Density

PhET Sim design document

Design Team:
Jonathan Olson
Sam Reid
Chris Malley
Kelly Lancaster (lead)
Kathy Perkins
Noah Podolefsky
Trish Loeblein
(Archie Paulson should be included in final posting, probably Carl and Wendy too?)

public URL: http://docs.google.com/View?docID=dfnd4k2x_50gxv7zxfw

---

Contents

Recent Changes

- May 19, 2009 (Archie)
  - 'fluid volume' changed to 'fluid height'
- Oct 17 (Andrea Bair)
  - comments from geology perspective (see end of Comments and Discussion section)
- Oct 2 (Archie)
  - decision made to go with the "perspective" view
  - force vector legend in mockups
  - comments and decisions moved to end of document
- Sept 29 (Archie) created third tab for fluid displacement visualization
- Sept 23 (Archie) alternate "flat" view mockups
- Sept 23 (Archie) some small changes after talking with Chris and Noah:
  - visual cue for object "snapping" added
  - option for labels on objects in the second tab
- Sept 22 (Chris and Archie) comments on 3D issues
- Sept 19 (Archie) substantial changes, including
  - new behavior and interface in both tabs (see Basic Sim Operation and Mockups)

    - new document section for Usage Scenarios
- Sept 11 (Archie) changed first tab to include Wendy's bowling ball idea
- Sept 10 (Wendy, Chris and Archie) new comments/discussion
- Sept 10 (Chris) comments on 3D issues
- Sept 10 (Archie)
    - made a section at the top here for miscellaneous comments and discussion
    - made a section for outstanding issues
    - small changes made to mockups (scale in water, no more 'remove object' button)
- Sept 8 (Carl) comments
- Sept 7 (Archie) first draft of basic sim operation with mockups
- Sept 4 (Archie) first draft of learning goals

Outstanding issues

The following issues require some decision by the design team.

SR Noted:
Normal force arrow? Why is the contact force not called normal force?
Show how buoyancy is created by pressure on top, pressure on bottom, etc.

Higher priority:

- Ideas for separate tabs:
    - Carl suggested a tab exclusively for a puzzle involving Archimedes and his gold crown problem (see his comments above).
    - A separate tab could show a geophysical scenario: mantle rock rather than pool water (with no walls), and the blocks as floating continents.
- The interface for selecting objects in the Properties box may not be clear about which object is selected.
    - However, an outline (or aura) around the selected object in the play area might be distracting while you're observing it float and sink while moving the density slider.
    - Also, changing any of the properties will immediately change the color/ size of the selected object, so it would then be immediately apparent which object is under control.
    - A colored text label on the selected object might suffice (similar to the "object labels" in the second tab).
    - Chris says: I think you're going to need some indicator on the selected object (a halo, outline, or whatever) that indicates that it's selected. How else is the user going to know that there's even the concept of "selected object" in this sim? You'll also need this when 2 objects look the same, in order to tell which object the Properties panel applies to -- the alternative is modify a property in order to see which object changes, but I don't think that's at all desirable.
- Should we allow for multiple instances of "block" and "ball" objects to be removed from the toybox?
- NP: One thing that I noticed in the density sim is that all the blocks have 4 "sub-blocks". It seemed like the teachers wanted to figure out volume by counting these sub-blocks. So it might be useful to make all the sub-blocks the same size.
  So, for example, a block that was 2x2x2 sub-blocks would have a volume of 8 "units", and another block that was 4x4x4 sub-blocks would have a volume of 64 "units".

Lower priority:

- Showing force vectors:
  - Should all objects in the play area show force vectors, or only the selected object? (Or should there be a drop-down menu to choose "all/ none/selected" objects?)
  - Should each type of force vector have a checkbox so that the user could (for example) see gravitational forces but choose not to see contact forces?
- Should the sim name include something about buoyancy as well as density?
- Should the first tab have a third radio button for constant-density objects?
- We need more (and more thorough) usage scenarios, linked to the learning goals.
- The graphics shown in the mockups should be improved (if we decide we want them). Things like a more realistic Archimedes (white-bearded man in a swimsuit), more realistic swimming pool (perhaps a diving board).
- Would some sounds be appropriate for fun (like splash, bloop-bloop bubble sound)?
- What should happen when fluid gets displaced above the top edge of the pool in the second tab?
- What should happen when the fluid level goes above the top edge of the small beaker in the third tab?

Issues that were decided have been moved to the Comments and Discussion section.

Learning Goals

Students will be able to...

1. [Density/done] define and differentiate the concepts of weight, mass, volume and density (all variants of the vague idea of "amount of stuff")
2. use the (metric and English) units of weight, mass, volume and density
3. [Density/done] describe how the concept of density relates to an object's volume and mass
4. [Density/done] explain how objects of similar volume can have differing mass, and how objects of similar mass can have differing volume
5. [Density/done] explain why changing an object's volume or mass does not affect density (ie, understand density as a characteristic property)
6. apply the definition of density to both liquids and solids
7. describe how buoyant force is related to relative density
8. [Density/done] relate relative density to the volume-above-liquid-surface of a floating object
9. predict the weight of a (completely or partially) submerged object of known mass and volume
10. predict how an object will move (sink, float or otherwise) when placed in a liquid, given densities of the object and liquid
11. describe the forces that act on a (completely or partially) submerged object at rest
12. relate the buoyant force on an object to the weight of liquid displaced by it
13. illustrate the principle of a boat's buoyancy, even when it holds objects with density greater than the underlying liquid
14. describe how a boat's properties must change to carry a heavier load
15. [Density/done] measure the volume of an object by observing the amount of fluid it displaces, and subsequently compute its density
16. [Density/done] identify an unknown material of arbitrary size by measuring its density (and comparing to a table of known densities)

17. describe the average density of an object comprised of two materials (eg, metal and air)

Density: goals 1-6
Buoyancy: goals 7-14
Challenge/Puzzle: goals 15-17

Tips: ideas to include- purpose of bottle,

Basic Sim Operation

General:

- context is a swimming pool, with its cross-section visible
- three tabs available (see below)
- objects
    - can be dragged around; they can be held underwater or in the air, but will fall/sink/float when released
    - object color
        - for an object with a "custom" material, it has a uniform color that corresponds to its density
        - for an object with a particular material (such as iron, see below for options), it will be colored by a texture pattern
    - object interaction
        - Objects can be stacked.
        - If the lower object has a flat top, the upper object will snap to the center of the lower object.
        - If the upper object is more than half-off the lower object, the upper object will snap to the left or right and continue to fall/ sink/float on its own.
        - The upper object will also snap away from the lower object if the lower object does not have a flat top (eg, the man, sphere, etc).
        - If two objects are stacked on a single lower object (ie, 2 on 1, not 1 on 1 on 1), both upper objects should snap to a position such that the load on the lower object is uniform about its center of mass.
        - While snapping (for maybe a half-second or so), the user should see a vertical line on the relevant objects through the center of mass of each object (these lines will become aligned during the snap). This will serve as a cue that the computer is doing something non-physical.
        - Objects should not snap in the depth dimension (ie, forward and back); they should always stay near the front (cross-section) plane. This is non-physical since there is no tilting, but that's ok.
- scales
    - one scale on the ground
    - one scale in the pool
        - can be moved around (by grabbing the scale's platform; the scale readout stays on the bottom of the pool)
        - the scale platform can be used to lift objects partially or fully out of the liquid
    - scales act like a "lower object" for purposes of object interaction (see "object interaction" bullet, above)
- features for animation
    - brief bobbing of a floating object after it's moved or changed

        - brief ripples in liquid after it's perturbed
        - Archimedes does something funny (eg, holds his breath when underwater)
- "Reset All" button goes back to default settings
- "Help" button will pop up text bubbles which differ according to tab (see below)
- help menu
    - include a table of common materials and their densities (for guessing the mystery materials)

First tab (see mockups):

- when the sim starts
    - four fixed-mass objects of different shapes and densities fall from the same initial height (all are labeled with their mass)
    - one falls onto the scale on land, the rest fall into the pool (kind of like the Masses and Springs startup)
- radio button allows for either four same-mass objects (default) or four same-volume objects (see mockup)
    - when the radio button changes state, all objects disappear and the other set appears and falls
- objects will be colored according to their density (like a "custom" material)
- object and liquid properties are not editable
- help button shows help bubbles
    - "objects can be stacked"
    - pool-scale: "move the scale platform to lift objects"

Second tab (see mockups):

- layout
    - there is a "Toybox" and a "Properties" box above the pool
    - the Properties box defaults to hidden (see mockup)
    - when the Properties box is hidden, the Toybox is larger; otherwise it's shorter (see mockup)
- Toybox behavior
    - holds all objects not in the play area
    - the contents of the toybox scroll by using the "More" buttons (as in "Eating & Exercise")
    - objects can be dragged out: they then land in the play area and are removed from the Toybox
    - objects from the play area can be dropped in the toybox: they are then removed from play area
    - objects can be selected while they're in the toybox, and their properties appear and can be altered (this is to allow the user to make a steel ball, for example, and *then* drop it in the pool)
    - objects can appear smaller when in the box (scaled to fit), but when they are dragged out they have the correct size dimensions
- Properties box
    - selecting objects
        - clicking any object (or the pool's liquid) will show its properties in the box
        - properties shown will change depending on the type of object (eg, radius instead of height,etc for a sphere)
        - all objects are also selectable in drop-down
    - "material" drop-down
        - materials types include "custom", "iron", "cork", "oak wood", "pine wood", "lead", "ice", "mystery"
        - if the liquid is selected, then options include "custom", "pure water", "seawater", "mercury", "gasoline", "mystery"
        - each material has a characteristic density

            - moving the density slider will change the material type to "custom"
        - "density" slider
            - changing density should change the color of the object to signify change of an intrinsic property
            - volume will remain constant; mass will change
            - moving the density slider will change the material type to "custom"
            - this slider will be greyed out for the "fixed density" objects (see "General", above)
        - "mass" slider
            - also changes volume (at constant density) by growing/shrinking all object dimensions uniformly
            - greyed out for the "fixed density" objects
        - size dimension sliders
            - also changes mass (at constant density)
            - greyed out for the "fixed density" objects
            - controls depend on object selected:
                - for blocks: three sliders (height, width, length)
                - for spheres: one slider (radius)
                - for the liquid: one slider (height)
        - "volume" readout shows current volume (not editable)
        - "percent volume submerged" readout shows what percentage of the volume lies below the fluid surface (not editable)
- "Show" controls:
    - "Force Vectors" (see mockup) -- when selected the following forces are shown for the selected object:
        - gravity
        - buoyancy
        - fluid drag (for sinking objects)
        - contact forces (floor, scale or inter-object contact)
    - "Fluid Height" (see mockup) -- when selected, the height of the liquid is shown (with many significant figures for measuring fluid displacement of immersed objects)
    - units selection: all displays/controls will change to selected unit system
    - "object labels"
        - drop down menu with the following choices:
            - none (default)
            - mass
            - volume
            - density
        - when something other than "none" is chosen, text labels appear on each object, displaying the value of the selected quantity
        - the labels appear as text on objects (like the text seen on objects in the first tab), the fluid will also be labelled (perhaps in the lower right corner)
- available objects (two types):
    - variable density:
        - Block A, Block B, C & D (four blocks total)
        - Ball A, Ball B (two balls total)
    - fixed density
        - Archimedes (ie, a person)
        - oak log
        - heavy bowling ball, labeled with weight
        - light bowling ball, labeled with weight
        - mystery object (all properties display as "?")
- additional object properties:

        - volume of any object can be changed by dragging its edges (like resizing a desktop window)
    - Help button shows the following help bubbles:
        - "objects can be stacked"
        - pool-scale: "move the scale platform to lift objects"
        - toybox: "remove objects by returning them to the toybox"
        - liquid in pool: "the type of fluid can be changed in the Properties box"

Third tab (note: mockups are a rough draft for now, drawn without perspective):

- layout
    - Same as second tab:
        - toybox
        - properties box
        - reset & help buttons
    - Different from second tab:
        - beakers instead of pool (see "beaker behavior" below)
        - objects cannot be moved to the right of the big beaker
    - starting behavior is having a box fall onto the scale
- beaker behavior
    - default fluid level is at the top edge
    - when a floating object is placed in the big beaker, fluid spills over to small beaker (and the small-beaker scale reads the same as the on-land scale with the floating object)
    - a sinking object will sink to the in-fluid scale, and fluid will spill over (then the sum of the two right scales will equal what the on-land scale read with the sinking object)
- "Reset Fluid Levels" button
    - objects in beaker will return to toybox (instantly)
    - fluid in big beaker will rise as the fluid in the small beaker goes to zero

---

Usage Scenarios

The following are scenarios illustrating how we hope users to interact with this sim. Scenarios should be linked to the learning goals.

- Bob wonders why they say that 90% of an iceberg lies below the water. He goes to the second tab, selects a block in the toybox, changes its material to "ice", then drops it in the pool. He plays with the dimensions of it and sees that it floats very low. (Learning goal 8)
- Alice wants to know how an object's weight changes when submerged. In the second tab, she pulls a block out of the toybox and places in on the scale, where it's weight shows as "1470.0 N". She then drags it to the pool and sees that it floats. She grabs the scale platform and lifts it to the bottom of the floating block. The scale reads "0.0 N" until it touches the object. As she lifts the object further with the scale platform, the scale readout begins to show values that increase from 0 to 1470 when the block if fully out of the liquid. (Learning goal 9)
- Bob has heard of "Plimsoll lines" and wants to experiment with how deeply a loaded boat will float. He goes to the second tab and drops a wooden block in the pool. He then makes the block longer and less dense, and sees that it floats very high. He looks at how much of the "boat" is below the fluid level (in the "percent volume submerged" readout) -- it's a low number, around 10%. He then finds an iron ball in the toybox and drags it out to a point above the boat block and releases the mouse. The iron ball falls onto the boat and snaps to the boat's center, and the boat then floats lower in the fluid. Bob looks at the

"percent volume submerged" readout and sees 0.0%, but realizes (because of the image under "selected object" in the Propoerties panel) that the readout number is for the iron ball. When he clicks the boat, he sees that it is now 50% submerged. He wants to watch this number as he changes the mass of the iron ball, so he selects "% volume submerged" from the "Object Labels" drop-down. The boat then reports that value even when the iron ball is selected and is changing mass. Bob feels all warm inside. (Learning goals 13 & 14)

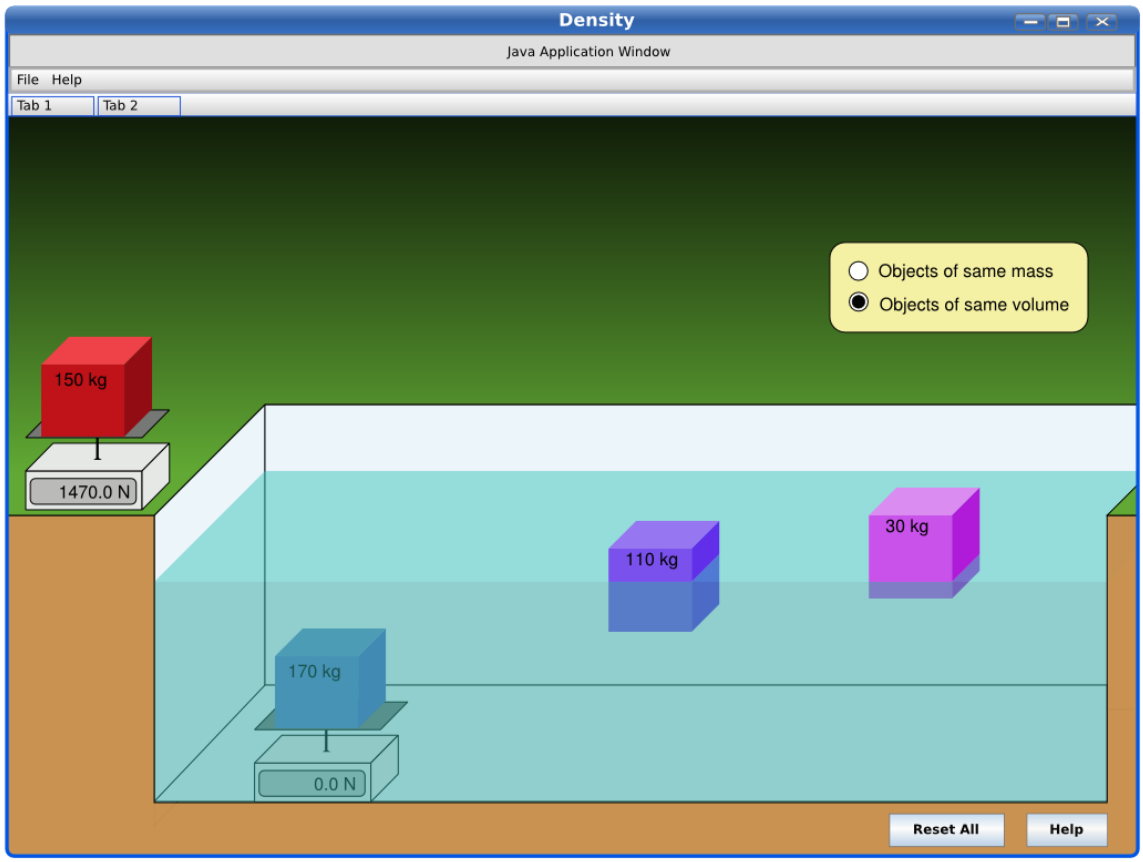
Should write up scenarios for these:

- Measure the density of the "mystery" object by weighing it and observing the volume of liquid it displaces.
- Measure the density of the "mystery" liquid similarly (with an object of known density).

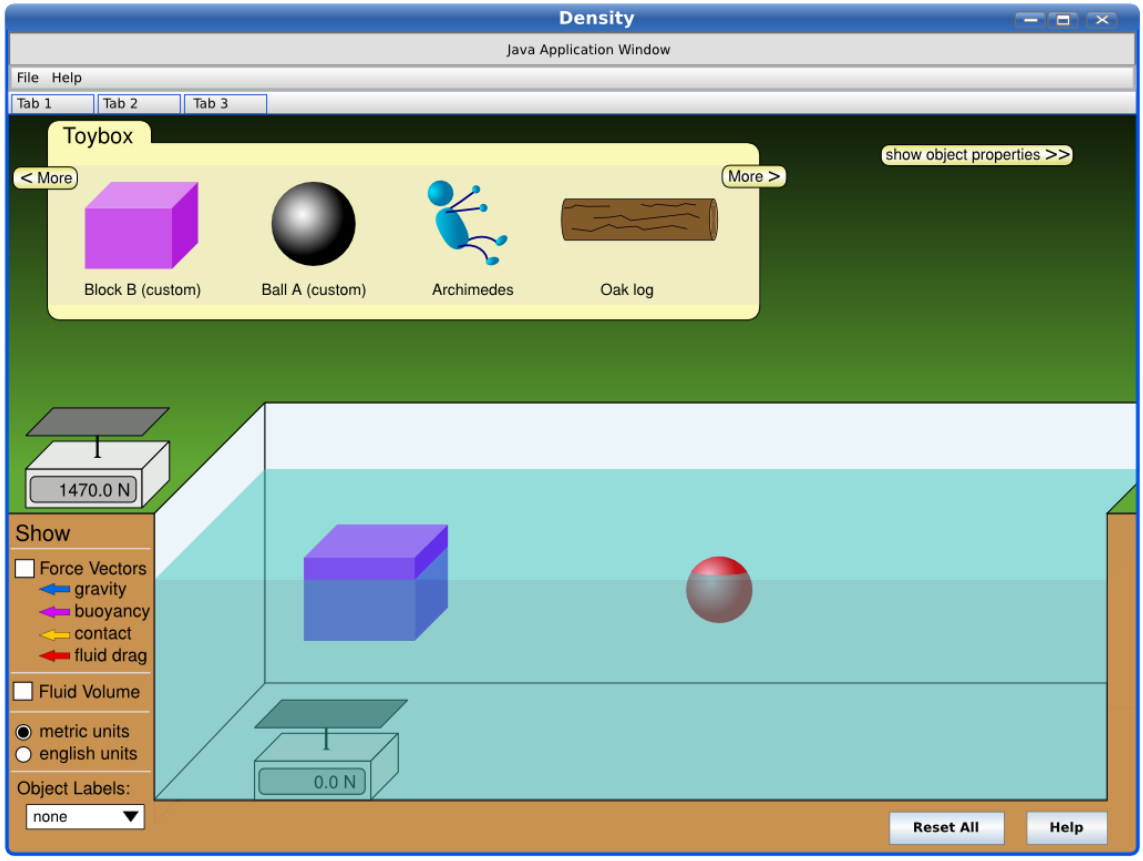
Mockups

First tab (default):

First tab (selected same-volume objects):

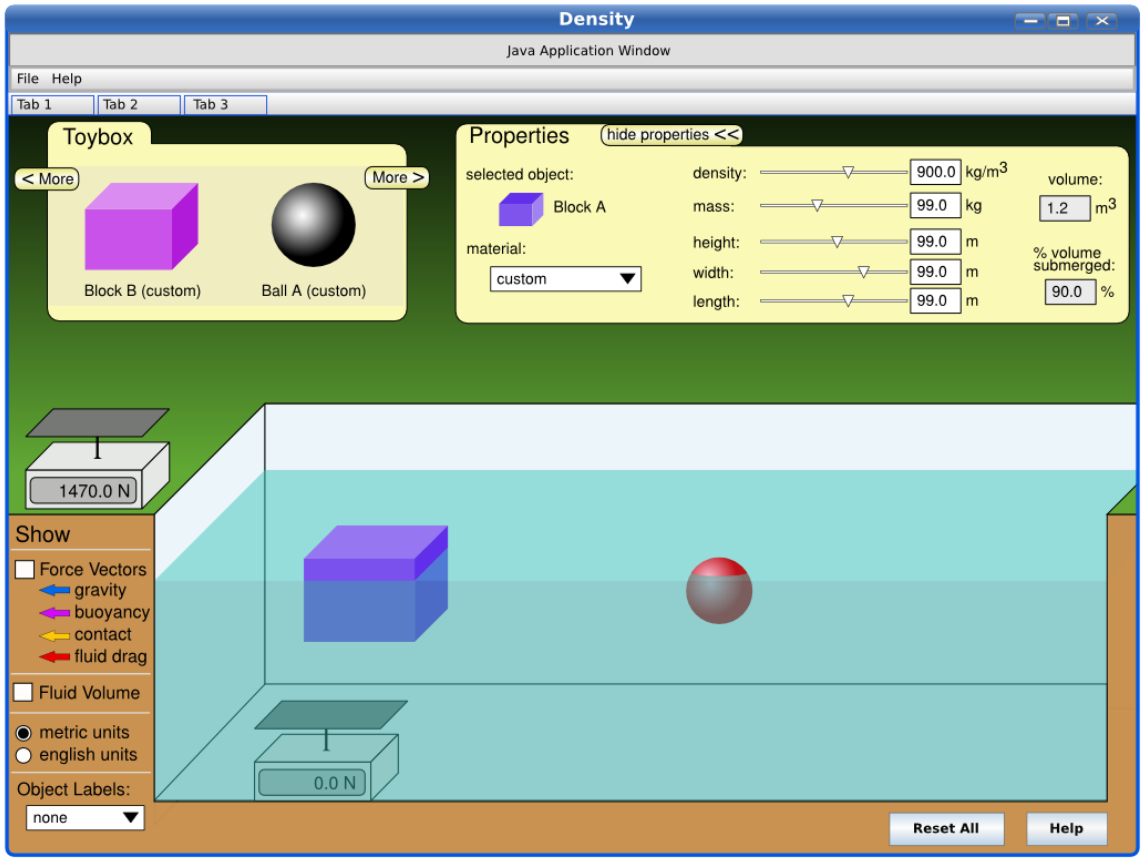


Second tab (default):

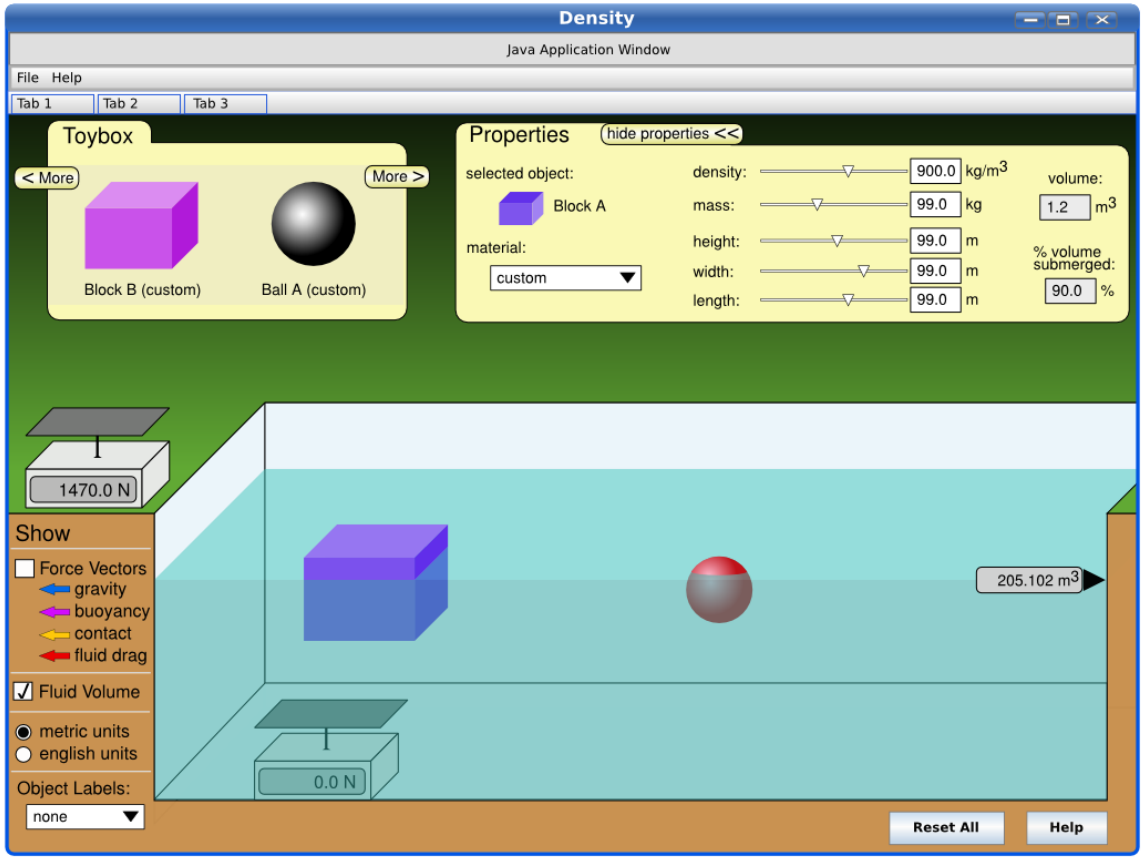


Second Tab (view Properties box):

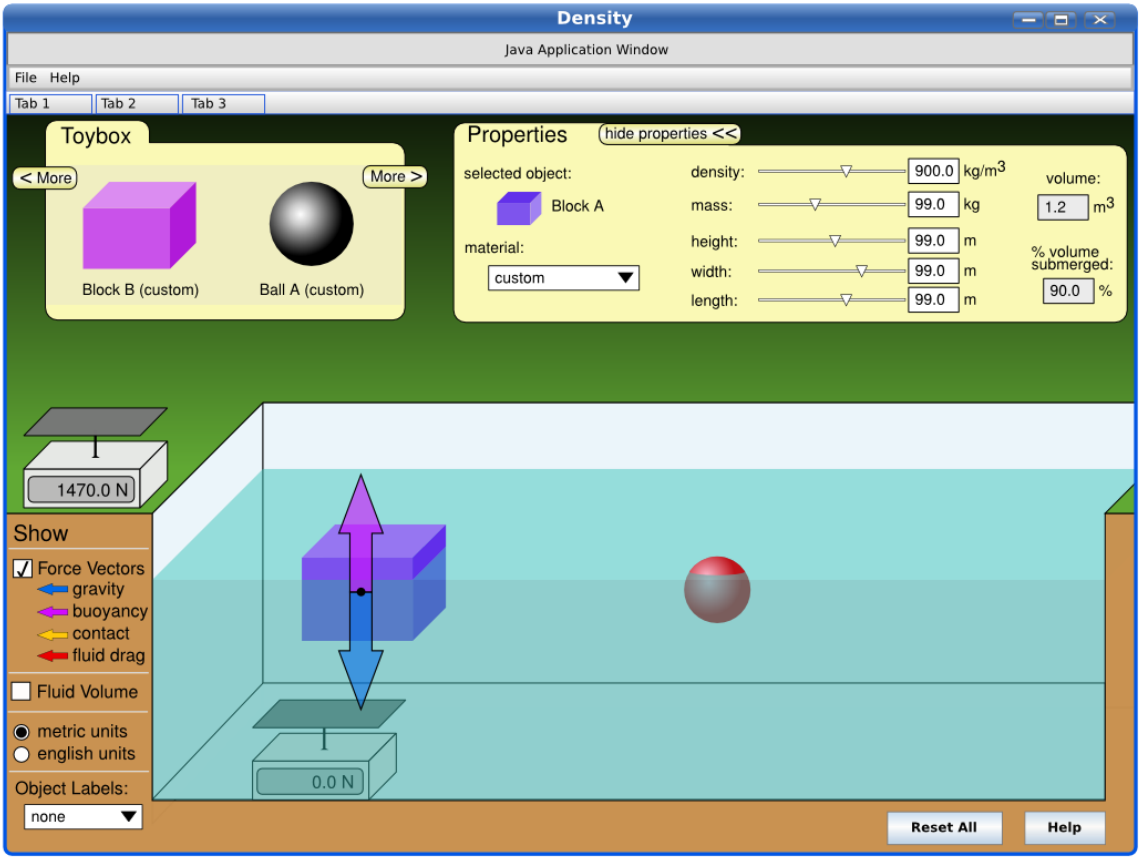
+



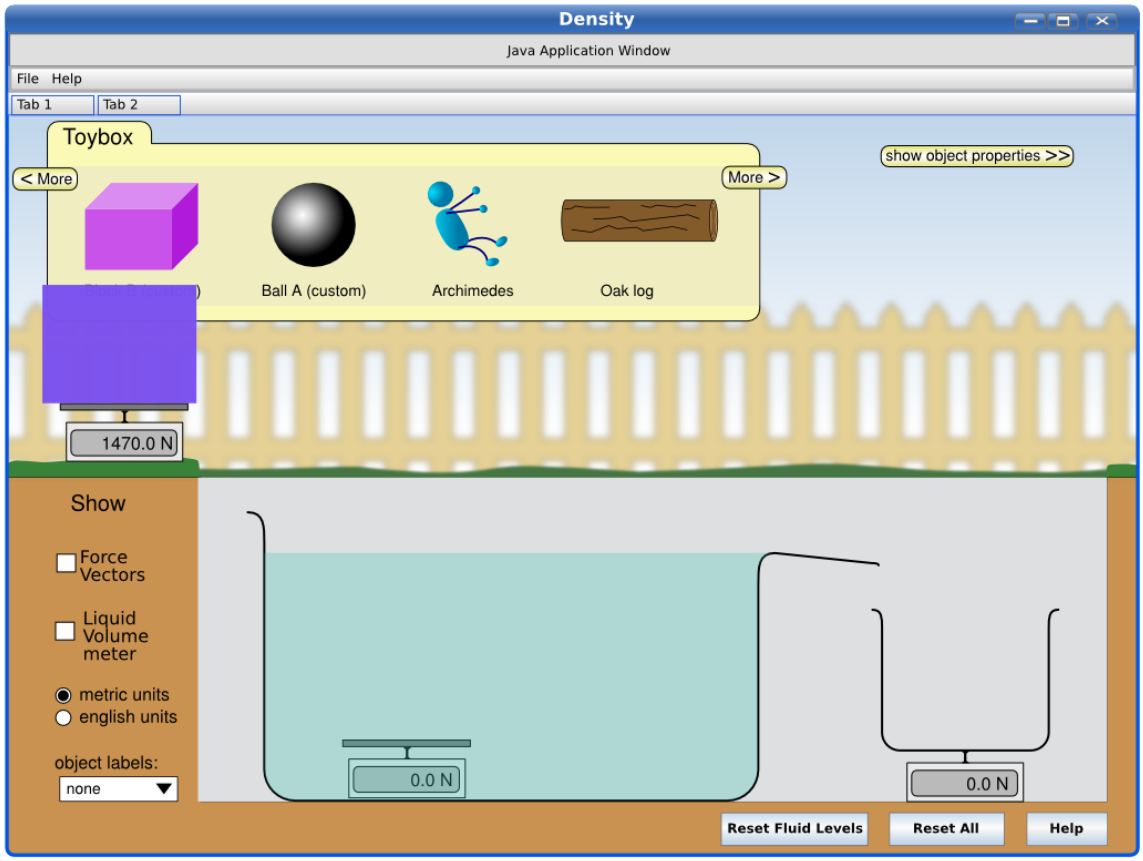
Second Tab (Fluid Height selected):



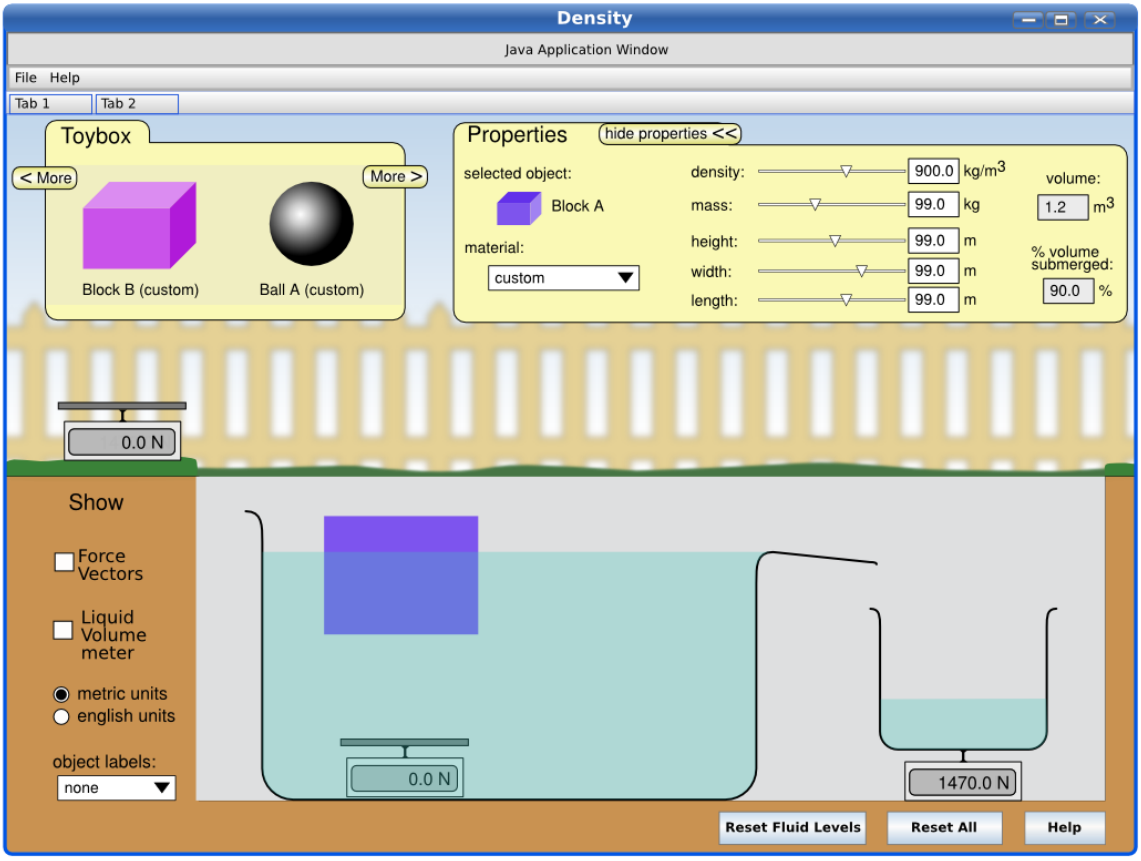
Second Tab (Force Vectors selected):


Density
Java Application Window
File Help
Tab 1
Tab 2
Tab 3
Toybox
< More
More >
Block B (custom)
Ball A (custom)
Properties
hide properties <<
selected object:
Block A
material:
custom
density:
900.0 kg/m3
mass:
99.0 kg
height:
99.0 m
width:
99.0 m
length:
99.0 m
volume:
1.2 m3
% volume submerged:
90.0 %
1470.0 N
Show
Force Vectors
gravity
buoyancy
contact
fluid drag
Fluid Volume
metric units
english units
Object Labels:
none
0.0 N
Reset All
Help

Third Tab (default), flat view:

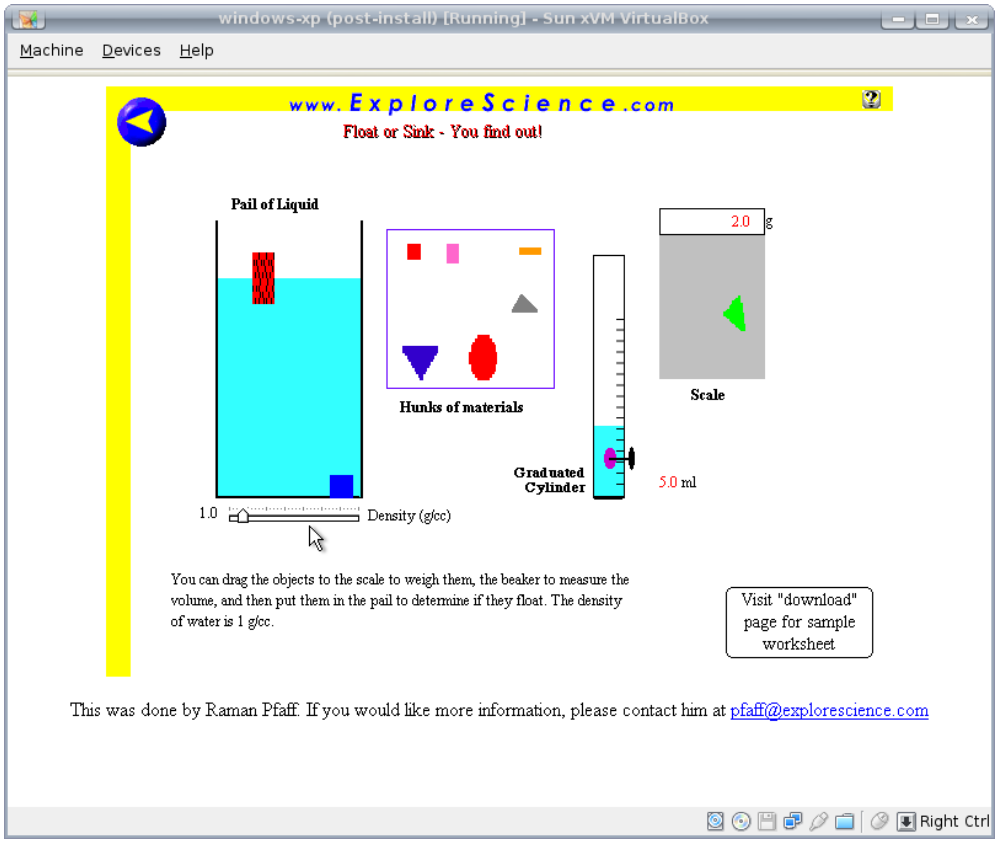


Third Tab (after moving object), flat view:



Other Resources

- University of Washington has two (coupled) papers on student understanding of Archimedes principle from 2003, and a tutorial on buoyancy:
    - Helping students develop an understanding of Archimedes' principle. I. Research on student understanding
        - common misconceptions include:
            - Objects sink to a depth that depends on their density (false for incompressible fluids).
            - The buoyant force depends on an object's depth or mass.
            - The volume of fluid displaced by a fully submerged object depends on its mass or density, or the density of the fluid.
        - Part of the difficulty stems from problems with balancing forces and applying Newton's 2nd Law.
    - Helping students develop an understanding of Archimedes' principle. II. Development of research-based instructional materials
        - based on labs in which students observed objects of varying mass, volume and density to discover rules for what floats
        - tutorial
            - Buoyancy tutorial is preceeded by a tutorial on hydrostatic pressure.

        - Sect I: "Guides students from the assumption of a linear variation of pressure with depth to the conclusion that a fluid exerts a net upward force that does not depend on depth or the mass of the object."
        - Sect II: addresses displaced volume misconceptions with questions.
        - Sect III: "The students are asked to evaluate a statement by a fictional student who claims that [Archimedes'] principle implies that the weight of the liquid displaced by an object is equal to the weight of the object itself."
        - Sect IV: Balance forces to predict floating or sinking.
    - Lab experiement also designed using spring scales.
    - Students who used the tutorial did better on questions related to the misconceptions (identified above).
- a simple flash sim that kind of does what we want:
    - www.sciencejoywagon.com/explrsci/media/density.htm (requires a shockwave plugin)
    - screenshot:
    - 

Comments and Discussion

This section is an archive of comments and responses by the design team. Issues that still require attention are in the Outstanding Issues section.

Decided issues (items previously in "outstanding issues", but now decided):

- Perspective or Flat view?

- See the previous versions of the design document for mockups of a rough sketch of this sim with just a plain flat view
- Reasons for flat view:
    - A "flat" view would be much easier to implement.
        - (Chris) The "flat" visualizations would be much easier, resulting in an implementation with complexity similar to ph-scale. The visualizations with 3D perspective are more similar in complexity to glaciers (regardless of whether we use a 2D or 3D library). We're looking at something like an order of magnitude difference in the effort required for these 2 approaches, so this is indeed an important decision. If we used the "flat" visualizations, do you still intend to show 3D objects in the toolbox? I guess this might be useful, to convey that the objects do have depth, but that we're not going to show the depth.
        - (Chris) Regarding the new 3rd tab... Visualizing fluid flow from one beaker to another would be a piece of cake in 2D. It's going to be very complicated with fake 3D, and even more complicated in true 3D (sinceit would involve creating a moving mesh of geometry).
    - There is a Washington tutorial on buoyancy that uses "flat" pictures, and they showed good learning gains (see their papers in Relevant Resources, below).
        - (Noah P) Fact that UW tutorial works in 2D is good evidence it can still work.
    - (Trish) Textbooks consistently mix both perspective and flat representations in their diagrams.
        - (Noah P) Use in textbooks provides no evidence that 3D is actually better.
    - (Trish) Students draw their own sketches in 2D 'flat' view, and so are more familiar with them.
    - (Trish) Drawing a force vector coming from the center of a block in 3D perspective could be confusing.
    - (Trish) Some of the favorite sims for kids are the Flash ones, which are 2D.
    - (Noah P) It is easier to see whether objects are floating at the same height, and it actually seems cleaner with less distraction by the cool 3D graphics.
    - (Noah P) Although you don't have a direct visualization of volume, the idea of "heavy for size" or "light for size" can still be completely conveyed in 2D.
    - If we use the "perspective" (not "flat") view, the types of objects available will be fixed once sim development begins (due to the difficulty of implementing special shapes). We must be sure that we're all ok with that.
- Reasons for perspective view:
    - Flat view does not allow for proper volume (and fluid displacement) visualization
    - Not sure if flat view can it satisfy our learning goals for understanding volume.
    - (Wendy) Responding to Chris's point on implementation of the 3rd tab (above): I would think the water part can be real 2-D not problem. I think you still could have the 3-D (fake 2-D) looking image of the block but all the spill over and containers in 2-D.
- As a possible compromise, there appear to be some possibilities for using the perspective view with 2D code, but borrowing some libraries for 3D modeling. A few hours (<10) of investigation might allow for some

prototyping to see what the options are. However, even with prototyping, using 3D libraries might possibly open up some unanticipated problems (recall the difficulties of moving to piccolo).
    - Other related, but less important issues:
        - If we go with flat view, should the objects in the toybox (or the icon in the properties) still appear in perspective? This might remind users that objects are still 3D though they appear flat (ie, that the absence of perspective in the play area is just a simplification for display).
    - Decision: we'll go with the "perspective" view, but will implement it with 2D code
- How should we allow for objects to rotate?
    - rotation will be desirable when:
        - a block acting as a "boat" would realistically tilt when asymmetrically loaded
        - it would be nice for a block to rotate as it falls off another object (eg, if the block is put on the land-scale, but is mostly hanging off, it should fall off the scale with some rotation)
        - to be realistic, the shortest dimension of a floating block should flip to be on the vertical axis (though we could probably ignore this subtlety)
    - Chris says object rotation will be difficult with a 2D sim (eg, changing lighting on object surfaces)
    - Cheap rotation idea:
        - Archie says we may be able to do this cheaply by storing an image of the object, along with an image of it tilted to the left and an image of it tilted to the right (or perhaps even 5 images: center, little-left-tilt, greater-left-tilt, little and greater right-tilt). Animation would then be choppy, but the idea would be there. Chris thinks this may look very bad. Perhaps he could make a test of it so we could see.
        - Chris: I'm not at all convinced that this will be an acceptable solution to rotation. It won't look at all physically correct, would require a large number of images to look smooth, and generating such images for anything other than simple shapes will be a daunting task. I think Carl's proposal to avoid rotation altogether is a better way to go.
        - Wendy: if we can avoid rotation all together..., which I think we could, it'd be better.
        - Archie: Wouldn't no rotation look far less "physically correct" than cheap rotation? I think it would. It need not look smooth, but it at least gets the idea across -- acknowledging the physics without having to show it accurately. This issue, after all, is not part of the learning goals, just appearance.
    - No rotation idea:
        - Carl says: I think we can have [tilting and rotation] be quite unrealistic, without it making any significant impact on effectiveness of sim.
        - Wendy suggests we have objects snap to being on or off an object (or scale) below if it lands partially on it. If a floating platform is loaded with another object, the top object should snap to the middle. This seems reasonable for a first attempt.
    - Decision: For now, we should have no rotation. (If this looks very bad we might need to reconsider.) Let's just snap objects horizontally such that
        - when stacked, the upper object snaps to the center of the lower object

        - when hanging more than half-off, the upper object snaps to the side
- 3D issues
    - Will we want 3D sims in the future? If so, maybe we should consider all the challenges listed by Chris in his comments.
    - can we live with the constraints of a 2D model?
        - difficult coding object rotation
        - object rotation may not look so good
    - Carl says: If we stick with simple shapes like rectangles or spheres and we do not allow them to rotate, can't we make them look 3D and use underlying volume and buoyancy calculations based on 3 d, but actual images will always be 2 d, as in other sims? Seems like we can satisfy all the learning goals with that.
    - Decision: We'll stick with 2D for this sim.
- object collision behavior
    - Need to work out collision behavior of the objects and the (in-fluid) scale platform. Should floating objects get pushed around when hit?
    - Decision: keep it simple for now.
- objects available
    - Are we ok with having mostly simple objects such as blocks and spheres?
    - including a new object in the sim will require
        - providing a 3D shape model (for fluid displacement)
        - images that allow for 3D rotation in the plane of the sim window
    - if we want more than 5 or 6 objects, we need an interface for adding and removing them
        - One possibility could be having many objects in the "select object" drop-down menu, and selecting one that's not visible will make it fall onto the heap of other objects. Then we'd also have a "remove object" button on the "properties" box.
        - Chris doesn't like the idea of having the same tool for object creation/deletion as for object editing.
    - Decision: new interface designed (scrolling toybox)
- How do we distinguish mass and weight in imperial (English) units?
    - Decision: use pounds for weight and slugs for mass.
- Which force vectors should be shown? For now we have gravity, buoyancy, fluid drag (for sinking objects), inter-object contact forces.
    - Decision: use gravity, buoyancy, fluid drag (for sinking objects), inter-object contact forces.
- Fluid displacement visualization (Wendy)
    - To see the volume/weight of liquid displaced I think it'd be nice if we had some sort of container that was full to the top. Then when you add an object, water/liquid spills over into some sort of container below. Then you can weigh the spilled over liquid and find it's mass. It may be a little hard to make a realistic scenario for this but students have a very hard time with considering the mass of the liquid that raise up and they often don't make the connection that the level the water raised is equal to what was displaced. The spilling over scenario should make that connection much more explicit.
    - Decision: added as the third tab

Objects and materials:

- Wendy says: Might be fun to use bowling balls. They're the wrong mass for your start up but might go elsewhere. That's a real life thing that is the exact same size but different masses and everyone knows it. You could put the weight labels on them just like they really have. 8lb bowling balls float in water and heavier ones do not.
    - Archie (Sept 11): I've sketched up something to this effect.

- Wendy says: I would suggest using Lead instead of iridium. Students have heard of lead but not iridium and lead is amazingly dense so should get the point across fine.
    - Archie (Sept 11): lead has only half the density of iridium. Perhaps use both? I like the idea of an exotic element, and Carl was asking for a very high-density object.

3D issues:

- decision made on this -- see it in the Outstanding issues, below
- Archie's original comments:
    - The sim should be seen in 3D perspective, since the visualization of volume is critical to the learning goals.
    - Object collision control may have to be 3D-aware; perhaps objects could be in either the background plane (falling on ground when released) or in a foreground plane (falling in the liquid or on scale when released).
    - We might need a real 3D model to implement object rotation features (see outstanding issues, below)
    - Considering Chris's comments (below) on 3D issues, I think we can and should do it in 2D.
- Chris says: If 3D visualization is indeed "critical to the learning goals" for this sim, then you will need an underlying 3D model, and a 3D software library. There is simply too much 3D here, and faking it in 2D will (a) be unnecessarily expensive and (b) result in code that is hard to maintain. I say this based on recent experience with glaciers.
- Chris says: So the critical decision is to either move forward with 3D or find a way to do this sim in 2D. If you choose to move forward with 3D, the first thing we'll need to do is find a third-party 3D library. A 3D library needs to satisfy these requirements:

1. Meets PhET licensing requirements. The "no GPL" issue immediately rules out some very good packages.
2. Programming API easy to work with and well-documented. Ideally, we'd like something of the same quality as Piccolo 2D.
3. Performance is decent for all PhET users. This is going to be almost impossible to satisfy. 3D requires more horsepower, and older machines aren't going to be able to keep up. I'm afraid that you'll need to live painful performance on older machines.
4. Meets PhET's general needs for 3D sims. It's difficult to know PhET's 3D needs until we've done a few 3D sims. But we don't want to end up supporting code based on multiple 3D libraries. We should expect to do a few prototypes with different 3D libraries, but settle on 1 library to satisfy all PhET needs.
5. Is platform independent. Again, difficult to satisfy. 3D packages often have a platform-specific component, to take advantage of hardware acceleration (see 3 above). Hard-specific components may include (for example) a Windows DLL. This complicates development, testing, delivery, build process, etc.
6. Is a reasonable size. JMonkeyEngine (for example) is somewhere around 10MB, very large relative to the size of PhET sims. Do we want to bundle this with every 3D sim? require users to download and install it separately?...
7. Fits with PhET "single jar" delivery model for sims. Related to 6 above. How do we deliver and install a 3D library?...
    - Chris says: There are also issues with 3D interfaces:
        - 3D user interfaces are harder to use than 2D user interfaces, and typically require instruction for the user
        - 3D user interfaces are harder to design than 2D user interface
        - user interaction is less intuitive, primarily because the mouse is a 2D input device
        - there is a large body of established techniques & conventions for 3D user interfaces, and PhET designers will need to become familiar with this prior work

- On the other hand, Chris says (Sept 22):
    - I don't think we should be trying to fake 3D with a 2D library.
    - Faking 3D in this sim is going to be much more difficult than it was in glaciers. Even without rotation, some of the stuff (eg, making objects look like they are partially submerged) is going to range from difficult to horribly complicated.
    - The need to "keep our options open" is at odds with the simplifying assumptions that I'll need to make in order to pull off the faking. If you decide at some point that you want (for example) more convincing rotation or interaction of objects, the probability is high that I will not be able to accommodate some changes with a 2D library.
    - Current requirements will be difficult to meet in 2D, and some of the stuff you might want later will likely be impossible (either not technically feasible or prohibitively expensive). This tells me that a 2D implementation is the wrong way to go.

Comments by Carl on Sept 8:

- A couple of minor things that occurred to me:
    - scales should be raised up a bit so that it is obviously in the water
    - unless there is some good reason not to, should keep the control panel on the right hand side to be consistent with most of the other sims.
        - Archie says: We need the horizontal space. This happened in Glaciers, too. A lot of the action happens in a wide area, with room to spare at the top and bottom. Imagine filling the pool with a few big objects (like the first tab) -- you run out of room quickly.
- Other passing thoughts to consider are a couple of game like things:
    - have set of very dense weights (almost no volume) on side that one could set on top of floating objects and could see how they float deeper, and eventually sink. (don't let them roll over so weight falls off, as always happened when I tried this as a kid).
    - an "Archimedes and the crown" panel where there is a big yellow crown, and they have to figure out how much gold it contains. The crown should have come up with some semi-random density based on reasonable possible ratios of silver to gold. In previous sim panels, they would be able to get density of gold and silver, and then in this panel, they would be able to put crown into water and measure its volume and also weigh it outside of the tub, so they calculate its density. Then they could put in a value for silver to gold ratio, and then gets suitable response if they are right or wrong.
        - Archie says: I like this idea. We also could, as a simpler alternative, just have "Archimede's Crown" as one of the mystery objects. Then we could put both gold and silver densities in the materials table, and the same puzzle would be do-able. Perhaps not as fun, though? Either way, this puzzle would require some guidance/instruction.

Miscellaneous comments (Archie):

- Concerning changing object density... There should be controls for changing the volume, mass and density of a given object. However, I don't think that we should be changing mass while holding volume constant (or vice versa) for two reasons:
    - The middle school learning goals that I've seen in Trish's textbooks emphasize that density is a "characteristic feature" of a given material. Thus you can change the shape and size of an object (of uniform material type), but you won't change its density. I think that we can have a density slider which will also change the colour (or some other visible aspect) of the object. But increasing the object's mass should

also increase it's volume. Similarly, changing an object's volume should change its mass (at constant density). Only by changing material type can the student change the density, and that can be done by either selecting a different material (from a drop-down menu, perhaps) or by explicitly changing the density with a slider and thereby changing to a "custom" material (kind of like the pH Scale).
    - I noticed a student difficulty with the Torque/Angular Momentum sim: you can change the inner and outer radii of the platform (in the Moment of Inertia tab) without changing the mass, thus rearranging the constant total mass. But the picture looks like you're just cutting away from or adding more to the platform. That's probably how it should work in that case, but students did not realize that the mass was constant.
- Do we want objects such as a hollow metal box (for learning goal 17)? Such an object should probably change density as its volume changes, since we would want to keep the wall thickness constant. This seems too hard to implement, so we'll skip it.

Geology perspective (Oct 17, Andrea Bair)

- For the intro level courses, learning goals 1-6 are appropriate, and we have very similar statements of goals. What you're describing and showing in the sim mockups for these goals for the most part are very appropriate for use in our intro geology courses.
- I notice you address the issue of changing object densities... I've been talking with faculty here about what they think would be most useful to have in a sim (not knowing that you guys have already done so much planning!). The two faculty members expressing particular interest are Alexis Templeton (specifically for her environmental geology course) and Greg Tucker (intro level course, geomorphology, and a new course he's developing called Fluid Earth). Both Greg and Alexis independently expressed the most interest in having a sim in which users could change variables (esp. temperature and pressure) and see a visual representation of density changing in a rock or other material (like a cube expanding or shrinking, in these cases).This is definitely a priority for our faculty, and a concept that they feel students would learn from a sim (and have struggled with in other contexts).
- Another issue that is difficult for students is the idea of density as a "unit area" property. Looks like this is implicitly dealt with here, but having a way to divide up an object into smaller cubes and observe behavior (or a similar scenario) would be welcomed too.
- There some more advanced goals in non-intro geology courses, some of which likely fit in well with physics goals represented here, and others with a context quite different than a swimming pool. I don't have a concise updated list of these at the moment, but as you are thinking about adding a tab specifically with a geology focus, I'll spend a little time updating this and have some suggestions as to what we think would be the most useful to build off the concepts you're illustrating in the rest of the sim. Will get back to you on this, with early next week my goal.

I am not sure if we have tossed the idea of a quantitative tab/sim or not, but this one seems sort of what I was imagining for a game. I don't like the fact that there are no units on the scale (or call it a balance since it measures mass). It certainly it's very fun either. I don't like the fact that the calculator puts in the values; I think the student should have to decide which number goes where. A fun game level might include "Given density and mass, estimate volume" , or "given density and volume estimate mass". In this level, a timer could make the game more challenging.
http://www.karlyoder.com/flash_density.html
here -pat loeblein 7/31/10 10:28 AM

Other archived discussion from the to-do document:
density notes:

Bottle size:

NP: I think the bottle will work better if

-it is drawn in 3D

JO: concerns about development cost and performance for it to be 3d.

-is dragable, so it can be moved out of the way (although I'm not sure where you would move it - everywhere you put it it is going to be in the way of something)

KP: I agree that the 2L bottle looks big ... when I make a 2L object square ... it looks smaller but I think its just that depth looks small for some reason. A small flat image of a mouse might be less precise scale, but also give you some sense?

The 2L bottle looks correct to me and JO; the diameter is approximately 10cm, so you can fit about 40 2L bottles in, then the space between the bottles accounts for the other 20L.

Also, the camera angle doesn't accentuate the depth of the pool, the actual pool dimensions are: 1m x 0.3m x 0.4 m.

NP: I thought the size of the 2L bottle was also odd, so I Photoshop'd two 1 L boxes next to the bottle. Each box is 10x10x10 cm. (See attached image.)

Appears that the bottle is basically a cylinder 10 cm in diameter and 20 cm high. That makes it's volume about 1.5 L, so in fact it is about 25% too small.

Of course, this is kind difficult to grasp because the bottle is round, and you are comparing it to cubes. Have to realize the discrepancy due to different geometries.

KL: How important is the 2L reference, since we do not allow the user to

change units? It seems like the soda bottle introduces more problems

than it solves.

TL: I understand the issue of the bottle, but something like a mouse could be more of a problem. We use a really big ladybug in some of the sims and we have the cartoon firedog in another sim, so the mouse might just seem cartoonish and not represent any sense of scale. I don't have a good solution, but it just so happened that some students were playing with the Ladybug sim today, so it came to my attention. A number 2 pencil is a pretty standard object for all students, would it be a possibility or is it too small? It just occurred to me that you might have meant a computer mouse, that is pretty standard. If that is what you were thinking, then just disregard my concern.

JO: Let's get a better 2L bottle graphic, perspective is wrong, label looks flat

Could try a blender object

Add a button or checkbox for show / hide "reference object", which would show hide the soda bottle.

SR: Maybe it's not important to try to indicate scale using a reference object in the

sim. Since wood will always float, etc, some properties of the sim and related learning goals are scale-independent.

SR: Another good way to address scale in the sim would be to make one of the default objects recognizable at a certain size. Like a rubick's cube?

_____

Can you change density when a different object is selected?

Kathy: Another question on the controls ... right now you can grab density and it automatically switches to "custom", and might be missed.

SR: Who will miss the texture changing from wood to non-wood? Also, now that the ticks are labeled, you will see that the density slider is no longer pointing at "wood".

KP: What are folks opinions about making Density slider non-grabbable unless you select custom yourself? This might address Ingrid's expectations about the objects becoming particular types when you slide the density slider - since now she would have to select custom to be able to do this. Also might have forced Ingrid to use the combo box since you could not move off of wood otherwise under this paradigm.

SR: I think this will idea just makes the user interface less convenient and less usable. I don't think making the density slider non-grabbable would have made Ingrid (or other users) more inclined to go to the drop down box. Why not be able to change the object type using the density slider? Now that it is labeled, they will see that

KL: [...] like Kathy suggests, maybe density could be a readout for the non-custom objects, so that the user must select a "custom" object to change the density.

TL: I like Kathy's idea that the density should only be changeable when Custom is selected. [...] But the color change of the object is subtle in my opinion.

SR: I don't believe it to be subtle: the first time it happens, it changes from wood to non-wood; the indication is in the texture of the object and the fact that the density silder no longer points to "wood" tick mark.

SR: What about getting rid of the drop down box, and making it so that dragging next to a tick mark snaps to the tick mark and changes to that texture?

KL: When did we decide to *label* the tick marks? The label makes it look like the object should change.

SR: We have wanted to label the tick marks since the original design of this tab; however, we only recently invented a practical way of doing this using the custom components.

SR: What about getting rid of the drop down box, and making it so that dragging next to a tick mark snaps to the tick mark and changes to that texture?

Decisions:

Allowing control over density:

The more I play with this, the more I am unsure about allowing direct control over density ... maybe we should just allow them to adjust mass and volume, and then have a slight space in the control panel, with Density just a bit below and as a readout only? I'm interested in what others think as they play with it - or we can just interview and see.

SR: I think this feature is valuable because it satisfies the learning goal "anything less dense than water will float and anything more dense will sink", and that it will help enforce the idea that density is a property of materials. But I do worry that it may give the misconception that "when you change an object's density, its volume will change too".

KP: I was suggesting that custom object allowed direct control over mass and volume - as it does now. But removing the control over density directly (which right now mimic's change in mass), and having density as a readout only. I was thinking that this might help better establish the relationship of how mass and volume effect density.

Christine also votes to be able to change density

Ingrid also votes to be able to change density

JO: nice to change density
SR: We will allow control over density based on overwhelming majority request. We will consider whether you should always be able to control density without selecting "custom" object first as a separate issue.

What to do about scale "calibration" being different when moving from land to sea? Putting the land scale underwater would cause it to read something like 5N even with no block on it.

JO: what about ignoring this problem, but making the scales immobile so we don't need to address it?

KP: Students should still be able to move the scales, the pressure differential problem would be small (based on the surface area of the scale connector part), and should be ignored.

KL: don't put a label for "water" so that students have to discover it for themselves.

Most massive Wood block causes water to leak out,

JO: I'd solve this using the same strategy we used for Styrofoam

JO: or we could slightly increase the pool size.

What are we doing with Units? Can't read out kg underwater.

SR: Always read out forces in the buoyancy sim

Tom: 3. In the "custom" version a styrofoam volume > 170 L causes the fluid to empty entirely? Other weird things happen for large volume choices (try ice at 164L)

The water sticks up out of the pool.

One solution would be to limit the size of the blocks

SR: Changing the ranges of the sliders has fixed this problem from the point of view of manipulating the sliders. However, you can still type in a large value in the text boxes to make the pool overflow. There should be max type-in values for each of the boxes, and they should be make it so the objects don't overflow the pool.

JO: Good solution would be to put a maximum volume on the cubes

JO: Or could have the fluid splash out , like in the 3rd tab of the design doc

SR: Max volume sounds easier

Could have ocean water as a choice

This makes the fluid density slider tick labels look far too crowded. We are recommending to leave this out (or could significantly increase the width of the slider, but that also has disadvantages).
