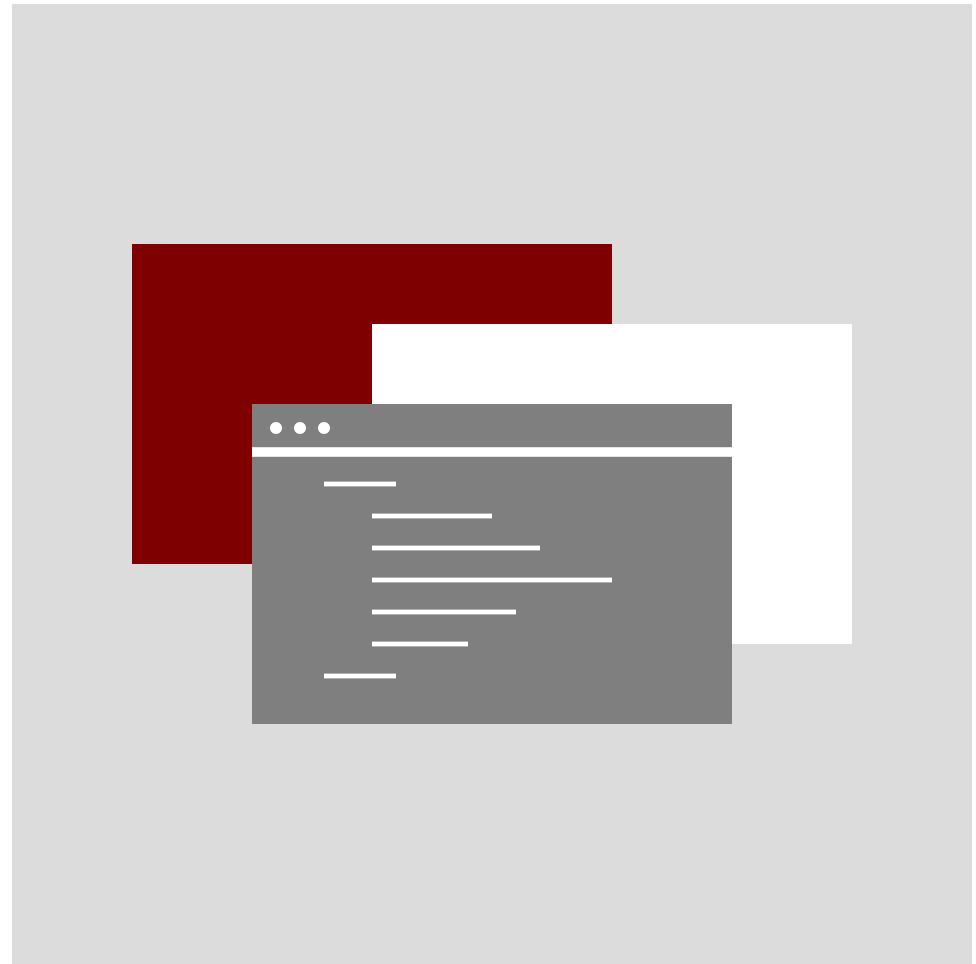


# decksh

a little language for decks



Anthony Starks  
@ajstarks

*A language is any mechanism to express intent, and the input to many programs can be viewed profitably as statements in a language. This column is about those “little languages.”*

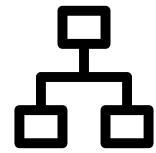
Jon Bentley, Little Languages, Communications of the ACM, August 1986

# Deck



a Go package for presentations

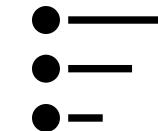
# *Elements*



Structure



Text



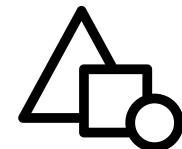
Lists



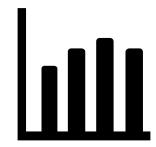
Arrows



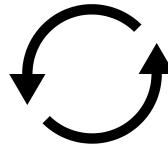
Images



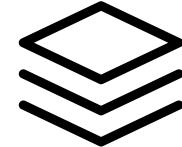
Graphics



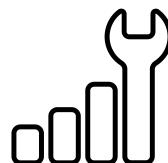
Charts



Loops



Data



Utility

# decksh

# deck markup

# SVG

# PDF

# PNG

```

deck
  slide "rgb(250,250,250)" "black"
    ctext "Deck elements" 50 90 5
    image "follow.jpg" 70 50 640 480 50
    blist 10 75 3
      li "text, image, list"
      li "rect, ellipse, polygon"
      li "line, arc, curve"
    elist

    gy=10
    rect 15 gy 8 6 "rgb(127,0,0)"
    ellipse 27.5 gy 8 6 "rgb(0,127,0)"
    line 50 gy 60 gy
    curve 80 gy 95 30 90 gy
    arc 70 gy 10 8 0 180 0.1 "rgb(0,0,127)"
    polygon "37 37 45" "13 7 10" "rgb(0,0,127)"

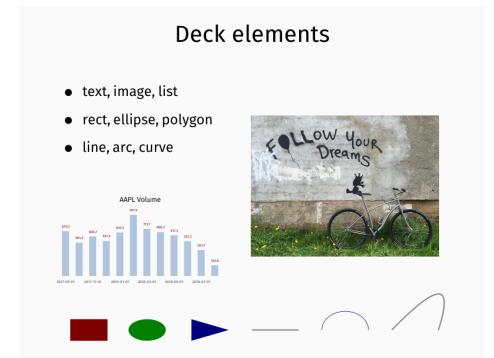
    opts="-fulldeck=f -textsize 1 - xlabel=2 -barwidth 1.5"
    dchart -left 10 -right 42 -top 42 -bottom 25 opts AAPL.d
  eslide
edeck

```

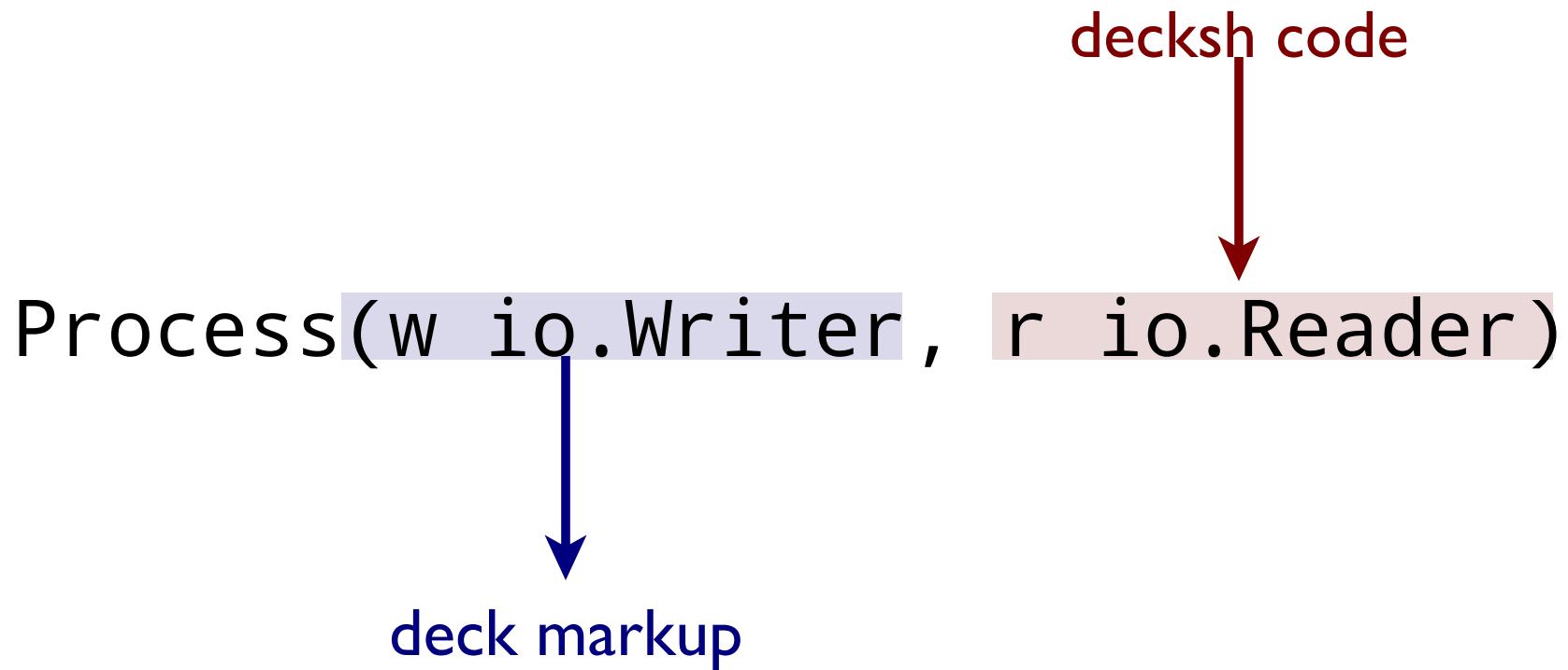
```

<deck>
<slide bg="rgb(250,250,250)" fg="black">
<text align="c" xp="50" yp="90" sp="5">Deck elements</text>
<image name="follow.jpg" xp="70" yp="50" width="640" height="480" scale="50" />
<list type="bullet" xp="10" yp="75" sp="3">
<li>text, image, list</li>
<li>rect, ellipse, polygon</li>
<li>line, arc, curve</li>
</list>
<rect xp="15" yp="10" wp="8" hp="6" color="rgb(127,0,0)" />
<ellipse xp="27.5" yp="10" wp="8" hp="6" color="rgb(0,127,0)" />
<line xp1="50" yp1="10" xp2="60" yp2="10" />
<curve xp1="80" yp1="10" xp2="95" yp2="30" xp3="90" yp3="10" />
<arc xp="70" yp="10" wp="10" hp="8" a1="0" a2="180" sp="0.1" color="rgb(0,0,127)" />
<polygon xc="37 37 45" yc="13 7 10" color="rgb(0,0,127)" />
<text xp="26.00" yp="45.60" sp="1.50" align="center" wp="0.00" font="sans" opacity="100.00" color="black" type="">AAPL Volume</text>
<text xp="26.00" yp="26.00" sp="1.50" align="center" wp="0.00" font="sans" opacity="100.00" color="lightsteelblue" />
<text xp="10.00" yp="38.46" sp="0.75" align="center" wp="0.00" font="sans" opacity="100.00" color="rgb(127,0,0)" type="">679.9</text>
<text xp="10.00" yp="23.00" sp="0.80" align="center" wp="0.00" font="sans" opacity="100.00" color="rgb(75,75,75)" type="">2017-09-01</text>
<line xp1="12.91" yp1="25.00" xp2="12.91" yp2="34.24" sp="1.50" opacity="100.00" color="lightsteelblue" />
<text xp="12.91" yp="35.24" sp="0.75" align="center" wp="0.00" font="sans" opacity="100.00" color="rgb(127,0,0)" type="">504.3</text>
...
</slide>
</deck>

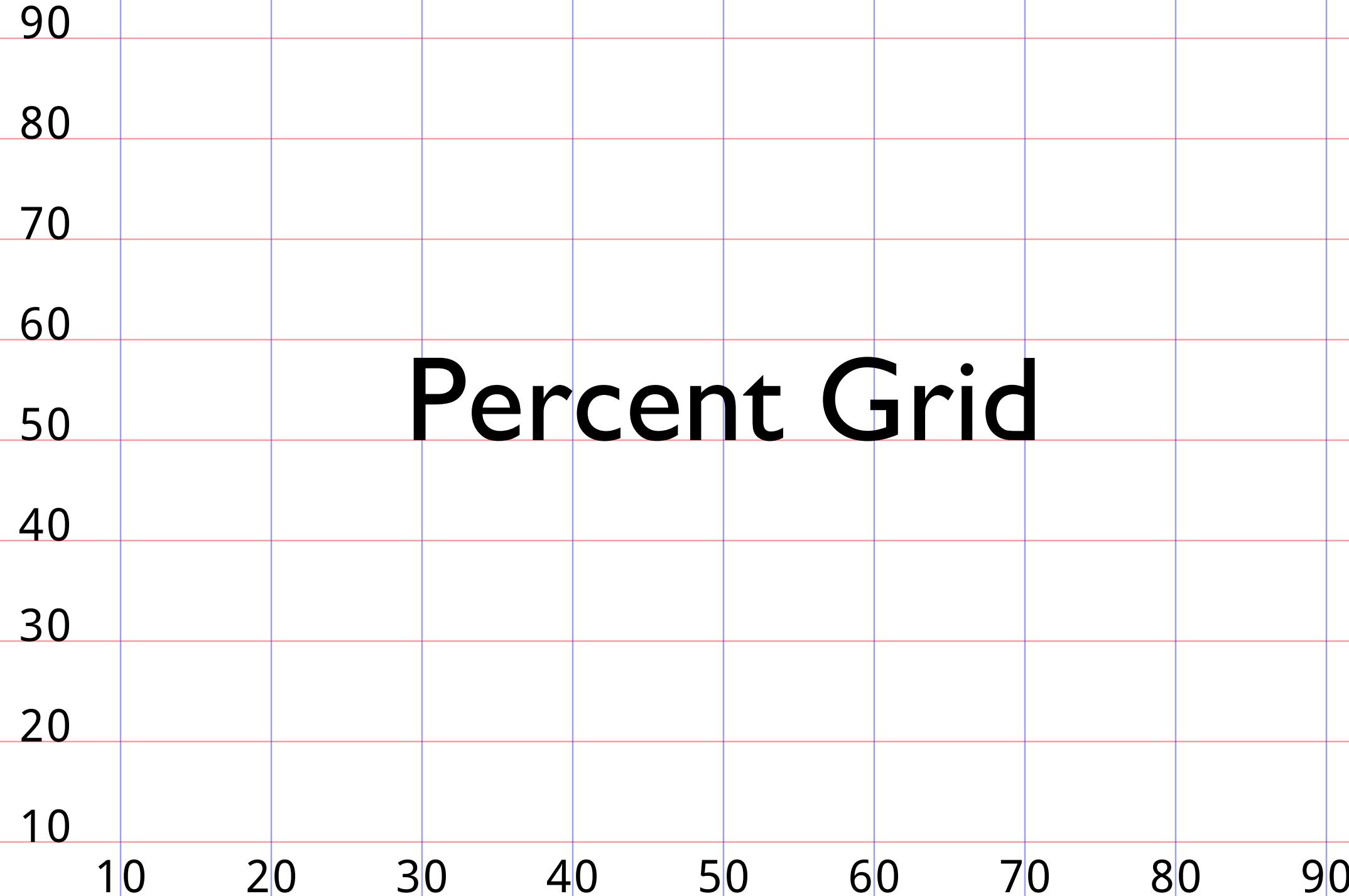
```



# *decksh API*



# Percent Grid



```
// hello world
deck
    slide "black" "white"
        ctext "hello, world" 50 25 10
        circle 50 0 100 "blue"
    eslides
edek
```

# hello, world

# *Running decksh*

decksh	<i>read from stdin, write to stdout</i>
decksh in.dsh	<i>read from file, write to stdout</i>
decksh -o out.xml	<i>read from stdin, write to file</i>
decksh -o out.xml in.dsh	<i>read from file, write to file</i>
chmod +x in.dsh; ./in.dsh	<i>executable deck with #!/path/to/decksh</i>

decksh example.dsh | pdfdeck ...

hw.dsh - Visual Studio Code

File Edit Selection View Go Debug Terminal Help

hw.dsh

```
// hello world
deck
    slide "black" "white"
        ctext "hello, world" 50 25 10
        circle 50 0 100 "blue"
    eslide
edock
```

PROBLEMS TERMINAL ...

1: bash

```
$ decksh hw.dsh | pdf
$ open f.pdf
$
```

Ln 8, Col 1 Tab Size: 4 UTF-8 LF Plain Text

master\* 0 0 0 0



hw.dsh - Visual Studio Code

File Edit Selection View Go Debug Terminal Help

hw.dsh

```
// hello world
deck
    slide "black" "white"
        ctext "hello, Mars" 50 25 10
        circle 50 0 100 "red"
eslide
edeck
```

PROBLEMS TERMINAL ...

```
1: bash
$ decksh hw.dsh | pdf
$ open f.pdf
$ decksh hw.dsh | pdf
$
```

master\* 0 0 ▲ 0 Ln 5, Col 29 Tab Size: 4 UTF-8 LF Plain Text ☺ 🔔



# *Keywords and arguments*

`text "string....." x y n [font][color][op]`

`text "hello, world" 80 50 2` *hello, world*

`text "hello, world" 80 40 2 "serif"` *hello, world*

`text "hello, world" 80 30 2 "serif" "red"` *hello, world*

`text "hello, world" 80 20 2 "serif" "red" 50` *hello, world*

# *Variables and Assignments*

```
x=10                                // number assignment  
y=20  
factor=2  
what="hello world"                   // string assignment  
  
size=x/factor                        // assignment with binop  
text what x y size                  // text "hello world" 10 20 5  
  
y-=10                               // assignment operation  
size+=factor                         // assignment op, substitute  
text what x y size                  // text "hello world" 10 10 7  
  
for v=0 100 5                        // loop from 0 to 100 by 5  
    line 100 v 0 v 0.1 "blue"        // blue horizontal lines  
    line v 100 v 0 0.1 "red"        // red vertical lines  
efor
```

# *Keywords*

## Structure    Text

deck	text
edeck	ctext
slide	etext
eslide	rtext
canvas	arctext
include	textblock
grid	textfile
	textcode

## Lists

list
blist
nlist
clist
li
elist

## Graphics and Arrows

acircle	pill	rbrace
arc	polygon	ubrace
circle	rect	dbrace
curve	rrect	arrow
ellipse	square	crarrow
hline	star	clarrow
line	vline	cuarrow
	lbrace	cdarrow

## Images

image
cimage

## Charts

dchart
legend

## Loop

for
efor

## Data

data
edata

## Utility

vmap	polarx	area
random	polary	format

# Structure

```
deck
  slide
    text "first slide" 50 50 2
  eslides
  slide "black" "white"
    include "file.dsh"
    eslides
  edeck
```

deck

slide

text "first slide" 50 50 2

eslide

slide "black" "white"

include "file.dsh"

eslide

edeck

ctext "hello, world" 50 25 10  
circle 50 0 100 "blue"  
for x=20 80 10  
 circle x 75 2  
efor

# *Text*

hello world

text

*x y size [font] [color] [op] [link]*

hello world

# ctext

*x y size [font] [color] [op] [link]*

hello world.

# etext

*x y size [font] [color] [op] [link]*

angle(315) named(25)  
rotate(45) about((35))

# rtext

*x y angle size [font] [color] [op] [link]*

hello there world

arctext

*cx cy radius beg-angle end-angle size [font] [color] [op] [link]*

# Text

## textblock

The quick brown fox  
jump over the lazy  
dog

## textfile

This is the contents  
of a file. it contains lines of text.  
Reading is fundamental.

"text" x y width size [font] [color] [op] [link]

## textcode

```
package main

import "fmt"

func main() {
    fmt.Println("hello, world")
}
```

"filename" x y size [font] [color] [op] [sp]

"filename" x y width size [color]

# Lists

First thing

- First thing

Second thing

- Second thing

Third thing

- Third thing

Fourth

- Fourth

I. First thing

2. Second thing

3. Third thing

4. Fourth

First thing

Second thing

Third thing

Fourth

list

li "..."

elist

blist

li "..."

elist

nlist

li "..."

elist

clist

li "..."

elist

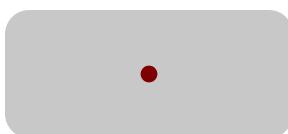
x y size [font] [color] [op] [spacing]

# *Graphics (shapes)*



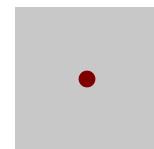
rect

`x y w h [color] [op]`



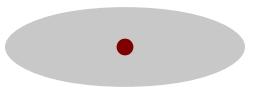
rrect

`x y w h r [color]`



square

`x y w [color] [op]`



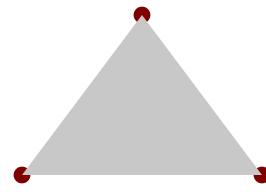
ellipse

`x y w h [color] [op]`



circle

`x y w [color] [op]`



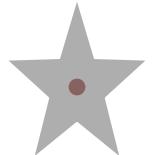
polygon

`"xc" "yc" [color] [op]`



pill

`x y w h [color]`



star

`x y nsides inner outer [color] [op]`

# *Graphics (lines)*



**arc**

*x y w h a1 a2 [lw] [color] [op]*



**curve**

*bx by cx cy ex ey [lw] [color] [op]*



**line**

*x1 y2 x2 y2 [lw] [color] [op]*



**hline**

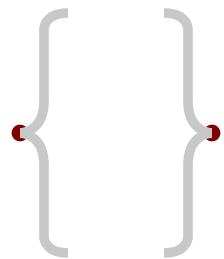
*x y len [lw] [color] [op]*



**vline**

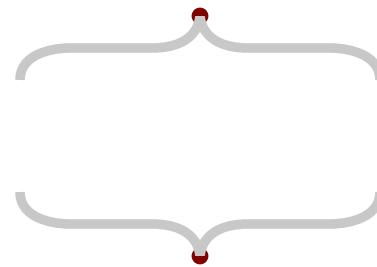
*x y len [lw] [color] [op]*

# Braces



[r-l]brace

*x y size aw ah [lw] [color] [op]*



[u-d]brace

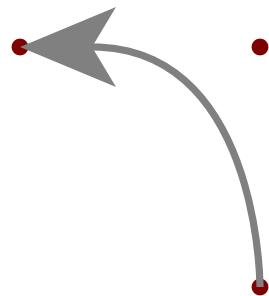
*x y size aw ah [lw] [color] [op]*

# Arrows



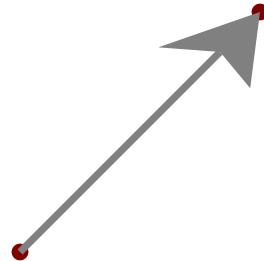
arrow

*x1 y1 x2 y2 [linewidth] [aw] [ah] [color] [op]*



larrow

*x1 y1 x2 y2 x3 y3 [lw] [aw] [ah] [color] [op]*



rcarrow

...



ucarrow

...



darrow

...

# Images



Up in the clouds

image

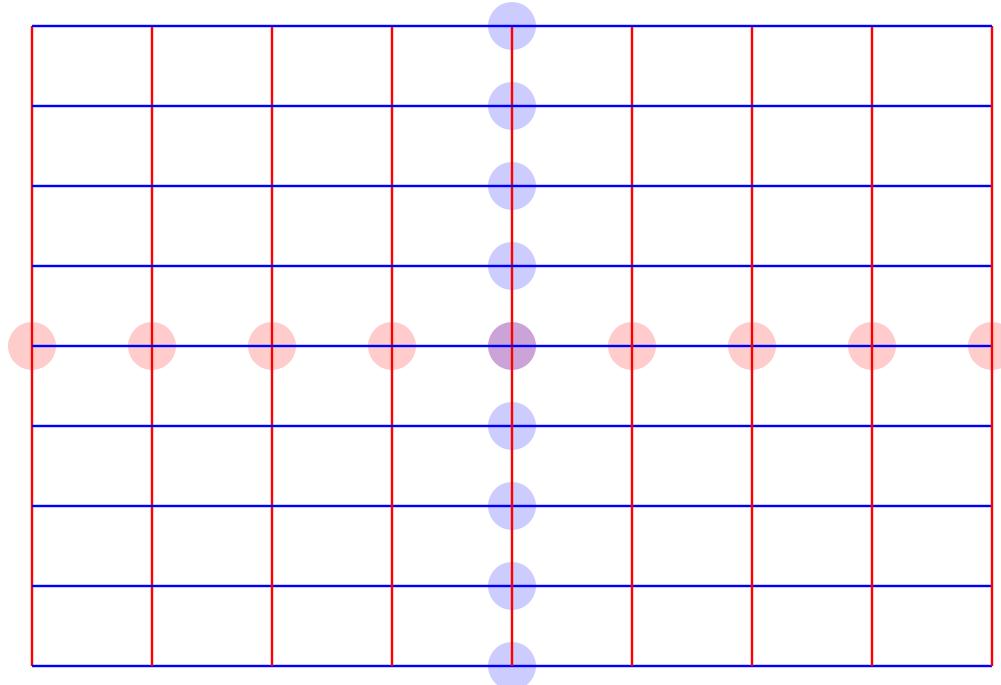
"filename" x y w h [scale] [link]

cimage

"filename" "caption" x y w h [scale] [link] [caption-size]

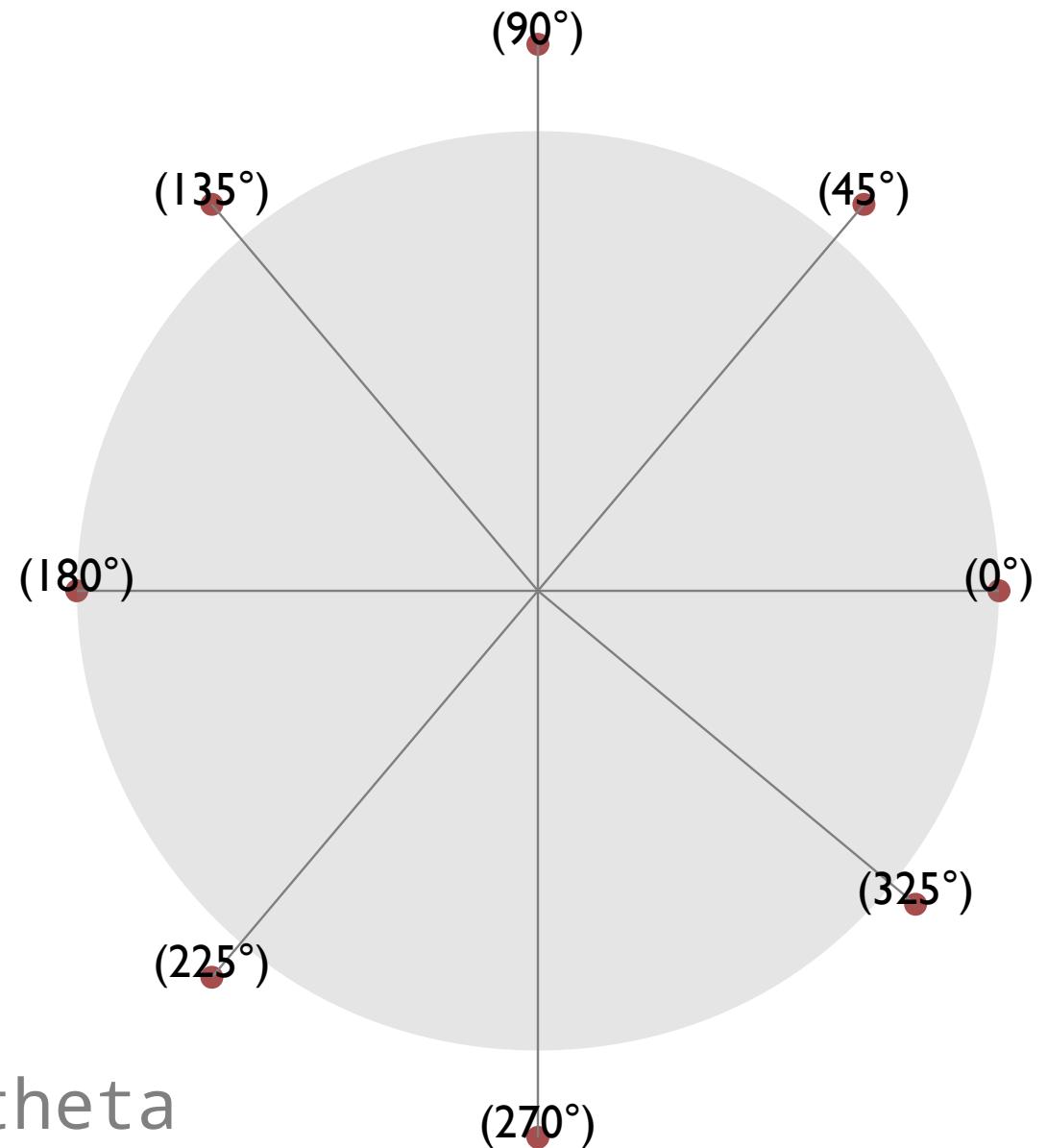
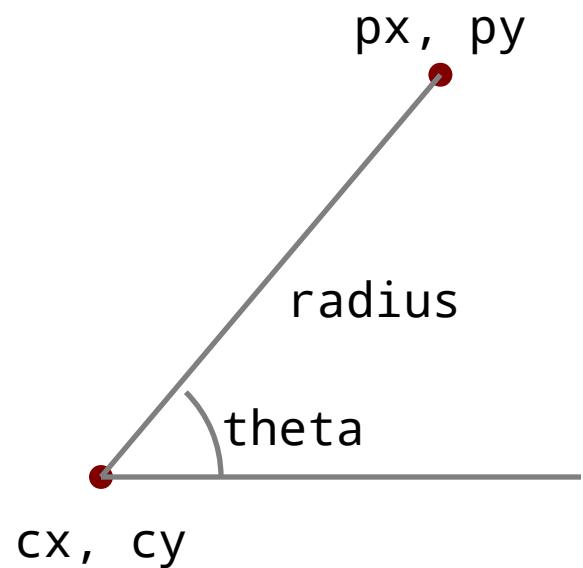
# Loops

```
for v=50 90 5
    vline v 50 40 0.1 "red"
    hline 50 v 40 0.1 "blue"
    circle v 70 2 "red" 20
    circle 70 v 2 "blue" 20
efor
```



**for** v=begin end [increment]  
...v...  
**efor**

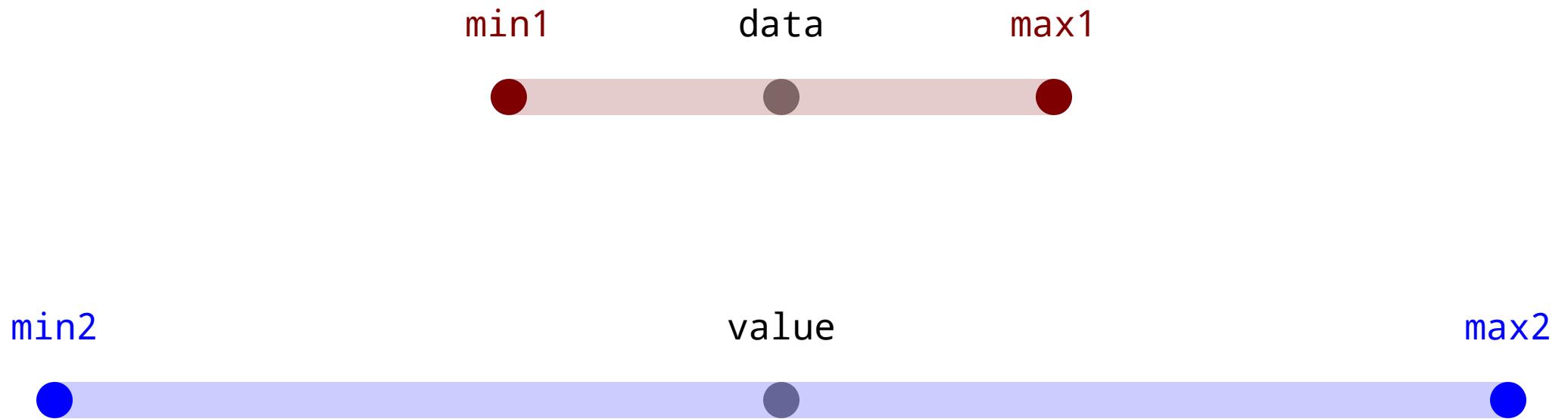
# Polar Coordinates



`px=polarx cx cy radius theta`

`py=polary cx cy radius theta`

# *Mapping Ranges*



`value=vmap data min1 max1 min2 max2`

# *Formatted Strings*

```
v1=100.3
```

```
v2=200.234
```

```
title=format "%.2f Million (USD)" v1
```

```
subtitle=format "Total value: %.2f" v1+v2
```

```
ctext title 80 70 4 "sans" "maroon"
```

```
ctext subtitle 80 60 3 "sans" "gray"
```

100.30 Million (USD)

Total value: 300.53

value=format fmt expression

# *Random Numbers*

```
x1=random 40 70
```

```
y1=random 60 70
```

```
x2=random 40 50
```

```
y2=random 50 60
```

```
x3=random 60 70
```

```
y3=random 35 45
```



x1 ,y1



x2 ,y2



x3 ,y3

value=random min max

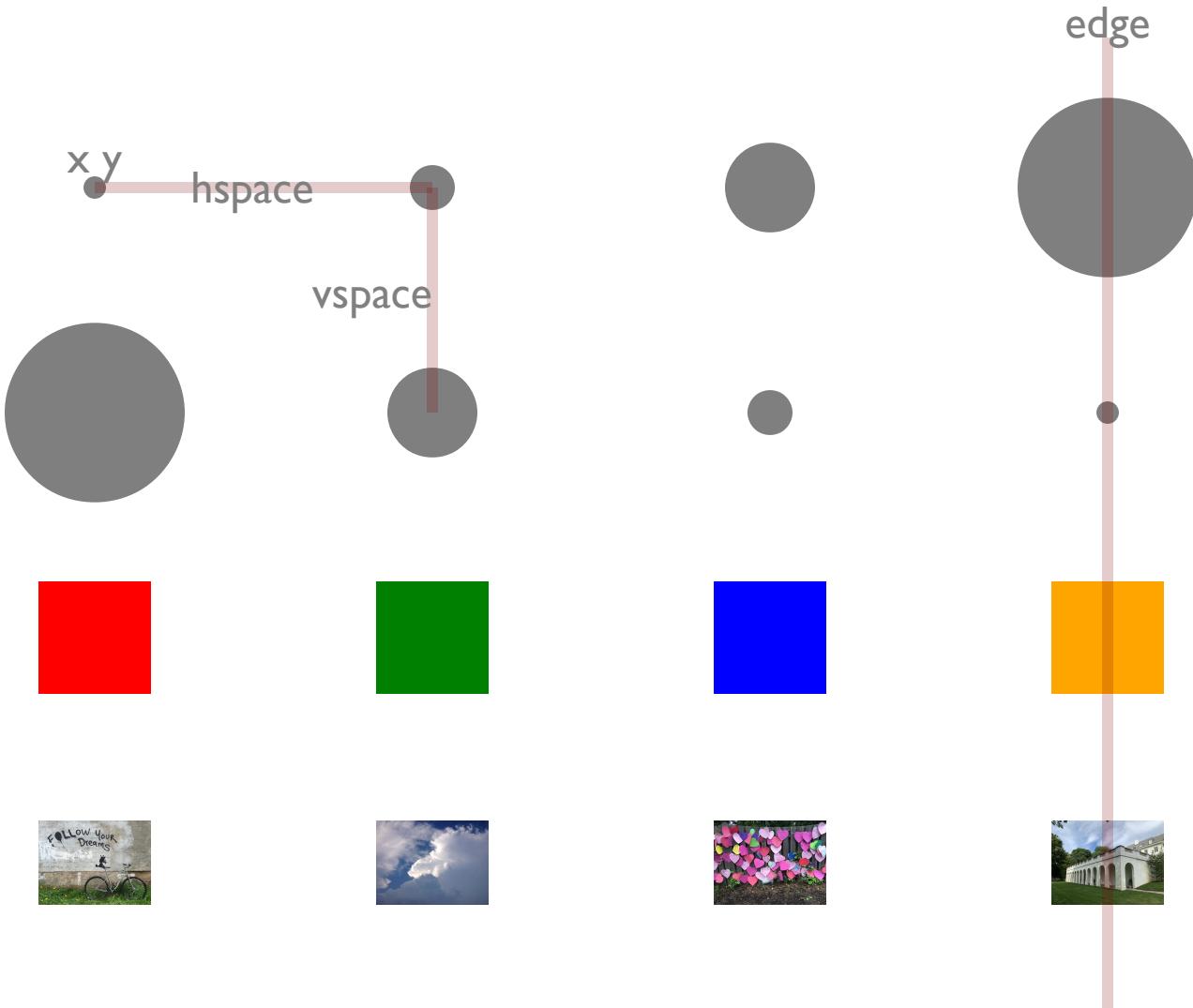
# Flexible Grid

```
circle x y 1  
circle x y 2  
circle x y 4  
circle x y 8
```

```
circle x y 8  
circle x y 4  
circle x y 2  
circle x y 1
```

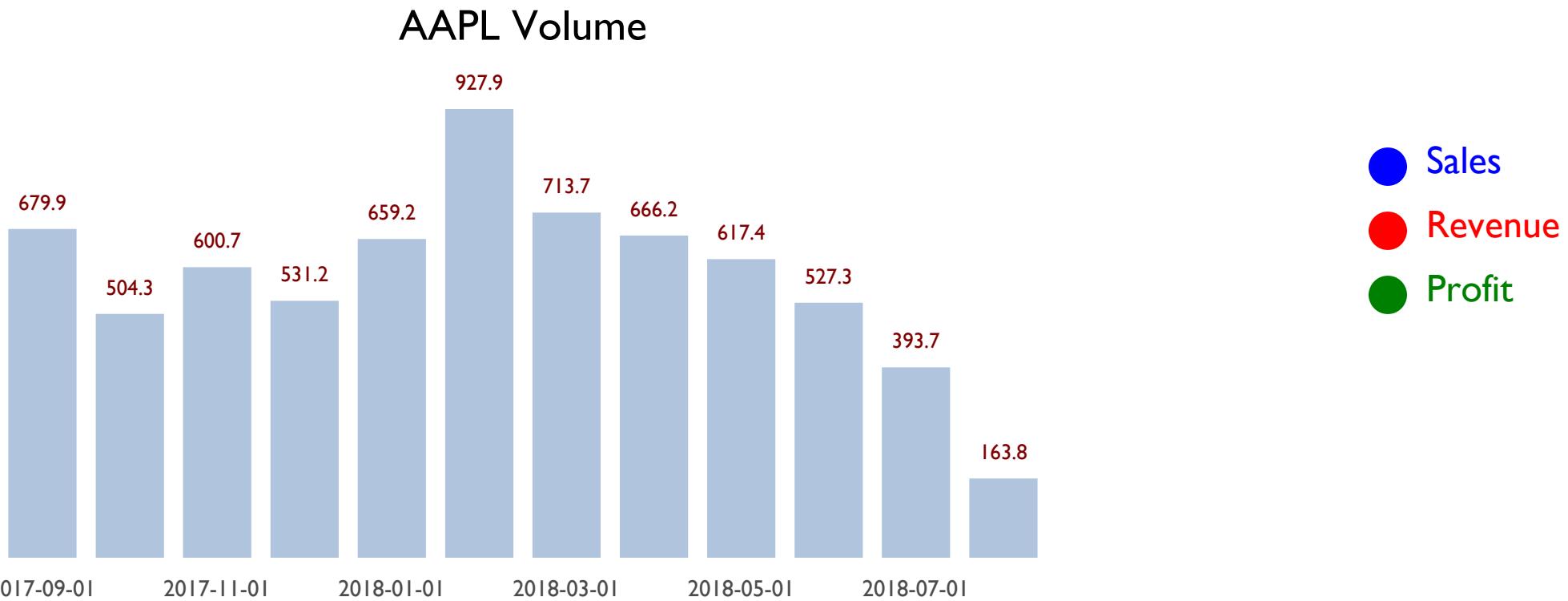
```
square x y 5 "red"  
square x y 5 "green"  
square x y 5 "blue"  
square x y 5 "orange"
```

```
image "images/follow.jpg" x y 640 480 7  
image "images/cloudy.jpg" x y 640 480 7  
image "images/hearts.jpg" x y 640 480 7  
image "images/oldfields.jpg" x y 640 480 7
```



**grid "foo.dsh" x y hspace vspace edge**

# Charts



dchart

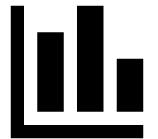
[args]

legend

x y size [font] [color]

- Sales
- Revenue
- Profit

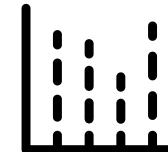
# *dchart types*



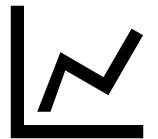
Column



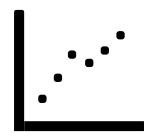
Bar



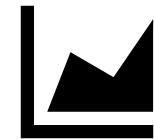
Dot



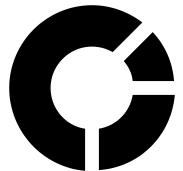
Line



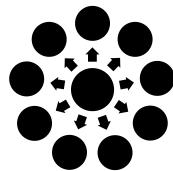
Scatter



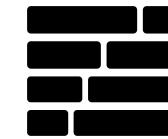
Area



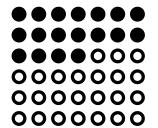
Donut/Pie



Radial



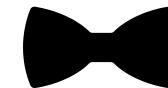
Pmap



Waffle/Lego

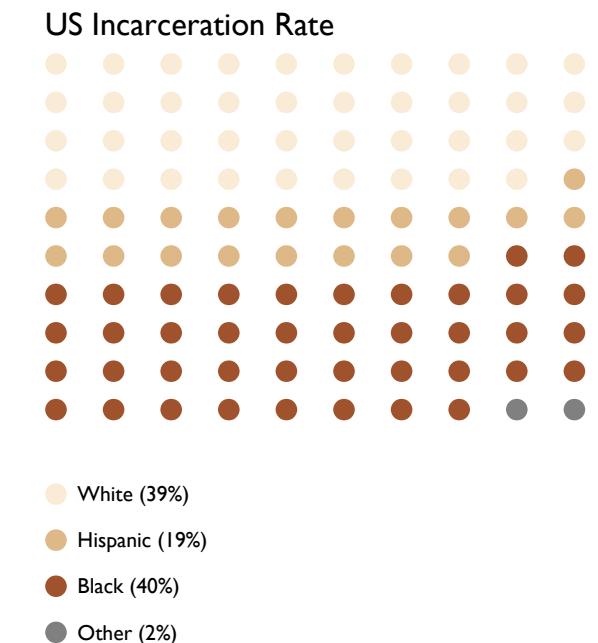
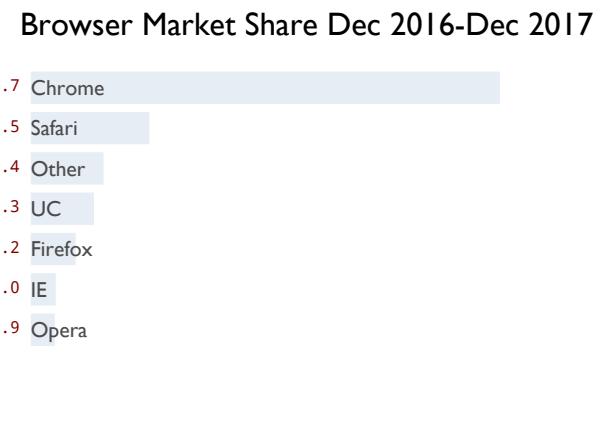
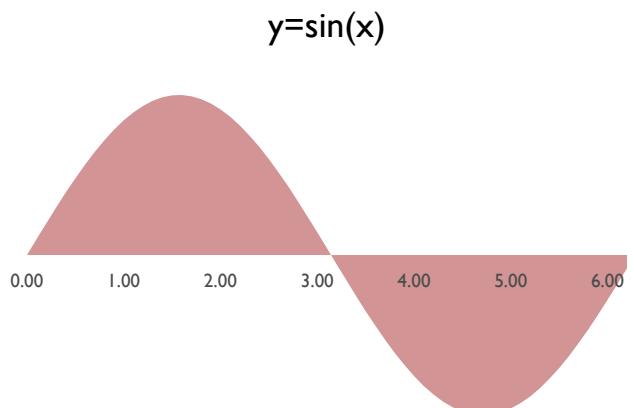
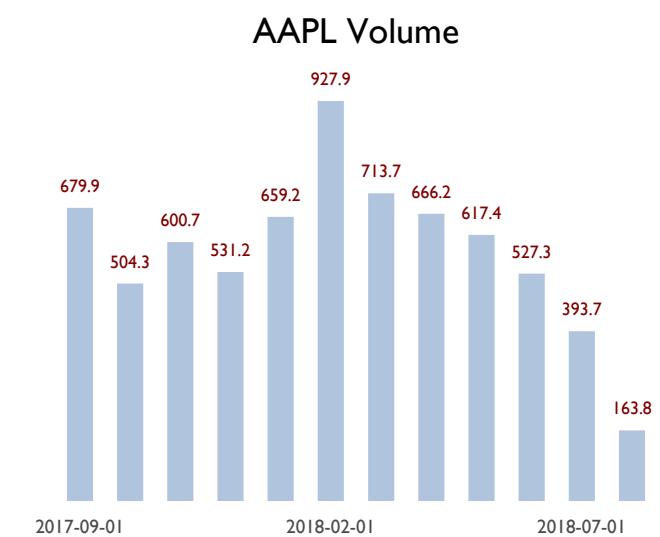
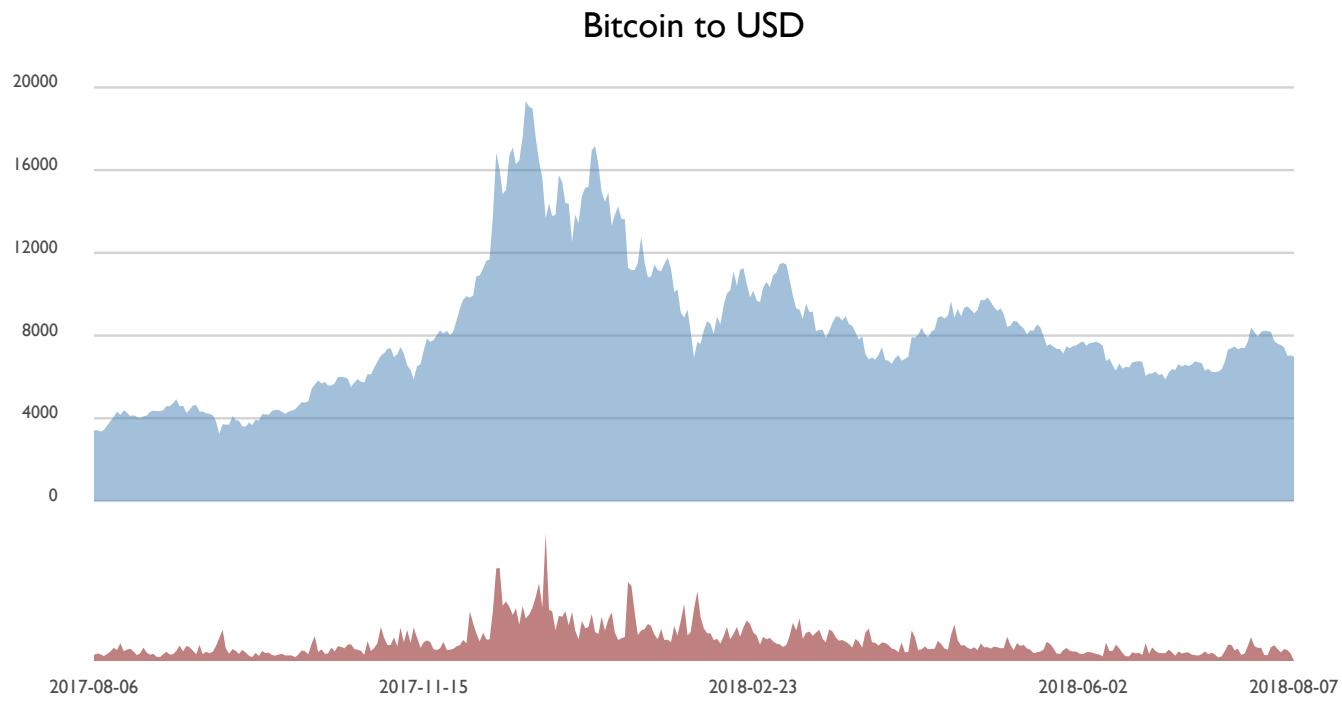


Fan



Bowtie

# *dchart: charts for deck*



```

deck
  slide "rgb(250,250,250)" "black"
    ctext "Deck elements" 50 90 5
    image "follow.jpg"    70 50 640 480 50
    blist 10 75 3
      li "text, image, list"
      li "rect, ellipse, polygon"
      li "line, arc, curve"
    elist

    gy=10
    rect 15 gy 8 6           "rgb(127,0,0)"
    ellipse 27.5 gy 8 6      "rgb(0,127,0)"
    line 50 gy 60 gy
    curve 80 gy 95 30 90 gy
    arc 70 gy 10 8 0 180 0.1 "rgb(0,0,127)"
    polygon "37 37 45" "13 7 10" "rgb(0,0,127)"

    opts="-fulldeck=f -textsize 1 -xlabel=2 -barwidth 1.5"
    dchart -left 10 -right 42 -top 42 -bottom 25 opts AAPL.d
  eslide
edeck

```

## Deck elements

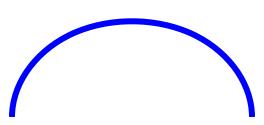
- text, image, list
- rect, ellipse, polygon
- line, arc, curve



# decksh example.dsh | pdf

# Deck elements

- text, image, list
- rect, ellipse, polygon
- line, arc, curve

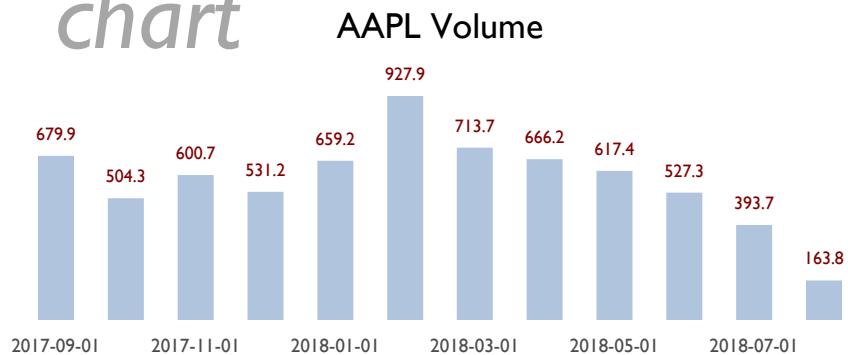


# *text* Deck elements

*list*

- **text, image, list**
- **rect, ellipse, polygon**
- **line, arc, curve**

*chart*



*image*



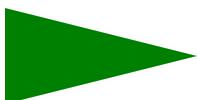
*rect*



*ellipse*



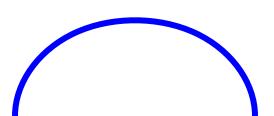
*polygon*



*line*



*arc*



*curve*



# Examples



# Anthony J. Starks

## Art + Code

---



+1 908.548.3403



ajstarks@gmail.com



@ajstarks



github.com/ajstarks



speakerdeck.com/ajstarks

```

deck
  mx=50          // midpoint
  tx=30          // text left
  ix=20          // image left
  ts=10          // base text size
  ss=ts*0.85    // sub-head text size
  cs=ts*0.55    // contact info text size
  ly=58          // line y

  slide "white" "rgb(100,100,100)"
    image "starx.png"           mx 87 512 512 7.5
    ctext "Anthony J. Starks"   mx 70 ts "sans" "black"
    ctext "Art + Code"         mx 62 ss "sans" "maroon"
    line ix ly 80 ly 0.3 "maroon"

    image "phone.png"          ix 50 1200 1200 1.2
    image "email.png"          ix 40 1200 1200 1.2
    image "twitter.png"        ix 30 1200 1200 1.2
    image "github.png"         ix 20 120 120 10
    image "sd.png"             ix 10 512 512 2.5

    text "+1 908.548.3403"     tx 49 cs
    text "ajstarks@gmail.com"   tx 39 cs
    text "@ajstarks"           tx 29 cs
    text "github.com/ajstarks"  tx 19 cs
    text "speakerdeck.com/ajstarks" tx 9 cs
  eslide
edeck

```



## Anthony J. Starks

### Art + Code

---

 +1 908.548.3403

 ajstarks@gmail.com

 @ajstarks

 github.com/ajstarks

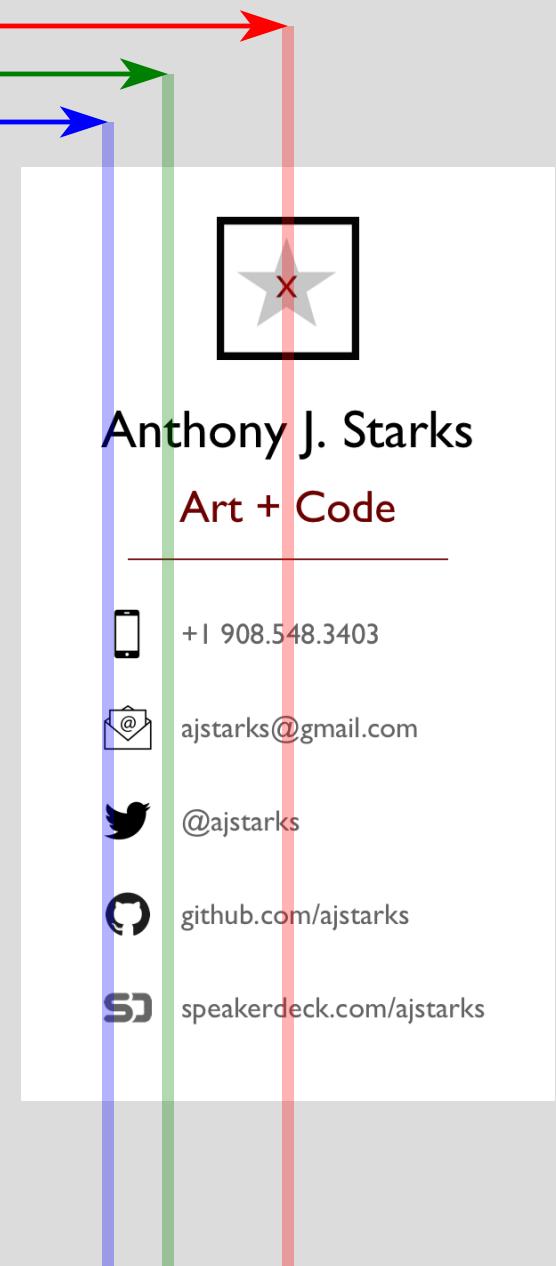
 speakerdeck.com/ajstarks

```
deck
  mx=50          // midpoint
  tx=30          // text left
  ix=20          // image left
  ts=10          // base text size
  ss=ts*0.85    // sub-head text size
  cs=ts*0.55    // contact info text size
  ly=58          // line y

  slide "white" "rgb(100,100,100)"
    image "starx.png"           mx 87 512 512 7.5
    ctext "Anthony J. Starks"   mx 70 ts "sans" "black"
    ctext "Art + Code"         mx 62 ss "sans" "maroon"
    line ix ly 80 ly 0.3 "maroon"

    image "phone.png"          ix 50 1200 1200 1.2
    image "email.png"          ix 40 1200 1200 1.2
    image "twitter.png"        ix 30 1200 1200 1.2
    image "github.png"         ix 20 120 120 10
    image "sd.png"             ix 10 512 512 2.5

    text "+1 908.548.3403"     tx 49 cs
    text "ajstarks@gmail.com"   tx 39 cs
    text "@ajstarks"            tx 29 cs
    text "github.com/ajstarks"  tx 19 cs
    text "speakerdeck.com/ajstarks" tx 9 cs
  eslide
edeck
```



```

deck
  mx=50          // midpoint
  tx=30          // text left
  ix=20          // image left
  ts=10          // base text size
  ss=ts*0.85    // sub-head text size
  cs=ts*0.55    // contact info text size
  ly=58          // line y

  slide "white" "rgb(100,100,100)"
    image "starx.png"           mx 87 512 512 7.5
    ctext "Anthony J. Starks"   mx 70 ts "sans" "black"
    ctext "Art + Code"         mx 62 ss "sans" "maroon"
    line ix ly 80 ly 0.3 "maroon"

    image "phone.png"          ix 50 1200 1200 1.2
    image "email.png"          ix 40 1200 1200 1.2
    image "twitter.png"        ix 30 1200 1200 1.2
    image "github.png"         ix 20 120 120 10
    image "sd.png"             ix 10 512 512 2.5

    text "+1 908.548.3403"     tx 49 cs
    text "ajstarks@gmail.com"   tx 39 cs
    text "@ajstarks"           tx 29 cs
    text "github.com/ajstarks"  tx 19 cs
    text "speakerdeck.com/ajstarks" tx 9 cs
  eslide
edeck

```



## Anthony J. Starks

### Art + Code

---

 +1 908.548.3403

 ajstarks@gmail.com

 @ajstarks

 github.com/ajstarks

 speakerdeck.com/ajstarks

```
deck
  mx=25          // midpoint
  tx=62          // text left
  ix=57          // image left
  ts=6           // base text size
  ss=ts*0.85    // sub-head text size
  cs=ts*0.50    // contact info text size
  lx=50          // line x

  slide "white" "rgb(100,100,100)"
    image "starx.png"           mx 75 512 512 7.5
    ctext "Anthony J. Starks"   mx 35 ts "sans" "black"
    ctext "Art + Code"         mx 22 ss "sans" "maroon"
    line lx 90 lx 10 0.3 "maroon"

    image "phone.png"          ix 80 1200 1200 1.2
    image "email.png"          ix 65 1200 1200 1.2
    image "twitter.png"        ix 50 1200 1200 1.2
    image "github.png"         ix 35 120 120 10
    image "sd.png"              ix 20 512 512 2.5

    text "+1 908.548.3403"     tx 79 cs
    text "ajstarks@gmail.com"   tx 64 cs
    text "@ajstarks"            tx 49 cs
    text "github.com/ajstarks"  tx 34 cs
    text "speakerdeck.com/ajstarks" tx 19 cs

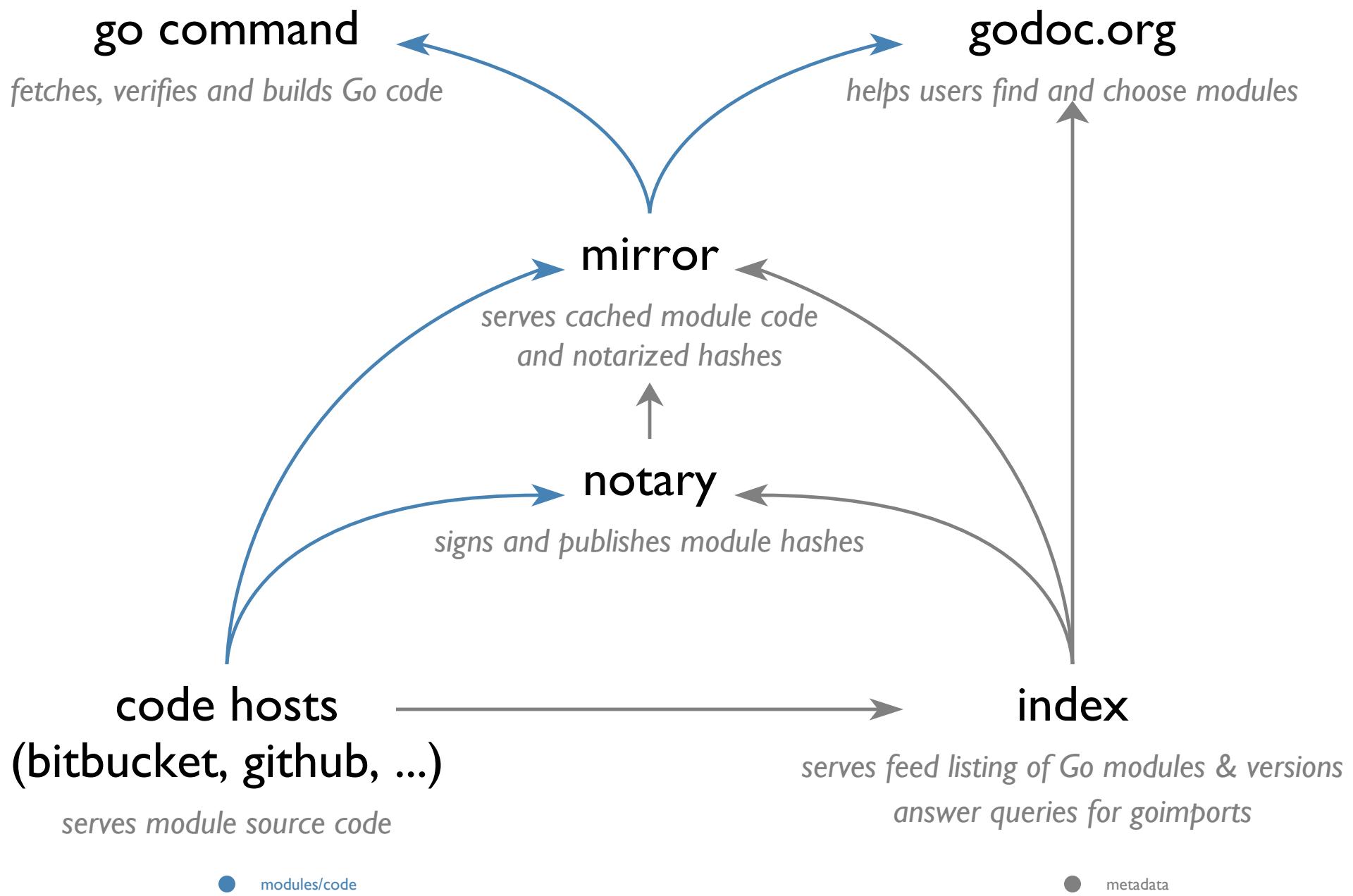
  eslide
edeck
```



**Anthony J. Starks**  
**Art + Code**

+1 908.548.3403  
ajstarks@gmail.com  
@ajstarks  
github.com/ajstarks  
speakerdeck.com/ajstarks

# Go Module Information Flows



BOS



SFO

Virgin America 351

Gate B38

8:35am

On Time

JFK



IND

US Airways 1207

Gate C31C

5:35pm

Delayed

# Flight Information

Los Angeles (LAX)  New York/Newark (EWR)



Distance Traveled

1,958 mi

3,151 km

Distance to Destination

596 mi

798 km



Time to Destination

1:20

Estimated time of arrival

12:14 am

Local time of arrival

12:14 am

Ground speed



Headwind



Outside Temperature

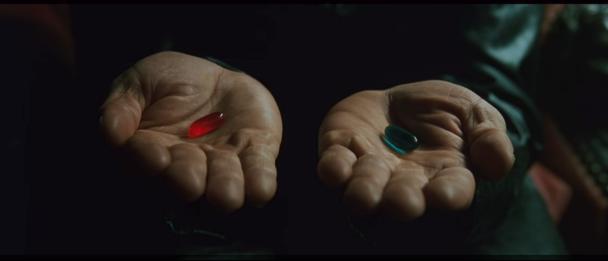


Current Altitude





Pulp Fiction (1993)



The Matrix (1999)



Roma (2018)

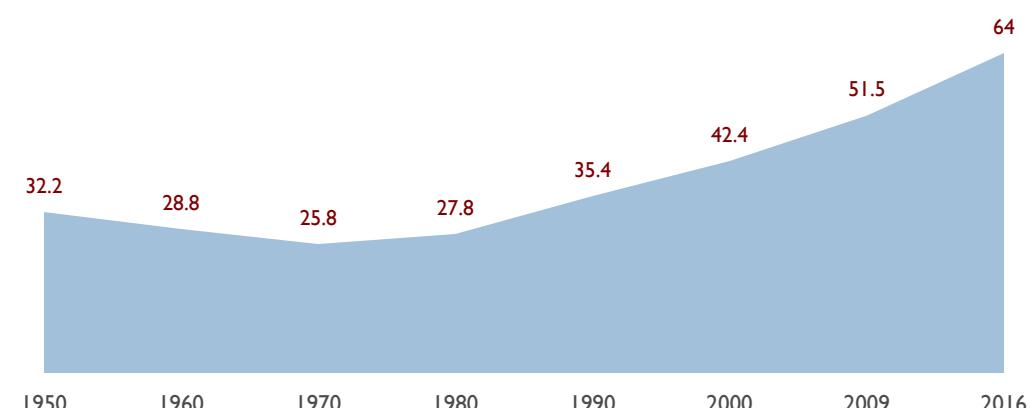
pulp04.png,1920,1080,Pulp Fiction (1993)  
matrix12.png,1920,1080,The Matrix (1999)  
roma04.png,1920,1080,Roma (2018)

caption movies.csv | decksh | pdf ...

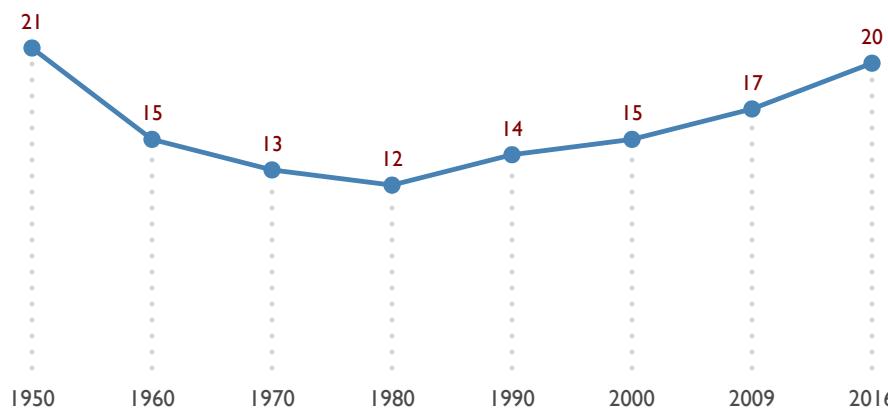
# A record 64 million Americans live in multigenerational households

The number and share of Americans living in multi- generational family households have continued to rise, despite improvements in the U.S. economy since the Great Recession. In 2016, a record 64 million people, or 20% of the U.S. population, lived with multiple generations under one roof, according to a new Pew Research Center analysis of census data.

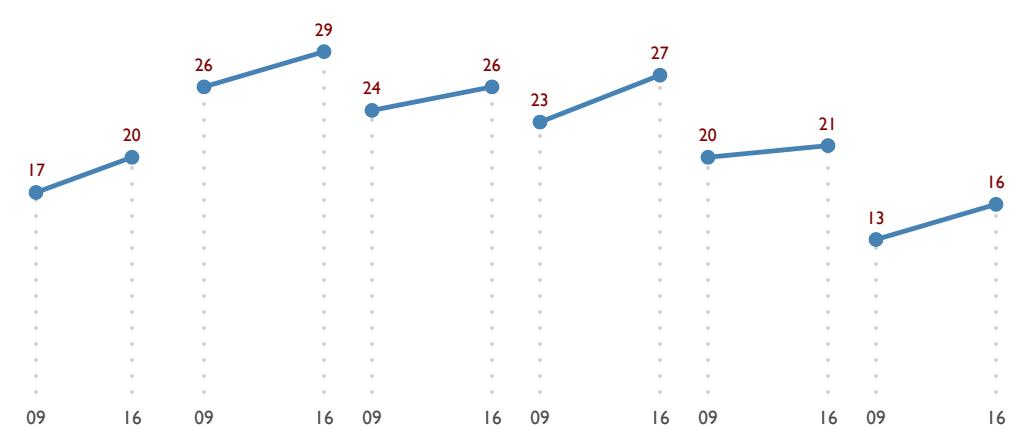
Multigenerational households (millions)



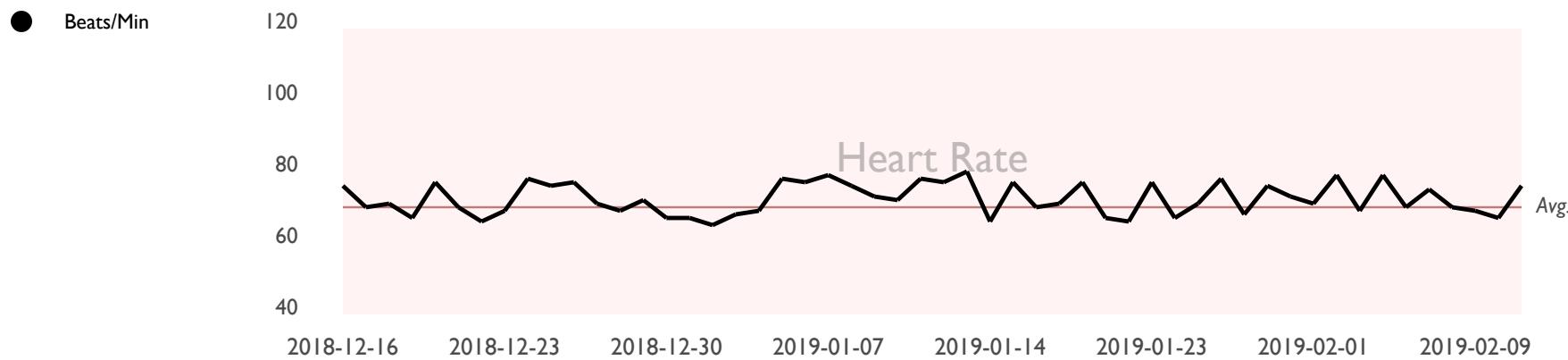
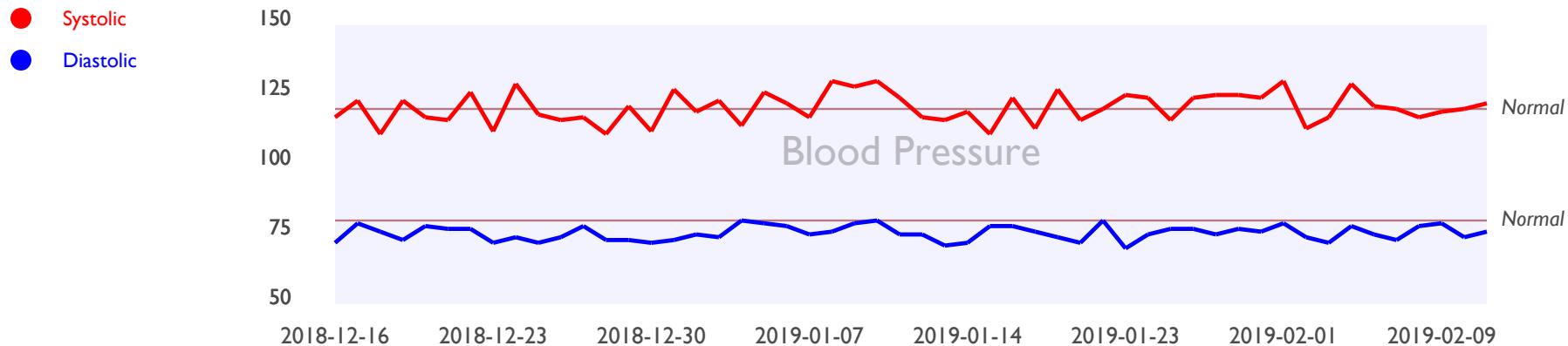
% of Americans in multigenerational households



Total      Asian      Black      Hispanic      Other      White

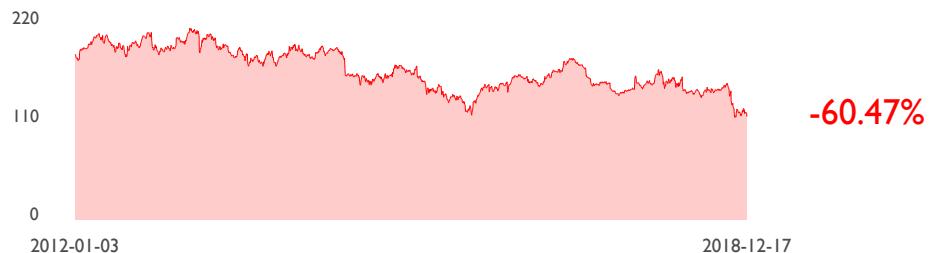


# Jane Doe





Rometty



The first woman to lead IBM, Rometty shifted IBM away from shrinking businesses such as computers and operating system software, and into higher-growth areas like artificial intelligence. Her tenure has also been met by fierce criticism relating to executive compensation bonuses, layoffs, outsourcing, and presiding over 24 consecutive quarters of revenue decline.



Palmisano



Palmisano's mandate was to move into new unique businesses with high profit margins and potential for innovation. This included purchasing PWC Consulting in 2002, so that IBM could go beyond selling computers and software and help customers use technology to solve business challenges in areas such as marketing, procurement and manufacturing.



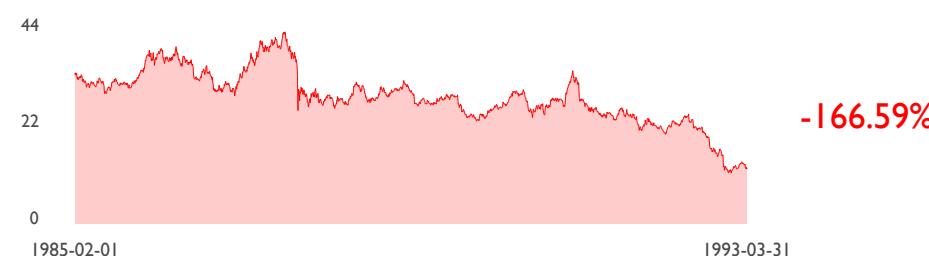
Gerstner



Gerstner's choice to keep the company together was the defining decision of his tenure, as these gave IBM the capabilities to deliver complete IT solutions to customers. Services could be sold as an add-on to companies that had already bought IBM computers, while barely profitable pieces of hardware were used to open the door to more profitable deals.

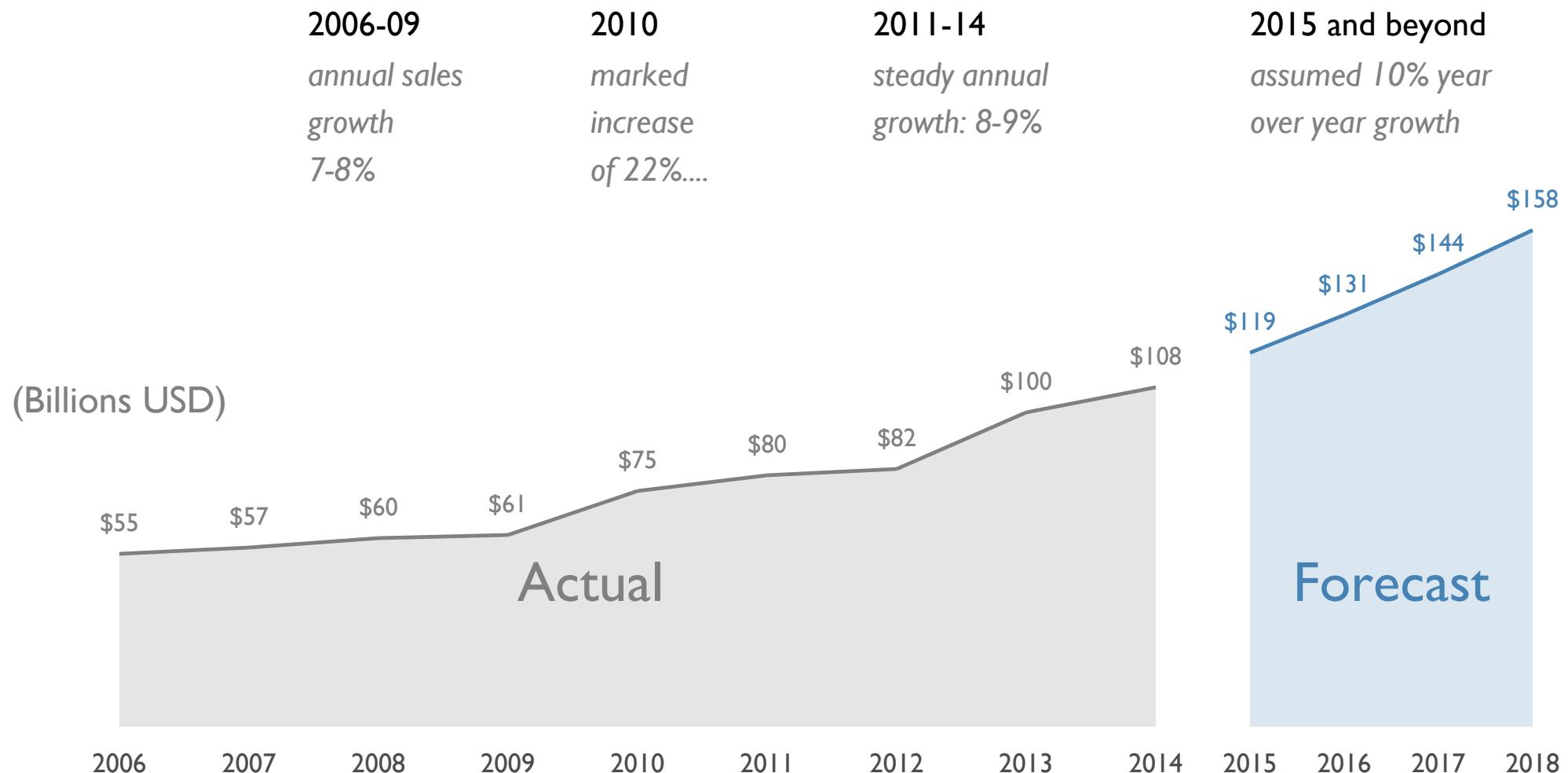


Akers

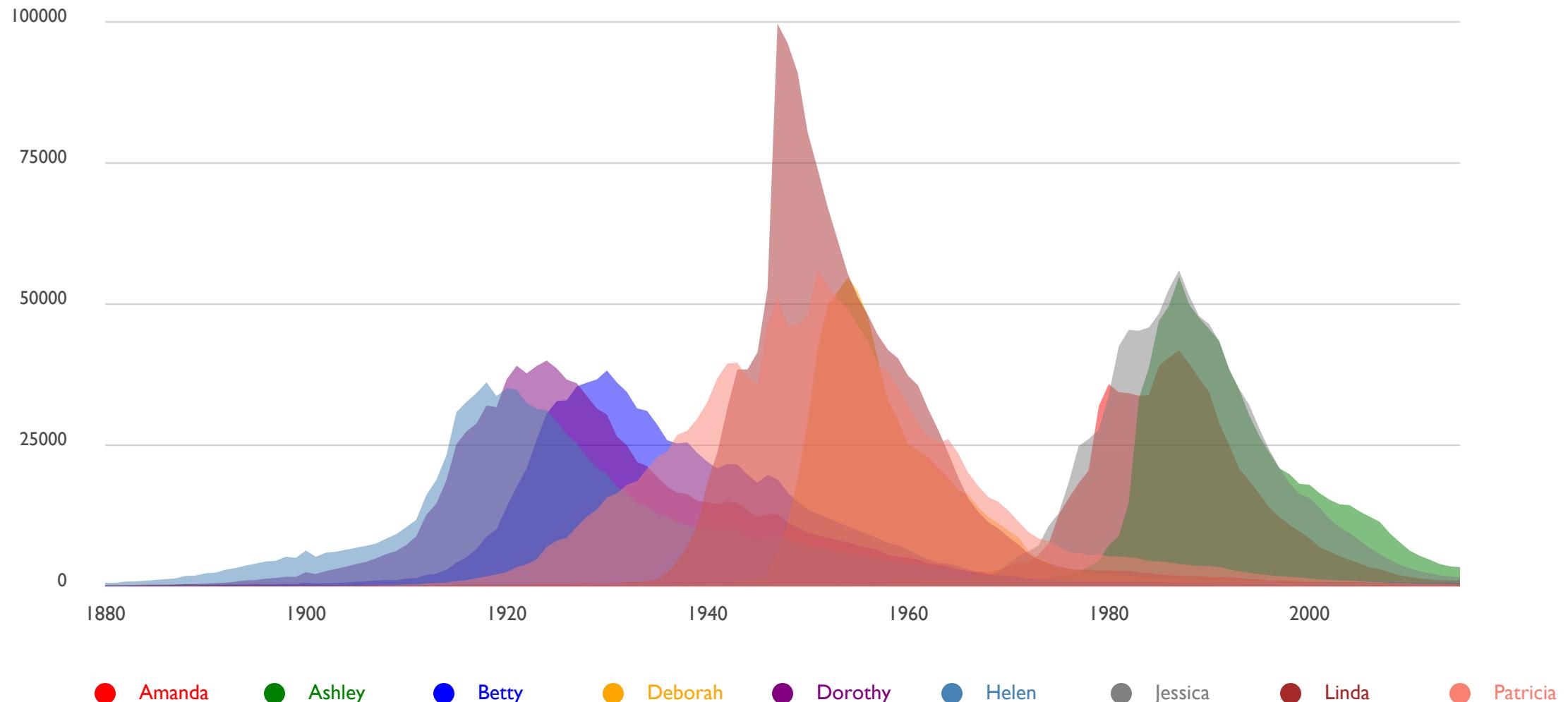


Akers was credited with simplifying the company's bureaucracy to focus more on profits. In a restructuring intended to reverse three years of disappointing performance, he created five new, autonomous organizations responsible for the company's innovation, design and manufacturing. Akers was forced to resign, after the company posted an unprecedented \$5 billion annual loss.

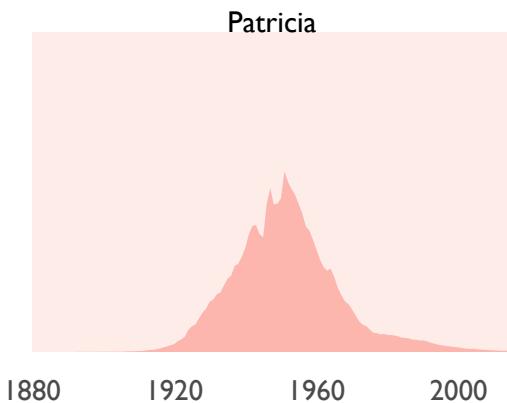
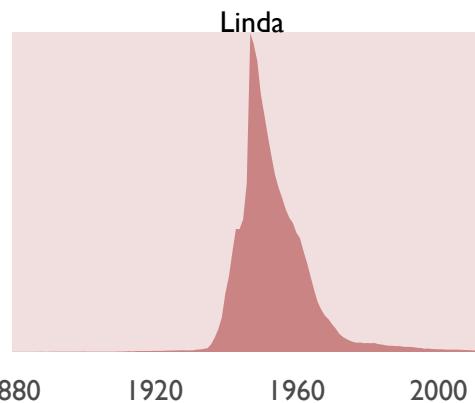
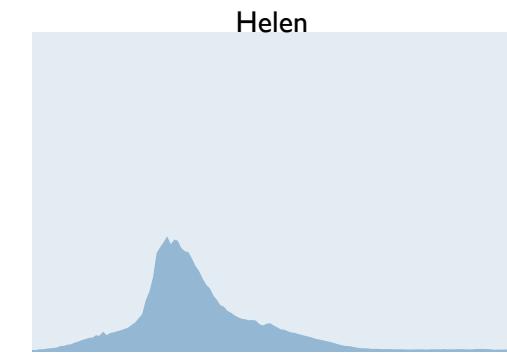
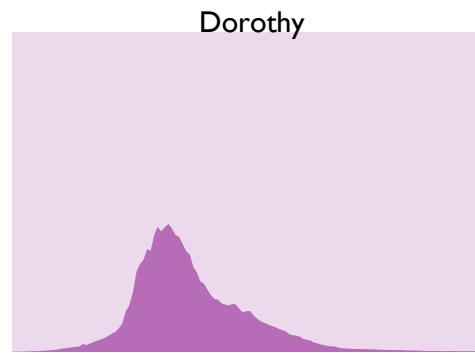
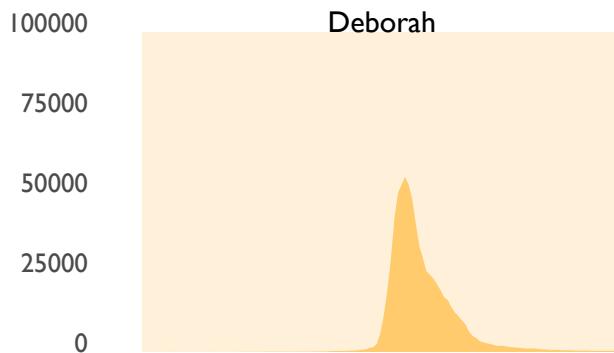
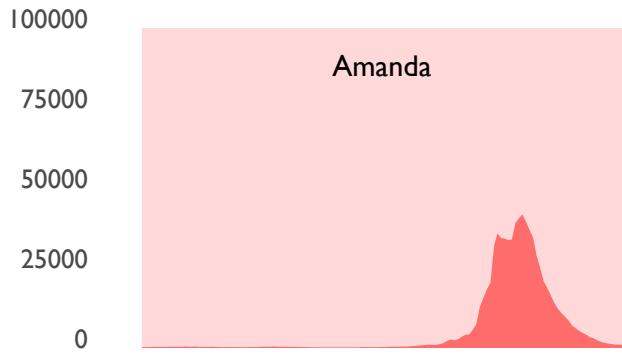
# Sales over time



# Evolution of Baby Names in the US: 1880-2015

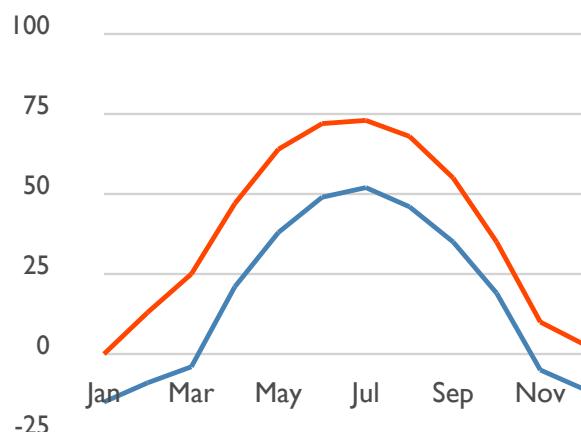


# Evolution of Baby Names in the US: 1880-2015

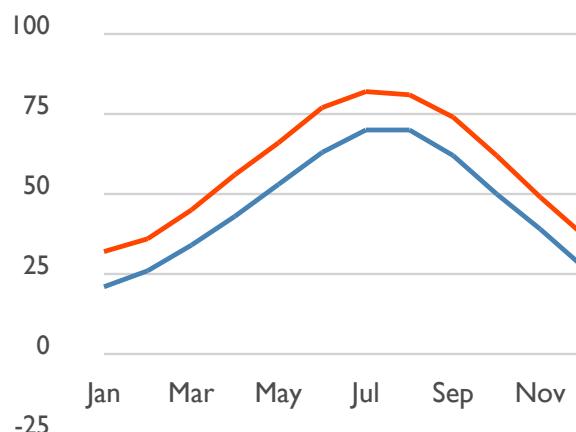


# Average High/Low Temperatures (°F)

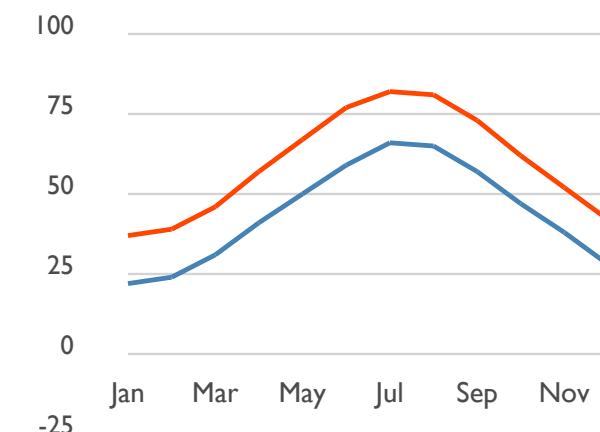
## Fairbanks



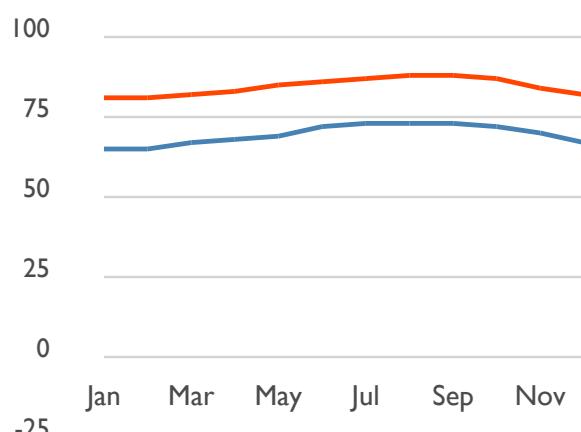
## Chicago



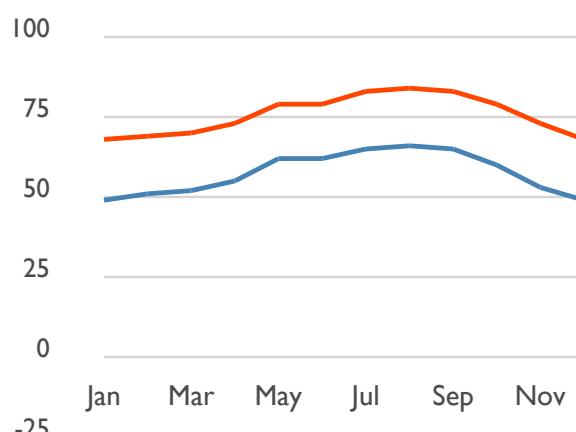
## Boston



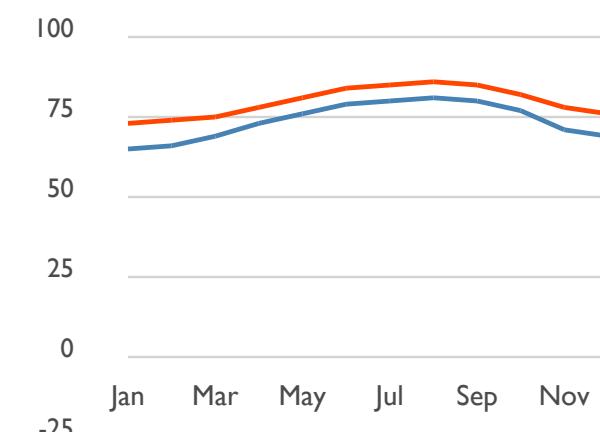
## Honolulu



## Los Angeles

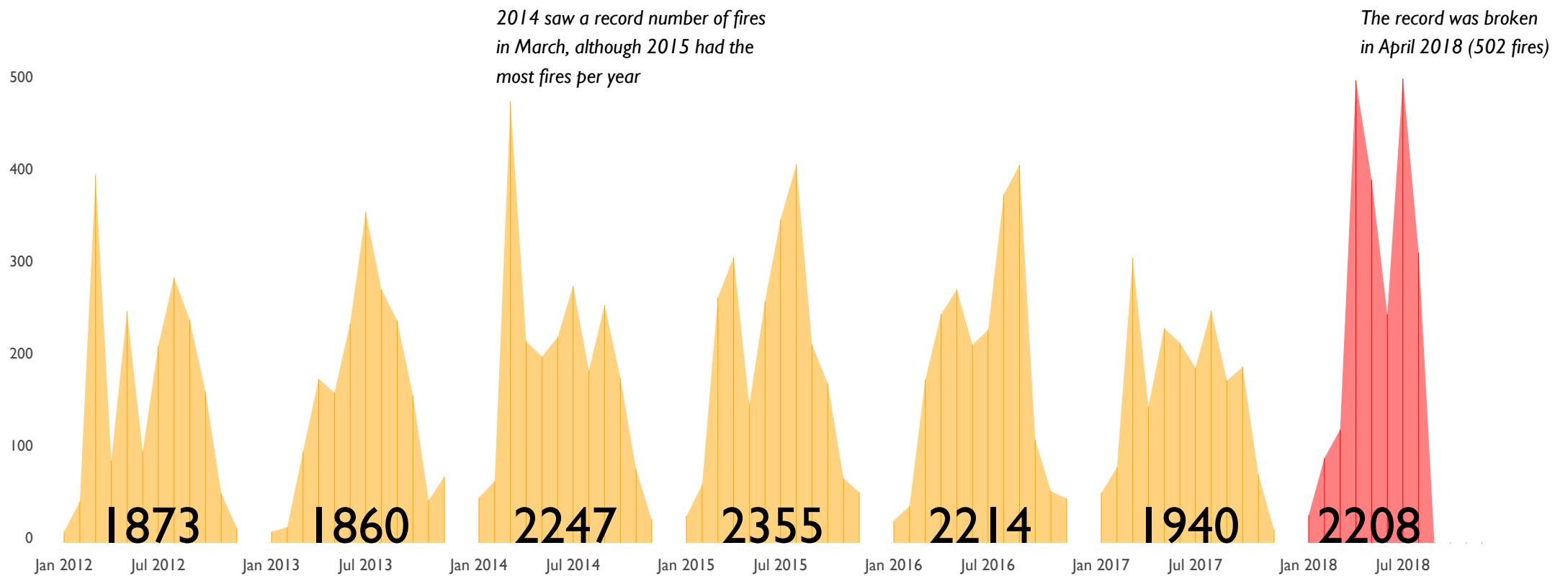


## Miami

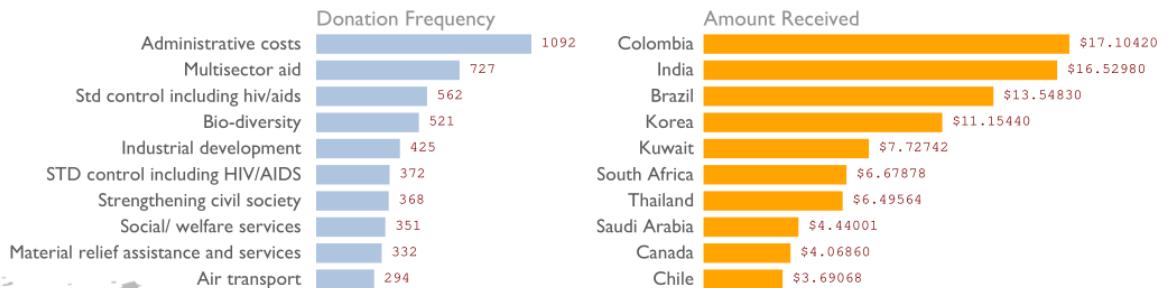
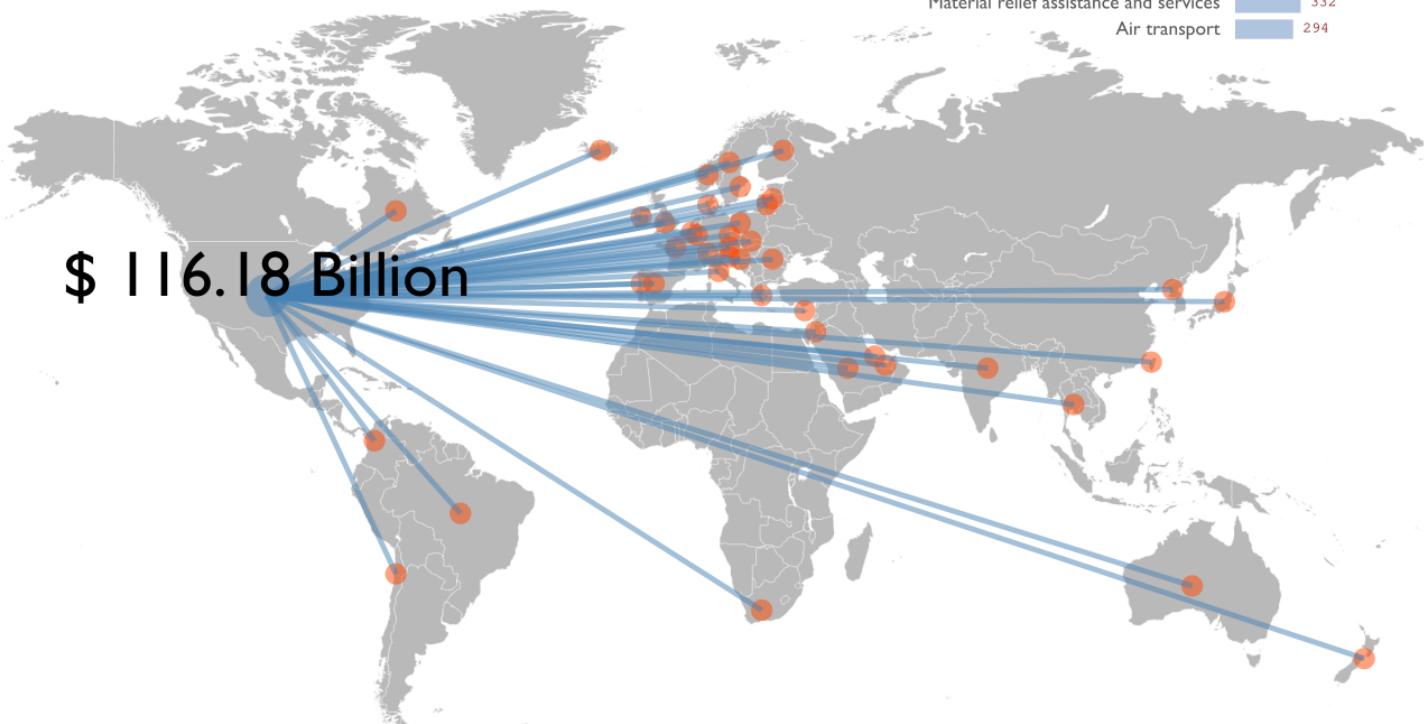


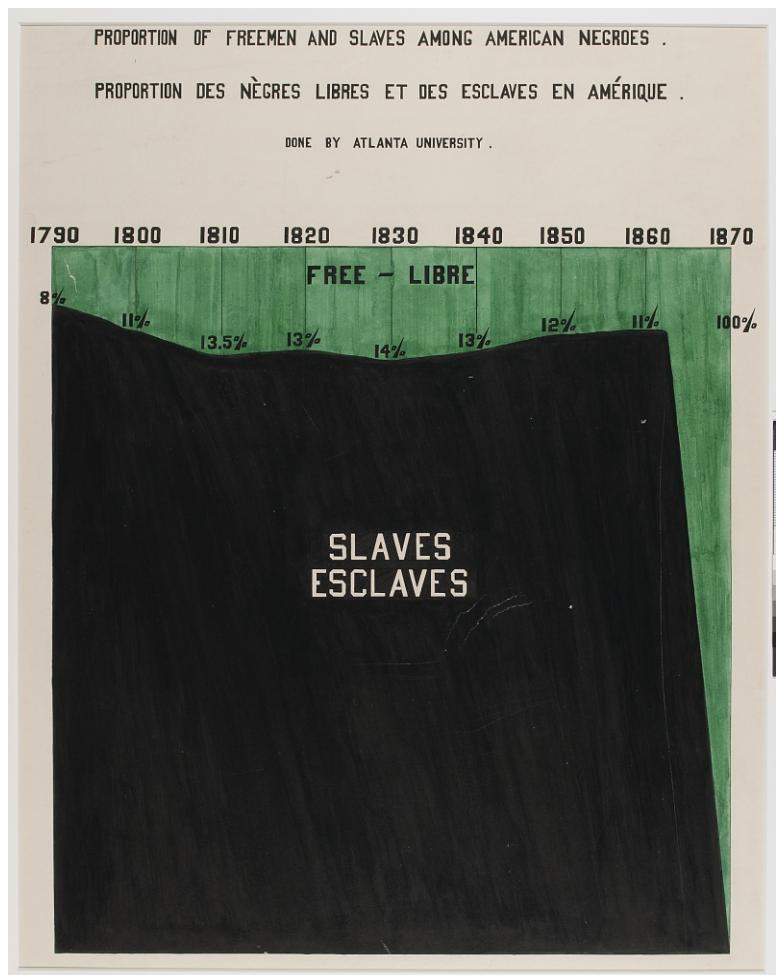
● Avg. High      ● Avg. Low

# German Wildfires 2012-2018



# United States

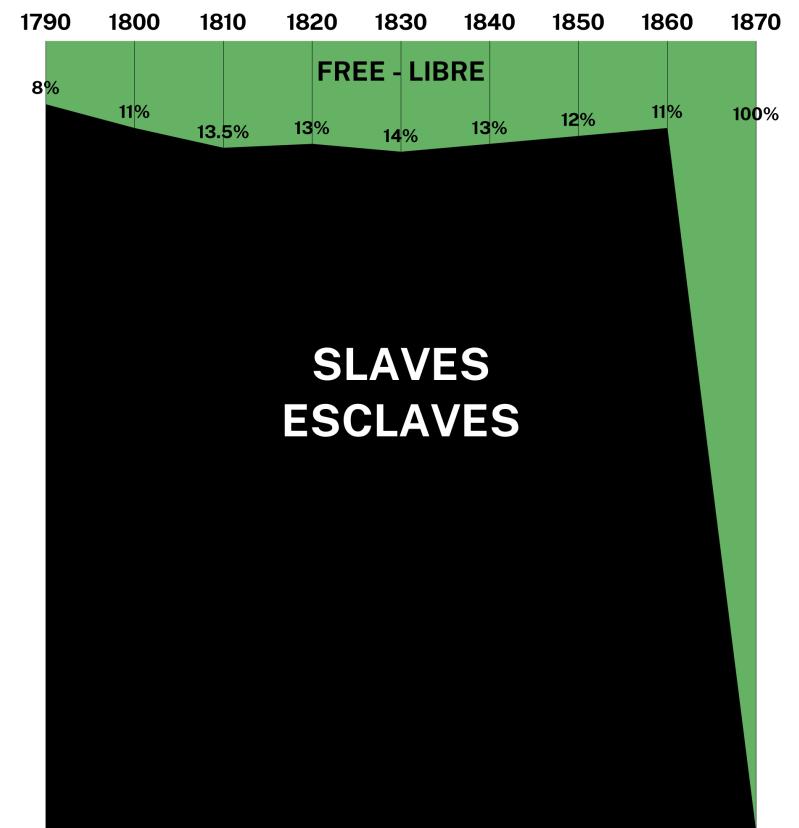




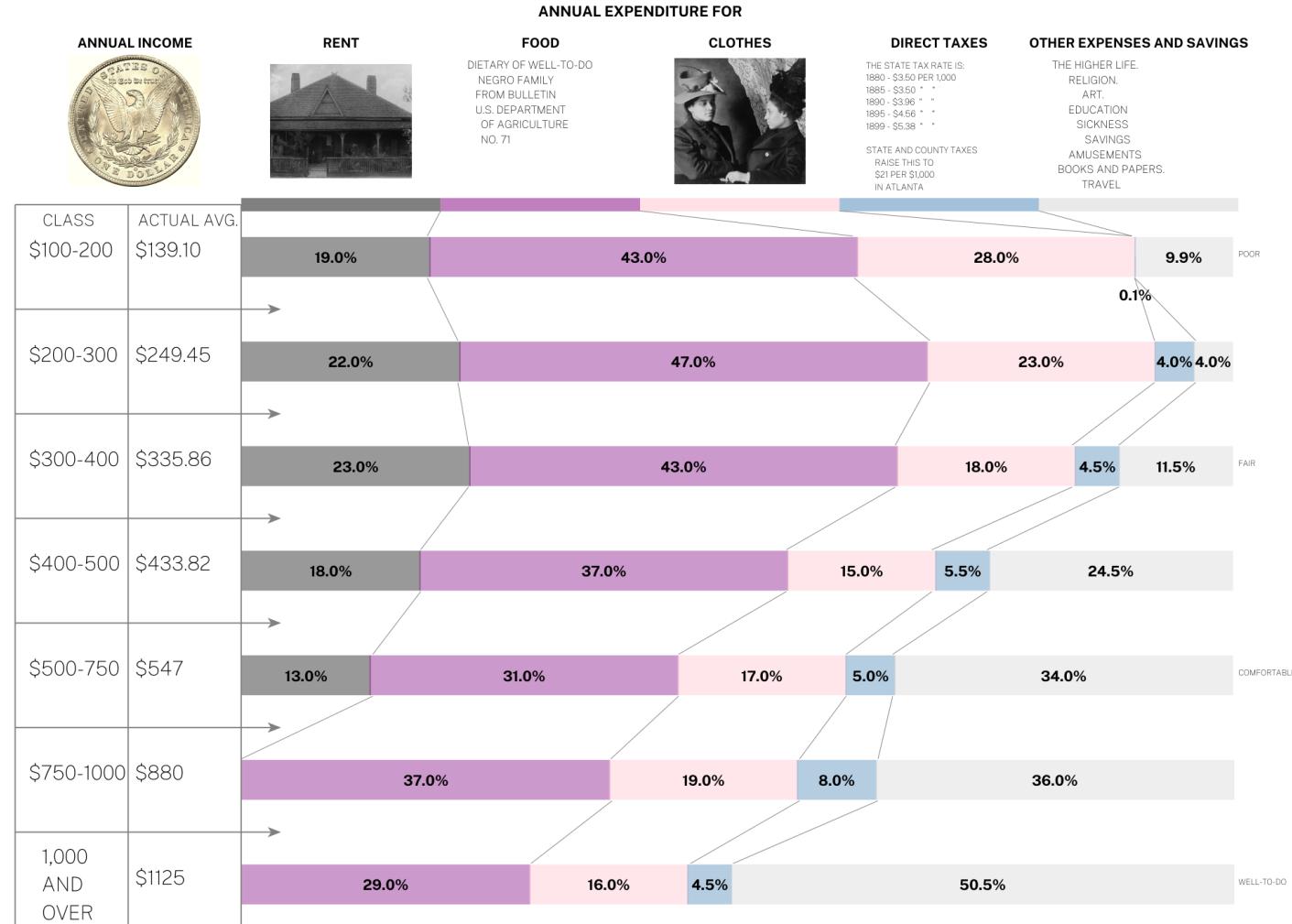
PROPORTION OF FREEMEN AND SLAVES AMONG AMERICAN NEGROES.

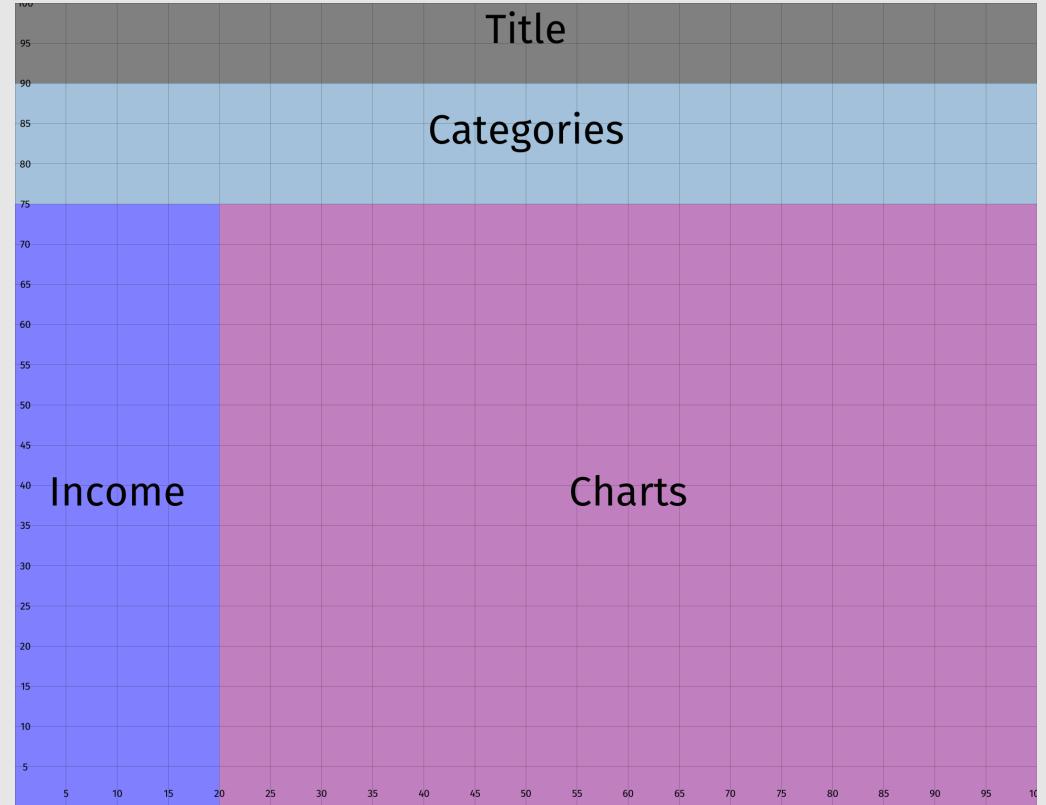
PROPORTION DES NÈGRES LIBRES ET DES ESCLAVES EN AMÉRIQUE.

DONE BY ATLANTA UNIVERSITY.



## INCOME AND EXPENDITURE OF 150 NEGRO FAMILIES IN ATLANTA, GA., U.S.A.





*go get it*

decksh

[github.com/ajstarks/decksh](https://github.com/ajstarks/decksh)

dchart

[github.com/ajstarks/dchart](https://github.com/ajstarks/dchart)

pdfdeck

[github.com/ajstarks/deck/cmd/pdfdeck](https://github.com/ajstarks/deck/cmd/pdfdeck)

examples

[github.com/ajstarks/deckviz](https://github.com/ajstarks/deckviz)

fonts

[github.com/ajstarks/deckfonts](https://github.com/ajstarks/deckfonts)

# Anthony J. Starks

## Art + Code



+1 908.548.3403



[ajstarks@gmail.com](mailto:ajstarks@gmail.com)



@ajstarks



[github.com/ajstarks](https://github.com/ajstarks)



[speakerdeck.com/ajstarks](https://speakerdeck.com/ajstarks)