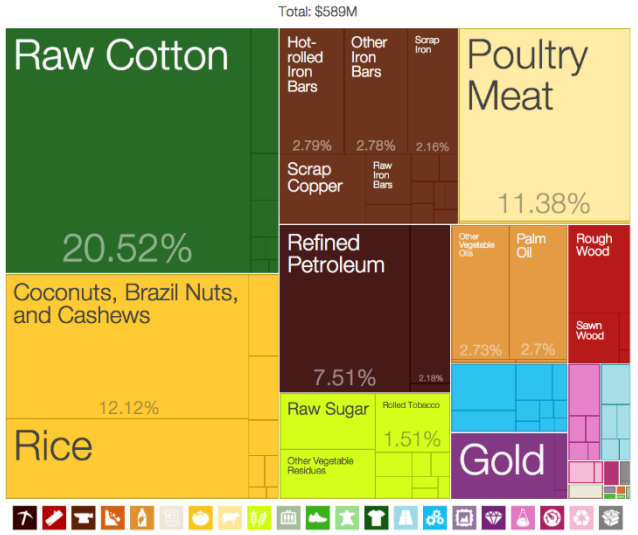
Treemaps

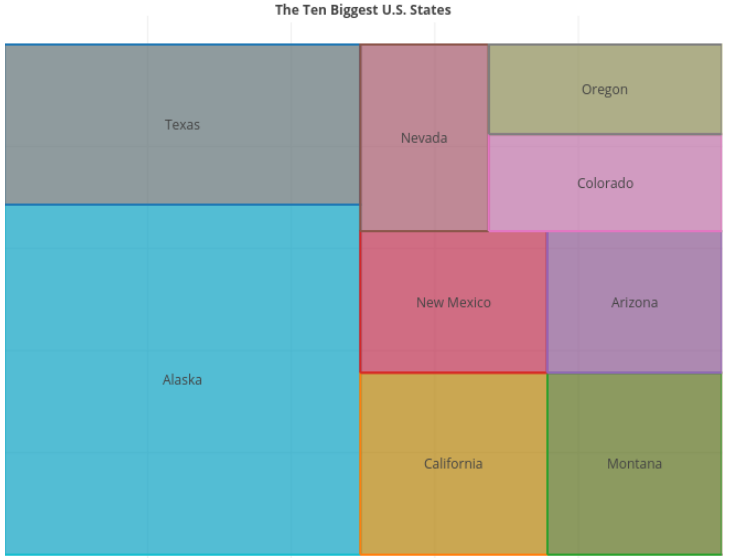
CPS 499/563 – Data Visualization

Dr. Tam Nguyen

What is a Treemap?

- Treemap is a **space-constrained** visualization of hierarchical structures.
- Treemap enables users to compare nodes and sub-trees even at varying depth in the tree, and help them **spot patterns**.





Where is it from?

- Treemaps: A space-filling approach to the visualization of hierarchical information structures
- (Johnson & Shneiderman ‘91)
- Paper available on ISIDORE

Paper "A space-filling approach to the visualization of hierarchical information structures"

Brian Johnson

Ben Shneiderman

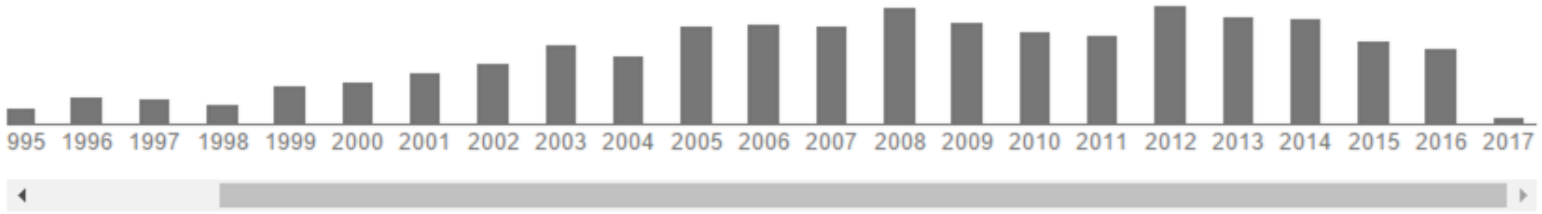
The paper

- Treemap:
  - Show a hierarchy as a 2D layout
  - Fill up the space with rectangles representing objects
  - Size on screen indicates relative size of underlying objects.

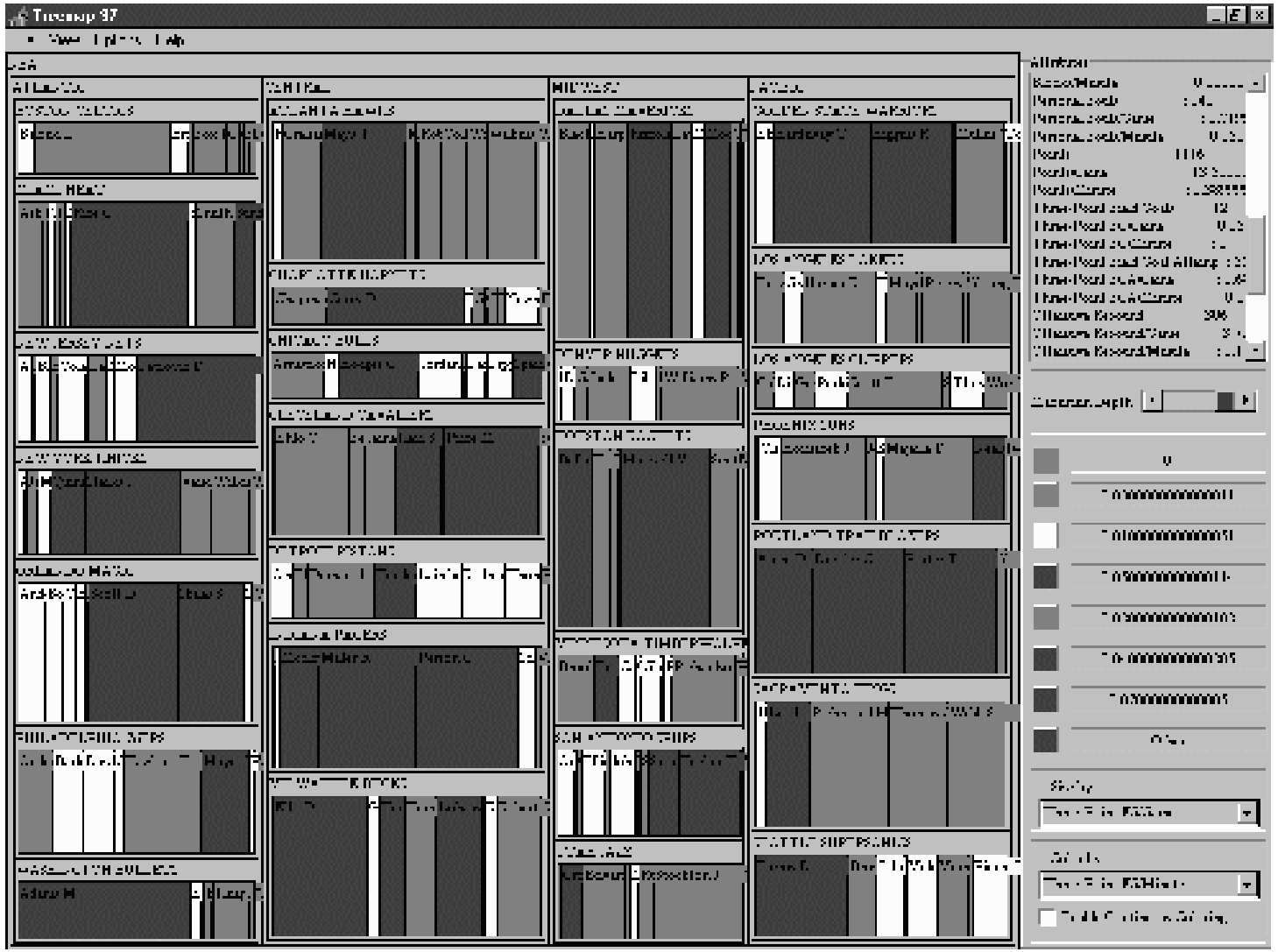
Tree-maps: A space-filling approach to the visualization of hierarchical information structures

[PDF] from umd.edu

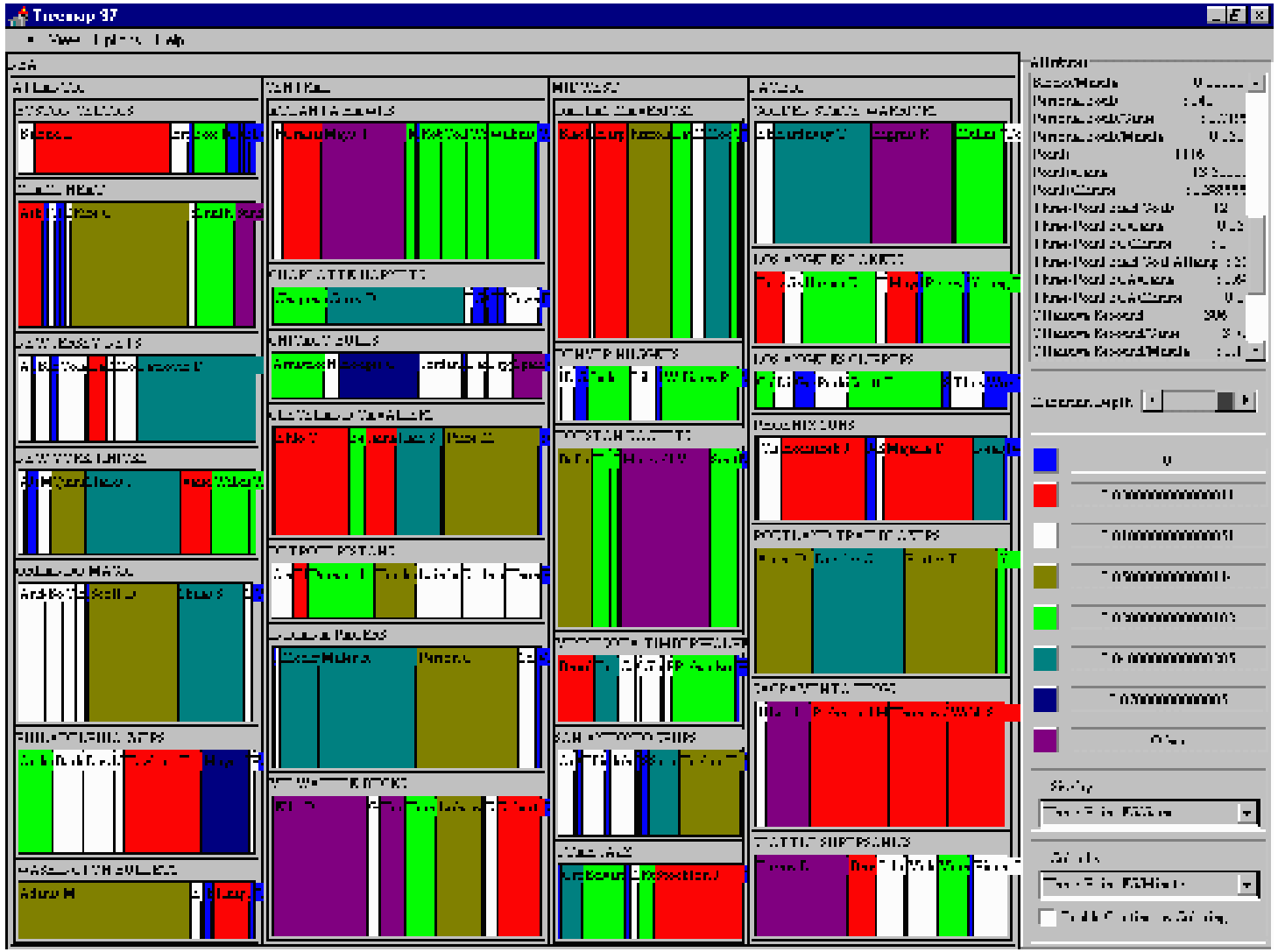
Authors	Brian Johnson, Ben Shneiderman
Publication date	1991/10/22
Conference	Proceedings of the 2nd conference on Visualization'91
Pages	284-291
Publisher	IEEE Computer Society Press
Description	Abstract This paper describes a novel method for the visualization of hierarchically structured information. The Tree-Map visualization technique makes 100% use of the available display space, mapping the full hierarchy onto a rectangular region in a space-filling manner. This efficient use of space allows very large hierarchies to be displayed in their entirety and facilitates the presentation of semantic information.
Total citations	Cited by 1671



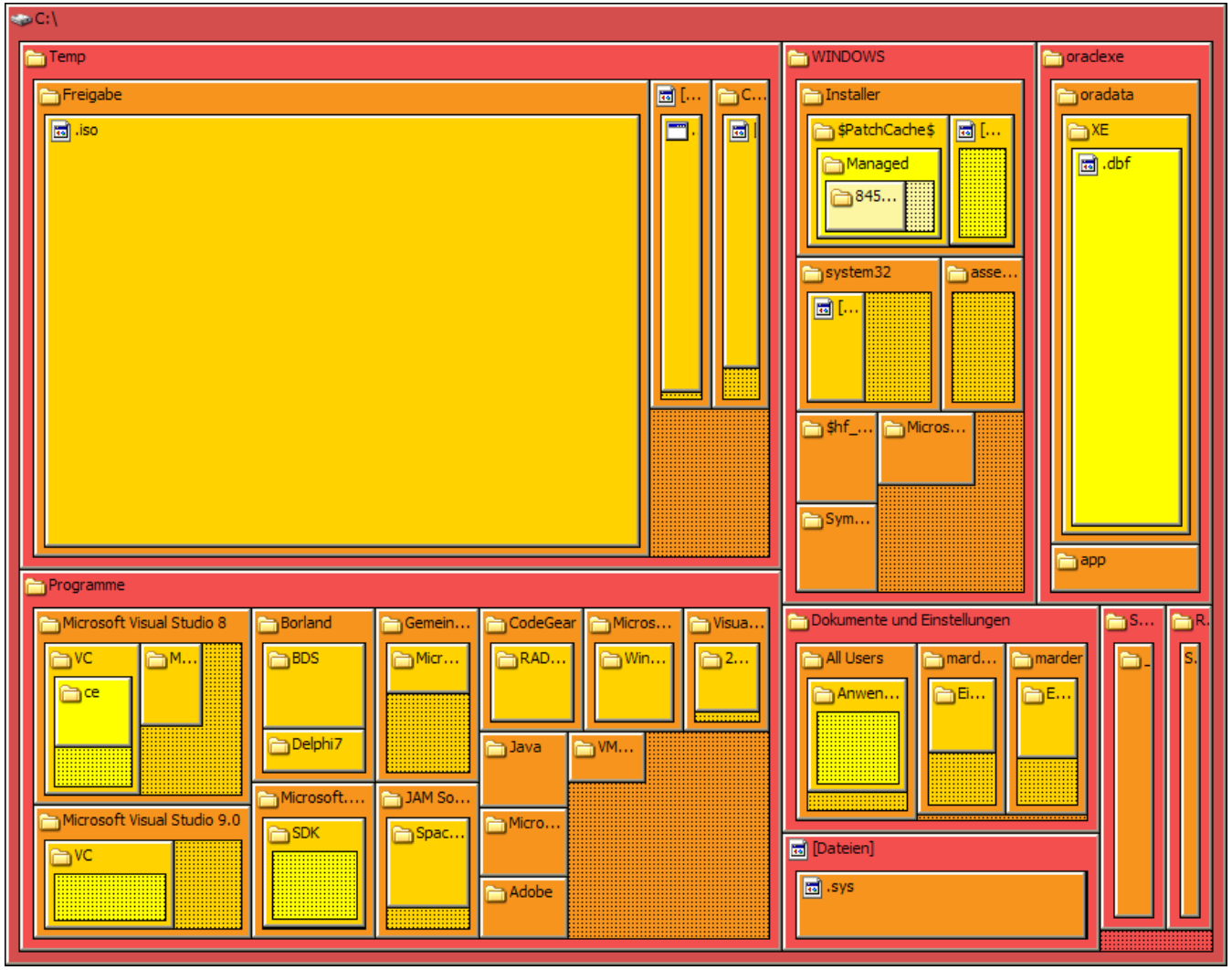
Early Treemap Applied to File System (1991)



Early Treemap Applied to File System



Early Treemap Applied to File System

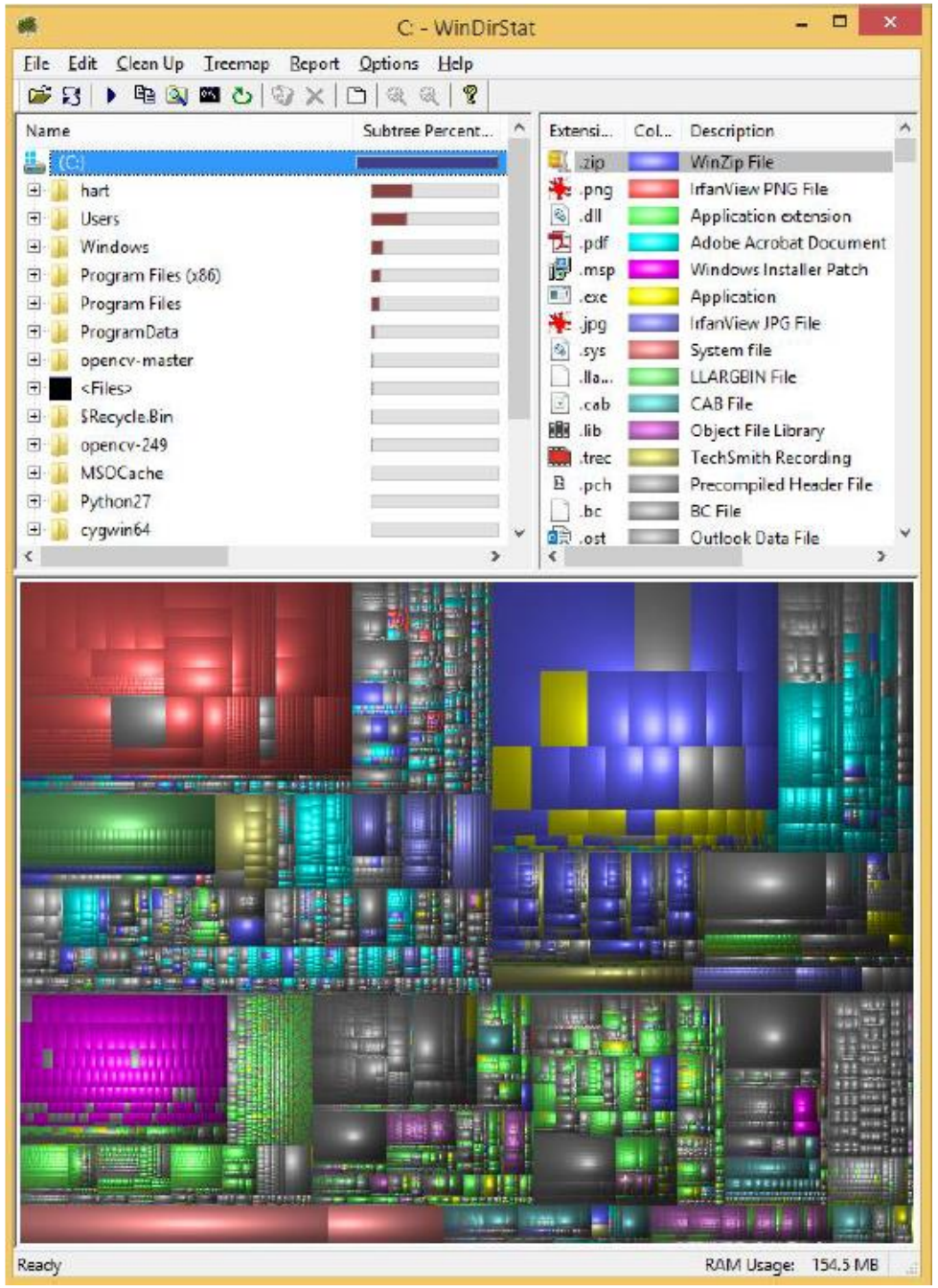


Visualization Factor

- Maps quantities to area
- Color used to differentiate areas
- Shading delineates hierarchical regions

Area

Color



Pros

- **Good idea**
  - Break into meaningful groups
  - Fix these into a useful aspect ratio

- **Provide excellent interactivity**
  - Access to the real data
  - Makes it into a useful tool

Cons

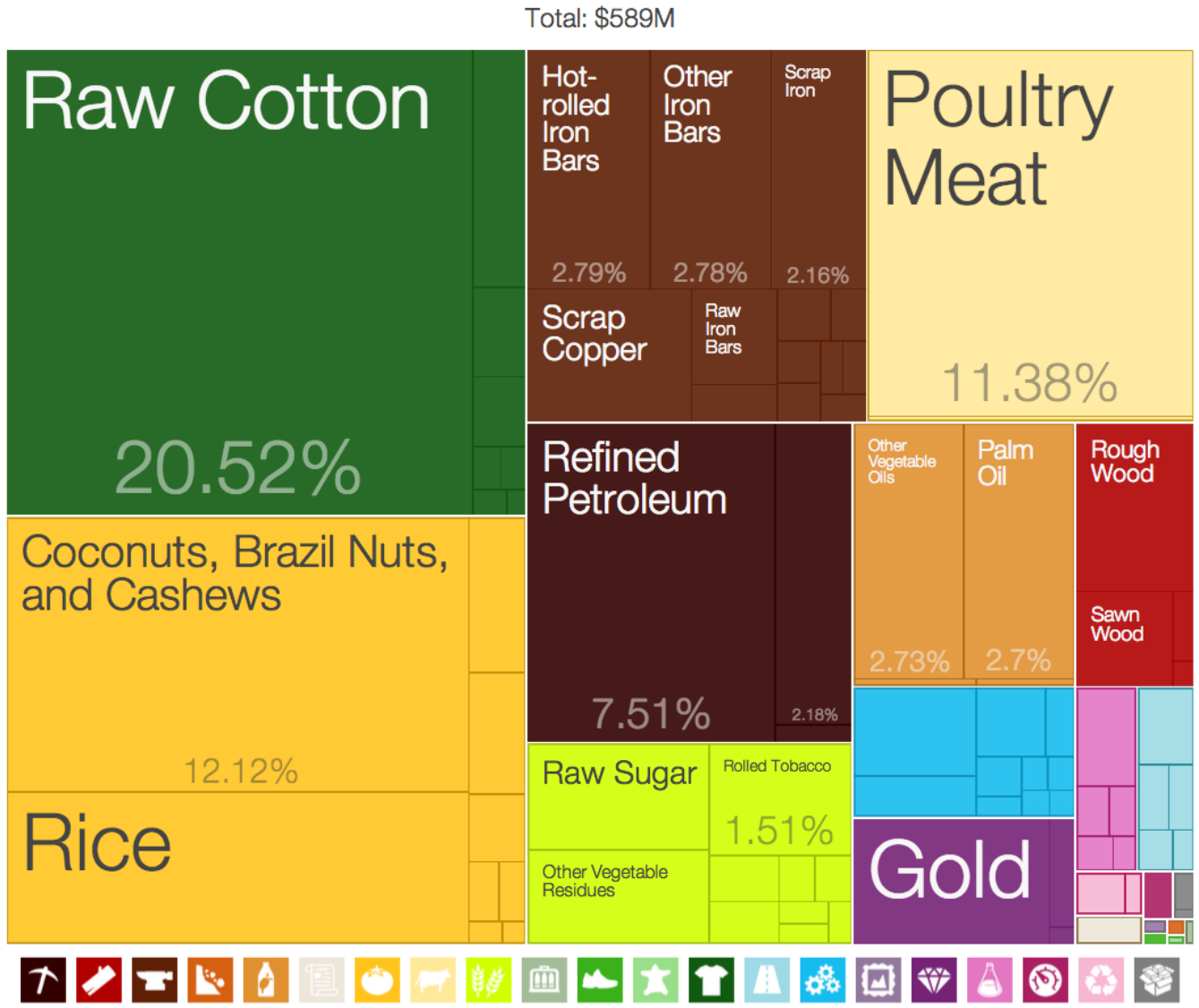
- **Too disorderly**
  - Aspect ratios uncontrolled leads to lots of skinny boxes that clutter
- **Wrong application**
  - Don’t need all this to just see the largest files in the OS

Treemap in different applications

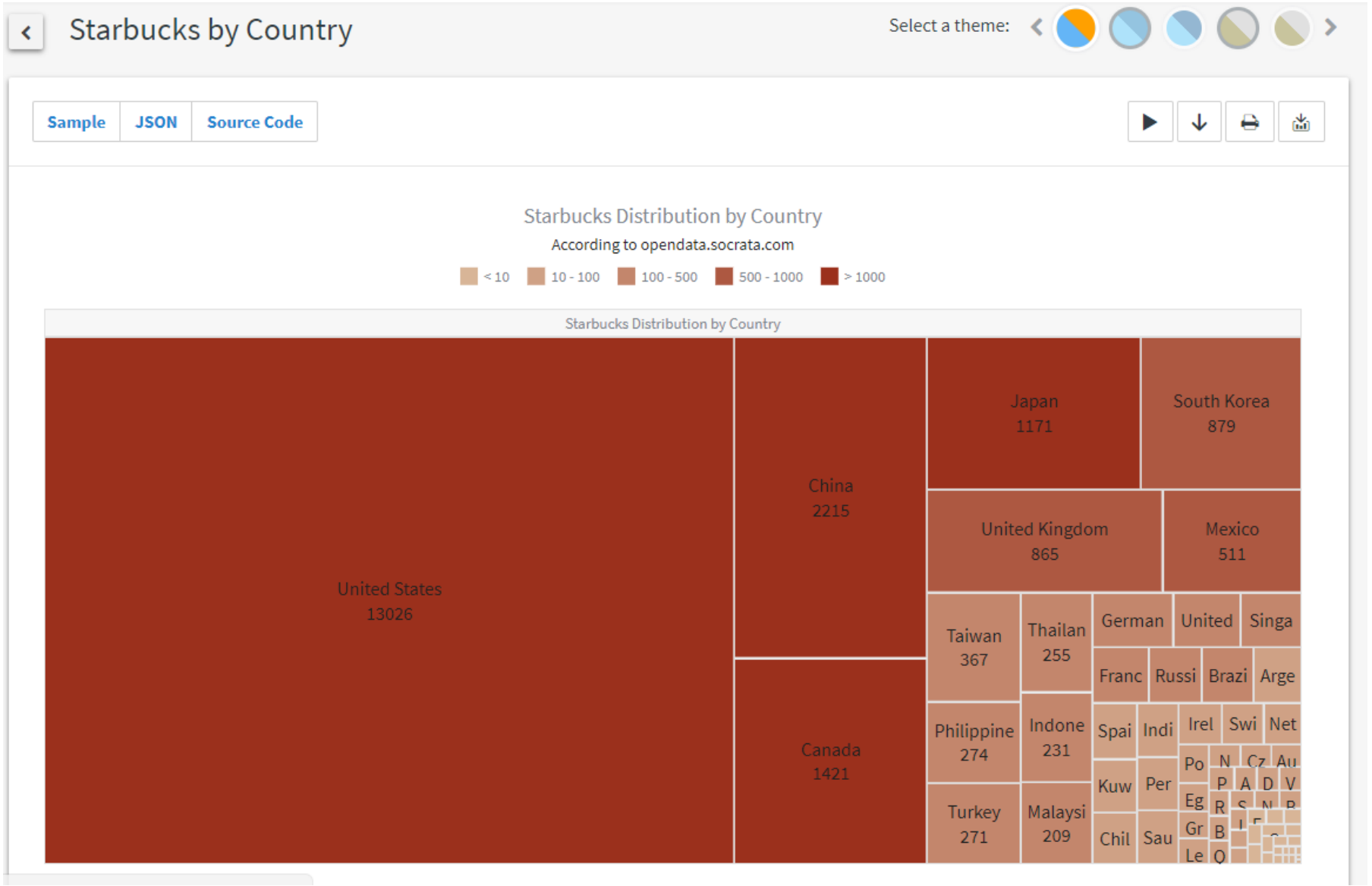
- Treemap of soft drink preference in a small group of people.
- Color and gradients are used to group items.

Treemap in different applications

- Treemap of Benin's exports by product category, 2009.



Starbucks by Country

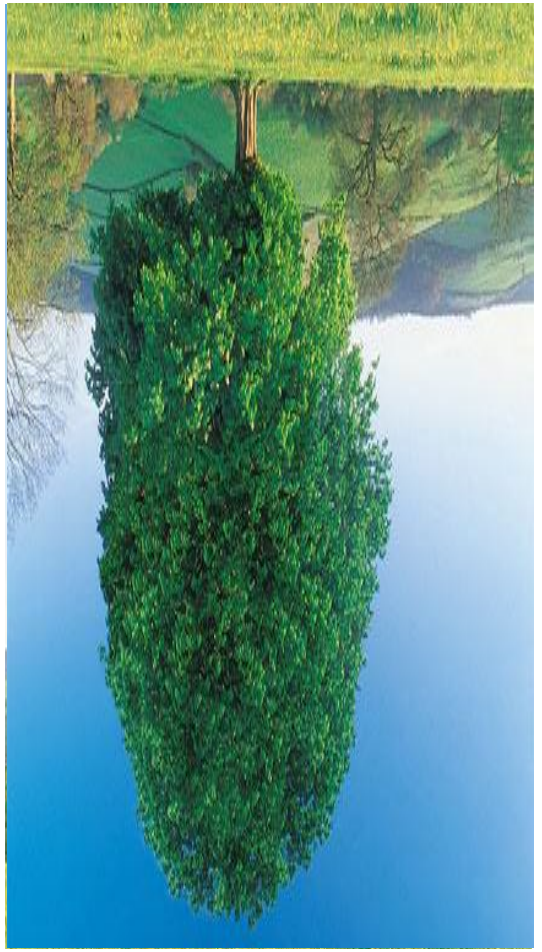


https://www.anychart.com/products/anychart/gallery/Tree_Map_Charts/Starbucks_by_Country.php

How to construct a Treemap?

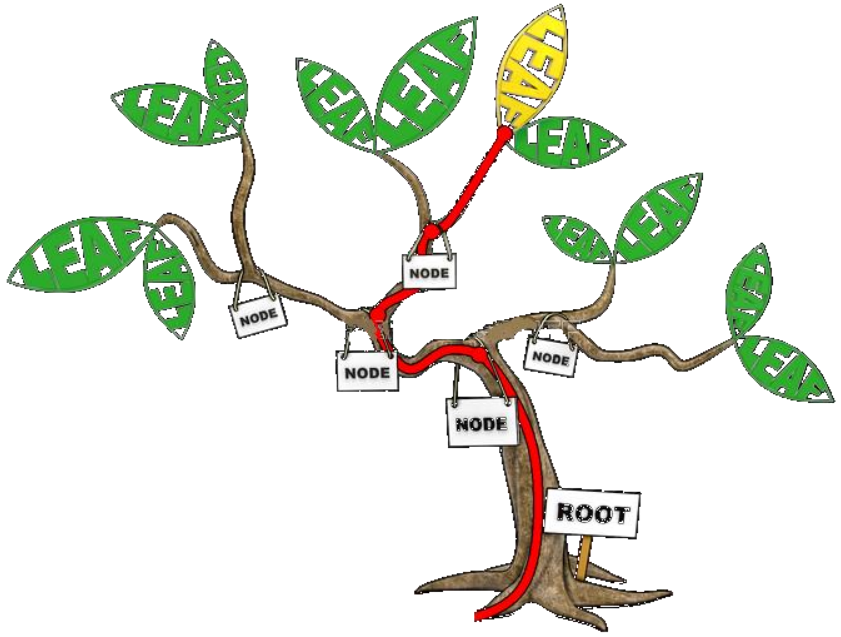
- Let’s start with a tree first

- "A tree may grow a thousand feet tall, but its leaves will return to its roots."
- -Chinese Proverb

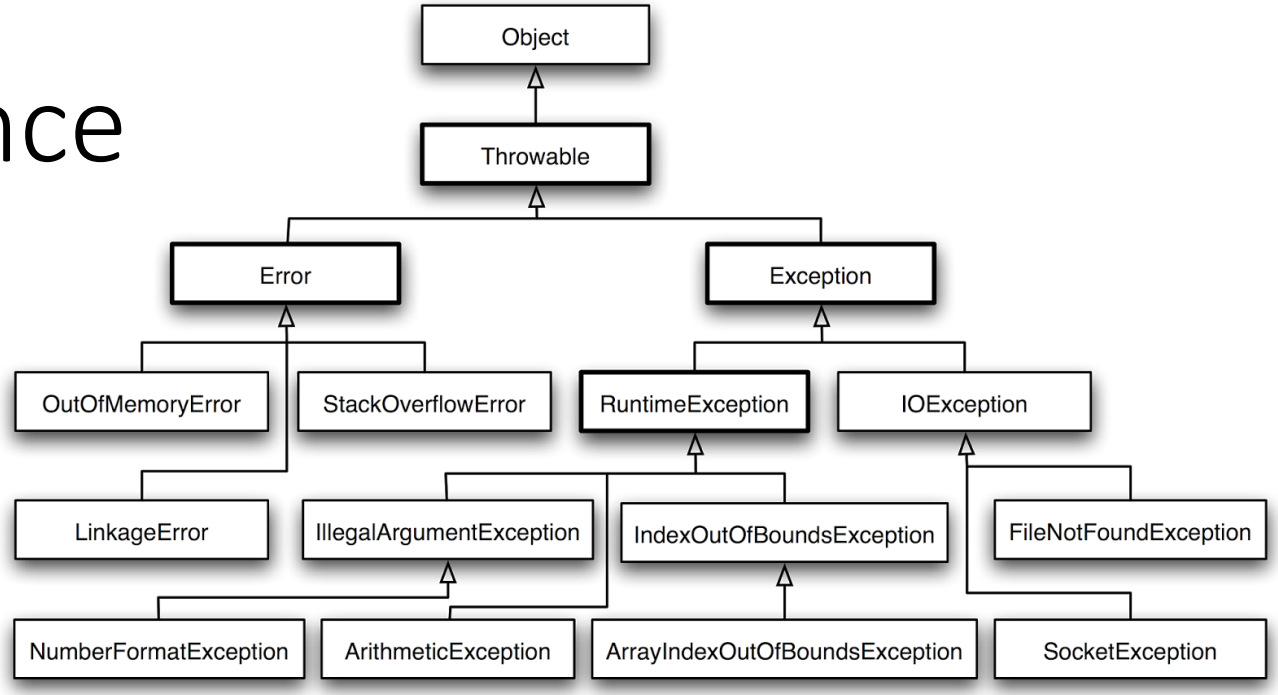


Trees in Computer Science

- Tree
  - n nodes, n-1 edges
  - single parent node can have multiple child nodes (siblings)
- Hierarchy
  - height-based layout used

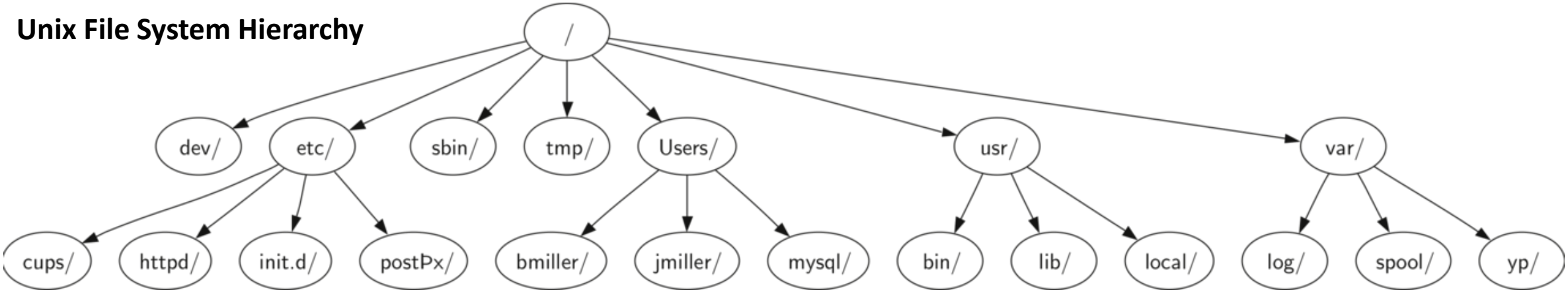


Tree in Computer Science



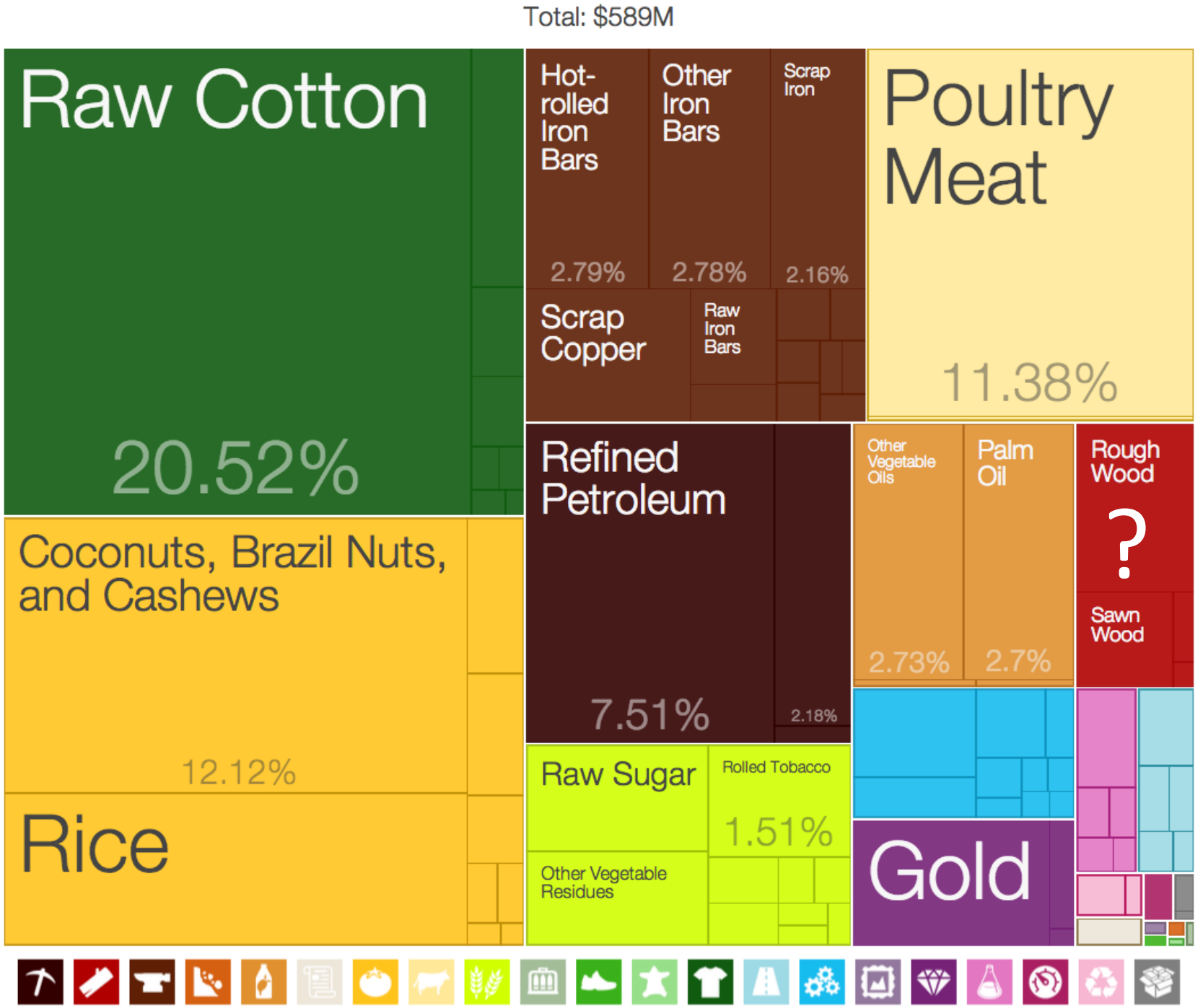
Java Class Hierarchies

Unix File System Hierarchy



Treemap

- Where is the tree?



How to construct a Treemap

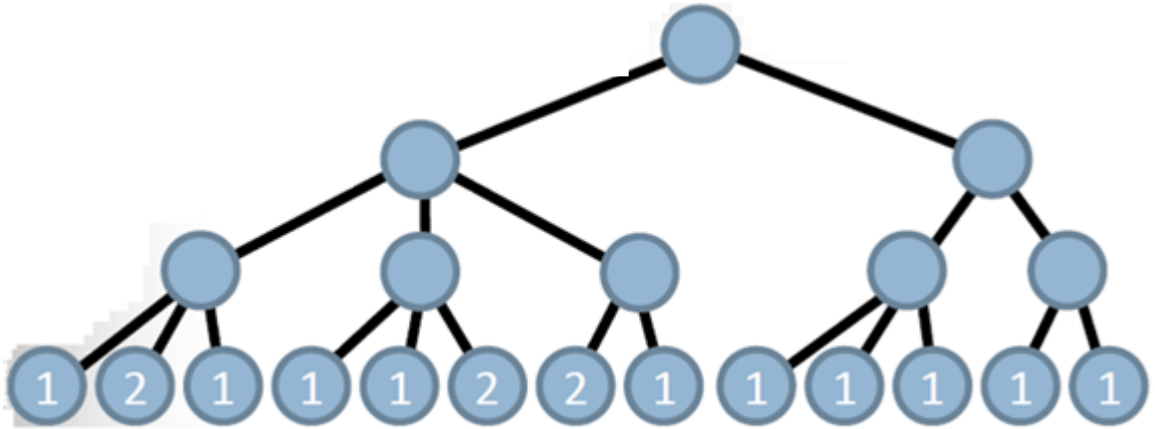
- Two steps:
  - Bottom-up: Building the tree hierarchy
  - Top-down: Building the map

Example:

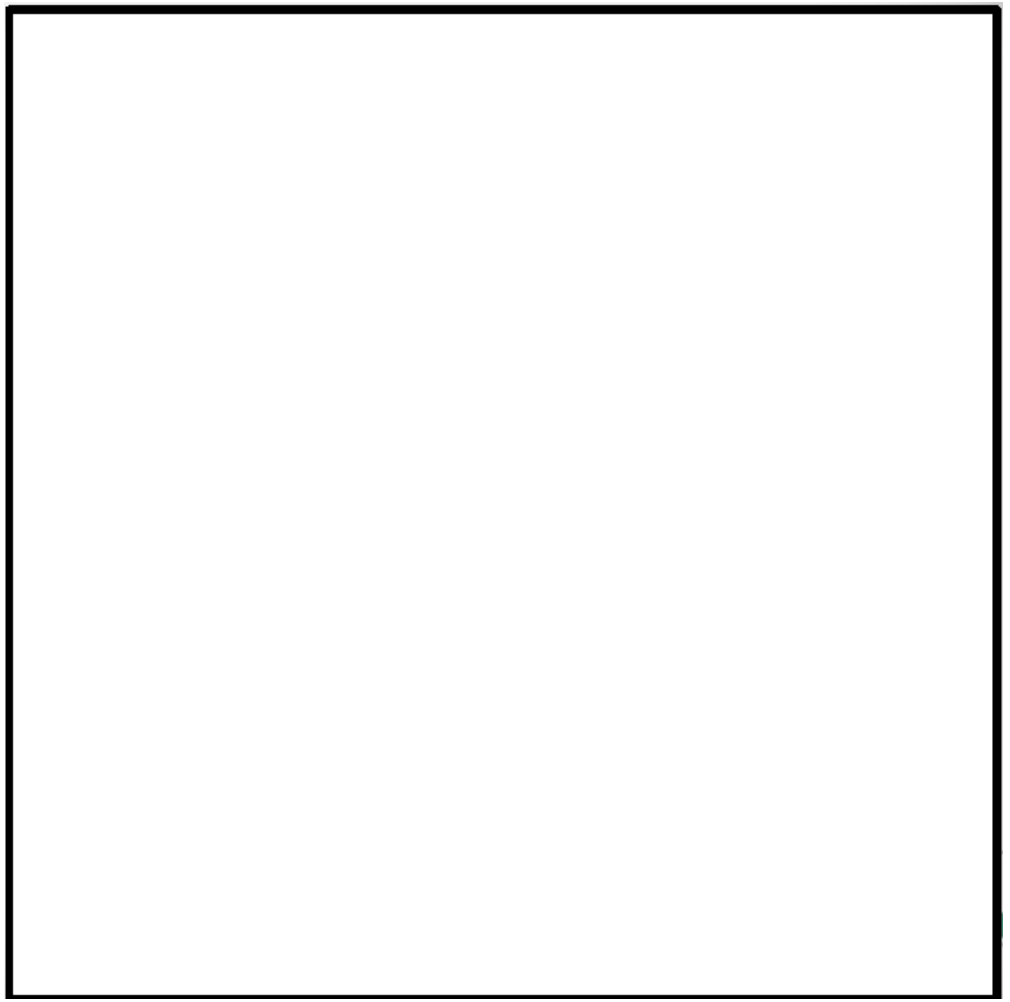
16 tennis players of 4th round of US Open from different countries

- Serbia: 1
- USA: 2
- Canada: 1
- Japan: 1
- Switzerland: 1
- Spain: 2
- France: 2
- Latvia: 1
- South Africa: 1
- Czech: 1
- Greece: 1
- Russia: 1
- Argentina: 1

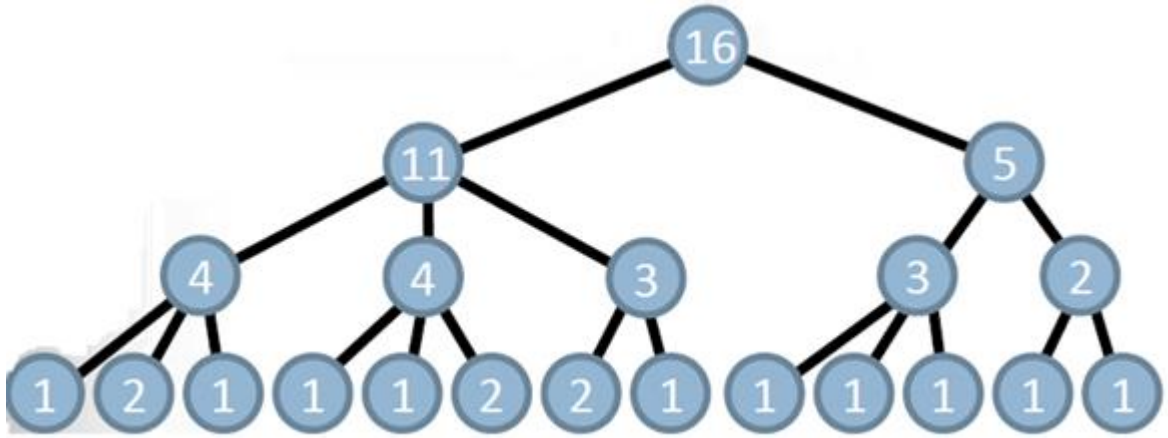
Example: How to construct a Treemap

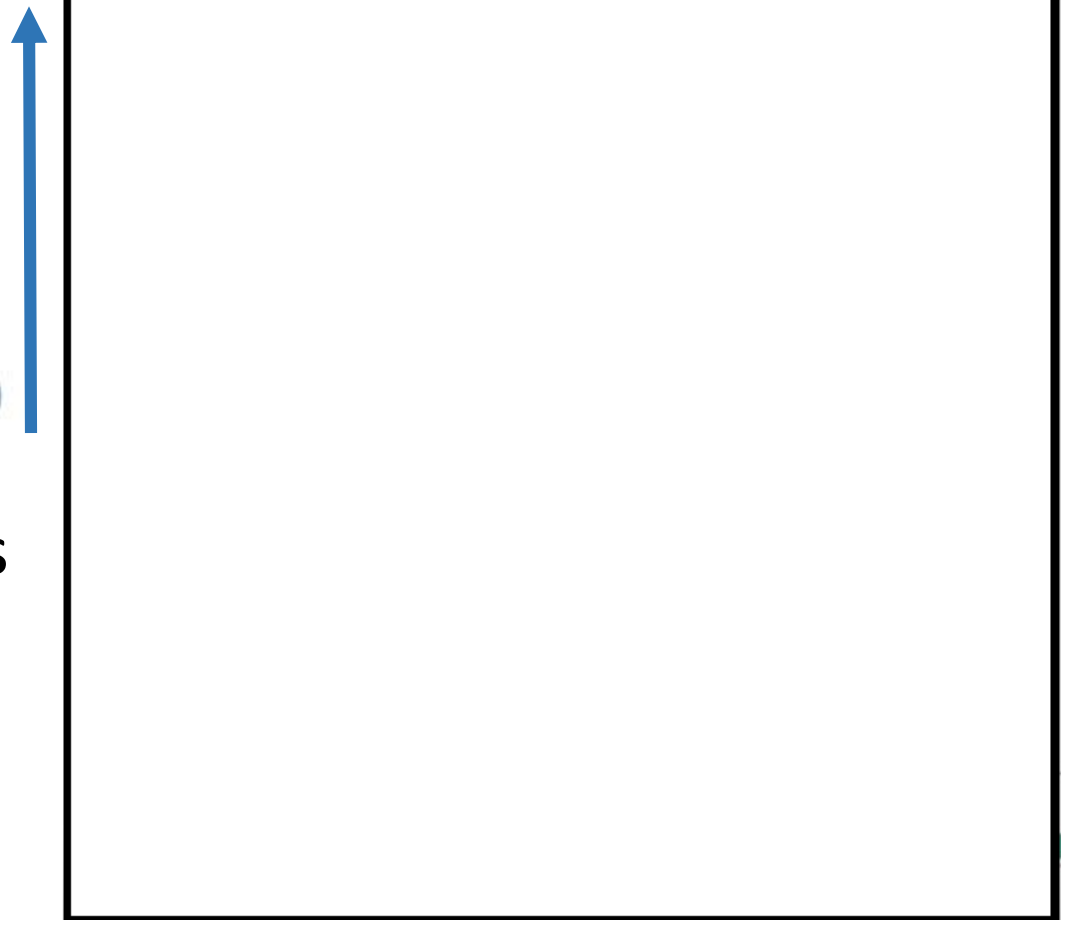


- Construct leaf nodes



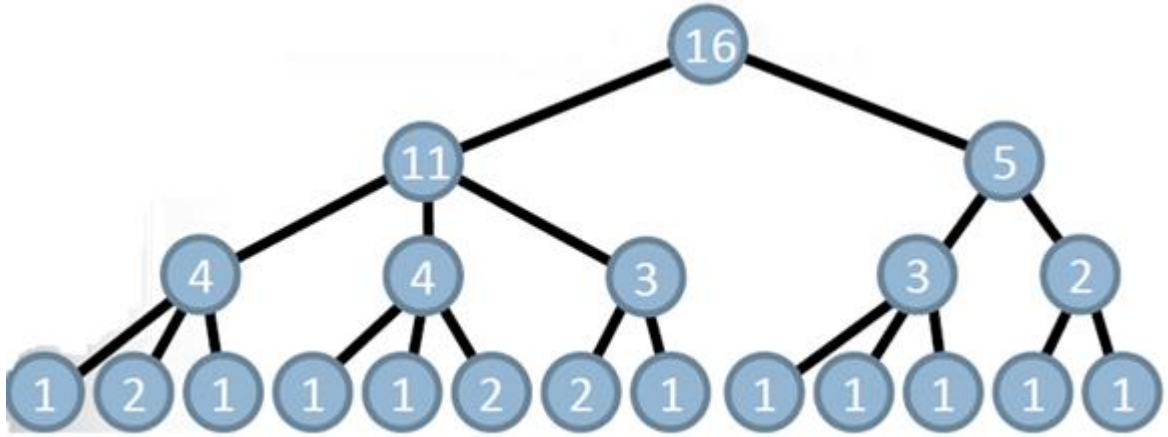
Example: How to construct a Treemap





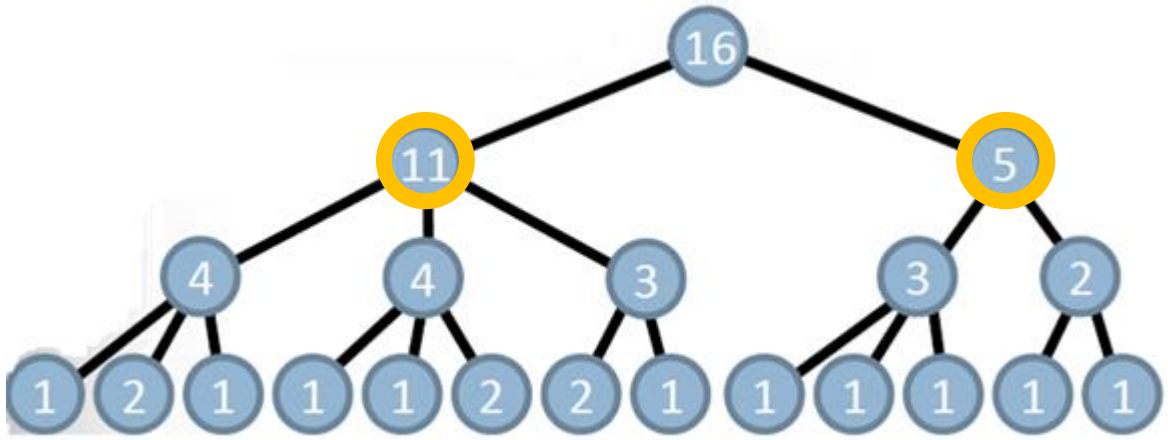
- Bottom-up: Set parents node values to sum of child node values.

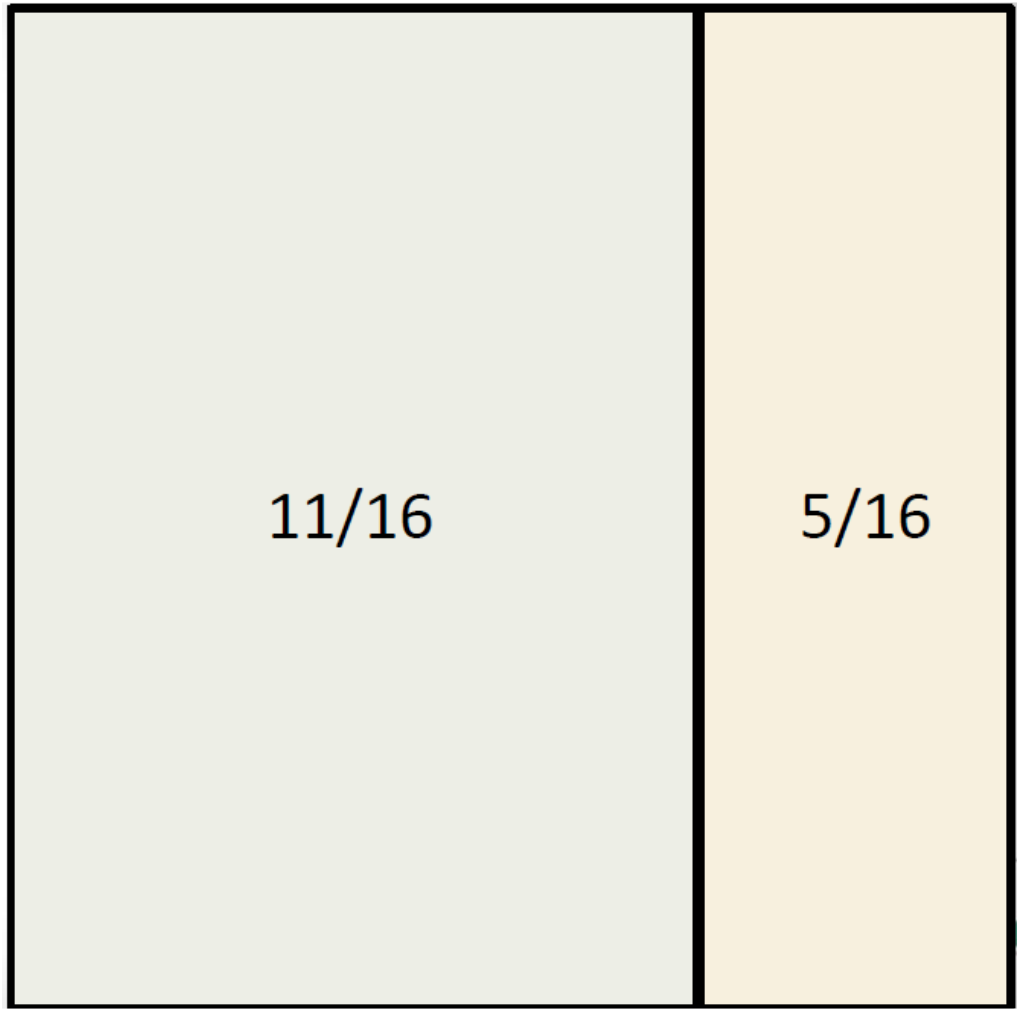
Example: How to construct a Treemap



- Top-down: Partition based on current node’s value as a portion of parent node’s value.

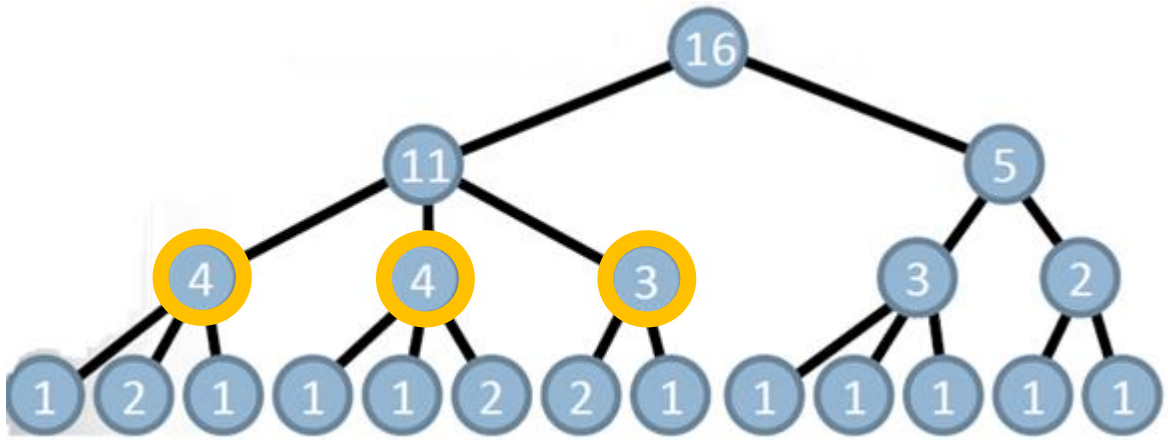
Example: How to construct a Treemap

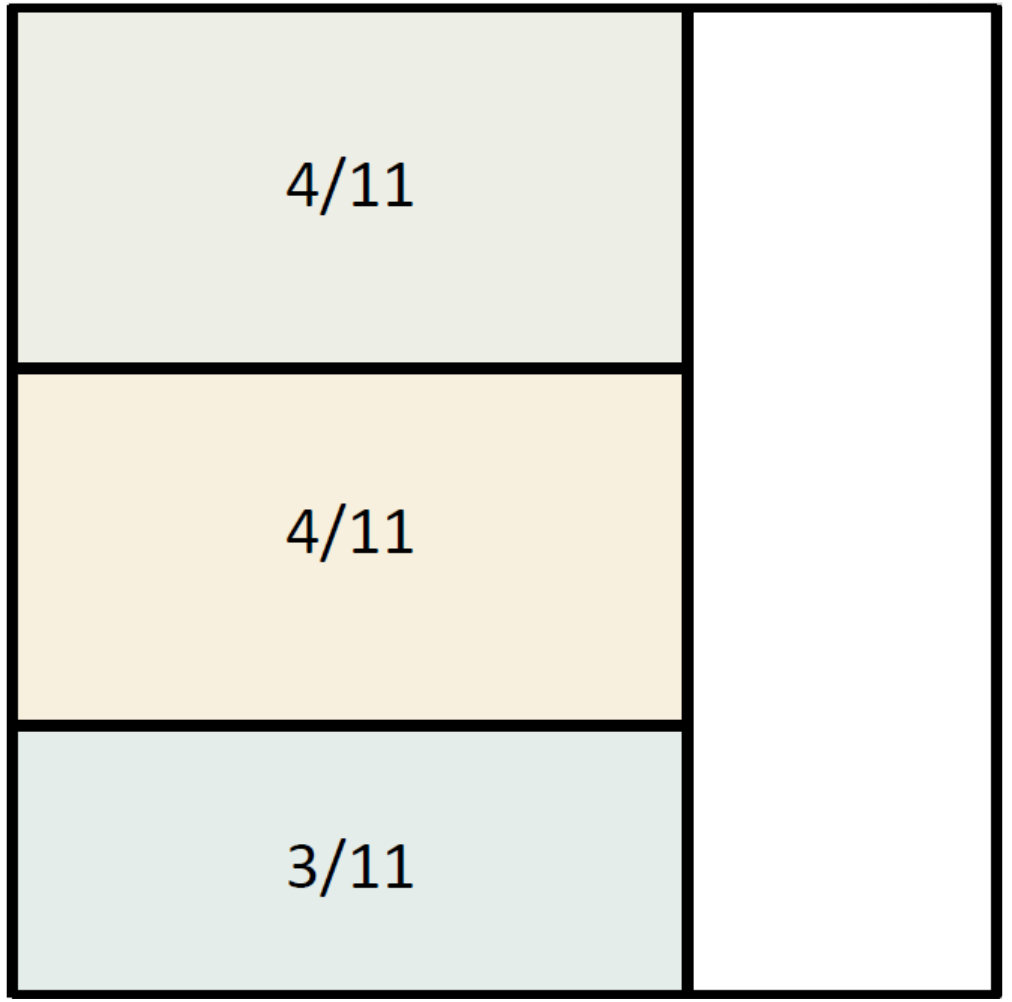




- Top-down: Partition based on current node's value as a portion of parent node's value.

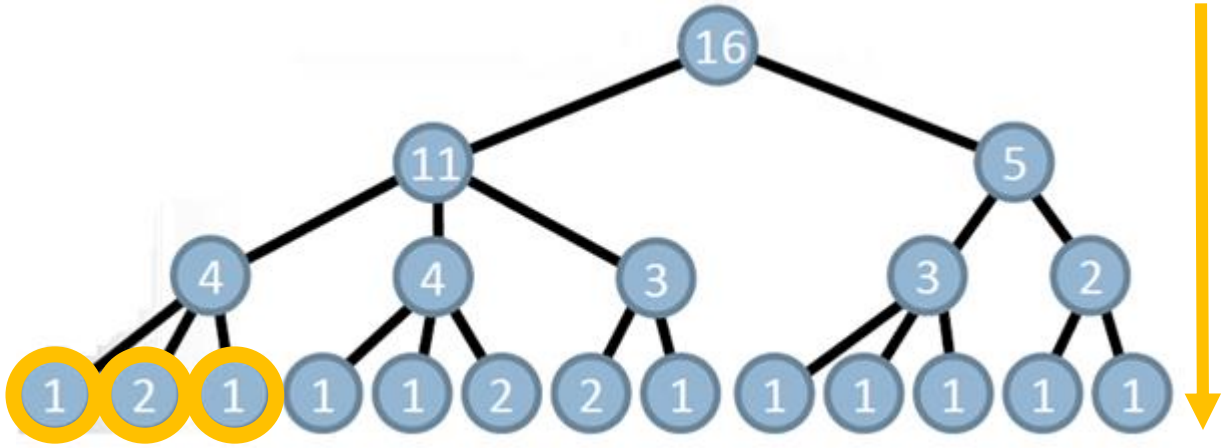
Example: How to construct a Treemap

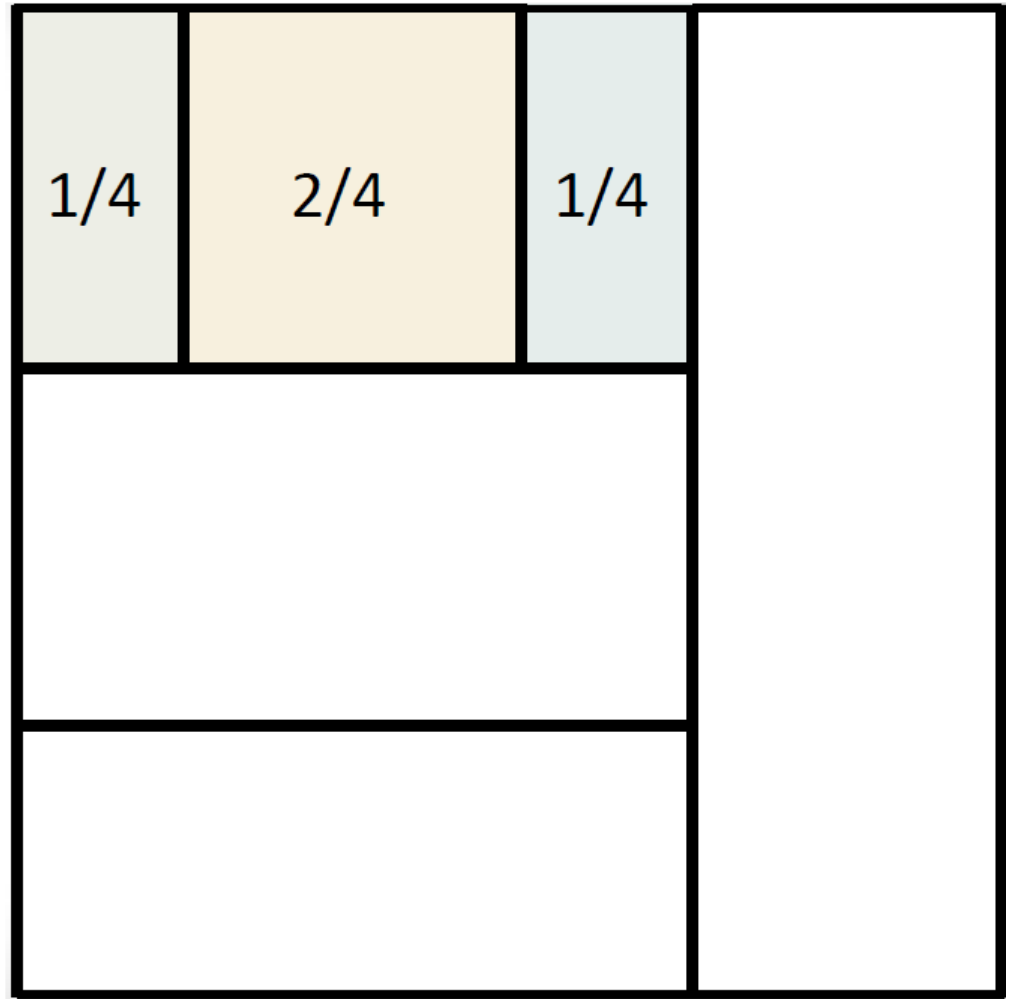




- Top-down: Partition based on current node's value as a portion of parent node's value.

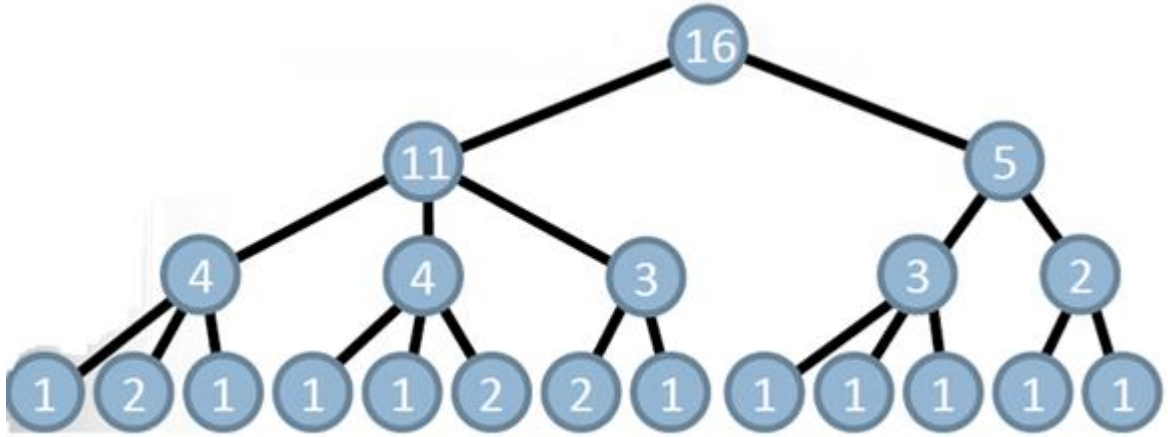
Example: How to construct a Treemap



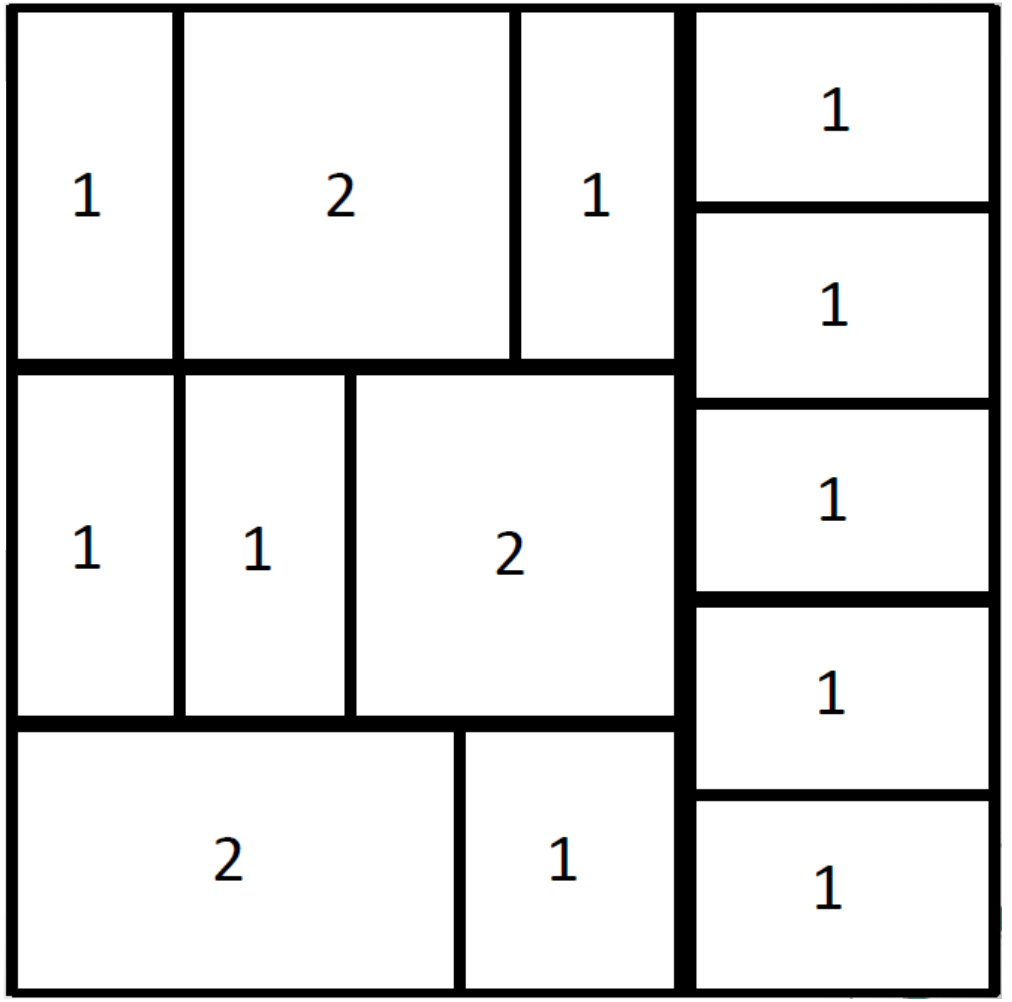


- Top-down: Partition based on current node's value as a portion of parent node's value.

How to construct a Treemap

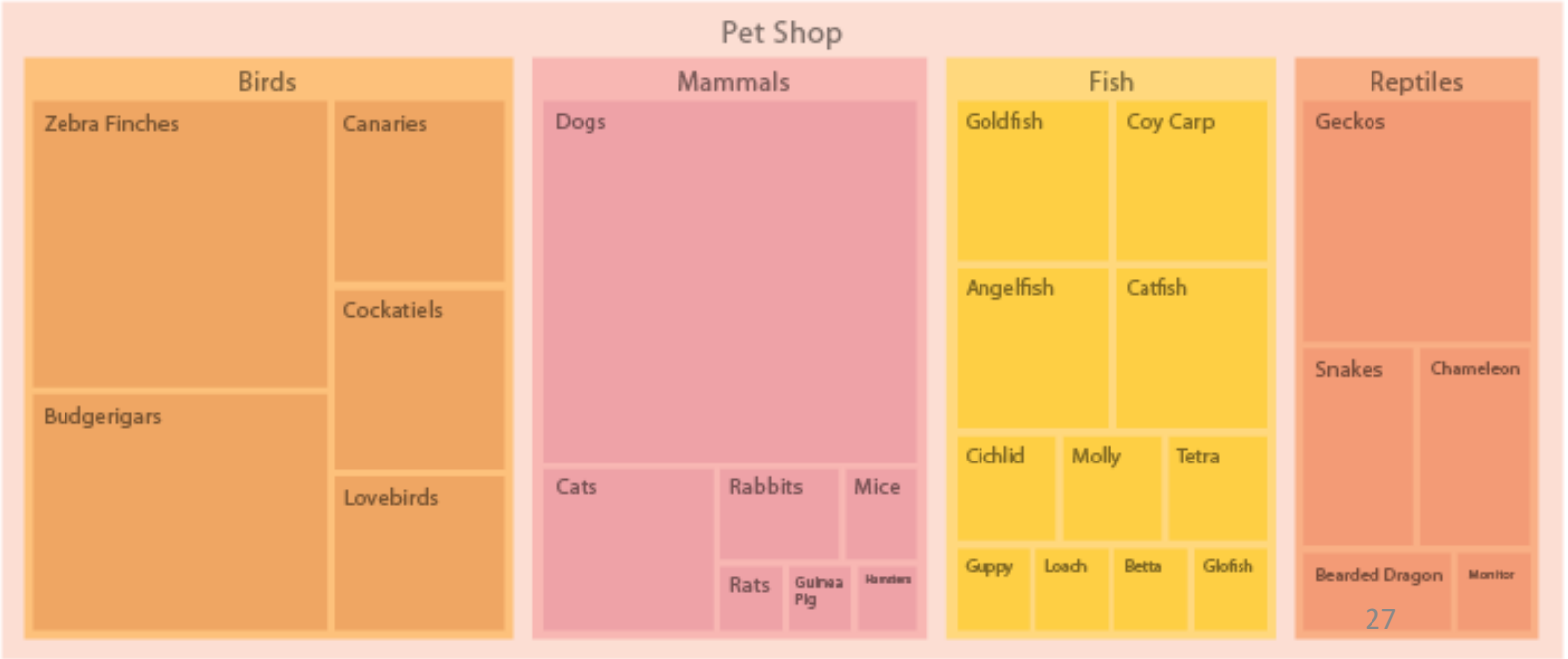


- Bottom-up: Set parents node values to sum of child node values.
- Top-down: Partition based on current node's value as a portion of parent node's value.



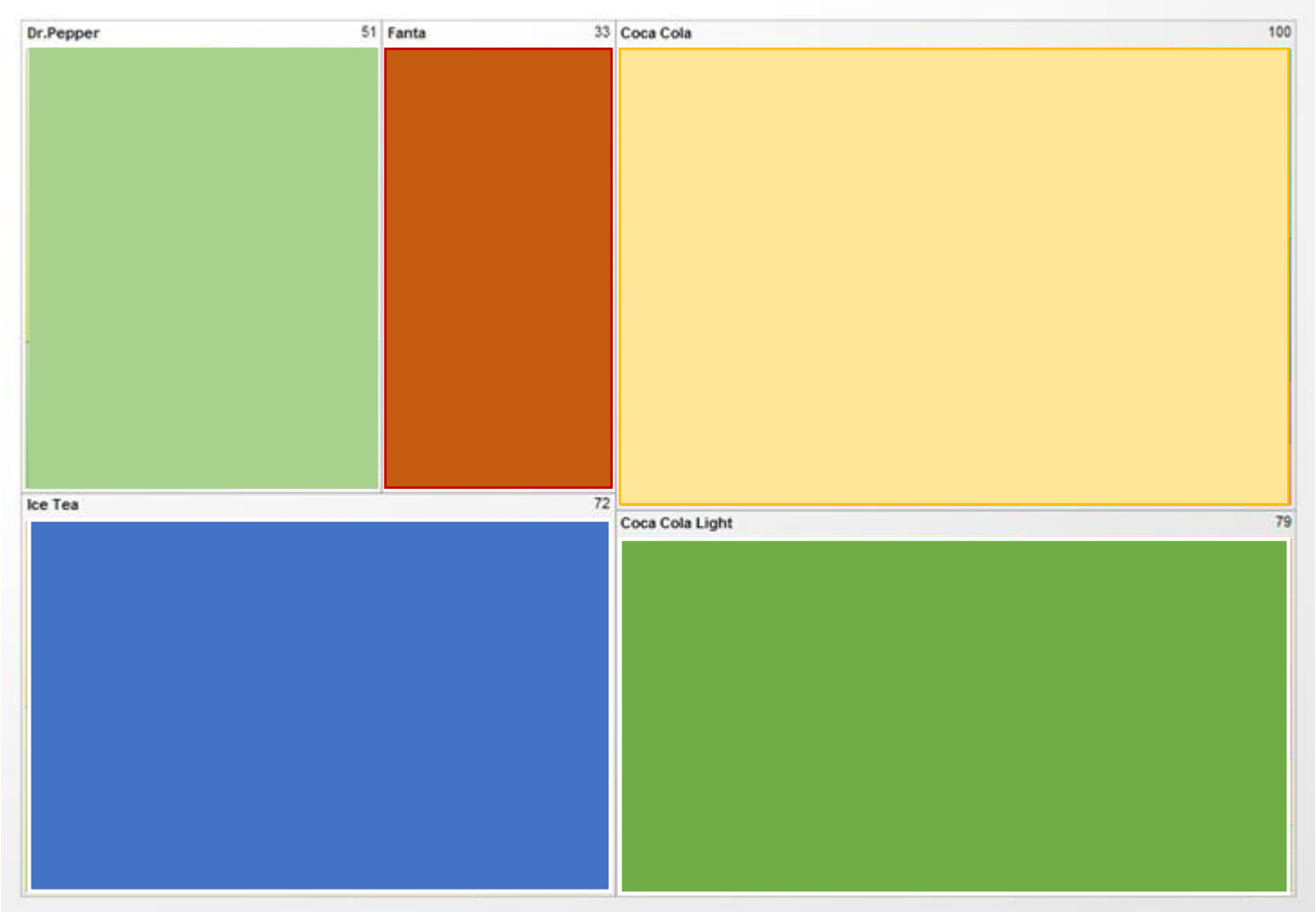
Nested Treemap

- If a treemap appears inside another treemap it is called a ***nested treemap***.



Nested Treemap

- Build the treemap for the higher level first

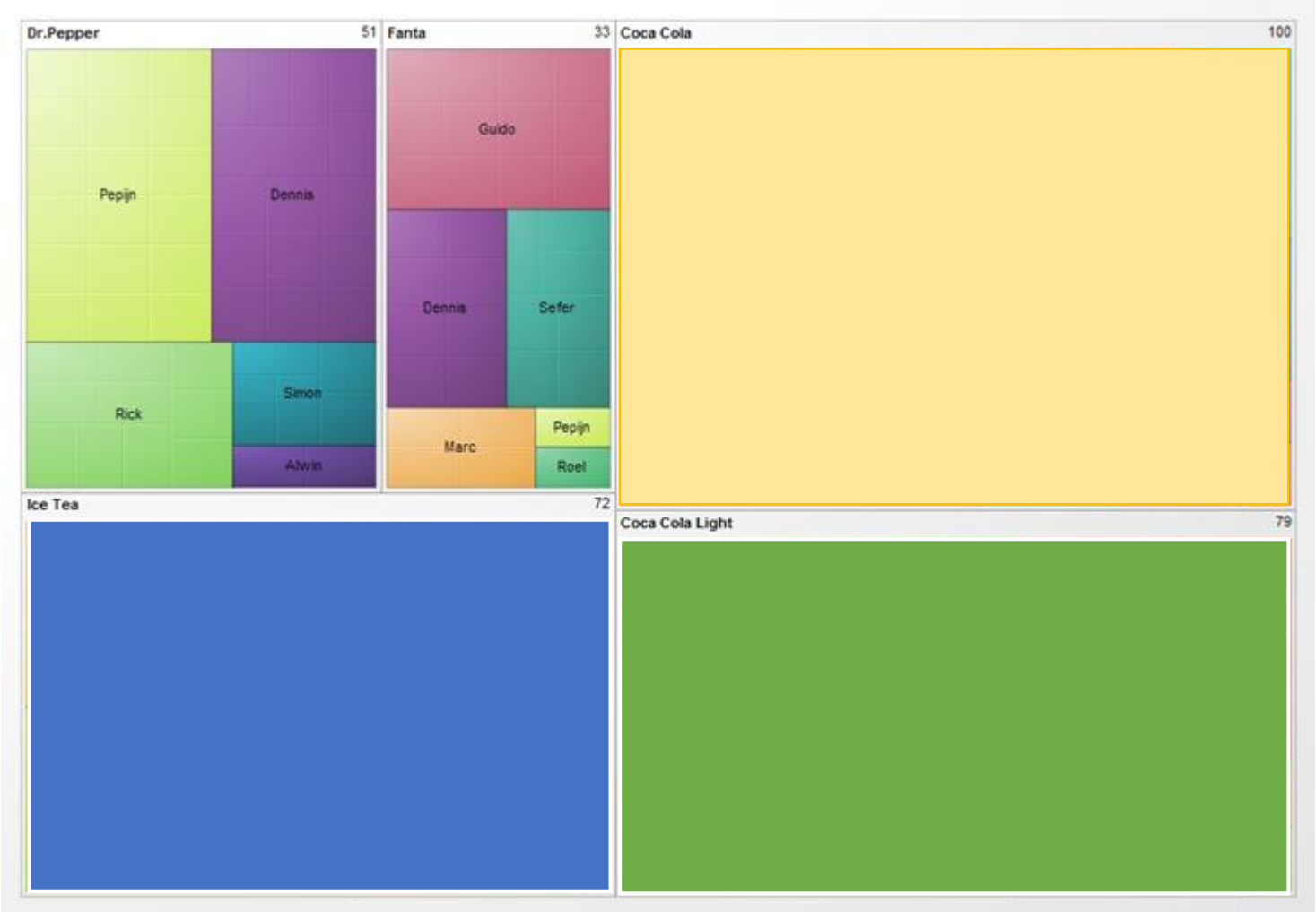


Nested Treemap

- Construct a treemap for each lower level

Nested Treemap

- Construct a treemap for each lower level



Nested Treemap

- Construct a treemap for each lower level

Nested Treemap

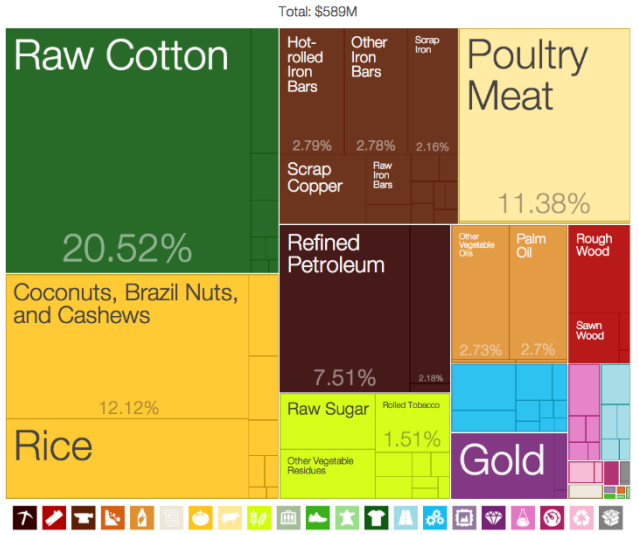
- Construct a treemap for each lower level

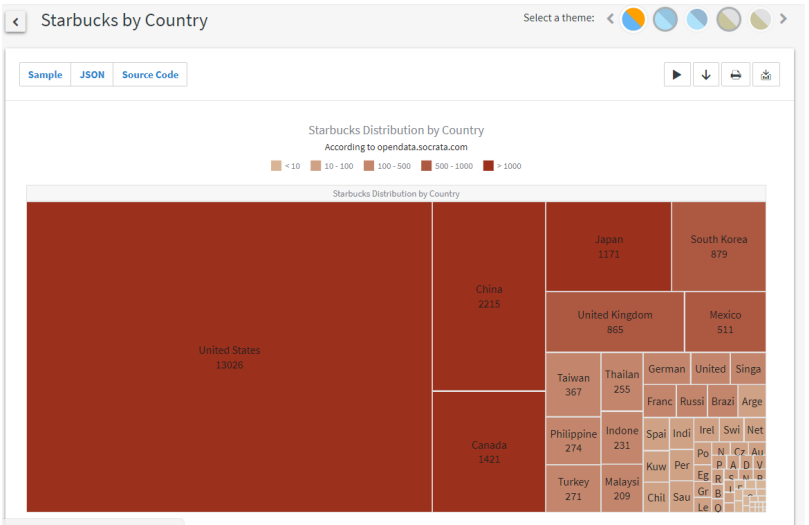
Nested Treemap

- Construct a treemap for each lower level

Note: Advantages

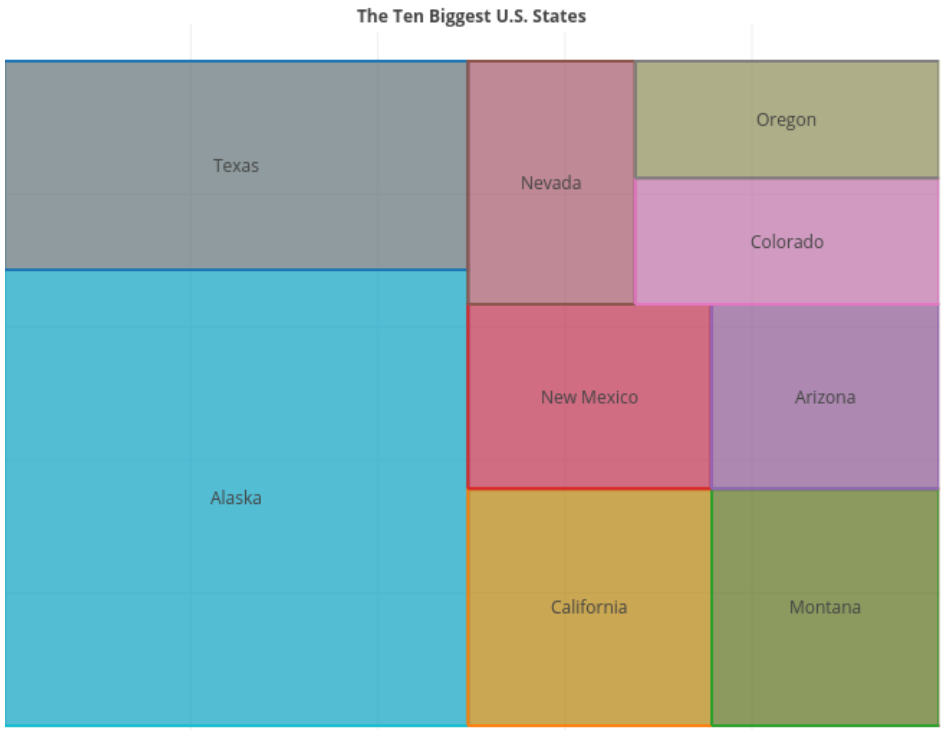
- Efficient utilization of display area
- Implicit display of structure
- Overview of entire hierarchy

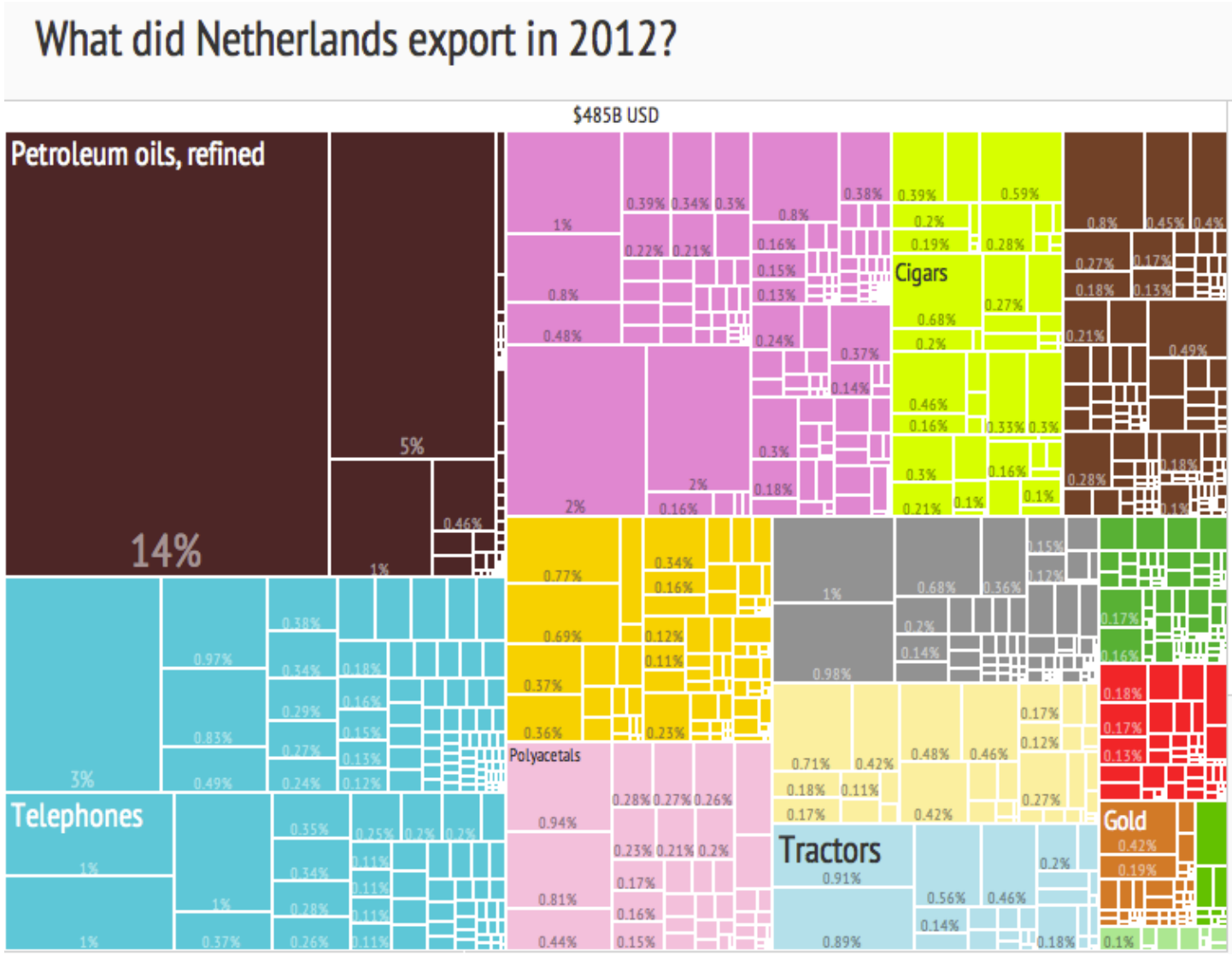




Note: Disadvantages

- The tree map is complicated when there are **many labels**.





Next class

- Treemap practice with MATLAB