Last Time

• Visualization Interactions
  – Overview + Detail
  – Focus + Context
  – Brushing & Linking
  – Filtering & Dynamic Querying
  – Details on Demand
Networks and Trees

• **Network** – A datatype that is useful for specifying that there is some kind of relationship between 2+ items
  – An item in the network is called a **node**
  – A **link** is a relation between two items
  – A network with a hierarchical structure is called a **tree**

• **Example:** In a social network graph, nodes are people and links represent friendship. The length/intensity of a link can represent the strength of that friendship
Node-Link Diagrams

- **Node-Link Diagram** – The most common visual representation of a network
  - Nodes are drawn as point marks
  - Links are drawn as line marks
  - (Think about the mathematical graph definition and representation)
You're looking at a map of mentions. Each user is connected to the people and hashtags they mentioned the most in recent tweets. Click a node to explore its neighborhood.
Flare imports
hierarchical edge bundling
Eurozone debt web: Who owes what to whom?

The circle below shows the gross external, or foreign, debt of some of the main players in the eurozone as well as other big world economies. The arrows show how much money is owed by each country to banks in other nations. The arrows point from the debtor to the creditor and are proportional to the money owed as of the end of June 2011. The colours attributed to countries are a rough guide to how much trouble each economy is in.

PORTUGAL

- GDP: €0.2 tn
- Foreign debt: €0.4 tn
- €38,081
  Foreign debt per person
- 251%
  Foreign debt to GDP
- 106%
  Govt debt to GDP

Risk Status: HIGH

Portugal, the third eurozone country to need a bail-out, is in deep recession. It is currently implementing a series of austerity measures as well as planning a series of privatisations to fix its shaky finances and reduce its debt burden. The country is highly indebted to Spain, and its banks are owed 7.5bn euros by Greece.

Back to introduction
Force-Directed Graphs

- Nodes repel each other with “antigravity,” while the links act like springs that hold them together.

  - **Pro:** Relatively easy to implement
  - **Con:** The layout is **nondeterministic**; no guarantee that it will be the same across multiple program runs
  - **Con:** Spatial position doesn’t necessarily imply a relationship, just that the nodes happened to repel into a faux-cluster
Force-Directed Graphs
Adjacency Matrix

- **Adjacency Matrix** – A Boolean 2D array:
  - Nodes are laid out in the rows and columns
  - Links are identified by TRUE values stored at the intersection of the (row, column) and (column, row) pairs that mark their endpoints
  - Can also be diagonal, only store (row, col) pairs

- **Pro:** Don’t have to deal with overlapping links
- **Con:** 1,000 nodes = 1,000,000 Boolean values
Any Questions?