

Multimedia Authoring System

Definition: Authoring System

An Authoring System is a program which has pre-programmed elements for the development of interactive multimedia software titles. Authoring systems vary widely in orientation, capabilities, and learning curve. There is no such thing (at this time) as a completely point-and-click automated authoring system; some knowledge of heuristic thinking and algorithm design is necessary. Whether you realize it or not, authoring is actually just a speeded-up form of programming; you don't need to know the intricacies of a programming language, or worse, an API, but you do need to understand how programs work.

Why use an authoring system

It generally takes about 1/8th the time to develop an interactive multimedia project, such as a CBT (Computer Based Training) program, in an authoring system as opposed to programming it in compiled code. This means 1/8 the cost of programmer time and likely increased re-use of code (assuming that you pass this project's code to the next CBT project, and they use a similar or identical authoring system). However, the content creation (graphics, text, video, audio, animation, etc.) is not generally affected by the choice of an authoring system; any production time gains here result from accelerated prototyping, not from the choice of an authoring system over a compiled language.

Multimedia Authoring tools

Tools for making a complete multimedia presentation where users usually have a lot of interactive controls.

Macromedia Director

- Movie metaphor (the cast includes bitmapped sprites, scripts, music, sounds, and palettes, etc.)
- Can accept almost any bitmapped file formats
- Lingo script language with own debugger allows more control including external devices, e.g., VCRs and video disk players
- Ready for building more interactivities (buttons, etc.)
- Currently in version 7.0, this popular general market product follows the cast/score/scripting paradigm, which makes it the tool of choice for animation content. Its roots as a cell- and sprite-animation program are unmistakable; and its inclusion of Lingo, its object-based scripting language, has made it the animation-capable program to beat. The AfterBurner compression Xtra creates Shockwave files, allowing Web playback.

Authorware

- Professional multimedia authoring tool
- Supports interactive applications with hyperlinks, drag-and-drop controls, and integrated animation
- Compatibility between files produced from PC version and MAC version

Other authoring tools include:

- Microcosm : Multicosm, Ltd. ; DOS, Windows Microcosm is a Hypermedia Linkage authoring system.
- Question Mark : Question Mark Computing Ltd ; DOS, Mac, Windows; WWW (via Perception) Question Mark is optimized for Electronic Assessment production.
- Emblaze Creator : Geo International ; JavaScript, Mac, Windows95, WWW.
- Emblaze Creator 2.5 is a cast/score/scripting tool which is designed for Web-based playback of interactive multimedia.
- Flash : Macromedia ; Mac, Windows95, NT, WWW (via Flash Player).
- Flash 3.0 is a cast/score/scripting tool, which primarily uses vector graphics (and can create vector graphics from imported bitmaps). It is optimized for Web delivery, and is especially common for banner adds and small interactive web deliverables.
- HyperCard : Apple Computer ; Mac, WWW (via LiveCard!).
- HyperCard is a card/scripting authoring system currently in version 2.4.1. It runs natively on both 68K and PowerMacintosh machines, and is widely used because of its easy availability at a low price. Its largest drawback is the lack of integrated color; current color implementation is via the ColorTools XCMD set (included) or via third-party XCMDs.
- HyperGASP : Caliban Mindwear.
- HyperGASP is a card/scripting authoring system currently in version 3.0; the newest version no longer requires HyperCard. Supports export to HTML for Web authoring.
- HyperStudio ; Roger Wagner Publishing ; Mac, Windows, WWW (via HyperStudio plug-in).
- HyperStudio is a card/scripting paradigm authoring system, optimized for and focussed on the educational market.
- IconAuthor : Asymetrix ; Windows, NT, Solaris, UNIX, WWW (via Windows).
- IconAuthor follows the iconic/flow control paradigm. It is notable for its SmartObject editor, which tags content files (still graphics, RTF text, etc.) for interactivity. It has the option to either embed content files or leave them external in specified directories. The biggest strength of this program is its included data handling, which makes it unparalleled for CBT data tracking. The latest version should also provide WWW porting of existing content. Avoid its internal "Move Object" path animation feature due to jerky response - use a .FLC or .AVI instead

Multimedia Programming vs Multimedia Authoring

- It should be noted that a distinction should be made between Programming and Authoring.
- Authoring involves the assembly and bringing together of Multimedia with possibly high level graphical interface design and some high level scripting.
- Programming involves low level assembly and construction and control of Multimedia and involves real languages like C and Java.
- Quicktime may also be programmed in C.

Key features that Multimedia Authoring System should provide

Authoring systems vary widely in orientation, capabilities, and learning curve. There is no such thing (at this time) as a completely point-and-click automated authoring system; some knowledge of heuristic thinking and algorithm design is necessary. Authoring is basically just a speeded-up form of programming --- VISUAL PROGRAMMING; you don't need to know the intricacies of a programming language, or worse, an API, but you do need to understand how programs work.

Paradigms of the Multimedia Authoring system

There are various paradigms, including:

Scripting Language

The Scripting paradigm is the authoring method closest in form to traditional programming. The paradigm is that of a programming language, which specifies (by filename) multimedia elements, sequencing, hotspots, synchronization, etc. A powerful, object-oriented scripting language is usually the centerpiece of such a system; in-program editing of elements (still graphics, video, audio, etc.) tends to be minimal or non-existent. Scripting languages do vary; check out how much the language is object-based or object-oriented. The scripting paradigm tends to be longer in development time (it takes longer to code an individual interaction), but generally more powerful interactivity is possible. Since most Scripting languages are interpreted, instead of compiled, the runtime speed gains over other authoring methods are minimal.

The media handling can vary widely; check out your system with your contributing package formats carefully. The Apple's HyperTalk for HyperCard, Asymetrix's OpenScript for ToolBook and Lingo scripting language of Macromedia Director are examples of a multimedia scripting language.

Here is an example lingo script to jump to a frame

```
global gNavSprite
on exitFrame
go the frame
play sprite gNavSprite
end
```

Iconic/Flow Control

This tends to be the speediest (in development time) authoring style; it is best suited for rapid prototyping and short-development time projects. Many of these tools are also optimized for developing Computer-Based Training (CBT). The core of the paradigm is the Icon Palette, containing the possible functions/interactions of a program, and the Flow Line, which shows the actual links between the icons. These programs tend to be the slowest runtimes, because each interaction carries with it all of its possible permutations; the higher end packages, such as Authorware or IconAuthor, are extremely powerful and suffer least from runtime speed problems.

Frame

The Frame paradigm is similar to the Iconic/Flow Control paradigm in that it usually incorporates an icon palette; however, the links drawn between icons are conceptual and do not always represent the actual flow of the program. This is a very fast development system, but requires a good auto-debugging function, as it is visually un-debuggable. The best of these have bundled compiled-language scripting, such as Quest (whose scripting language is C) or Apple Media Kit.

Card/Scripting

The Card/Scripting paradigm provides a great deal of power (via the incorporated scripting language) but suffers from the index-card structure. It is excellently suited for Hypertext applications, and supremely suited for navigation intensive (a la Cyan's "MYST" game) applications. Such programs are easily extensible via XCMDs and DLLs; they are widely used for shareware applications. The best applications allow all objects (including individual graphic elements) to be scripted; many entertainment applications are prototyped in a card/scripting system prior to compiled-language coding.

Cast/Score/Scripting

The Cast/Score/Scripting paradigm uses a music score as its primary authoring metaphor; the synchronous elements are shown in various horizontal tracks with simultaneity shown via the vertical columns. The true power of this metaphor lies in the ability to script the behavior of each of the cast members. The most popular member of this paradigm is Director, which is used in the creation of many commercial applications. These programs are best suited for animation-intensive or synchronized media applications; they are easily extensible to handle other functions (such as hypertext) via XOBJS, XCMDs, and DLLs.

NB: Macromedia Director uses this.

Hierarchical Object

The Hierarchical Object paradigm uses an object metaphor (like OOP) which is visually represented by embedded objects and iconic properties. Although the learning curve is non-trivial, the visual representation of objects can make very complicated constructions possible.

Hypermedia Linkage

The Hypermedia Linkage paradigm is similar to the Frame paradigm in that it shows conceptual links between elements; however, it lacks the Frame paradigm's visual linkage metaphor.

Tagging

The Tagging paradigm uses tags in text files (for instance, SGML/HTML, SMIL (Synchronized Media Integration Language), VRML, 3DML and WinHelp) to link pages, provide interactivity and integrate multimedia elements.

Techniques of formatting and delivering a content in a Multimedia Authoring System.

- i. Scripting (writing)
Standard Text --- say what you want with word
- ii. Graphics (illustrating)
“A picture is worth a thousand words” say what you want with a graphic illustration
- iii. Animation (wiggling)
Now we approach multimedia --- say what you want with a graphic animation or video
- iv. Audio (hearing)
Sounds can convey alerts, ambience and contents say what you want with a narration
- v. Interactivity (interacting)
True multimedia immerses yourself in an interactive presentation, possibly more instructive. Interactive actions can start animations, audio, move to new parts of presentation, control simulations etc.

Types of Audio in Multimedia Applications:

- i. Music - set the mood of the presentation, enhance the emotion, illustrate points
- ii. Sound effects - to make specific points, e.g., squeaky doors, explosions, wind, ...
- iii. Narration - most direct message, often effective

Types of Interactive Multimedia Applications:

- i. Menu driven programs/presentations often a hierarchical structure (main menu, sub-menus, ...)
- ii. Hypermedia

Less structured, cross-links between subsections of the same subject -> non-linear, quick access to information +: easier for introducing more multimedia features, e.g., more interesting "buttons".

- iii. Simulations / Performance-dependent Simulations e.g., Games - SimCity, Flight Simulators

Factors that affect the Quality of Service of a multimedia application

The QoS measure probably is the ultimate measure of a multimedia system.

The four essential parameters are:

- Bandwidth -- capacity of the transfer mechanism between source and destination.
- Delay -- the time a multimedia unit spends in transmission from source to destination.
- Delay Jitter -- Variation in delay delivery of data
- Loss Probability -- the ratio of units of information that an application can afford to lose.
- Hardware problems:

Network Connection

Traffic Analysis --- scheduling, buffer design, congestion control and synchronisation.

Multimedia data storage

Factors considered for storage facilities for Multimedia Systems:

- Large volume of data
- Real time delivery
- Data format
- Storage Medium
- Retrieval mechanisms

The RAID technology

RAID --- Redundant Array of Inexpensive Disks

RAID System:

Set of disk drives viewed by user as one or more logical drives

Data may be distributed across drives

Redundancy added in order to allow for disk failure

Offers:

Affordable alternative to mass storage

High throughput and reliability

Levels of RAID functionality:

Level 0 – Disk Striping --- distributing data across multiple drives

Level 1 – Disk Mirroring --- Fault tolerancing

Level 2 – Bit Interleaving and HEC Parity

Level 3 - Bit Interleaving with XOR Parity

Level 4 – Block Interleaving with XOR Parity

Level 5 - Block Interleaving with Parity Distribution

Level 6 – Fault Tolerant System --- Error recovery

Level 7 – Heterogeneous System --- Fast access across whole system