Cocoa™ Project Pages

Resources

Introduction

This site is about Apple's Cocoa™ DR3 software, the ways we used Cocoa in the Islander Middle School Multimedia Club, and some resources to help you get started using Cocoa, or to help you consider the educational components of Cocoa. This Cocoa is different from the application environment for Mac OS X. Cocoa is an example of "Programming by Demonstration." This essay, written by Henry Lieberman at MIT, explains the concept of "programming by demonstration," as well as providing a look at Cocoa and other applications that support programming by demonstration.

Cocoa is a unique and fun program that students ages eight and above can use to create multimedia simulations, games, or puzzles, which can then be configured to work over the internet as imbedded plugins on HTML pages or as AutoPlayers, which behave like standard Macintosh applications. To quote the Cocoa FAQ:

> Cocoa is Internet Authoring for Kids. Cocoa allows children (and everyone else too) to create interactive web pages, simulations, and video games with animation, sound, and interactive control. Cocoa achieves this using Programming by Demonstration, where instead of writing arcane code, you just put the system into record mode and show it what should happen in a given situation. When this situation is present again, the system replays what you showed it. Cocoa worlds can be delivered as Web pages using a Netscape Plugin or as standalone AutoPlayers.

The Cocoa authoring environment consists of three pieces:

> Cocoa Authoring Tool, used to build and edit Cocoa Worlds

Cocoa AutoPlayer Engine, used to deliver standalone, double-clickable Cocoa World Applications usable with practically any Macintosh.

Cocoa Plugin, used to allow Cocoa Worlds to be viewed inside web pages.

Apple stopped developing Cocoa after DR3 (Developer's Release 3). The technology was sold to Stagecast, which developed its Creator program from Cocoa. For the most part Cocoa is complete; sometimes it will drop out on you with error type 2, but that can usually be solved by allocating more memory to the application. The software is easy to master while still providing hours of challenges for the programmer.

Another important Cocoa site on the Internet is the Project Cocoa pages. Here you can find the Cocoa installers for both 68K Macs and PPCs. There are also a few sample files to get you started, as well as links to a couple of sites using Cocoa.

Note that Cocoa is a Classic Macintosh application, and the plugin will work only with browsers running in the Classic Environment. However, I have run Cocoa extensively in Classic and have built AutoPlayers and ran them as well. It is a little disconcerting having the monitor turn blue as it steps down to 256 colors, but Cocoa works fine in OS X 10.2.6 with QuickTime 6.2. If only there were a way to get the plugin to work with a modern browser....

Resources

Programming Resources

Programming with Cocoa is easy once you get the knack of it. With an intuitive graphic user interface, a flexible graphics editor, and the ability to serve up your projects on the internet for the world to see, or to build an application usable on most any Macintosh, Cocoa is a hit with students. These resources will help you get started with Cocoa:

First, watch the Cocoa Tutorial Movies. These four QuickTime movies will lead you through the basics of Cocoa and whet your appetite for the fun ahead. Additionally, there is a QuickStart Movie that combines the lessons of all four movies into one larger movie.

Next, download the Cocoa Basics Tutorial, the Cocoa Advanced Tutorial, and the Cocoa Reference PDFs, available from the Project Cocoa Page.

While many students enjoy creating their own Types, or characters, for the Basics Tutorial, others might want the art provided for them. Stop by the Cocoa Basics Tutorial Parts Page and grab the graphics for yourself!

Cocoaban is an effort at collaborative World Building. The World is available for download (as the Cocoaban authoring kit). Visit the Cocoaban Challenge page and learn to customize this World with your own levels. You can send in your levels and have them posted on the Cocoaban page!

Gauges and Counters keep track of different parts of a simulation, keep score in your game, or even determine when a specific Type "expires" and is removed from your World. This informative article, "Adding Counters and Gauges to a Simulation," explains different uses for counters and gauges in a simulation World, and explains how to create the Rules that will make your counters and gauges work.

"Click Responses" add complexity to your Worlds, allowing users to provide input into your Worlds. DoubleClick, a World put together by Cameron Hayne, illustrates how double-clicking works in a Cocoa World, and shows how Cocoa can be programmed to look for specific types of input: in this case, a slow, medium, or fast double-clicking pattern. I have taken apart Cameron Hayne's DoubleClick World to show Cocoa programmers how he used Appearances and Variables to do the "impossible" in Cocoa: wait for and acknowledge a user's double-click. This Double-Clicking Tutorial will lead you through the code and explain how this World was created.

Sounds can make your Cocoa Worlds more entertaining, or help to clarify an issue in a simulation World by offering feedback to your user. Sounds are recorded into Cocoa using the standard Record dialog box, like that in SimpleText. This World, Sounds, from the Project Cocoa page, shows how a "Click Response" Rule can be used to play the sounds you create for your Cocoa World.

Creating complex Worlds or Worlds with many different Boards can oftentimes create problems when you build an AutoPlayer version of your World. When you double-click on your World it might not open at all, or the World might crash once it is open. Take a look at how to solve these problems that Cocoa has with managing memory usage.

Cocoa and Education Resources

As a former Islander Middle School Multimedia Club advisor, I first started using Cocoa because somebody had left an old copy on the Technology Specialist server. However, I quickly realized the appeal that this program would have to some of my students who were eager to create something on the computer but lacked the programming skills to make a conventional application. Besides, students seem to love anything that can be accessed through a web browser.

Typically, my sixth grade students got the most out of Cocoa. They used Cocoa every week during Multimedia Club, creating short, single Board games or "scenes" that they rarely completed or saved. The games were oftentimes modeled on existing video games, with updated plots or different characters. Sometimes students would build scenes with characters that acted independently from the person using the software, creating semi-autonomous characters with limited rule sets that they would set in a World of their creation. While some students did develop more complex programming styles and learn to take advantage of a limited palette of programming options in Cocoa to create complex Worlds, most students seemed content to keep their Cocoa programs simple. The

experience seemed more for the moment, as students built Worlds that they played with in Cocoa's viewer window and tweaked the rules to make them work correctly.

> Degrees of Comprehension: Children's Understanding of a Visual Programming Environment is an paper by Cyndi Rader, Cathy Brand, and Clayton Lewis, from the Department of Computer Science at the University of Colorado, evaluating their use of Cocoa in an educational environment. The Abstract reads as follows:
>
> > A new generation of innovative, highly visual children's programming environments is under development. In this paper, we consider the instructional requirements for children learning to program in a visual environment. Based on our year-long experience using Apple Computer's KidSim/Cocoa prototype [2] and the results of a year-end assessment, we conclude that the children failed to grasp many aspects of the program operation. The children readily mastered drawing and animating characters in imaginary worlds, but struggled to achieve more complex behaviors. Lack of explicit instruction on program functionality hindered these children in their attempts to create more sophisticated science programs. We explore the prospects for more effective instruction and suggest some guidelines for designing visual programming environments.

My experiences using Cocoa in my Multimedia Club closely mirror the findings of these researchers. I saw many students enter a project enthusiastic about the possibility of creating their own game, but who later became frustrated by the not-always-intuitive nature of programming in Cocoa. True, it is very easy to program using Cocoa: when you first start programming using Cocoa you are literally putting the software into "record mode" and programming by demonstration. However, once your programming style matures and you are trying to create more complex Worlds, programming using Cocoa becomes more of a means of manipulating the shortcomings of the programming environment to your advantage, "recording" less and rather constructing by hand the many interlocking parts of the Rules that govern the World. As my work with the Multimedia Club concluded, I had taken to building authoring kits from which students could take an already built set of complex rules, governing perhaps a character's movements, and change the art, add more boards, or add additional rules. This got students past the initial hurdle of creating more complex Worlds while allowing them to take apart and examine the Cocoa "code," which in turn gave them a better understanding of programming using Cocoa. This website attemps to fill in some of the gaps in the documentation to better educate the user about the intricacies, and fun, of Cocoa.

There are a few other articles about the educational role Cocoa might play in a technology-rich educational setting.

> Kurt Schmucker, who wrote the Cocoa FAQ and served as a Principal Scientist in Apple's Learning Technologies Group, wrote a paper about Apple's "Worlds of Science Contest," featuring Cocoa technology.
>
> > Mr. Schmucker's References are particularly interesting from an educator's standpoint. They are mainly case studies about KidSim, the predecessor to Cocoa (yet identical in its workings and intentions). My favorite is David Canfield Smith and Allan Cypher's KidSim: Child

Constructible Simulations article from the Proceedings of the Imagina '95 Conference, Monte-Carlo, February 1995, pp. 87-99.

Henry Lieberman has written a book on "Programming by Demonstration" called Watch What I Do: Programming by Demonstration. He has a web page that includes the full text of his book as well as an explanation of Programming by Example; places to learn more about this programming style; and examples of Programming by Example, including Cocoa.

Nothing, however, speaks as much about the fun and excitement of Cocoa than the various Worlds people have created in Cocoa.

Worlds

Since I no longer have access to a properly configured web server, I have chosen to make available the stand-alone application version of each of the Cocoa projects below. These are all built Worlds that have been **Saved for AutoPlay** from the Cocoa application. If you would like the "uncompiled World" file for use with Cocoa, email me.

Games

Alien Conspiracy: Programmed by Ryan Foltz, this multi-board adventure challenges you to steal the coordinates of a crashed space ship, battle your way across the sea and desert, and finally plant a charge on the alien ship in an effort to destroy it. See also the Read Me and Instructions, included in the download section.

Space Command: Written by Chris Zacherer, this is a Cocoa version of the arcade classic, Asteroids.

Quack!: In this Cocoa World you are a duck who must work its way through a series of challenging puzzles.

Happy Hot Dog: Written by a six year old, this Cocoa World challenges you to keep a hot dog alive by feeding it.

Cocoaban: Join in the fun with the newest Cocoa World from Josh! Take part in the Cocoaban Challenge. Visit this web page, download the Cocoaban authoring kit version 2.0 World, and customize it to include your new level. Everything you need to know and do can be found here.

The Ninja Series: Dan, who contacted me through this page, has spent a considerable amount of time working with Cocoa and has written many exciting games as he works to improve his Cocoa skills. These two games are from his Ninja Series.

Ninja 1: The Manebrane Nemesis starts the adventure.

Ninja 2: Trouble at Saurus Keep picks up with a new story line that is equally thrilling and even more difficult.

BashTwins: Dan describes this game as an update to the classic Macintosh game, Brickles. Use your paddle to knock out the bricks in pursuit of the taunting demons above you. There is an Easter egg in this game; can you find it?

Cola Wars: Pepsi™ has stolen the secret for Coca-Cola™. As a Coke employee, you must infiltrate the Pepsi headquarters and recover the recipe. Another exciting game from Dan!

Jumpball: A back to school offering for the Cocoa Projects Pages! I found a Newton version of this game on a Japanese web page and fell in love with it. Here is my Cocoa version. It's an early beta: the game works but needs some improvement.

Action0.9b: I found this game in the MIT Mac Archives, and included it here because of the hilarious graphics and sound effects. My friend Chris helped me translate the Japanese, which aided in figuring our how this cool game works. Please see the Read Me for information on which keys to use.

Tomb Raider: Help Lara Croft (from video game fame) navigate successfully through this challenging and exciting game!

Jeff's Adventure: Full of cool animation and puzzle-solving, this exciting game is a hint of things to come from this Cocoa programmer!

Moving Maze: Guide the Wanderer through an ever-changing maze in pursuit of the Crop Circle. I whipped this World together in under an hour after being inspired by the work of a Multimedia Club student. Small but complex, this World has only a few rules but proves to be quite challenging..

Juniper's Big World: The complete version of a game in which you help Juniper the dog navigate out in the big world. This game utilizes both click and key responses, and has three complete levels.

Adventure Game: Distributed with Cocoa DR3, this World features Will and Fido. There are two different boards: an Easy board, where Fido has a limited Rule set, and a Hard board, where Fido has a much more complex Rule set.

Cameron Hayne's ApplePuzzle. Cameron describes the puzzle as such: This is a reimplementation in Cocoa of the old Apple desk accessory called "Puzzle." The idea for this type of puzzle originated with Sam Loyd and it is usually referred to as a "15 puzzle."

Cameron Hayne's Tictactoe.

Wacko, the Virtual Pet: This is a world put together by the Cocoa Development Team. It is a bit like the Tamagatchi, where you raise a virtual pet, play with it,

feed it, and entertain it so that it survives and prospers.

Gogo Bananas!: A game in which you play Gogo the Gorilla, who you must navigate through a maze in search of tasty bananas. Developed by the Cocoa Team.

Master Guess: Kind of like Memory, but a bit more complex.

Wall Climber: A demonstration World programmed by the Cocoa Development Team. You can make the walls shorter or taller, but Wacko, the Cocoa mascot, will always be able to climb them.

Language Arts Worlds

Tiny KanjiWorld: This simulation, built by the Cocoa Development Team, shows how the combination of two characters results in a third character, which represents a third word. This is from the original About:

> KanjiWorld shows one of the ways in which the ideographs used in the Chinese and Japanese languages can be generated. The red ideographs are part of the 214 Chinese 'radicals' - simple ideographs that are used as building blocks of more complex ideographs. In KanjiWorld, these radicals move around randomly, but when they come together in just the right way, these radicals combine to form a more complex ideograph. These new generated ideographs are shown in black. The ideograph 'well' at the bottom of the world absorbs the generated black ideographs and randomly adds new radicals to the swirling mix.

Mathematics Worlds

Knight's Tour: Cameron Hayes, who used to run a wonderful Cocoa site with many fabulous examples of programming in Cocoa, got in touch with me about his Knight's Tour Worlds.

> The "Knight's Tour" is an old problem of recreational mathematics. The idea is to find a sequence of knight moves that visit every square on the board once and only once.This Cocoa program solves the Knight's Tour problem by using a very simple heuristic. At each move, it chooses the square which has the least number of available moves. If there are several squares with the same number of available moves, it just chooses the first one. You can download the Cocoa source code and look at how it was implemented.
>
> The algorithm used is amazingly robust. You can edit the board even while the knight is moving and it manages to function fairly well. For example, remove the two middle squares from the top 6 rows of the

board to make a U-shaped board. Even for such edited boards, it often comes close to a solution.

There is also a more basic Knight's Tour World, called Knight's Tour Random. This Cocoa program does not solve the Knight's Tour problem - here the knight just choses randomly from the available moves. I find it interesting that most of the time, this random strategy ends up visiting a large proportion of the board.

Finally, there is a Knight's Tour Partial World that shows it working (somewhat) on a partial board. It is slowed down to make it easier to see what is happening.

Because of the complexity of Cameron's Knight's Tour Worlds, I am including the Cocoa Source in the download area, so you can better examine the workings of this complex World. You can use Cocoa DR3 to save these Worlds as AutoPlayers.

Ask Marilyn: A challenging little puzzle, adapted from the Ask Marilyn column.

Pachinko: Like the Japanese pinball game of the same name, you watch the balls fall into the slots below. This World introduces the basic concept of probability and probability distribution to elementary and middle school students.

Roman Numerals: It would have been so much easier back in the day if the Romans had this handy calculator for computing the sum of two different Roman Numerals.

Waiting In Line: A multi-board simulation showing how people move through lines.

Science Worlds

Virtual Canary: This wonderful Cocoa World arrived in my email from a Cocoa-coder who I met through the Cocoa Mailing List (now the Creator Mailing List). The object of this simulation is to raise a canary from an egg to an adult. Excellent graphics and a fun premise for a World.

Smoke Rings: A simulation that shows how air molecules and smoke molecules interact to produce smoke rings.

Josh's Network Construction Kit: Build your own network topographies in this Cocoa World! First you'll explore the parts that make up a network, then you'll see a constructed network that you can walk Josh through to have him explain the different components. Last, you can construct your own network. This is my most complex Cocoa World to date. Note that this World was put together on a Mac with a large monitor; adjust your monitor depth to 1024x768 for best results.

Josh's Ocean Life: This is what you end up with if you follow the advanced tutorial (except I left out the diver). Click responses and random aquatic movement make for a simple undersea simulation.

Spread of a Rumor: This simulation was originally released as a sample world by the Cocoa Development. This simulation makes good use of counters. After I went into a classroom to fix a computer and observed that a teacher was using the same type of simulation with her students to show how the flu is spread, I converted this simulation into a standalone player for her class.

Ants: This simulation shows how ants use chemicals to trace a path from a food source back to the nest.

Plumin': A Plumber's Construction Set. Put the pipes together and turn on the water.

Mendel: Explore the world of genetics with this interesting simulation.

Sugar Water Evaporation: This simulation shows how the amount of sugar dissolved into a beaker of water affects the evaporation rate of the water.

Flower Garden: This is a sample World distributed with Cocoa. This World is a very simple simulation that demonstrates the interconnectivity of an ecological system.

Swarming Flies: Written by Gordon Worley and distributed under the GNU General Public License, this simulation examines the tendency to swarm exhibited by flies.

TrafficSim: Written by Cameron Hayne, this simulation lets you create the conditions that cause traffic jams. You can click on the highway to add a rock, which might cause an accident, or to add more cars to create more traffic.

Ratmaze: Another Cameron Hayne World, this simulation features rats trying to work their way through a maze.

Pressure & Piston: Developed by the Cocoa Development Team, this World presents a piston and gasses. You can adjust the pressure of the gasses contained in the piston and examine how it affects the piston's performance.

Butterfly Life Cycle: Featuring beautiful art work, this World, developed by the Cocoa Development Team, leads the user through the life cycle of a butterfly.

Cell Cycles: Another Cocoa Development Team World, this one shows the user how cells divide and reproduce.

Pond Life: A simple simulation that shows life in a pond. Frogs eat bugs and produce more frogs, while the bugs themselves also reproduce.

The A-mazing Wetlands: Part simulation, part game, this Cocoa World, programmed by students ages 8 to 11, leads the user through the rich ecology that makes up a wetlands.

If you are interested in getting your hands on the original source so you can see how these Worlds work, email me and I'll see about emailing you a copy of the Cocoa World you want.

Links

Cocoa resources are getting more and more difficult to find on the Internet. Here are some Cocoa Links that still work:

The Cocoa FAQ: Check here for answers to your Cocoa questions.

Cocoa Games: Chris Zacharer has a collection of Cocoa games on his web site. "Karoline in Paris" is particularly good. You can play the games online, or download the standalone player versions.

The DMOZ Open Directory Project maintains a page of Cocoa links.

Essential Cocoa provides a brief overview of getting started programming Cocoa.

The Google Web Directory has an entry for Cocoa.

Project Cocoa: you can download Cocoa for the PPC or a 68K Mac here, as well as a few examples with commented "code" to get you started.

Kurt Schmucker has a page with links to his Worlds of Science paper, as well as instructions on configuring a web server to serve up Cocoa Worlds.

Stagecast, who bought the technology behind Cocoa from Apple and developed its own Stagecast software, has Cocoa for download from their site. You can get the installers, tutorials, and movies here.

Triton Toys is a site developed as part of a school lesson. Some of the links are dead, but the site provides some good Cocoa resources. The Teacher Page also contains many resources.

Wecman's Big Adventure: A complex Cocoa game authored by Greg Miller. You can download a Demo version on this web page. Tenadar Software has a few cool Cocoa Worlds to purchase; unfortunately, their page seems to be down. I took a class taught by Mark Miller, Greg Miller's father, and I won the door prize: The Tenadar Software Combo CD!

updated 7/30/2003 by Josh Burker (remove the (remove) in my email address on this page to email me)