The Gumball Machine: Hidden Machinery

If you have not already done so, **Import** the **Gumball Machine** into IntelliTools Classroom Suite® and try it out. Click the coin buttons, both the one matching the price and an incorrect coin, and see what happens. Buy gumballs repeatedly. How many different colors are there? Notice that there are different comments with each color. How does the **Gumball Machine** know what coin you put into it? How does it produce gumballs of different colors?

We are about to take the **Gumball Machine** apart and find out all its secrets. When our investigation is over, you'll have acquired a bunch of new tricks for working with regions and hidden toolbars. You'll also have a chance to customize the **Gumball Machine** to make modified versions. Be sure you have downloaded and unzipped the **Gumball Machine Art** file. It has the clip art you'll need to make modifications.

This activity employs tricks with graphics, toolbars, and regions, so we'll take a look at all of those features. Follow the steps in this tutorial to discover the hidden structure of the **Gumball Machine**!

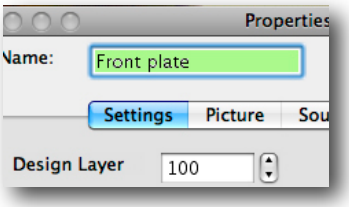
A Trick With Graphics

The **Gumball Machine** page has only two graphics on it. Its full-page **background picture** is a vintage gumball machine with lots of colored gumballs displayed in the glass globe top, a curved slot where the coin goes in, a price permanently molded into the metal front plate, and a scoop where the gum comes out.

The second graphic is hidden. It seems to be part of the background, but functions as **foreground scenery**. To examine it, choose **Select Object** under the **Edit menu** and open the pull down the menu to its right. Several regions are listed that we'll look at later, and one object called **front plate.** Choose **front plate** and let the **Edit** menu close.

Now you can see the selection outline of **front plate**. It covers the area from the curved bottom of the coin slot down to just above the bottom scoop. There is a small notch cut out of the bottom in the center to let the top of the gumball show. I made **front plate** by selecting and copying part of the background image. It fits perfectly over the same part of the background, and the colors match perfectly, too.

Without losing this selection, open **Edit** again, and choose **Properties...** In the **Properties dialog**, notice that **front plate** is on **Design Layer 100,** the highest or front layer. Any other items added to this page slide behind **front plate**. It's set so it **does not respond to mouse clicks,** is **not included in a scan,** and is **locked** in place.

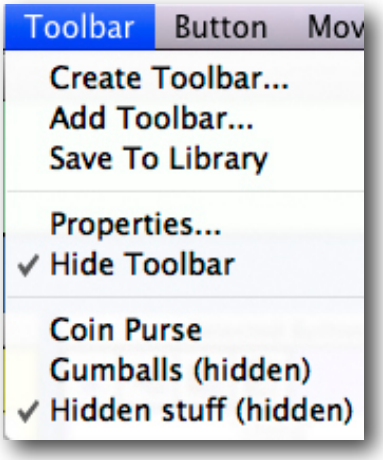


Users don't suspect that **front plate** is there, so coins appear to really fall into the slot. With the help of realistic sound effects played at

key moments, interacting with the **Gumball Machine** feels very much like a real world experience!

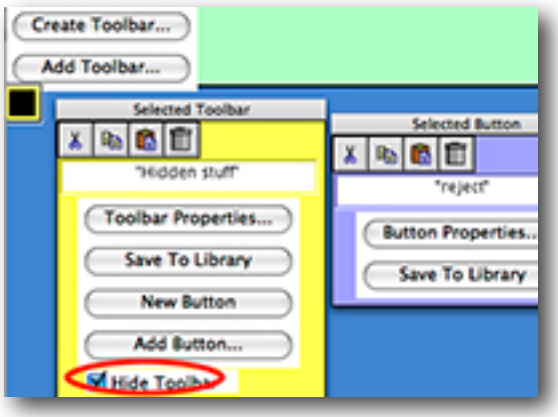
Hidden And Visible Toolbars

In analyzing any activity, you definitely want to take a look at all the toolbars in it because much of the action will be controlled through buttons. Like many activities, **Gumball Machine** has hidden as well as visible toolbars. A quick way to locate all toolbars is to choose **Custom Toolbars and Buttons** from the **Edit** menu, so do that now.

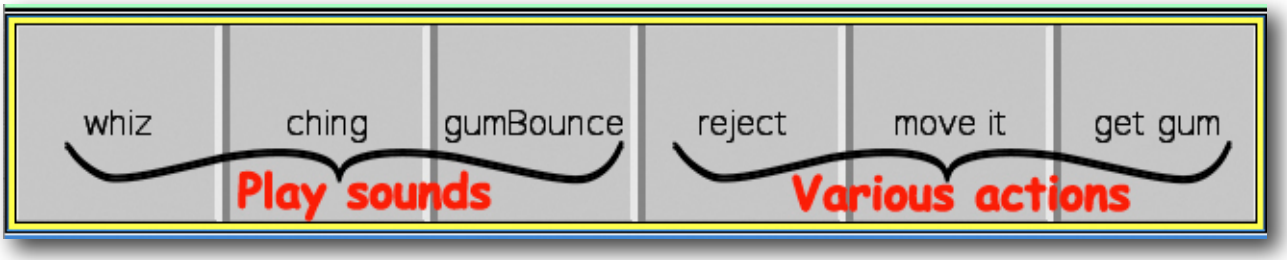


Notice that the menu bar at the top of the page changes once you are in toolbar editing mode. The second word from the left is now **Toolbar**. Click on **Toolbar** and a new menu drops down with various editing functions for toolbars and, what we need right now, a bottom section that lists all the toolbars in the activity. The list even tells you which ones are hidden! **Gumball Machine** works with just three toolbars, and two of them are hidden.

Let's take a quick tour of the three toolbars, checking out the actions of the buttons in each one. To explore a hidden toolbar, select it (it will be a tiny square) and remove the checkmark in the **Hide Toolbar** box on the yellow **Selected Toolbar** palette.



Control-click each button to quickly open its **Properties** and see what actions and settings are there. Alternately, you could select each button and open its **Properties** from the blue **Selected Button** palette.



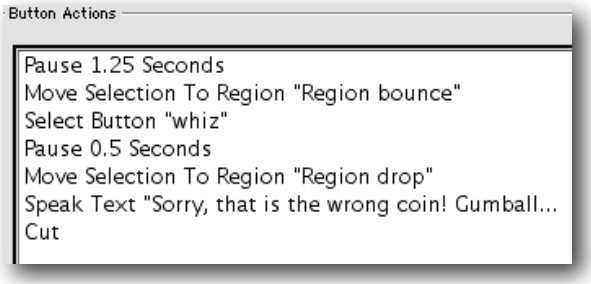
1. **Hidden stuff** (hidden) Select and unhide **Hidden Stuff** first. Although the user never suspects it exists, this toolbar includes three buttons for sounds and three buttons that provide much of the action in the **Gumball Machine**. Let's take a close look at both the sound and action buttons. Besides finding out how **Gumball Machine** works, we can learn several good strategies for building our own new activities.

Sound Buttons: Open the **Properties** of one of the first three buttons on the **Hidden stuff** toolbar, either **whiz**, **ching**, or **gumBounce**. All three are set to play sounds when selected without waiting for other actions to finish. The only difference between these three buttons is the particular sound assigned to each.

> **Quick Tip:** When you need a sound to play as the result of a user's action rather than automatically, it should be in a button, and the button should be on a hidden toolbar, unless the user needs to see the button to click it. I usually group all the hidden sound buttons on a single hidden toolbar so it's easy to find them later, if I need to edit them.

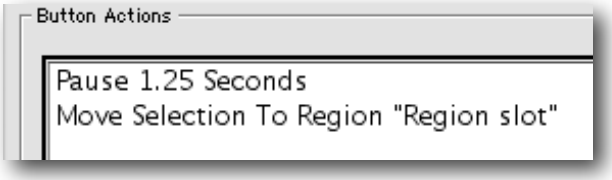
Action Buttons: Each of the remaining three buttons on **Hidden stuff** has a unique set of actions, so we'll open the **Properties** of all three.

The first action button, **reject**, is used if a wrong denomination coin is inserted. This button moves the incorrect coin twice, to two positions (**Region bounce**, then **Region drop**),

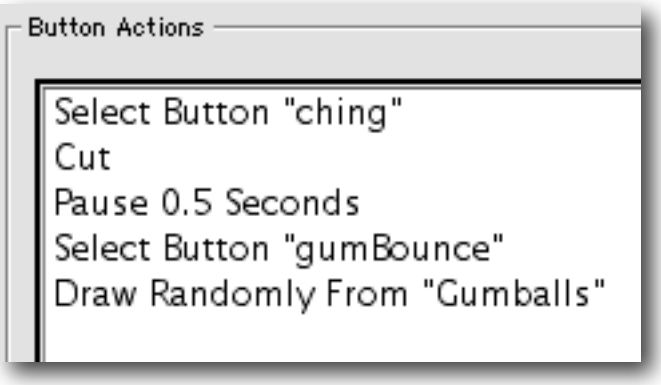


selects a sound button **(whiz)**, speaks a "Sorry...wrong" feedback sentence, and then deletes the coin (**Cut** action).

The next button, **move it,** is selected when a correct coin is inserted. Using **region slot** as a target, this button moves a correct coin down into the slot, so that the coin disappears behind the **front plate** graphic. From there, **region slot** takes control. and activates the sequence to choose a gumball.



The third action button, **get gum**, controls the main functions of the **Gumball Machine**. It selects the sound button **ching** (which plays the dropping coin and ching sounds), deletes the coin from behind the front plate, selects the sound button **gumBounce** (the rattle of the gumball dropping down), and randomly selects a button from the **Gumballs** toolbar.

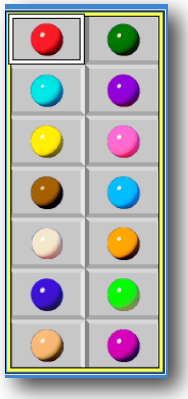


Quick Tip: Hidden buttons like **get gum** can select other hidden buttons, either on the same toolbar (**Hidden stuff**), on a different hidden toolbar (**Gumballs**), or both! A hidden button also can select a button on a visible toolbar, go to another page, load a picture, save the activity, or all four actions—the possibilities are endless!

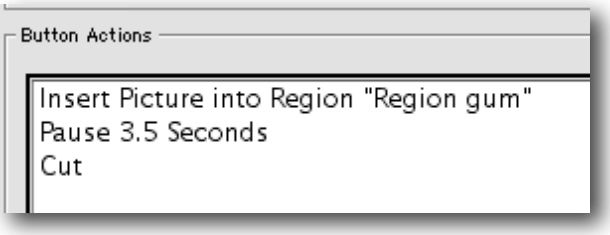
I used the **Draw Randomly** action rather than **Pick Randomly** so that all fourteen colors of gumballs are used before the cycle repeats. Once one of the gumball buttons is selected, the actions in it finish the process of buying the gum.

Quick Tip: If you've built or modified activities, probably you've used the ability to hide toolbars as a means of putting away tools until they are needed. The Gumball Machine presents another option: Use hidden machinery to do all kinds of tasks behind the scenes! There are many advantages to having sets of toolbars and buttons located in a hidden utility room. The user isn't bothered by a lot of unneeded clutter, yet the hidden options can be utilized from any page of an activity.

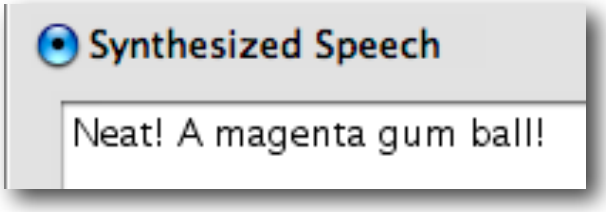
2. **Gumballs** (hidden) Below **Hidden Stuff**, along the left edge of the screen, select another of those little squares. It's the second hidden toolbar, **Gumballs**. Make it visible by removing the checkmark on **Hide Toolbar**.



The 14 buttons each have a different color gumball picture but identical actions, so just open the **Properties** of any one of them. Each button inserts the gumball picture to **Region gum** (in the scoop), pauses, and deletes the gumball picture.



Under the **Sound** tab, each button of **Gumballs** has **Synthesized Speech** marked and a feedback sentence about the color of the gumball. The sound plays automatically when the gumball picture is inserted. No **Play Sound** action is needed.

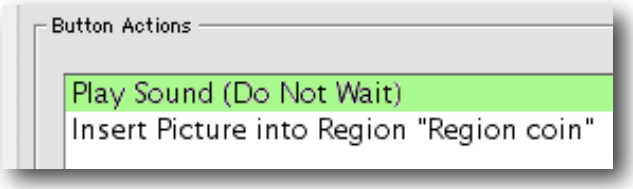


The **Gumballs** buttons complete the action sequence begun by clicking a coin button matching the current price of gum. It's a complex set of actions and involves many different objects in a sort of digital Rube

Goldberg machine, but the entire ballet plays behind the scenes. For the user, the **Gumball Machine** is a simple cause and effect experience: Insert coin, get gum!

Quick Tip: Often you'll find it convenient to hide a toolbar with a set of similar items from which to pick randomly. If these buttons all have mostly the same actions, scan settings, attributes, etc, save time by making the first button, duplicating it however many times you need, and then making the few changes needed in each. Just be sure to make a checklist of what needs to be unique about each button!

3. **Coin Purse (visible)** This is the only toolbar a user sees and interacts with. It has five different coin buttons. The actions of all the coin buttons are the same: Insert the coin picture to **Region coin** (above the coin slot) and play a click sound of coin hitting metal. Notice the **Singular Name** attribute for each button. That's what **Region coin** checks to determine if it's the correct coin.



Quick Tip: I took more thought about the appearance and placement of **Coin Purse** than with the other two toolbars, since the user will see it. For hidden toolbars I use the default **Appearance** settings, a real timesaver. Another advantage of using hidden toolbars where possible!

Gumball Machine has a very simple interface, although it certainly has a complex set of actions. By planning ahead and using a flow chart, I was able to get by with this one visible toolbar and nothing scannable on the page, yet it is a very rich virtual environment.

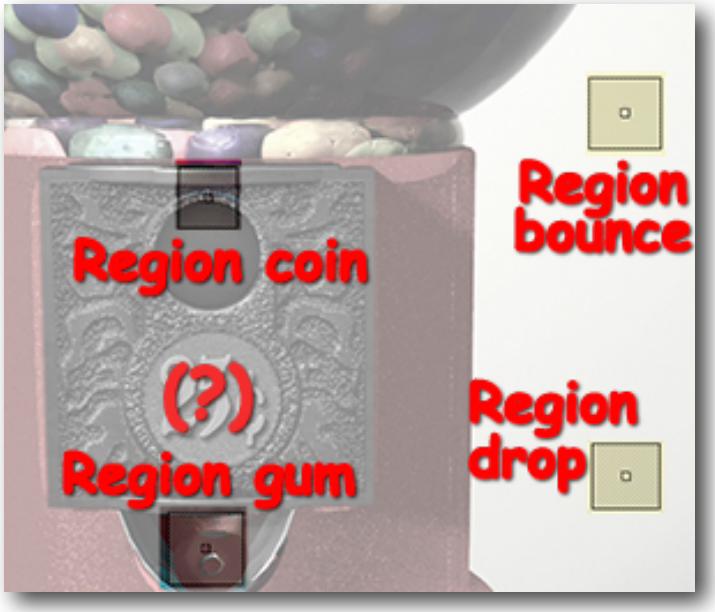
We will look further into how all these buttons and toolbars interact to bring the **Gumball Machine** to life. For now, hide **Hidden Stuff** and **Gumballs** by selecting each toolbar and restoring the checkmark beside **Hide Toolbar** on the yellow **Selected Toolbar** palette. Then click the **Done** button in the upper right corner to leave toolbar editing mode.

The Hidden Power Of Regions

Many of the button actions we just explored refer to regions, so let's check out the regions on the page next. Regions can have several different functions. They can simply be targets to which items are moved or inserted. They can be points where items are sorted into correct and incorrect categories.

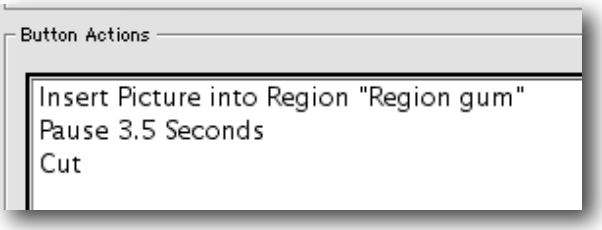
Most important, they can select buttons when they accept or reject items, which means that they can trigger any set of actions you can add to a button. In the **Gumball Machine**, regions are used in all three ways, sometimes all at once.

Regions are one of the most powerful tools in ICS, but usually they are invisible on the page. To see the regions on the **Gumball Machine** page, choose **Show Regions** from the **IntelliPics Studio** menu. Now the regions should appear as cross-hatched rectangles.



Regions Used As Targets

Two regions to the right of the gumball machine serve as simple targets, **Region bounce** (upper) and **Region drop** (lower). Remember the **reject** button on the **Hidden Stuff** toolbar? If an incorrect coin is inserted, **reject** uses the **Move Selection To Region** action twice: to move the coin up to **Region bounce** and quickly down to **Region drop**.



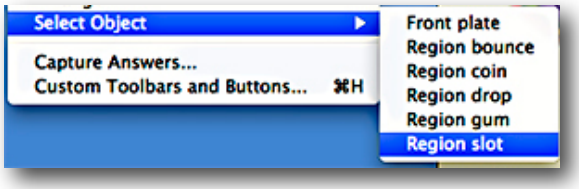
Next, examine the region atop the metal scoop where the gumball appears. That is **Region gum**, and it is the target of all the buttons on the hidden toolbar **Gumballs**. The gumball buttons insert their pictures directly to **Region gum** as they speak a feedback sentence. Just think, an invisible button selected remotely is putting a picture in place with the help of a hidden target. Pretty sneaky!

All three of these target regions are set to **accept all objects**. They don't need to do any sorting, because regions further up the line have already taken care of that task.

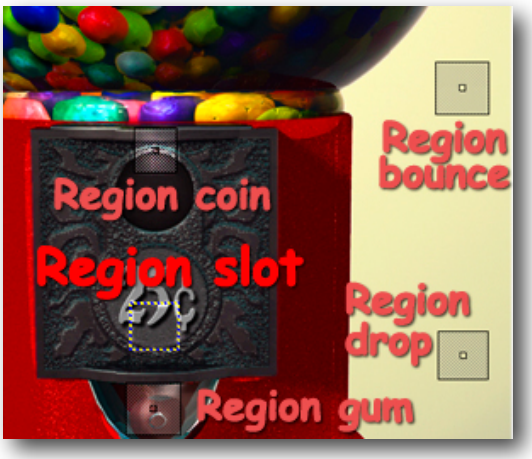
Hidden Target: Region slot

One region is still hidden from us, even with **Show Regions** active. That is **Region slot**, located behind the price sign on the **front plate** graphic. All regions are sitting on the default **Design Layer 0**, while we saw that **front plate** is in front of everything else on **Design Layer 100**. That's why **front plate** is obscuring our view of **Region slot**.

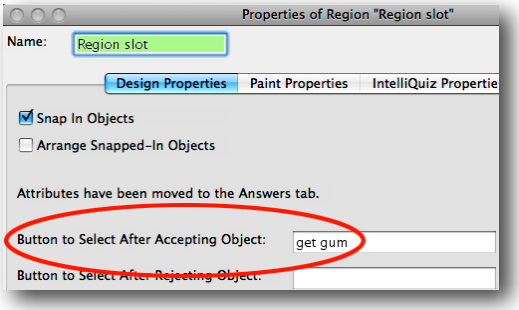
To select any item that is behind something else, we can choose it from the pull-down menu to the right of **Edit—>Select Object**. Choose **Region slot** from the list, and you will see the selection outline of **Region slot**. Go back up to **Edit** and choose



Properties... The **Properties** dialog opens, and you can now see all the settings for **Region slot**, even though it's hiding behind that graphic.



Notice that, like the other three regions used as location targets, **Region slot** accepts all objects. It doesn't have to do any sorting. By the time a coin is moved to **Region slot**, it has already been checked and found to be the correct value.



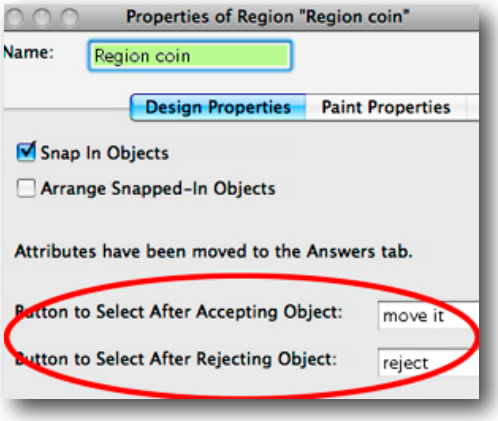
However, **Region slot** not only functions as a location target but also activates a button, ***get gum***, when it accepts an object. We've already seen that it's the hidden button ***get gum*** that actually chooses the gumball by drawing randomly from the **Gumballs** toolbar and then inserts it to the scoop of the **Gumball Machine**, using **Region gum** as the location target.

One Region Used For Sorting And Actions

The most critical region, **Region coin**, is located at the top of the front plate just above the coin slot. It is the target of all the coins in the **Coin Purse** toolbar.

When a coin button is selected, the clink-on-metal sound plays and the coin picture is inserted directly into **Region coin** at the same time. **Region coin** screens each inserted item by checking its **Singular Name** attribute. If the **Singular Name** of

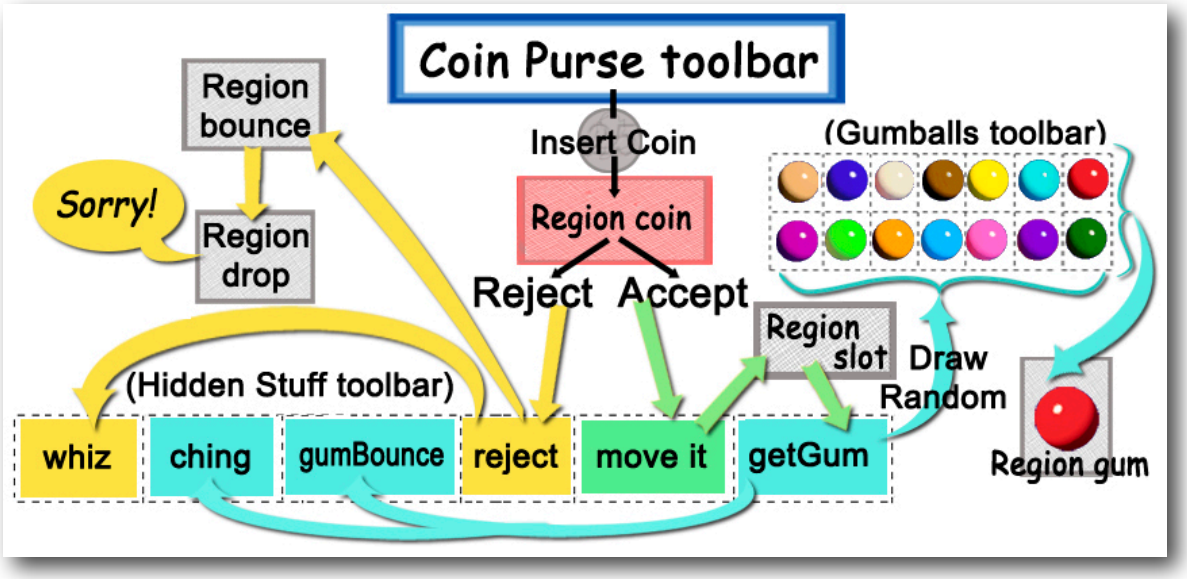
the inserted coin matches the **Singular Name** listed as correct in **Region coin**, that coin is accepted; otherwise, it's rejected.



Region coin has a designated button to select if the object is accepted (**move it**), and a different button to select if the object is rejected (**reject**). We have seen what the **reject** button does: It throws the coin off to the right and gives feedback before deleting the coin. The button selected when an object is accepted, **move it**, moves the accepted coin down into **Region slot** behind the **front plate** graphic.

A Powerful Option: A Branch Point

We have explored all the pathways in the hidden machinery of Gumball Machine. Now let's look at a flow chart of the entire activity.



You can see that **Region coin** is the main branch point for actions in this activity. What makes it possible to have such a branch point is that regions have the **special option** to designate separate buttons that will be selected when accepting and/or rejecting items.

In this activity, the branch point filters out incorrect coins and gives feedback, while the correct coin causes a gumball to be released, but with the added feature that it may be any of fourteen colors. You could set up a store in this way, with every item having a reject button for the wrong price.

Quick Tip: Designating buttons for accept and reject is a very powerful option, since either a correct object or an incorrect object can trigger any set of actions you can add to a toolbar button, including selecting even more buttons. Even better, the triggered buttons can be hidden, so that the resulting cascade of effects seems to occur naturally because of the user's manipulation of the object.

This branching option can be used to create many types of activities. You could use such a branch in an adventure story, where offering the correct item opens a door, or a secret passage or maybe a treasure chest, while an incorrect item is rejected, sends the user into a trap, or even ends the game.

In a test, you could have an item as the correct answer set off a victory song, then move on to the next question, while incorrect item takes the user through some review or gives a clue, then returns to the same page so the user gets another try. Without cluttering up the interface, being able to branch off depending on user interaction gives you a way to make much more interesting and effective activities.

This concludes our investigation of the **Gumball Machine**. Before we turn the activity over to students to play with, let's reflect on all the lessons we have learned.

What Have We Learned?

- Consider using foreground scenery to create a more realistic interface. A simple example is copying part of a background picture, pasting it in **Design** mode as a floating graphic, lining it up with the background, and locking it on **Design Layer 100**. Any inserted item will go behind the foreground graphic by default. To really blend your foreground scenery so it won't be detected, remove it from scans and set it not to respond to mouse clicks.

- Add ambient and logical sound effects to create a realistic soundscape. Ambient sounds most likely will be page sounds, but also consider adding sounds to buttons that can be played, possibly remotely (**Select Button** action), to add realism.

- Get into the habit of grouping hidden sound and action buttons on a single hidden toolbar, to make it easy for you to edit your activity. It also helps you find the machinery when you haven't looked at the activity for a while :) It doesn't matter about appearance and number of buttons on such a behind-the-scenes toolbar.

- Use hidden sound buttons if you need more than one sound on a page (perhaps a **Page Sound** plus a **Select Button** "my hidden sound" action) and for sounds that occur as a result of user actions (Example: clink when coin is inserted, rattle and ching sound when it drops into the **Gumball Machine**, rattle when the gumball is released).

- A hidden toolbar with a set of similar objects, including numbers, can be used with the **Draw Randomly From "toolbar name"** action for all kinds of activities. The **Draw Randomly From** command could be in a button activated by a region. Suggestions: Use to simulate rolling dice, to vary feedback sentences, to pick random colors of spaces in a game, to tell fortunes, to choose riddles.

- Buttons on hidden toolbars can select and thus activate other hidden buttons on the same or on other toolbars. They also can select and activate visible buttons, and do anything you could control from user-clicked buttons or from **Page Action**s. Using hidden machinery enables you to design far more complex activities and still retain an uncluttered, simple interface.

- Besides serving as a way to assess right and wrong answers and report a test score, regions can be used as invisible targets and as a point where the actions and effects in an activity branch. As targets, regions enable the insertion of items to a specific point on the page (via **Insert Into Region**) and, with the help of the **Move Selection To Region** action, to move those items along a simple path. As a branch point, regions can select designated buttons upon accepting or rejecting an item, triggering any set of actions you can put into a button.

- Very sophisticated activities are made possible by combining the use of regions as targets and branch points with hidden toolbars that serve as the main machinery of an activity. Using these two kinds of hidden features also keeps the user interface clean and uncluttered, and often contributes to the realism of the user experience.

Insert Picture...? or Move Selection...?

Use the **Insert Picture Into Region** action instead of the **Move Selection To Region** action if you intend the region to sort items into accepted or rejected status by attribute.

In my experience, if an item is sent to a region by the **Move Selection to Region** action, the region fails to activate the button it should select upon rejecting the item. Moving an item into the same region with the mouse, using scanning directional arrows, or with the **Insert**

Picture Into Region action all successfully activate the designated button upon rejection.

There may be a bug in **Move Selection**, but it only is a problem when selecting a button upon rejecting an item. For all other purposes, such as moving an item to a region that accepts all and selects a button upon accepting an item or to a position via a target region, **Move Selection To Region** works very well.

Gumball Machine Crashes

I built the **Gumball Machine** in ICS v. 3.0.7, and it never crashed. However, I went back and tried it on v. 3.0 and found that there are numerous problems with using the two regions actions, and the finished activity crashes. I've tried to find work arounds to no avail. I've decided to post the **Gumball Machine** anyway, figuring that almost nobody would be using such an early version ICS.

Optional: Modifications To The Gumball Machine

We've learned quite a bit by analyzing the **Gumball Machine,** and it's a fun activity to use with children to practice recognizing coins and colors, but there is more we can do with it. The following are instructions for modifying the **Gumball Machine** so that you have a whole family of related activities. At the same time, you'll be reinforcing all the new tricks you've learned.

This is where you will need the clip art from the **Gumball Machine Art** folder, so be sure you have downloaded and decompressed it.

Variation 1: Changing The Price Of The Gum

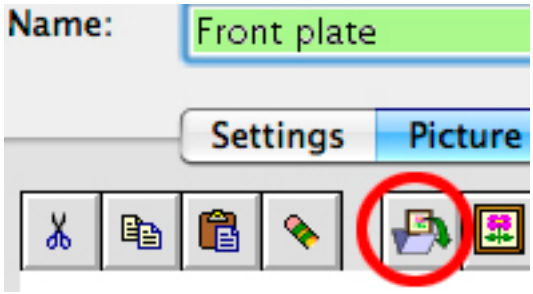
To make certain you understand how regions are used to make the **Gumball Machine** work, try changing it up to accept a different coin. With only 3 changes, you can sell gumballs at a different price.

Changing The Price Sign

First, you need to change the price sign. Not only does the **front plate** graphic add to the realism of the experience by serving as hidden foreground scenery, it also makes changing the price sign very easy.

Just as you did when exploring the settings of front plate, go under **Edit**, choose **Select Object** and open the **drop-down menu**, and choose **front plate** from the list of items on the page. Let **Edit** close and, without losing the selection, open **Edit** again and choose **Properties...** The **Picture Properties** dialog for **front plate** will open again.

This time click the **Picture** tab and locate, just under it, the **folder icon** for loading a picture from a file. Click it, and navigate to the **Gumball Machine Art folder**. In it you will see a collection of **front plate** graphics, with prices



ranging from one cent to two dollars. Choose the one with your new price, click **Open**, click **OK** to close the **Properties** dialog, and your new price should be in place on the front of the **Gumball Machine**.

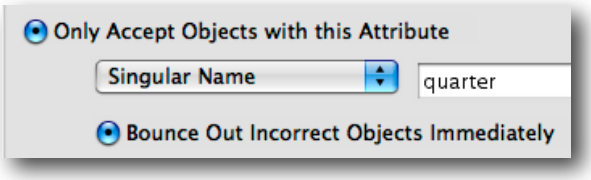
> **Quick Tip:** You've just seen the advantages of using a floating graphic as an image placeholder. Since you **loaded a new image into the graphic instead of loading a new graphic into the page**, it's already exactly in its place with all the settings intact.

Changing the Accepted Coin

Next, you must change the **Singular Name** that **Region coin** accepts as correct. Regions are VERY picky about **Attributes**, so it's a good idea to open the **Properties** of the coin in the **Coin Purse** toolbar that matches your new price and check its **Singular Name** under **Attributes**, for wording, spelling, and capitalization. I'm paranoid, so I just copy it!

Then under the **IntelliPics Studio** menu choose **Show Regions**. Regions, except for the one hiding behind **front plate**, will show as **cross-hatched rectangles**. **Control-click** the region you see just above the coin slot (**Region coin**) to open its **Properties**.

It should have two buttons listed, **move it** for **Accepting Objects** and **reject** for **Rejecting Objects**. Click the **Answers** tab at the top, and to the right of **Singular Name** type or paste the name of the coin that will be your new price.

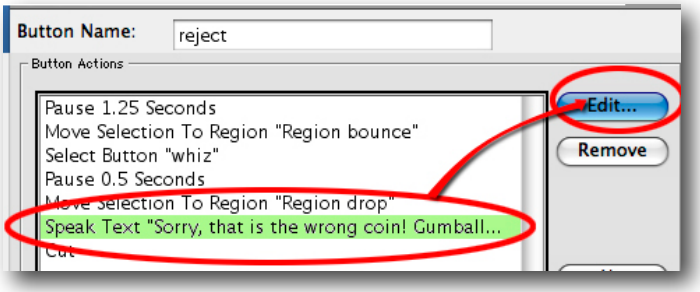


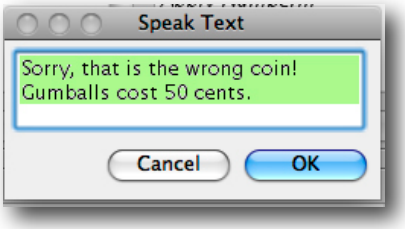
Click **OK** to close the dialog. Since **Region coin** is the branch point for filtering correct and incorrect responses, that's all you need to do to make the **Gumball Machine** accept a different coin!

Changing the Feedback

While the **Gumball Machine** will now sell gum for the new price, and reject other coins, there is one change left to make. When the **Gumball Machine** rejects a coin, it gives the user feedback that includes the current correct price. You need to edit the feedback so it speaks the new price, but the **Speak Text** action is located on a hidden button, **reject**. We'll have to make that button visible to change it.

Under **Edit**, choose **Custom Toolbars And Buttons...** Check that the yellow palette says **Hidden Stuff**, and remove the checkmark in the lower left to make that toolbar visible. Choose the **reject** button, and on the blue floating palette, click **Button Properties**. Under **Actions**, search down the list for **Speak Text** and click the **Edit** button beside it.





A small dialog will open in which you can change the sentence so that it states the new price. Click **OK** twice to close the little dialog and the **reject Button Properties**. Click the checkbox in the lower left of the yellow palette to **hide** the **Hidden Stuff toolbar** again, and click **Done** in the upper right of the screen to close toolbar editing mode.

Test And Save The Modified Activity

Test your modified **Gumball Machine** and see if it will sell you gum at the new price, and that it rejects all other coins. When you've run these two tests, save your modified activity under a new name.

Notice how that branch point simplifies editing. If you only changed which **Singular Name** is accepted in **Region coin**, all would work

correctly! The other two changes simply affect the appearance and feedback comments. It pays to rough out a flow chart so that you can design your activities in as modular a fashion as possible.

Variation 2: Changing The Contents Of The Coin Purse

Deleting Coins

I set up the **Gumball Machine** with five U. S. coins in the **Coin Purse** toolbar, but you can easily modify it to have more, fewer, or different coins. If five coins is too many, simply go to **Edit->Custom Toolbars And Buttons...** and delete one or more buttons from the **Coin Purse** toolbar, being careful to retain the one that will be accepted as the price of a gumball. As always, save the modified activity under a different name so that you keep both versions.

Changing Coins

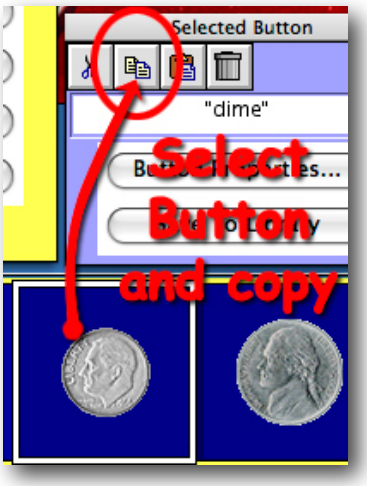
You can easily substitute Canadian coins for U. S. coins in the visible **Coin Purse** toolbar. **Control-click** each button to open its **Properties**. Under the **Picture** tab, replace the original image of the U. S. coin with a clip art image of a Canadian coin of similar value from the collection of clip art coins included in the **Gumball Machine Art** folder.

Probably you should change the button names to the more usual Canadian names at the same time, because students using scanning will hear these button names read aloud.

It's not necessary to change the **Singular Names** under **Attributes** in each button, unless you just feel better if everything matches. If you make that change, be certain that the **Singular Name** in **Region coin** matches whatever **Singular** Name you have used in the coin equal to the current price.

Adding More Coins

Perhaps you want to use one or more coins with values not initially included in the **Coin Purse** toolbar, such as the Canadian $1 (loonie),



Canadian $2 (toonie), or U. S. $1 (Sacagawea) coins. For that you should go to toolbar editing mode by choosing **Custom Toolbars And Buttons** from the **Edit** menu.

In editing mode, select any button on the **Coin Purse** toolbar, and use the icons on the blue floating palette to **copy** and **paste** it, thus duplicating the button. The duplicate will appear at the right end of the **Coin Purse** toolbar. Select the duplicate, and open its **Properties** by clicking **Button Properties** on the blue **Selected Button** palette.

In the **Properties** dialog of the duplicate button, **change the name** to that of the coin you want to add, and **load its image** under the **Picture** tab. Under the **Attributes** tab, check that the **Singular Name** attribute is correct for this new coin button. Click **OK** to close the **Button Properties** dialog. If you need to change the position of the new button in the toolbar, hold down **Shift** and slide the new button into its place. Click **Done** in the upper right to leave toolbar editing mode.

Notice that you did not have to change actions or other settings to add a new coin button. All the buttons in the **Coin Purse** toolbar have the same actions and general appearance. They only differ in the button and **Singular** names, and the picture that will be inserted. By starting with a duplicate instead of making a new button, you saved yourself

some time and also took advantage of the fact that the actions had already been tested.

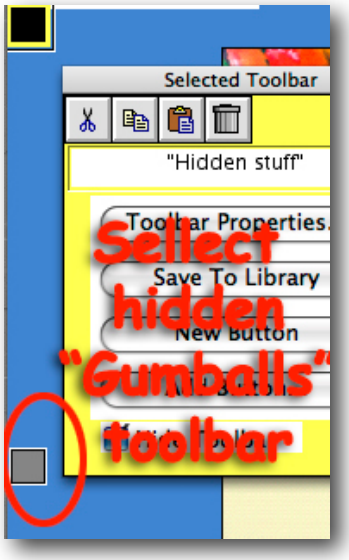
Add And Subtract Coins

You might decide to have just four coins, but want one of those to be one of the extra coins. In that case you could delete one button, then decide which of the remaining four you don't want, and convert that one into the new $1 or $2 coin. Just follow the usual steps for adding a new coin, but don't make a duplicate. Start by opening **Properties** and changing the name and picture.

Other Modifications

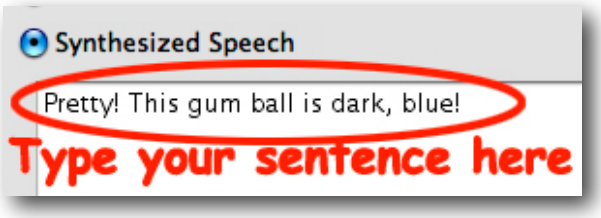
Fewer Gumballs And Colors

Besides changing the number of coins, you also can have fewer gumballs if you prefer. The **Gumballs** toolbar is hidden, so start by choosing **Edit—>Custom Toolbars And Buttons**. Select the small square about halfway down on the left edge, or choose **Gumballs** from the **Toolbar** menu. Remove the checkmark on the yellow floating palette for **Hide Toolbar** to make the **Gumballs** toolbar visible, delete one or more buttons, hide **Gumballs** again, and click **Done**. Since the **get gum** button **Draws Randomly** from the **Gumballs toolbar,** you can delete buttons from it without affecting anything else.



Change The Colors Feedback

You might want to change the feedback sentence in one or more of the gumball buttons. To do this, select any button of **Gumballs** and open its **Properties**. Under the **Sound** tab, notice that **Synthesized Speech** is active and under it is a space enclosing a feedback sentence. Delete the current sentence and



type in your own feedback sentence to replace it. Click **OK** to close the button's **Properties**. Be sure to hide the **Gumballs** toolbar again before leaving toolbar editing mode, and save your modified activity under a new name.

Gumball Of A Different Color?

It would be possible but tricky to add or substitute gumballs of other colors. You could load one of the gumball clip art files into a graphics program such as **PhotoShop®**, select just the gumball, use an editing function to change its color, then save the new gumball art as **.gif** with **transparency** and **dithering**. That's what I did to make the background of the gumballs disappear in ICS. I'm not sure what would happen in other graphics programs.

If you were successful in producing a graphic that loaded into ICS with a nice clean edge, you could substitute it for the default image in one of the gumballs. You would also need to change the **Speak Text** action to give feedback about the new color, and change the **button name**. To keep all the original gumballs and add a new one, duplicate a button, then load your new graphic, change the feedback, and change the button name.

Build New Activities

The tricks used in the **Gumball Machine** aren't restricted to this particular activity. Looked at as a pattern, the **Gumball Machine** is an example where different outcomes occur depending on which object a student chooses and conveys to a particular target. Instead of a coin into a slot, the same machinery might handle a story where at this point the user has to give someone a particular clothing item to help them get dressed! Take a moment to think of some other types of activities where you could utilize a branch point.

The other distinctive thing about the **Gumball Machine** is that outcomes are further varied by having any of fourteen color gumballs

appear in response to a correct coin, and for each color there is a different comment. That's more like real life, and lots less boring. Here's a challenge: Can you also vary the feedback for an incorrect coin? The easiest way involves setting up another hidden toolbar with a buttons to contain the set of feedback sentences, and, in the reject button, substitute a "**Draw Randomly From (feedback choices toolbar)**" action for the **Speak Text** action.

The option to vary the outcomes enables all kinds of activities. Take a few minutes to imagine some games using this trick, or maybe have the prize for some work vary each time a student works through a problem. A nice touch to keep kids engaged!

I hope you and your students have as much fun using and learning from the **Gumball Machine** as I did in building it, and that it leads to many new activities!

Quick Links

Credits And Sources

Activity concept, structure, and artwork are original by Ann Brundige.

Photo Credits

Gumball Machine And Gumballs

The background picture for the **Gumball Machine** and the gumball images were built in Vue Complete® and edited in PhotoShop®. The following photos were used for reference and for the gumballs texture:

2552026699_b831868956_o.jpg by Julie Alexander (2008) (CC BY 2.0) Vintage gumball machine via Flickr.com Used for reference

http://www.publicdomainpictures.net/view-image.php?image=106316&picture=gumballs-assorted-colors Gumballs all jumbled together randomly by Julie Gentry Free Download (Public Domain)

About the coins clip art: I referred to numerous photos of both U. S. and Canadian coins, then did extensive editing in PhotoShop®, in some cases adding grunge layers and slight color fills. The resulting clip art is purposely low resolution (in some cases 20X reduction from the photos!) to avoid copyright and potential counterfeiting issues and retains just enough detail to be recognizable in ICS.

Coin Reference Photos

50cent_designs.jpg by G. Edward Johnson (2010) (CC BY 3.0) Different designs of the 50 cent piece. From left to right, the 1976

Bicentennial coin with Independence Hall on the back, the John F. Kennedy coin with the presidential seal on the back, Ben Franklin with the liberty bell on the back, and walking liberty with an eagle on the back. via Wikimedia Commons

6056727301_b946d0bc91_b.jpg by fdecomite (2011) (CC BY 2.0) Canadian Coins 7 around 1 Quite Best Fit via Flickr.com

6056720935_2d06b115ab_b.jpg by fdecomite (2011) (CC BY 2.0) Canadian Coins 6 around 1 Best fit via Flickr.com

Sound Credits

coinClick: button_toggle_3.wav from ICS Media Library

Ching sound of coin dropping and registering in the machine: 75235_creek23_cha-Ching from freesound.org

Whiz sound of rejected coin: fweettap-tap.mp3 adapted from Magic2.wav plus part of putdown2.mp3 from ICS Media Library and beachware.com respectively.

gumBounce sound of gumball dropping: adapted from CLIPCLIP.mp3 from beachware.com