



**Tribeworks**

# Chapter 1

## How to Use This Book

### Introduction

This is not intended to be a user manual for iShell 3. Instead, it documents the processes and techniques we used to create a major project for the United States Naval Academy. We originally created this project in iShell 2, and have converted it to iShell 3, using many of the new features found in iShell 3. We have tried to write this document as if we were sitting across the table from you talking about how and why we created each of the documents in the project. It shows, in many cases, step by step techniques for utilizing the great power of iShell 3.

We invite you to use the templates, code and artwork in your own programs. We have included a folder full of the original Photoshop files we used to create this project. The only restriction is that images specific to the US Navy and the US Naval Academy can not be used without permission.

more to come....

### How to use this CD-ROM

more stuff here.

#### Table of Contents and Index

All of the table of contents items are linked to their respective pages. Click on any entry in the table of contents to go to that page. The index is likewise hot . Click on any page number to navigate to that page.

#### Hot Links in the Text

All of the cross references to other sections or Chapters are hot-linked. Click on any of the references and you will be taken to that page. Likewise, all of the HTTP references are linked and clicking on them will launch your web browser and open the linked URL.

#### Open the iShell Document

All of the file names shown in the text in bold are linked to their respective documents on the CD-ROM. For example, if you see a reference like **Main.kd**, you can click on it and open it in the iShell Editor. Likewise, **ARTWORK/Main Screen/mainScreen.psd**, will open the file in Photoshop.

## **Embedded Movies**

Many of the Chapters include embedded Quicktime movies in the left margin. These movies show some of the step-by-steps instructions as a screen-captured video. Simply click on the small video icon and the movie will enlarge to full size, start playing and return to icon-sized when done.

## **Chapter Contents**

The following describes the contents of each Chapter. This will tell you the key iShell concepts that are discussed in detail in that Chapter. You can click on any of the bold headings and go directly to that page. We would recommend that you print this chapter out and circle those topics that most interest you and then proceed to those sections and read them. You may have to back up and read a few sections before as most of the discussions are in the context of the entire project. We also highly recommend that you open the source files in the iShell Editor and follow along as you read the text.

### ***Chapter 2, “Planning the Project”***

This Chapter gives an overview of the entire project. It details some of the thought processes we used to arrive at our overall design approach.

#### ***Project Definition, page 7***

Learn a little about the project and our experiences with an enthusiastic but naive customer. This set the stage for major project decisions.

#### ***System Specifications, page 9***

Here, we detail the system requirements for this project on both the Windows and Macintosh platforms. Systems performance and screen resolution are discussed.

#### ***Media, page 11***

We discuss where we obtained the raw media for this project. This includes, photos, video, audio and Quicktime VR panoramas.

#### ***Project Structure, page 13***

There are a number of different ways to develop a project structure that allows the user to navigate through the content. We discuss, in general terms, several different navigation schemes and then describe our choice for a navigational structure.

#### ***Implementing the Structure, page 18***

Once we decided on a structure, we needed to come up with a top-level design for our implementation. That design is described here.

**Project Setup, page 20**

We discuss the directory structures we used to organize both the raw assets as well as the files used in the Media folder. Naming conventions for both files and variables are discussed. Finally, we show how we created the Project file (.kp) and customized the Windows Settings and URL Settings for our project.

**Chapter 3, “Designing the Main Screen”**

We do a lot of design in Photoshop prior to writing any iShell code. This Chapter discusses how we do some of the design and how we build a bridge between the artwork and final iShell code.

**Creating the Artwork, page 28**

We use Photoshop almost exclusively to build our artwork. Here, we describe how we build all of the interface elements for one screen in a single Photoshop file and then extract the pieces for the runtime media assets.

**Flipbooks, page 30**

We use a lot of Flipbooks for buttons. This details, step-by-step, how we create Flipbooks in Photoshop.

**Volume Control Knob, page 34**

This section discusses how to avoid using transparency (alpha channels) in a button so as to limit the impact on processor performance.

**Titles, page 36**

Here we discuss some techniques that we use to create multiple images that will replace each other in the same area of the screen. We make use of Photoshop layers and saved selections to ensure that the artwork lines up properly in the final iShell implementation.

**Artwork File Formats, page 37**

This is a short discussion of two of the picture formats we use, PICT and PNG.

**Alpha Channels, page 37**

This short discussion of the pros and cons of using transparency, or alpha channels, in your project. We follow with a full description of how we create transparent images using Photoshop. This is found in *Creating the Image with Transparency*, page 38.

**One More Thing..., page 40**

Even though we used transparency, we wanted minimize the impact on the computer system. This connects the artwork to the code and shows how we prepared the artwork to minimize the processor penalty for using alpha channels.

## **Chapter 4, “The Main Screen Document”**

We actually start laying our first iShell document in this Chapter. Here we create our master document and discuss how it acts as a container for all of our content documents. This document is loaded once and remains on the screen for the entire presentation.

### ***Creating the Main Document, page 43***

Step-by-step instructions for creating a new document in iShell 3.

### ***Laying Out the Basic Interface Elements, page 45***

- Using a Box Element as an organizational container (Parent/Child Elements)
- Naming an Element so it doesn't show up in a target list
- Relative Positioning
- Setting Transparency

### ***Adding the Click Sound, page 48***

This details the use of a Sound Element. Learn why you need to use Buffers to ensure a smoothly playing sound. When should you load a complete sound into memory.

### ***Create a dummy document, page 50***

No, this isn't a stupid document, but shows how we embed another iShell document in our master document. It discusses how to layer the document so that the transparency of the frame provides shadows and round corners.

### ***Communicating between Documents, page 52***

This is an introduction to inter-document communications. That is, how does an Element in the contained document tell an Element in the master document to do something.

### ***Playing a Sound Using a Global Target, page 54***

This is an efficient alternative to using Messages to communicate between documents. It describes the use of a Global Variable as a Target for a command.

### ***Using Messages, page 55***

This is a complete discussion of Sending and receiving Messages as they are used for inter-document communications.

### ***Messages vs. Global Targets, page 58***

This gives some hints as to when it makes sense to use Messages and when it is more efficient to use Global Target Variables.

### ***Using Properties and Templates, page 59***

We used a lot of Templates in this project. This section gives an overview of Templates and how Properties are used within them.

***Creating the Template, page 60***

This contains step-by-step instructions for creating a template and defining Properties. We used this template for multiple buttons on the right side of the screen.

***Button Behavior, page 64***

As part of our template creation, we discuss the standard button behaviors we use in most our projects.

***Adding the Template to the Main Screen, page 68***

After completing the template, we discuss how to add multiple instances of it to the main screen. This implements all of the main navigation buttons and shows how to set Property values for each instance of the template.

***Using Former Positioning, page 73***

Ever wonder where it would be practical to use the Former mode of positioning an Element? Here tis.

***The Bitmap Option, page 75***

When do you check Bitmap in an Image Element? A discussion of the pros and cons and how to calculate how much memory and image takes up.

***Back Button, page 80***

This contains a discussion of the new Back Command with a Target in iShell 3.

***Adding the Volume Slider, page 82***

We had a volume slider in our code library, here we add it to the project.

***Chapter 5, "The Volume Slider"******Artwork, page 85***

This is a very quick discussion of how we created the artwork for the slider and the slider s knob.

***Implementation, page 86***

We begin this discussion with a discussion of how to use the Mouse Moved Event and then limiting the movement of the volume knob with a Drag Command that is constrained by the containing Element.

***The Math Behind the Movement, page 87***

We provide a comprehensive discussion of the math required to change the volume.

***Attribute Getters, page 88***

We discuss in detail the new Attribute Getters for Position and Size, such as Absolute Width, Media Width and Width.

### ***Calculate the Volume, page 94***

This shows how we calculated a new volume and set it. This deals with setting the System Volume, not the volume of individual sound elements.

### ***Initial Settings for the Volume Knob, page 95***

When this screen is first loaded, the knob needs to be set to the current system volume. This shows how to get the volume and then set the knob. It includes a short discussion of Runtime Getters.

## ***Chapter 6, "Introduction Screen"***

This is the first content screen we created. It contains a short slide show that uses the new timeline feature of iShell 3.

### ***Creating the Media, page 100***

We discuss creating the slides and narration for this introductory sequence.

### ***Creating Chapter Tracks, page 103***

The bulk of the media discussion is focused on step-by-step instructions on how to create Chapter tracks in Quicktime movies using just Quicktime Player Pro.

### ***Timeline Sequence, page 110***

We used the new Timeline feature of iShell 3 to create a sequence of images that were timed to some music. This details how to use the Timeline and the Trigger Event, including how to set key frames and assigning values to the Triggers.

### ***Adding Mission Buttons, page 117***

This is another example of using a Template with Properties. This time, we create buttons out of invisible Box Elements. It also shows how to add the template to the introduction screen.

### ***Adding the Narration, page 121***

We created a Chapter track in an audio movie earlier. This section shows how to use the individual Chapters to trigger a Chapter Event.

## ***Chapter 7, "Home Screen"***

This screen is the equivalent of the home screen in a web browser. It is where everything starts. This was, by far, the most difficult screen to create, not from an iShell standpoint, but from a content standpoint.

### ***Design Considerations, page 128***

The user wanted a full screen movie to run in the background. Here, we discuss some of the design trade-offs we had to make in preparing this screen. We discuss how iShell renders movies direct-to-screen and why we didn't use a layered Quicktime movie.

***Building the Movie, page 129***

We show how we created a graphic overlay for the movie that contains the text of the mission statement. It discusses the filters we had to use to get the text to show over the video.

***Compressing the Movie, page 132***

The shows the techniques and decisions we made in compressing this movie using the Sorenson 2 video codec. A more complete discussion of video compression is found in the next Chapter.

***Adding the Overlay, page 135***

We show how we added the text overlay created earlier. Performance is optimized when using pre-multiplied black alpha. We walk through an example of creating a source file in Photoshop using pre-multiplied black alpha.

***Add the Buttons, page 140***

We used the same buttons we created in the introductory sequence. Here we add them to the movie. Since the buttons are transparent boxes, they don t impact the movie performance.

***Chapter 8, "Movie Screen"***

All of the video created for this presentation are shown on this one screen. It has a full set of movie controls and can show movies at normal size (320x240) or full-screen (640x480).

***Artwork, page 143***

We prepared an image that would be used as a Offset Image kind. In addition, we created some more Flipbooks.

***Movie Preparation, page 145***

This contains an extensive discussion on many aspects of preparing video for display from CD-ROM. The following topics are discussed:

- Source Movie formats - what gives the best resolution?
- Capturing Video
- Compressing Video - in general.
- How codecs work
- Our compression settings for this project
- Sorenson compression
- Audio compression

***Creating a New Source File, page 155***

We faced some interface issues when displaying the movie at full screen, this details the issues we face, the solution is described later.

### ***Selecting the Font, page 156***

We created a Title field on the movie frame. Here, we have a short discussion about fonts and cross-platform issues.

### ***Clipping the Cornerpage 158***

This contains a complete discussion of how to get rounded corners on a video by using iShell clipping regions. It shows how we turn the clipping on and off depending on the size at which the movie is displayed.

### ***Adding the Bottom Frame, page 160***

This is how we solved the interface issues with the full-screen movie. We describe a content document that is larger than it should be and how we went back to the main screen to fix it.

### ***Adding Buttons, page 161***

We used the Offset Image option to add control buttons for the movie to the bottom of the screen.

### ***Playing the Movie, page 167***

This is a full discussion of how to implement the movie controls as separate buttons. It includes buttons that start, stop, rewind, fast forward and fast rewind the movie. Learn how to play a movie at 300% of its normal speed (backwards too). Learn about the Mouse Idle Event.

### ***Full Screen Video, page 174***

This describes the commands we used to change the size of the movie from 320x240 to 640x480. We also discuss the use of a Private Property, which is different from the way we used Properties in Templates.

## ***Chapter 9, "QuickTime VR"***

This chapter contains everything you ever wanted to know about Quicktime VR in iShell, but were afraid to ask. We considered this the centerpiece of the presentation.

### ***VR Production Tools, page 182***

We begin with a comprehensive discussion of how we photographed the VR scenes. It includes discussions of:

- Planning the photo shoot
- Camera
- Tripods
- Film and Developing
- Camera Leveling
- Exposure
- Tips on documentation

**Rendering the Scenes, page 188**

Once we had the photos developed and in hand, we had to stitch them into usable scenes. This section describes the tools we used to render the scenes. It includes discussions of;

- Stitching and Rendering Software
- VR Utility Software
- The Stitching process
- Creating the Panorama - tiling and compression
- Placing and Connecting Nodes
- Adding Hotspots

**Node Notes, page 197**

This is a short discussion of RTF text and how we created it.

**Vertical Scrollbars, page 198**

This contains a discussion and examples of how to create and use Scrollbars in iShell 3. This is completely new to iShell 2 users.

**Basic Elements, page 201**

Here, we add the basic panorama to the iShell document.

**VR Controls, page 203**

This is a very comprehensive discussion about how we added custom panning and zooming controls for the panorama. We used very large flipbooks (27 and 81 cells) to control the VR. We show how to do some extensive math calculations to place both the Flipbooks and the Panoramas in proper orientation. The discussion includes:

- Field of View calculations
- Saving the Maximum Zoom Values
- Mouse Drag Command (with constraint)
- Mouse Idle Event
- Mouse Moved Event
- Set Cursor Command
- Log Command and the Debugger

**Navigate with Hotspots, page 220**

Here we used the Data Element as Tables to navigate from node to node in the VR scene. We used data from the tables to switch to a new VR node and to control the text that is displayed with each node. It includes discussions of the following:

- The Data Element loaded as a Table
- Set String functions
- Tel Table Commands
- Private Properties
- Hot to show text when the user rolls over a hot spot
- On Panorama/Enter Spot Command with no argument

### ***Node Information, page 232***

When the user clicks on the info button, the panorama is hidden and a text frame with scroller is shown that displays an RTF text file. This section discusses:

- A click shield - how to quickly disable a lot of buttons
- Using Mask/Unmask in the Editor
- Adding a Scrollbar to the Project
- Adding a Scroller Element
- Adding text to the Scroller

### ***Prepare for the Map, page 239***

We used a full screen map to help the user navigate through the scene. The next Chapter describes the map, this section describes how to use a Global to save a critical value that would be needed by the map screen.

## ***Chapter 10, "VR Tour Map"***

In order to help the user navigate through the 39 nodes, we provided a full screen map of the Academy in a separate file. Each of the areas on the map that contained a node was identified when the user rolled over it with the mouse. Clicking the node reloaded the VR tour with that node.

### ***Creating the map, page 244***

We started with an .eps map and converted it PICT format. We faced challenges getting it to fit on a 640x480 screen.

### ***Setting up Hot Regions, page 246***

Using the Hot Region magic wand selection tool in iShell can be difficult. Here, we show how we prepared an alternate version of the map art to make it easier.

### ***Updating the Node Table, page 249***

We used a little indicator light to show where the current node was located. Here we discuss how we updated our tab-delimited table with the iShell X,Y coordinates for the light.

### ***Placing the Light, page 251***

We needed to place the light at the proper place when the map screen was loaded. This section discusses:

- Animating a Flipbook as a blinking light
- Using a Set Tweener Command
- Using the Tell Table Command to get the light coordinates
- Dragging the light to the correct position

**Creating a Map Button Template** page 253

Once again, we turned to a template to speed production. This template was used to create rollover buttons for each of the nodes on the map. It used a Filter Lite Box Element. Tell Table Commands were used to get the name and number of the node.

**Implementing the Template** page 257

This shows how we added the 39 node buttons to the map, setting the proper Properties for each of the nodes. Each of the buttons used a custom Hot Region that was selected by using the Magic Wand tool of the Hot Regions tool. Here we temporarily substituted the custom artwork for the map that made it easier to select the regions.

**Chapter 11, “Career Content Screen”**

This is a content screen for one of the major categories in the project. This screen leads to a number of other content screens. Here we describe so of the features of this screen.

**Artwork, page 262**

The artwork on this screen reflects the action that the customer desired. We discuss the basic background artwork and detail how we created the rollover effects for each of the buttons. In addition, we did a slide show similar to that found in the introduction sequence.

**Using a List, page 267**

Each of the content screens plays an narration with a slide show. We needed some way to keep track of which narration has already been played. Here we discuss how we use the iShell data type of List to keep track for us. It includes discussions of:

- Target Identifiers - what is a Target
- Initializing a List
- Adding data to a List
- Checking the contents of a List
- If/Else Commands
- Message In/Message Out Parameters

**Stopping the Narration** page 270

This shows how to use a Keyboard Event to capture any key press.

**Timing the Slideshow, page 271**

This is a short review of using Chapter Events to time slides to the narration.

**Adding the Buttons, page 272**

We added invisible Box Elements as buttons over the text. Previously created templates didn't work, but were close. We show how we duplicated a template and modified slightly for use here.

## Chapter 12, "Slide Show"

Many of the content screens were simply a slideshow of photographs that were accompanied by music. The slides were 600x400 images that either advanced automatically or the user could step through one by one. This Chapter describes the template we created for viewing these slide shows.

### **Slides, page 277**

We used Photoshop batch tools to process and name our slides. This describes that process and the very important naming conventions we used to name each slide.

### **Image Processing Tools, page 279**

This is a very, very short discussion of the three tools we use most in our production work.

### **Set up Properties, page 281**

Since this document is a template document, we once again needed to create some Properties that define some of the variables for the show. We also describe the use of a Private Property that is used only on this screen.

### **Run the Show, page 282**

This shows how the Autorun checkbook works and how we set a Duration on a regular Image Element so that it acts as a timer. It discusses important information about the placement of the Loaded Events.

### **Change the Slide, page 282**

This highlights the processes we used to change the slide at a defined interval, it includes:

- Incrementing the slide count
- Building a URL dynamically using Set String Commands
- Switching the Slide

### **Testing, page 285**

We make liberal use of the Log Command and the Debugger. It includes a discussion of how to set and use Breakpoints.

### **Slide Controls, page 287**

The user can start and stop the slide show as well as advance forward and backward slide by slide. Our discussion includes:

- Using the new iShell 3 Image Kind, Button
- Overriding Button Kind behavior
- Starting and Stopping the Slide show
- Advancing one slide at a time - forward and backward
- Testing to see if you are at the end of the slide show

## Chapter 13, “Implementing the Slide Show”

In the previous Chapter, we only created the template for the slide show. This Chapter shows how we implemented it.

### **Adding Sound, page 296**

This screen contained two Sound Elements. We discuss the benefits of using the Sound Element vs. using a Quicktime Element to play the sound.

### **Setting up the Basics, page 296**

We discuss some of the common set up tasks that we needed to do on every content screen.

### **Adding the Slide Show, page 300**

This shows how we added the template for the slide show (created in the previous Chapter). It shows how we set the Properties for this particular implementation.

### **Adding the Web Links, page 302**

This screen contains links to various US Navy web sites. We created yet another template to create buttons for the Launch URL Command. This sections details how to:

- Adding Template Properties
- Use a Box Element with Frame
- Turn the Frame on and off
- Use the Launch URL Command
- Implement the button template

## Chapter 14, “Content: Career Pages”

This Chapter deals with the additional content screens for Careers that are not covered with the slide show template. We did not go into as much detail in this Chapter, but will point out here, things that will be of interest.

### **Summer Training Top-Level, page 309**

Here we discuss the Summer Training screen and the buttons used on the screen. While the behavior of the buttons is pretty much standard, we go into some detail about how to use Image Offset to save loading time for Image Elements.

### **Summer Training Slide Show Template, page 312**

Yes, another template. There are five self-running slide shows associated with each of the buttons in the main Summer Training screen. This describes how we created a simple self-running slide show template. It does not permit advancing slide-by-slide.

### **Music Events, page 314**

This section is of note because it shows how to play two sounds and once, music and narration. It shows how to reduce the volume of the music while the narration is playing

and then gradually increase the volume when the narration is done. It shows how to use a While Command.

### ***Using the Chapter Track, page 315***

This shows a new way to use a Chapter Track. It uses the On Movie/Chapter Event without an argument. Any Chapter encountered will trigger the Event. It also includes discussions of the following:

- Private Properties
- Building URLs dynamically with Strings
- Using a Counter and incrementing it
- Converting from a regular document to a template
- Adding a Connection to a Property
- Wiring the Connection

### ***Implementing the Template, page 329***

We implemented five instances of this new slide show template, setting unique Properties for each.

### ***Service Assignment, page 321***

This is a very simple screen, but we used some old and new features here:

- Playing two sounds at once
- Increasing the volume with a While Loop
- Using any Element as a Timer
- Timescale of a Duration and what it means

### ***Notable Graduates, page 325***

Again this is a simple screen that makes use of the following:

- Yet another template
- Box Element with Frame
- Box Element with Filter Lite/Invert
- Using a Global Variable as a Target
- Switching Movie and Image Elements
- Sizing a Button created from a Template

## ***Chapter 15, "Content: Athletics"***

The Athletics section contains the Athletics Main Screen, plus a number of sub-screens. We didn't have much content provided so we had to improvise.

### ***Athletics Main Screen, page 332***

At the time we implemented this screen, we were worried about processor performance. This discusses some of the steps we took that we thought would optimize that.

### ***Positioning Using the Former Modifier, page 333***

This shows how we used Former to help position Elements on screen.

### ***Audio and Slide Show, page 334***

The main Athletics screen used audio linked to a slide show using Chapter Tracks. Once again we used Chapter Events without an argument and built the URLs for the slides using String functions. We also needed a Private Property to keep track of the current slide.

### ***Adding Buttons, page 337***

The buttons down the left side of the Athletics Main Screen were implemented with a template. We discuss the following:

- Templates and Properties
- Properties and Connections
- Global Target Variables instead of Messages
- Implementing a Template
- Using Mask/Unmask in the Layout window

### ***Varsity Sports, page 341***

Varsity Sports is a screen that is linked to the Main Athletics Screen. In it, we use the following:

- Button Image Kind with different On Click Go to -(new to iShell 3)
- Radio Buttons (Push Buttons) - how to implement
- Another Template for buttons (with Properties)
- Nested Containers
- Box Elements with Filter Lite/Transparent
- Web links and Launch URL
- Activate Event and stopping sound

### ***Club Sports Screen, page 353***

The Club Sports screen is even less exciting than the Varsity Sports Screen. It s so dull that we won t even describe it, other than to say that the functionality is identical to Varsity Sports with the exception that there are no web links and the photos are nothing more than a text description of the sport.

### ***Intramural Sports Screen, page 353***

The Intramural Sports Screen is likewise unexciting. All of the intramural sports are listed and when you click on a sport, the rules for that sport are displayed. We ll list a few of the features on this screen as they pertain to things we haven t discussed in earlier chapters.

- Click shield to quickly disable buttons
- Using HTML Text instead of RTF
- Yet another Template
- Using a List to pass more than one item in a Message In Parameter

### ***Athletic Facilities, page 360***

This screen is a composite of the functionality of the VR Map and the Intramural Sports we just discussed. Instead of seeing a VR node, the user gets a picture and narration of the sports facility on the map. It uses the following:

- Data Element - Table
- Click Shield
- Another Template
- Hot Regions
- If/Else Commands
- Tell Table Commands
- Drag Command
- Using a Layered Quicktime movie and when it makes sense
- Refresh Command