

Types of Authoring Tools

- Each multimedia project you undertake will have its own underlying structure and purpose and will require different features and functions.
- E-learning modules such as those seen on PDAs, MP3 players, and intra-college networks may include web-based teaching materials, multimedia CD-ROMs or web sites, discussion boards, collaborative software, wikis, simulations, games, electronic voting systems, blogs, computer-aided assessment, simulations, animation, blogs, learning management software, and e-mail. This is also referred to as distance learning or blended learning, where online learning is mixed with face-to-face learning.
- The various multimedia authoring tools can be categorized into three groups, based on the method used for sequencing or organizing multimedia elements and events:
 1. Card- or page-based tools
 2. Icon-based, event-driven multimedia and game-authoring tools
 3. Time-based tools

Card- and Page-Based Authoring Tools

- **Card-based** or **page-based** tools are authoring systems, wherein the elements are organized as pages of a book or a stack of cards. Thousands of pages or cards may be available in the book or stack.
- These tools are best used when the bulk of your content consists of elements that can be viewed individually, letting the authoring system link these pages or cards into organized sequences.
- You can jump, on command, to any page you wish in the structured navigation pattern.
- Page-based authoring systems such as LiveCode from Runtime Revolution (www.runrev.com) and ToolBook (www.toolbook.org) contain media objects: buttons, text fields, graphic objects, backgrounds, pages or cards, and even the project itself.
- The characteristics of objects are defined by properties (highlighted, bold, red, hidden, active, locked, and so on).

- Each object may contain a programming script, usually a property of that object, activated when an event (such as a mouse click) related to that object occurs.
- Events cause messages to pass along the hierarchy of objects in the project; for example, a mouse-clicked message could be sent from a button to the background, to the page, and then to the project itself. As the message travelled, it looks for handlers in the script of each object; if it finds a matching **handler**, the authoring system then executes the task specified by that handler.
- Following are some typical messages that might pass along the object hierarchy of the LiveCode and ToolBook authoring systems:

LiveCode Message	ToolBook Message
closeCard	leavePage
closeStack	leaveBook
idle	idle
mouseDown	buttonDown
mouseStillDown	buttonStillDown
mouseUp	buttonUp
newBackground	newBackground
openCard	enterPage
openStack	enterBook

- Now let's look at specific examples.
- To go to the next card or page when a button is clicked, place a message handler into the script of that button.
- An example in RunRev's LiveCode language would be:

```
on mouseUp
  go next card
end mouseUp
```

An example in ToolBook's OpenScript language would look like:

```
to handle buttonUp
  go next page
end buttonUp
```
- The handler, if placed in the script of the card or page, executes its commands when it receives a "mouseUp" or "buttonUp" event message that occurs at any location on the card or page—not just while the cursor is within the bounds of a button.

- Card- and page-based systems typically provide two separate layers on each card: a **background layer** that can be shared among many cards, and a foreground layer that is specific to a single card.

Icon- and Object-Based Authoring Tools

- **Icon- or object-based, event-driven** tools are authoring systems, wherein multimedia elements and interaction cues (events) are organized as objects in a structural framework or process.
- Icon- or object-based, event-driven tools simplify the organization of your project and typically display flow diagrams of activities along branching paths.
- In complicated navigational structures, this charting is particularly useful during development. Icon-based, event-driven tools provide a visual programming approach to organizing and presenting multimedia.
- First you build a structure or flowchart of events, tasks, and decisions, by dragging appropriate icons from a library. These icons can include menu choices, graphic images, sounds, and computations.
- The flowchart graphically depicts the project's logic. When the structure is built, you can add your content: text, graphics, animation, sounds, and video movies.
- Then, to refine your project, you edit your logical structure by rearranging and fine-tuning the icons and their properties.
- With icon-based authoring tools, non-technical multimedia authors can build sophisticated applications without scripting.
- In Authorware from Adobe, by placing icons on a flow line, you can quickly sequence events and activities, including decisions and user interactions.
- These tools are useful for storyboarding, as you can change sequences, add options, and restructure interactions by simply dragging and dropping icons.
- You can print out your navigation map or flowchart, an annotated project index with or without associated icons, design and presentation windows, and a cross-reference table of variables.

Time-Based Authoring Tools

- **Time-based tools** are authoring systems, wherein elements and events are organized along a timeline, with resolutions as high as or higher than 1/30 second.
- Time-based tools are best to use when you have a message with a beginning and an end. Sequentially organized graphic frames are played back at a speed that you can set.
- Other elements (such as audio events) are triggered at a given time or location in the sequence of events.
- The more powerful time-based tools let you program jumps to any location in a sequence, thereby adding navigation and interactive control.
- Each tool uses its own distinctive approach and user interface for managing events over time.
- Many use a visual timeline for sequencing the events of a multimedia presentation, often displaying layers of various media elements or events alongside the scale in increments as precise as one second. Others arrange long sequences of graphic frames and add the time component by adjusting each frame's duration of play.
- **Flash** : Flash is a time-based development environment. Flash is also particularly focused on delivery of rich multimedia content to the Web.
 - With the Flash Player plug-in installed in more than 95 percent of the world's browsers, Flash delivers far more than simple static HTML pages.
 - ActionScript, the proprietary, under-the-hood scripting language of Flash, is based upon the international **ECMAScript** standard derived from Netscape's original JavaScript.
- **Director** : Adobe's Director is a powerful and complex multimedia authoring tool with a broad set of features to create multimedia presentations, animations, and interactive multimedia applications.
 - It requires a significant learning curve, but once mastered, it is among the most powerful of multimedia development tools.
 - In Director, you assemble and sequence the elements of your project, called a "movie," using a Cast and a Score.
- The **Cast** is a multimedia database containing still images, sound files, text, palettes, QuickDraw shapes, programming scripts, QuickTime movies, Flash movies, and even other Director files.
 - You tie these Cast members together using the **Score** facility, which is a sequencer for displaying, animating, and playing Cast members, and it is made up of frames that contain Cast members, tempo, a palette, timing, and sound information.

- Each frame is played back on a **stage** at a rate specified in the tempo channel. Director utilizes **Lingo**, a full-featured object-oriented scripting language, to enable interactivity and programmed control.

Objects

- In multimedia authoring systems, multimedia elements and events are often treated as **objects** that live in a hierarchical order of **parent and child relationships**.
- Messages passed among these objects order them to *do* things according to the **properties** or **modifiers** assigned to them.
- In this way, for example, Teen-child (a teenager object) may be programmed to take out the trash every Friday evening, and does so when they get a message from Dad. Spot, the puppy, may bark and jump up and down when the postman arrives, and is defined by barking and jumping modifiers.
- Objects typically take care of themselves. Send them a message and they do their thing without external procedures and programming. Objects are particularly useful for games, which contain many components with many “personalities,” all for simulating real-life situations, events, and their constituent properties.
- Object-based authoring programs typically provide objects pre-programmed with sensible properties, messages, and functions.
- A video object, for example, will likely have a duration property (how long the video plays) and a source property (the location of the video file) and it will likely accept commands from the system such as “play” and “stop.”

Chapter 11

The World Wide Web and HTML

- The World Wide Web (www.w3.org/) started in 1989 at the European Particle Physics Laboratory (CERN) as a “distributed collaborative hypermedia information system.”
- It was designed by Tim Berners-Lee as a protocol for linking a multiplicity of documents located on computers anywhere within the Internet.

- This new Hypertext Transfer Protocol (HTTP) provided rules for a simple transaction between two computers on the Internet consisting of:
 - (1) establishing a connection
 - (2) requesting that a document be sent
 - (3) sending the document
 - (4) closing the connection.
- It also required a simple document format called Hypertext Markup Language (HTML) for presenting structured text mixed with inline images.
- An HTML document could contain hyperlinks or anchors that referred to other similar documents. With browser software, users could then click on designated areas of hot text in one document and jump to another, which itself might have more hot text pointing to yet other documents.
- Users could surf from document to document across the Web, with HTML as the underlying buoyant framework.
- Berners-Lee is currently developing the next evolution, the **Semantic Web**, which “provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries.”

Dynamic Web Pages and XML

- HTML is fine for building and delivering uncomplicated *static* web pages. But you will need other tools and programming know-how to deliver *dynamic* pages that are built on the fly from text, graphics, animations, and information contained in databases or documents.
- JavaScript and programs written in Java may be inserted into HTML pages to perform special functions and tasks that go beyond the vanilla abilities of HTML—for mouse rollovers, window control, and custom animations.
- Cold Fusion and PHP are applications running side by side with a web server like Apache; they scan an outgoing web page for special commands and directives, usually embedded in special tags.
- If they find a special tag in the page, the software will do what the tag tells it to do, like “get today’s date and put it into that table cell” or “search this database for all customers with balances greater than \$100 and, after alphabetizing, put that list into a table on the web page being served.”

- Working hand-in-hand with these application servers, Oracle, Sybase, and MySQL offer software to manage **Structured Query Language (SQL)** databases that may contain not only text but also graphics and multimedia resources like sounds and video clips.
- In concert with HTML, these tools provide the power to do real work and perform real tasks within the context of the World Wide Web. Flash animations, Director applications, and RunRev stacks can also be called from within HTML pages.
- These multimedia mini-applications, often programmed by Web developers, use a browser plug-in to display the action and perform tasks such as playing a sound, showing a video, or calculating a date.
- As with Cold Fusion and PHP, both use underlying programming languages. With the introduction of HTML5, browsers can play multimedia elements such as sound, animations, and video without requiring special plugins or software.
- **XML (Extensible Markup Language)** goes beyond HTML—it is the next evolutionary step in the development of the Internet for formatting and delivering web pages using styles.
- Unlike HTML, you can create your own tags in XML to describe exactly what the data means, and you can get that data from anywhere on the Web.
- In XML, you can build a set of tags like


```
<fruit>
<type>Tomato</type>
<source>California</source>
<price>$.64</price>
</fruit>
```

 and your XML document, according to your instructions, will find the information to put into the proper place on the web page in the formatting style you assign.
- For example, with XML styles, you can declare that all items within the <price> tag will be displayed in boldface Helvetica type.
- In development as a technique to deliver more pleasing web experiences, AJAX (Asynchronous JavaScript and XML) uses a combination of XML, **CSS (Cascading Style Sheets)** for marking up and styling information), and JavaScript to generate dynamic displays and allow user interaction within a web browser.
- Multimedia on the Web During the coming years, most multimedia experiences on the Internet will occur on the World Wide Web, programmed within the constraints of HTML, then stretched by the enhanced capabilities provided by XML, Java, JavaScript, AJAX, and

special plug-ins like Flash and QuickTime to enable browsers to exceed their limits. These tools are used to build “**Web 2.0**” sites where there is collaboration and information sharing such as seen in blogs, on wikis, and at social networking sites such as Facebook and Twitter.

- To design and make effective multimedia for this environment, developers need to understand not only how to create and edit the elements of multimedia, but also how to deliver it for HTML browsers and plug-in/ player vehicles.
- Well-crafted, professionally rendered sites on the Web include text, images, audio, and animation presented in a user-friendly interface that balances the bandwidth deficit against user patience. Inside the event horizon of the amazing World Wide Web explosion are many uncertainties and unsolved challenges.
- The bandwidth deficit will certainly be met with technology solutions that will reach the last mile into homes and businesses.

For any doubt contact 9873961590.