Opening Discussion

- Minute essay comment:
  - Almost all of you have spatial data, but it isn't clear any of you have enough to need spatial data structures.
Quad-tree

- As the name implies, this type of tree has four children at each node.
- Typical implementation starts with a box and recursively splits it in the center along x and y.
- Data goes in leaves and splitting stops when you get down to a certain number of particles.
- Octree is the equivalent in 3-D. Approach doesn't scale well above that.
This is a move flexible option that splits on a particular axis at each node. (e.g. x=3)

Internal nodes have two children.

Split direction can change regularly or go in the direction of greatest spread.

Scales well to high dimensions. Can be built in a way that is perfectly balanced.

Let's write one.
Other Options

- There are lots of options for these.
- Point based trees keep data in all nodes and split on points.
- BSP-trees used in games/graphics. (Binary Space Partition) Like kD-trees, but split can be an arbitrary plane.
- etc.
Do you have questions about our quad-tree?