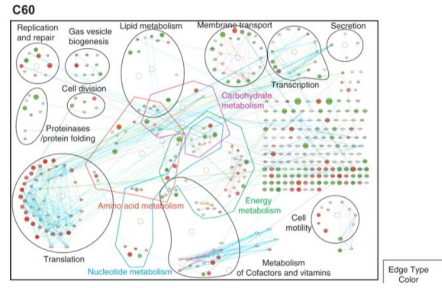
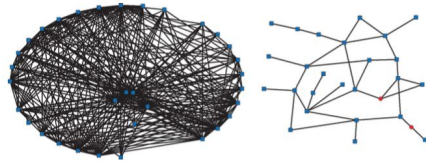
### 8. [RECIPES](#)

1. [Microbial Genome](#)
2. [Link Geometry – Detailed Bezier Control](#)
3. [Labeling Karyotype Bands](#)
4. [Image Maps](#)
5. [Image Transparency and Background](#)
6. [Complex Histograms](#)
7. [Variable Radius Link Ends](#)
8. [Stacked Histograms](#)
9. [Transparent Links](#)
10. [ID Fields](#)
11. [Heat Map Links](#)
12. [Inverted Links](#)
13. [Copy Number Data](#)
14. [Directed Links](#)
15. [Pattern Fills](#)
16. [Automating Tracks](#)
17. [Automating Heatmaps](#)
18. [Circular Stacked Bar Plots](#)
19. [Cortical Maps—Connectograms](#)
20. [Cell Cycle—Part 1](#)
21. [Cell Cycle—Part 2](#)
22. [Nature Cover Encode Diagram](#)
23. [Naming Names](#)

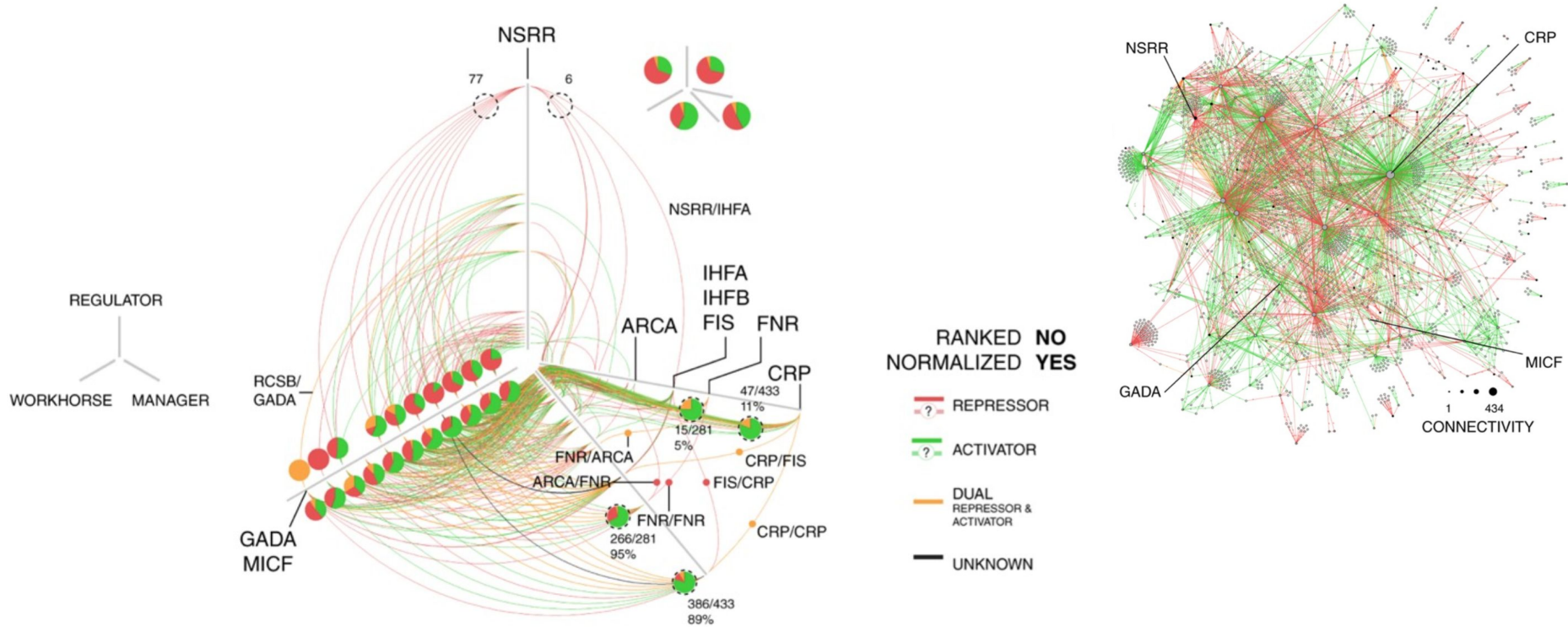
### 9. [HELPER TOOLS](#)

1. [Generating Random Link Data](#)
2. [Reordering Ideograms to Minimize Overlap](#)
3. [Bundling Links](#)
4. [Filtering Links](#)
5. [Visualizing Tabular Data](#)
6. [Generate Link Density Tracks](#)
7. [Visualizing Categorical Data](#)

# Visualizing Networks – The Other Problem

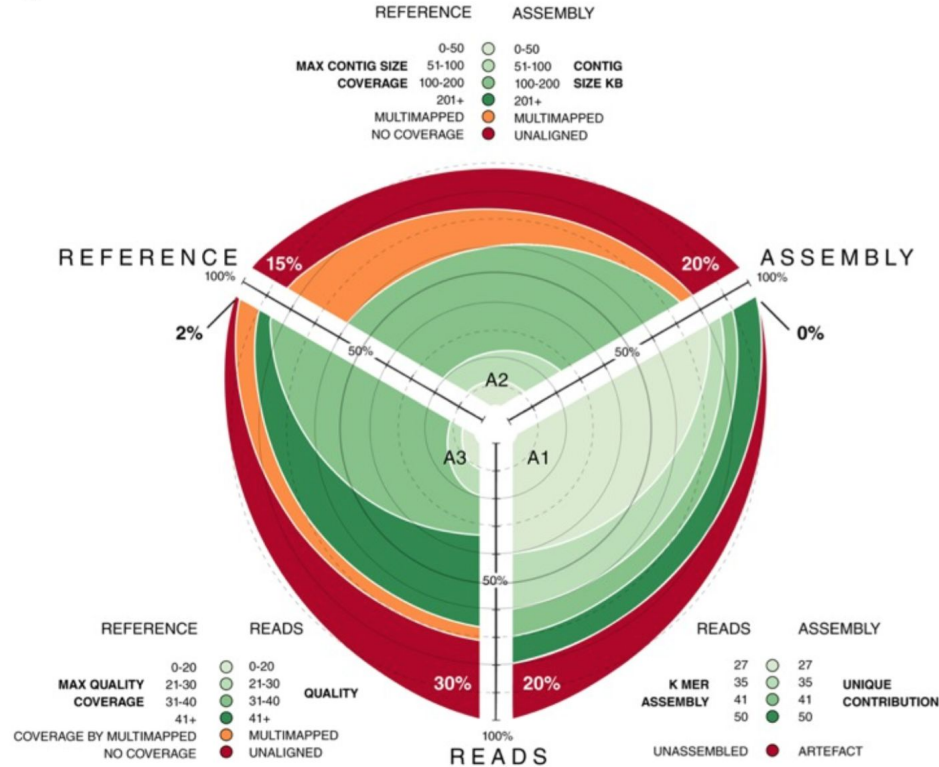


# Visualizing Networks – Hive Plots!



# Visualizing Networks – Hive Plots!

A



# Visualizing Networks – Hive Plots

Martin Krzywinski // [Circos](#) / [Genome Paths](#) / [Genome Informatics 2010](#) / [Presidential Debates](#) / [HDTR](#) / [Schemaball](#) / [4ness of π](#) / [GSC 10th](#) / [clock](#) / [photography](#) / [spam poetry](#) / [ascii](#) / [LOTRO](#)



## HIVE PLOTS

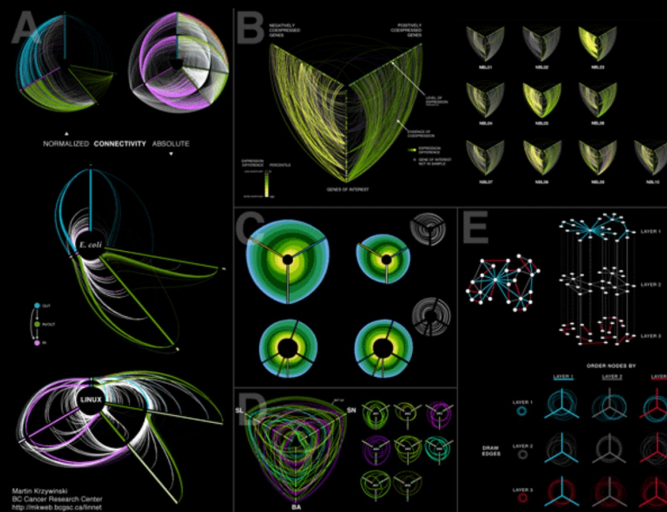
### RATIONAL NETWORK VISUALIZATION — FAREWELL TO HAIRBALLS

Martin Krzywinski, Genome Sciences Center, Vancouver, BC



#### PUBLISHED IN BRIEFINGS IN BIOINFORMATICS

Krzywinski M, Birol I, Jones S, Marra M (2011). [Hive Plots — Rational Approach to Visualizing Networks](#). Briefings in Bioinformatics (early access 9 December 2011, doi: 10.1093/bib/bbr069). ([download citation](#))



THE HIVE PLOT IS A PERCEPTUALLY UNIFORM AND SCALABLE LINEAR LAYOUT VISUALIZATION FOR NETWORK VISUAL ANALYTICS

#### UNDERSTANDING NETWORK STRUCTURE WITH HIVE PLOTS.

(A) Normalized (top) and absolute (bottom) connectivity of *E. coli* gene regulatory network and Linux function call network (Yan *et al.*)

(B) Gene co-regulation networks in neuroblastoma samples.

(C) Network edges shown as ribbons creating circularly composited stacked bar plots (a periodic streamgraph).

(D) Syntentic network of three modern crucifer species to ancestral genome.

(E) Layered network correlation matrix. In each cell two layers  $u, v$  are depicted with  $u$  used to order axes and nodes while links for  $v$  are shown.

[ZOOM](#) [GET SLIDES](#)

We're interested in how you apply this network visualization method to your data set—[let us know](#).

**NEW** [Join the discussion](#) (Rich Morin) about hive plots in d3.js ([demo](#), [github](#)). New to hive plots? See this [Useful d3.js + hive plot intro](#) by Mike Bostock.

**ABOUT**

<http://www.hiveplot.com/>

# Hive Plots



- Online: HiveGraph 1.0 @ <http://wodaklab.org/hivegraph/graph>
- Python API: pyveplot 0.6 @ <https://pypi.python.org/pypi/pyveplot/>
- R library: HiveR @ <http://academic.depauw.edu/~hanson/HiveR/HiveR.html>
- Java GUI: jhive @ <https://www.bcgsc.ca/wiki/display/jhive/home>

# Further Reading



NATURE METHODS | METHAGORA

## Data visualization: A view of every Points of View column

30 Jul 2013 | 8:08 AM | Posted by [Daniel Evanko](#) | Category: [Featured](#), [Visualization](#)

**We've organized all the Points of View columns on data visualization published in *Nature Methods* and provide this as a guide to accessing this trove of practical advice on visualizing scientific data.**

As of July 30, 2013 *Nature Methods* has published 35 Points of View columns written by Bang Wong, Martin Krzywinski and their co-authors: Nils Gehlenborg, Cydney Nielsen, Noam Shores, Rikke Schmidt Kjærgaard, Erica Savig and Alberto Cairo. As we prepare to launch a new column in our September issue we felt this would be a good time to collect and organize links to all the Points of View articles together in one place to make it easier to navigate this wonderful resource that the authors have provided us. For the month of August we will be making all the columns free to access so everyone can benefit from this practical advice on data visualization.

This should not be the end of the Points of View column though. We will be inviting new visualization experts to author articles on new topics that have not been covered so far or which can be expanded on. This page will be continuously updated whenever a new article is published so stay tuned. If you have a suggestion for a topic you would like to see covered in a future points of view article please comment below.

**Update of March 28, 2015:** A PDF eBook of the 38 Points of View articles published between August 2010 and February 2015 is now available at the [Nature Shop](#) for \$7.99 under the title "Visual strategies for biological data: the collected Points of View". The article summaries below provide a nice overview of what is contained in that eBook collection.

<http://blogs.nature.com/methagora/2013/07/data-visualization-points-of-view.html>