CHAPTER 12: FLOWCHARTS, STORYBOARDS AND RAPID PROTOTYPING

"What I hear, I forget; what I see, I remember; what I do, I understand." ---Confucius

---

Introduction

The tasks associated with the analysis and design phases of the systematic design process are fairly consistent across delivery systems. Whether designing instruction for delivery via interactive television (ITV), video, mail correspondence, the Web, or stand-alone computers, it is necessary to conduct *goal*, *instructional*, *learner* and *context analyses*, *define objectives*, *determine learner assessment methods*, *develop instructional strategies*, and *prepare instructional treatment plans*.

During the development phase, a variety of tools and techniques are utilized in conjunction with the delivery system. For example, with instructor-led ITV distance education, instructional technologists may create instructor guides (including PowerPoint presentations, lesson plans, and instructor notes), assessment instruments and student materials. On the other hand, instructional videos require developers to prepare shot sheets and scripts, and flowcharts, storyboards, and prototypes are generated for Web-based training and computer-based training (CBT).

In this Chapter you will learn how to develop instructional units by creating *flowcharts*, *storyboards*, and *working prototypes*. In this way you can illustrate the sequence of instructional content provided to target learners.
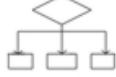

Flowcharting

Overview of Flowcharts

A *flowchart is a pictorial representation depicting the flow of steps in a program, people in an organization, or pages in a presentation*. In instructional design, flowcharts provide a bird's-eye view of the structure and sequence of a lesson. Elements are represented by simple icons (circles, rectangles, diamonds, or other shapes), with lines and arrows representing connections between events and the direction or order in which they occur. A flowchart indicates sequences and decision points as well as starting and stopping points. Since it is easier to grasp relationships in a visual form than in a verbal description, flowcharts can prevent the omission of steps in a process. Flowcharts are particularly useful for instructional designers who are novice or occasional programmers.

Flowcharting Conventions

While boxes, circles, diamonds and other symbols are not required to construct a flowchart, they convey information about the events in the process clearly and efficiently. Standard symbols that are applicable to most situations complex are described below:

Rounded Box		Represents an event that occurs automatically, triggering a subsequent action.
Rectangle or Box		Represents an event that is controlled within the process, typically a step or action taken. In most flowcharts this will be the most common symbol.
Diamond		Represents a decision point in the process, usually requiring a `yes' or `no' response, then branching to different parts of the flowchart.
Circle		Represents connection with another process. A reference to the new process should appear within the circle.
Directional Arrow		Indicates direction of flow. Horizontal arrows depict procedures or linear sequence of events, and vertical arrows depict sub- or super- ordinate skills and knowledge.
Program Decision - 3 Or More Options		Represents multiple choices for the user, with the chosen option determining the user's path through the remainder of the program.

Why Flowchart?

The ability to construct, read, and interpret flowcharts is a necessary skill for all instructional technologists. *Flowchart symbols are universally recognized across multiple disciplines, and will allow an instructional designer to communicate a great deal of information accurately, efficiently, and succinctly.* In addition, programmers and software and multimedia designers routinely use flowcharts to illustrate software capabilities, required user performance (input), and computer response (output). The high level of interactivity and branching involved in educational software in particular belies the need for instructional technologists to be comfortable with flowcharting protocol.

The following text conveys the same instructions as the flowchart that follows it. Which do you think provides a better understanding of the instructional flow?

Dial 0, then your friend's number. You will hear a tone. If you have a calling card number, dial your home phone number and the PIN number. If you don't have a calling card number, dial the 800 collect number. When the computer voice asks you to say your name, say your name. Wait for your friend to answer. If you have a calling card number, leave a message on your friend's answering machine. If you don't have a calling card number, hang up.
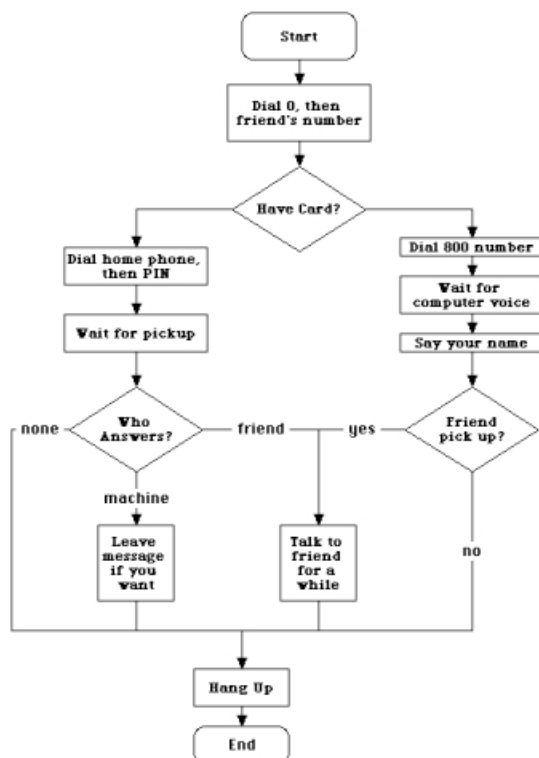
Drawing Flowcharts

There is not a rigid set of rules for constructing flowcharts, but the following six steps serve as guidelines to ensure consistency and accuracy. Note how each step has been applied in the example, and consider the implications for the different phases in the systematic design process.

1. Describe the process to be charted in a one-line statement. Example: "How to fill the car's petrol tank."

2. Start with the 'trigger' event that initiates the process. Example: "Low fuel warning light comes on."

3. Note each successive action in concise terms, avoiding ambiguous descriptions. Example: "Locate and drive into service station," etc.

4. When the flowchart branches into a complex number of options, choose the most important alternative and continue. Additional detail can be provided in other charts without subverting the efficiency of the main idea. Example: "Decision required: Quantity of fuel to be put into tank."

5. Make cross references to supporting information. Example: A cross-reference may be made to a list of cars requiring unleaded fuel, a table of preferred brands of fuel.

6. Continue describing each event, action, or decision as it occurs in sequence, until the process is concluded (a 'target' point). Example: "Fuel is paid for, tank is filled, customer exits the station."

How to fill the fuel tank
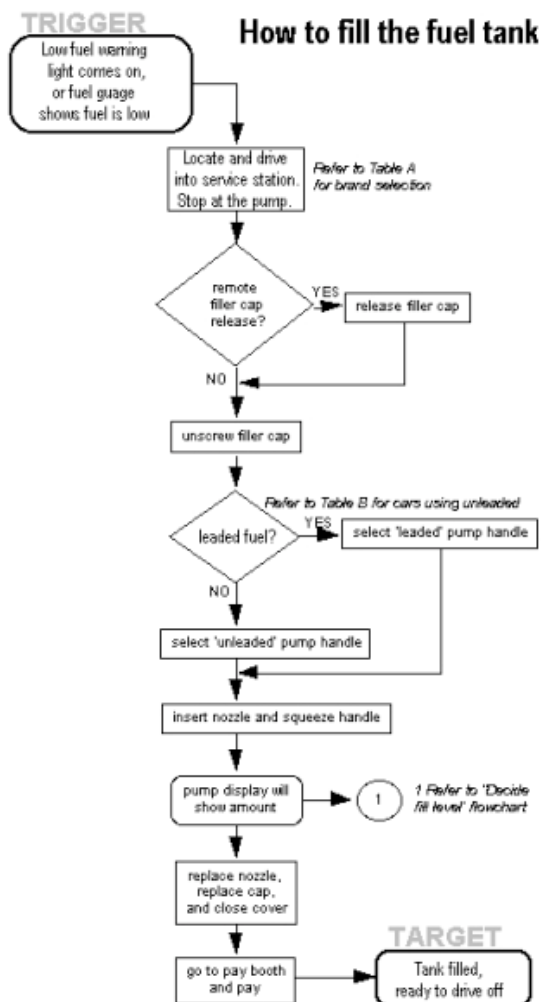
Flowcharting Software

Many think of flowcharts in terms of their original form, depictions of an algorithm's logic path, hand-drawn by computer programmers. However, a flowchart's usefulness goes well beyond mathematical functions. As a means of representing a process from beginning to end, whether that *process is an algorithm*, *procedure*, or *an instructional unit*, the tool once used exclusively by programmers has become a fundamental part of instructional design.

When flowcharts were constructed by hand, making changes was difficult and messy. As graphic programs became easier to manipulate, and the educational and business communities realized flowcharting software's potential, enterprising developers responded with a variety of packages for desktop computers.

Applications that include real-time or time-critical processes require dynamic process modeling. Programs such as ProcessCharter and Optima! allow users to create complex interactive models and observe how they perform over time. A process flowchart is created, and resources and calendars are added, culminating in a simulator running the process to indicate the movement of information.

For more information about flowcharting software, check the following Web sites:

CorelFlow	http://www.corel.com
Inspiration	http://www.inspiration.com
LotusNotes	http://www.lotus.com
Microsoft Project	http://www.microsoft.com
Primavera Project Planner	http://www.primavera.com
ProcessCharter	http://www.scitor.com
SmartDraw	http://www.smartdraw.com
TeamFlow	http://www.teamflow.com
Visio	http://www.microsoft.com/office/visio/

Guidelines for Flowcharting in Instructional Design

To flowchart an entire course, it would be necessary to establish a complete sequence of frames and a navigation scheme. Depending upon the scope of the course and the number of user options, this can be a time consuming process, requiring extensive organization and foresight. Without a clear and intuitive structure and navigation system, visitors to the site will have difficulty maneuvering and will either get lost or leave in frustration. One benefit of authoring systems is that this job is usually simplified by a pre-existing framework.
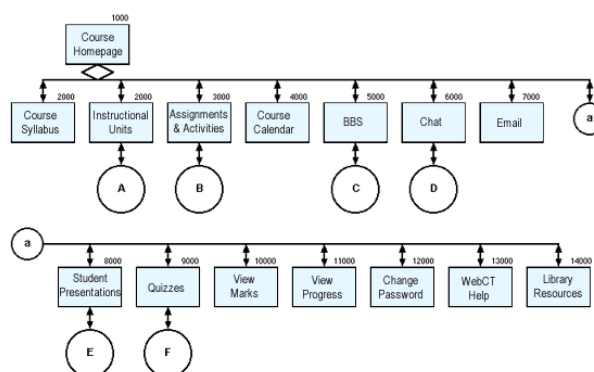

Figure 11.1: An example of Flowchart in Instructional Design

Distance learning instruction typically consists of multiple, linked pages, with a clear and intuitive navigation scheme. In creating flowcharts for instructional units, you examine the content to determine the number and sequence of sections,

using the events and interactions defined in the Instructional Treatment Plan (resuly of Goal Analysis). Each event or related group of events is usually represented by one box. As you construct your flowcharts, check for clarity and accuracy in the depiction of the sequence and interrelationship of the Web pages.
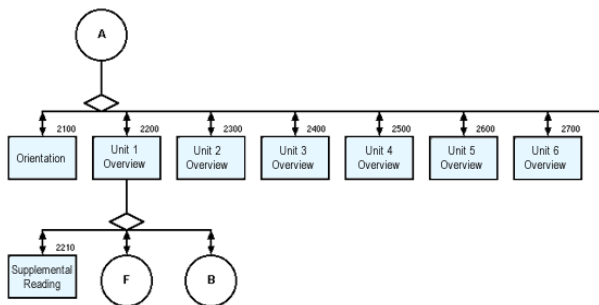

Figure 11.2: Example of Flowchart in Instructional Design

The flowcharting process should be informed by the specifications, events, and interactions identified in the instructional treatment plan and the instructional strategy. In the sample instructional strategy created in Table 11.1, an additional column has been added to illustrate the grouping of information and events. Typical courses would also include separate pages for each event as indicated in the "Media Selection" column.
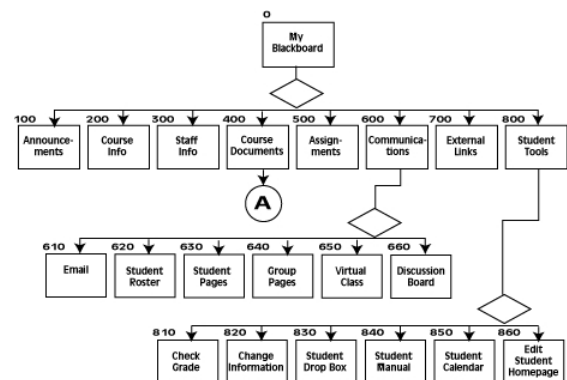

Figure 11.3: Example of Flowchart in Virtual School software

Page	Event	Description	Interaction	Media Selection
1	Provide Task Overview	*Set analysis in context of previous tasks and systematic design process. Describe analysis tasks, provide rationale for completing tasks and delineate time requirements.*	Learner-Content	Web Page
1	Inform Learners of Objectives	*List terminal and enabling objectives.*	Learner-Content	Web Page
1	Inform Learners of Prerequisites	*Describe desired prior skills and knowledge related to systematic design (goal, instructional, learner and context analyses). Also list required knowledge or access subject matter expertise, target audience and context.*	Learner-Content	Web Page
1	List Required Materials	*List Required Materials.* 1. Read Dick, Carey and Carey (2000) Chapter 1 Introduction to Instructional Design (p. 2-13). 2. Alternative forms of performance support for each type of analysis; and 3. Training modules provided for each analysis.	Learner-Content	Web Page
1	Describe Process including use of Performance Support Interface	*Inform learners to start by reviewing the performance criteria, job aids and some examples associated with the task. Tell learners that if they feel confident in their ability to complete the task and meet the performance criteria, they should go directly to the assignment. However, if learners are not confident in their ability to complete the task, suggest that they discuss the task with others (using the communications link) and/or go through the available training.*	Learner-Content	Web Page
1	Provide Direct Access to Performance Support Tools	*Present learners with direct links to:* • Task • Performance Criteria • Examples • Goal statement and analysis chart • Instructional analysis map	Learner-Content	Web Page E-Mail BBS

|				
		- Learner analysis report - Context analysis report - Communication links - Email to instructor - Email to follow classmates - Email to others - Course BBS - Job Aids - Job aid for clarifying fuzzy goals - Flowcharting conventions - Learner analysis template - Context analysis template - Training - Goal statement & analysis - Instructional analysis - Learner analysis - Context analysis		
1	Promote Social Interaction	*If learners are working in a group, instruct them to review and discuss the individual analysis reports generated by teammates and determine process for generating and posting one report for your team.*	Learner-Learner	E-Mail Student Web Sites
1	Promote Social Interaction	*Inform learners to post at least one question or comment about goal, instructional, learner or context analyses and respond to at least one other posting in the appropriate location of the course bulletin board system.*	Learner-Learner	BBS
2	Elicit Performance	*Instruct learners to complete assignment and provide directions for submitting and/or posting their work.*	Learner-Instructor	Web Page Student Web Sites
3	Performance Support	*Present performance criteria for assessing the quality of completed assignments.*	Learner-Content	Web Page
20	Assess Performance & Provide Feedback	*Assess learners' assignments and provide feedback based on performance criteria as soon as possible.*	Learner-Instructor	Student Web Sites Email

Table 11.1 - Instructional Strategy Example

Flowcharts should depict the sequence of the instruction in the instructional unit in a clear and unambiguous manner. Figure 11.4(a) illustrates frames or pages presented to students in a linear format, while Figure 11.4(b) depicts a non-linear navigation scheme. A linear format decreases the chances that students will accidentally miss or skip information. However it also restricts students to viewing and reviewing content in a particular order and format. Non-linear navigation provides students with greater flexibility, but requires careful planning to ensure that information is divided into logical and meaningful "chunks." It is also important to ensure that students can navigate through all of the pages without getting lost or overwhelmed by the sheer volume of information.

Figure 11.4(a): Linear navigation	*Figure 11.4(b): Non-linear navigation*
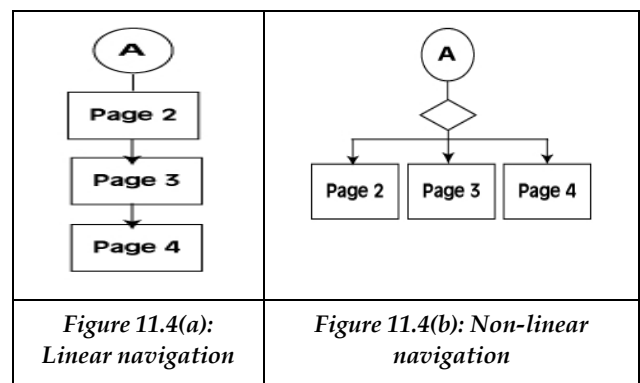
Organizing content into hierarchies (Figure 11.5) facilitates effective design and management of information. With a typical, multiple page site, determining the most appropriate way for students to progress through the information is critical. A thoughtful analysis of the flowchart provides perspective on the balance of depth (the number of levels of content) and breadth (the number of aspects or topics covered) of the content.
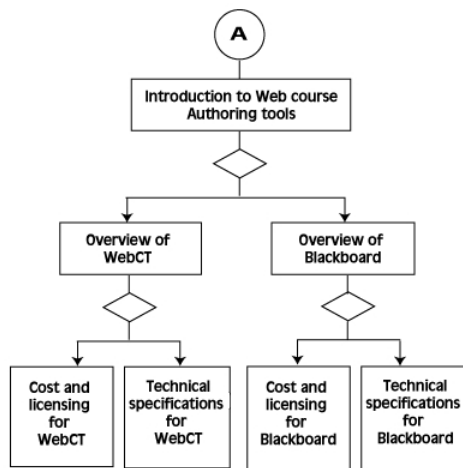
Figure 11.5: Example of a content hierarchy

Summary

Since humans are primarily a visual species, a picture of a process conveys information more efficiently than a written or verbal description. Making a process visible also renders it easier to manage and captures the subtle interactions among components. Flowcharts identify all key tasks, and the completed chart can be used:

- as a springboard for further discussion of the process;

- to connect with other flowcharts explaining related activities;

- to identify points where data can be usefully collected and analyzed;

- to isolate possible problem areas; and

- to communicate the process to those unfamiliar with it.

Producing a clear and concise flowchart begins with a brief overview of the process and progresses to the level of detail necessary for your instructional unit. The complexity of the lesson determines the level of detail to include; simple tutorials require little detail, while complex lessons such as simulations require a significant degree of detail. In general, the more time and effort devoted to constructing flowcharts, the less time and effort required correcting errors later.

Storyboarding

Overview of Storyboarding

Storyboarding provides a rough outline of how the presentation will appear, including a conceptual idea of the location of images and text, related links, and general layout. It provides a simple means of creating a visual representation of the user interface developed for each page, illustrating its relationship to the overall scheme.

Artistic ability is not a prerequisite for producing a storyboard. Sketching in outlines and simple shapes with explanations such as, "This image, top right hand corner, heading, level 2 and centered," communicates all necessary information. Files are named in relationship to the topic or page (e.g., "intro1.jpg"). Copy, image style, and size of graphics should be indicated for clarity, efficiency, and consistency. The developers should be able to take your storyboard and put together the same instructional site you would produce. With a large unit, clear storyboards are essential, as the work will often be delegated to other team members. When the portions are assembled, pages should exhibit consistent design despite having different authors. Storyboarding is a time-consuming process, with one study stating that approximately 19% of development time is spent creating and revising storyboards.

Storyboards for Interactive Multimedia

In the past, selecting a storyboard format was easy - there was only one choice! However, traditional storyboard formats are inadequate for designing today's interactive training.

Figure 11.6: Traditional, linear storyboard used primarily for video production

Figure 11.7: Example of Super Media Storyboard

With the proliferation of highly interactive multimedia courses and presentations, the two column format is no longer sufficient. Current and future multimedia training and education deal with more than audio and/or video. Today's courseware contains dynamic interactions, such as animated technical drawings, learner control of video sequences, click-and-drag exercises, matching games, role-playing scenarios, virtual world navigation, and much more.

The inadequacies of the two-column storyboard (script) prompted developers of interactive training to adopt the concept of the "super" storyboard. Using this approach, a document is created containing detailed information on each student interaction, with one full page dedicated to each. Super storyboards contained information about every media element and programming logic used in the interaction as well as a visual representation of the screen's appearance. These storyboard forms contain dozens of fields related to media development and multimedia authoring, including event name, graphic, and video description, video shot list information (location, subject, props), SMPTE time code, on-screen touch areas, branching logic, programming notes, and many other related fields. Designers reasoned that comprehensive storyboards containing this degree of detail would produce forms and templates that would hasten the scripting process. Instructional developers complete the storyboard process by filling in the appropriate fields.

Who Uses Storyboards?

The three primary users of storyboards are developers, graphic artists, and SMEs (subject matter experts). Storyboards provide inclusive programming instructions for the developer. Unless a separate list of graphics needed for production is provided, graphic designers will review storyboards to ascertain the number, location, and type of graphics needed for each screen or as part of animation.
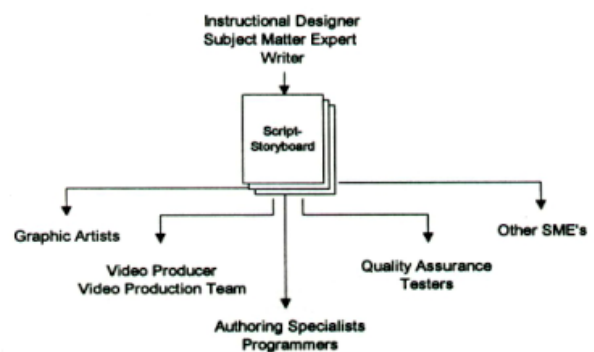

Figure 11.8: Team Members Reviewing Storyboards

For SMEs, storyboards are the main deliverable to be reviewed and approved before building begins. It is vital that storyboards are clear and unambiguous to ensure that SMEs gain an accurate understanding of the proposed product.

In addition to communicating information, storyboards also serve as documentation, facilitating changes and revisions.

Storyboarding Process

Seven steps for storyboarding:

1. Write and revise primary text

2. Write and revise secondary text

3. Produce storyboards

4. Check the fit of overlaying displays

5. Draw and revise graphic displays and plan other output

6. Check graphics and simultaneous text for fit

7. Review the flowcharts and storyboards

Step 1. Write and revise primary text

In this step, the text that students will see in the lesson is written. The primary text contains the essential instructional content. In tutorials, primary text usually contains new information, such as definitions, descriptions, and principles, the questions asked and the feedback given after responses. In drills, the primary displays are stimuli to which students respond and the feedback they receive.

Primary text displays do not include directions, help messages, hints, or cues. Observe the following guidelines while writing primary text information:

- Keep presentations concise and to the point.

- Use formatting features to emphasize what is important. Bolding, underlining, or coloring text draws the user's attention to the information, but also be careful not to overuse these features.

- Use analogies and metaphors, and incorporate frequent active student participation to reinforce understanding.

- Organize the content so that it is clear.

- Arrange information in lists when appropriate and state applicable rules.

- Include brief but obvious statements of transition when changing the topic or moving from one point to another.

- Be attentive to grammar, spelling, and sentence structure.

- Maintain a general writing style and choice of vocabulary that is appropriate to your intended audience's experience and reading level.

- Vary the types of questions within the instruction (multiple choice, matching, completion) and ensure that questions are concise and the method of response is clear to students.

- Compose feedback that is clear, informative, and positive for both correct and incorrect responses.

- Feedback for incorrect responses should not be more entertaining than feedback for correct responses

Step 2. Write and revise secondary text

Secondary text messages are written after the primary text has been completed, and consist of directions, menus, transitions, prompts, hints, review material, help, score and progress information, and lesson exit messages. Secondary text messages should also be clear, concise, and accurate.

Step 3. Produce storyboards

At this point, both primary and secondary texts are in draft form. While ordinary 8 1/2 by 11 paper can accommodate about eighty characters per line and sixty-six lines, most computer screens have between one-half to one-sixth the display capacity of a sheet of paper. Primary and secondary text displays must therefore be designed to accommodate the capacity of computer displays. Dividing text into smaller segments will also provide opportunities for student interaction with the instructional content by inserting questions, displaying pictures, and allowing navigational flexibility.

Storyboarding grids are created on paper that corresponds to the size of a computer display, with horizontal and vertical lines depicting dimensions across and down the page (Figure 11.9).

Figure 11.9: Storyboarding Grid

Using a storyboard grid provides ease in centering and spacing text and restricts the amount of text placed on each line. Graphic displays can be drawn on paper that has been

scaled down, but for maximum clarity it is best to use the same dimensions as your computer screen.

It is often unwieldy to assemble the entire collection of storyboard grids developed for a course or program, and with computers that use proportional fonts (e.g., Macintosh), grids are problematic. Designers often substitute index cards, placing an entire lesson on a bulletin board for evaluation.

The usefulness of storyboard grids is further limited by the advent of scrolling text fields and frames. Virtually unlimited amounts of text may be displayed in one screen. If more text than can be viewed on a single computer display or storyboard frame is to be included, additional pages of text information must be attached to the storyboard.

Step 4. Check the fit of overlaying displays

Many of the separate storyboards will be "overlaid" on single computer displays. A common example occurs when a student is asked a question. One storyboard contains the question as well as the space in which the student will type a response. Another storyboard contains a feedback message presented when the student gives an incorrect response. During the lesson presentation these two displays may appear on the screen at the same time. However, they are not written on the same storyboard because:

- Different messages appear on the screen depending on whether the student answers the questions correctly or incorrectly. Instead they are placed in identical locations on separate storyboards, ensuring that each message will fit simultaneously with the original question.

- The same message may be used as feedback for different questions.

- Establishing a one-to-one correspondence between storyboards and segments of a lesson, such as subroutines is advantageous.

- Since computer displays within a lesson are frequently programmed separately, then assembled, it is best to storyboard on different cards.

Storyboard overlays occur primarily in question-answer-feedback displays, user control messages, and directions or help message displays. They also occur in incremental text, text that is not presented on the screen all at once, but is added a little at a time, and in replaced text, in which a portion of text is erased from a larger display and replaced with different text. Using storyboards that have grids and are all the same size, or storyboarding on the computer makes checking for overcrowded or overlapped text simple.

Step 5. Draw and revise graphic displays and plan other output

"Graphic displays" refers to a variety of non-text presentations, including line drawings, cartoons, animations, geometric figures, and bar and line graphs. Observe the following guidelines in storyboarding graphic elements:

- Use graphics to present or elaborate important information.

- Avoid unnecessary detail or ambiguity.

- Use color sparingly, to distinguish or emphasize portions of pictures or graphs; avoid using too many colors simultaneously.

- Always allot space for simultaneous text.

Animations require special consideration because it is difficult to capture the nature of movement on paper. In addition to producing the essential elements in still form, make marginal notations that include a description of which elements change and the nature of the change, such as direction, size, distance, and rate of change. Drawing a few still pictures to show the change at different stages is often helpful. Other output consists of any presentation that does not appear on the computer screen, most frequently referring to sound. Sound requires special consideration because it is difficult to describe on paper. A storyboard for sound would consist entirely of marginal information describing the sound in terms of its nature (e.g., music, speech), content, and duration.

Step 6. Check graphics and simultaneous text for fit

Graphic displays, like text displays, are sometimes overlaid with one another or with other text. As with text, check that there is no overlapping or overcrowding in the composite displays, and that graphics and corresponding text are displayed simultaneously. Superimposing one storyboards on another to check for fit is a typical feature of graphics software. If overlapping or overcrowding exists, corrective

revisions can involve changing the proximity, size, or detail of graphics.

| Step 7. Review the flowcharts and storyboards |

- Assemble the storyboards in the order a student would encounter them in the lesson, as indicated by the flowcharts, placing them side by side or tacking them on a bulletin board so they can be viewed all at once.

- Review flowcharts and storyboard layout for sequence, style, completeness, student control, and length.

- Assume the role of an average student. Proceed from card to card, interacting with the lesson, answering each question correctly and incorrectly, encountering the appropriate presentations, choices, questions, hints, feedback messages, and remediation.

- Anticipate any problems and mark for revision.

The most frequently encountered problems include:

- Missing or incomplete directions

- Insufficient student interaction

- Inadequate coverage of topics

- Overlapping, overcrowded, or poorly spaced displays

- Redundant or irrelevant displays

- Question loops in which students may get stuck

- Poor transitions

- Poor student control, such as displays that cannot be reviewed

- Text passages with no graphic enhancements

Flowcharts and storyboards should be reviewed by other instructional designers, content experts, your client, and programming professionals to assess pedagogical quality, correctness of content and sequence, attainment of objectives, aesthetics, hardware and software suitability. Students should also review the documents to assess level of difficulty, clarity of text and pictures, and overall impression of the lesson. After synthesizing the feedback from reviewers, revise the documents.

Tools for Creating Storyboards

Depending on its complexity, a single application may require a number of storyboards. For example, a game would include multiple changes of direction and action by the user, multiple graphics, and multiple representations and feedback. In a non-sequential instructional unit, the user chooses any of multiple destinations and paths with a mouse click. In an application produced with a multimedia-authoring package such as Toolbook, a different background could represent each of the storyboards. The selection of storyboarding tools reflects the level of complexity required. Options include:

Word processor. Many designers use the drawing function in Windows or MacOS to rough out graphics and locations of various elements, including text.

Graphics application. A graphics application can be used to draw simple mock-ups of the screens with areas designated for additional notes.

Database management application. In complex applications (e.g., military simulations), the cost of mistakes can be serious. Databases are constructed with detailed explanations of components and with forms and reporting tools providing the content to users. In organizations that generate large numbers of applications, this can be a cost-effective approach

Storyboarding Software

This is a partial list of software applications that help developers produce storyboards (Note: Multimedia authoring packages can also produce quick designs to be printed out and annotated).

Canvas	http://www.deneba.com
Scriptwerx	http://www.originalvision.com
IVD Toolkit	http://www.electronicvision.com
StoryBoard Quick	http://www.powerproduction.com

Summary

Storyboards are the blueprints of the interactive courseware design and development process. The primary purpose of storyboarding is to communicate the intended design to each member of the team and provide documentation throughout development. The effectiveness of the communication and documentation is in direct relationship to the level of detail included in the

storyboards. Ensuring completion and accuracy of the storyboarding process minimizes confusion and errors that result in delays and major reworking.

Rapid Prototyping

Overview of Rapid Prototyping

In designing technology-based instruction, it is an efficient practice to use storyboards and flowcharts that are indicative of the content, layout, and organization of the unit to develop a small portion of the program. The representative portion is tested, evaluated, and revised before the resources necessary to produce a complete course are expended. This process of developing a sample to facilitate the identification of organization or design problems is called rapid prototyping.

The goal of prototyping is to provide a workable example of features of the system to users, decreasing the complexity of performance by focusing on limited components. There are two types of prototypes: horizontal and vertical (Figure 11.10). Horizontal prototypes present users with a wide range of program features that are not fully functional. Vertical prototypes present users with a small number of fully functional program features.
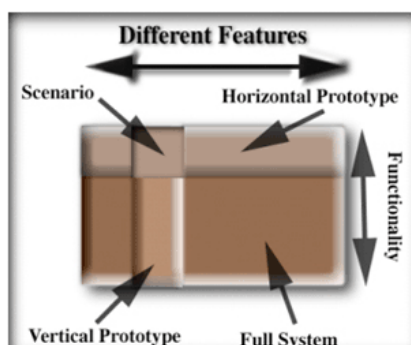
Figure 11.10: Vertical And Horizontal Prototype Structure

Horizontal Prototyping

A horizontal prototype would include program features accessible from the homepage, allowing developers to test the intuitiveness of icons and user interface features.

Vertical Prototyping

Vertical prototypes include one or more fully functional course features, as illustrated in Figure

11.11. Producing a vertical prototype of this course would include all aspects of one instructional unit and its related components.

Figure 11.11: Vertical Prototype Structure

Course developers test the specific features of the user interface before beginning full-scale production. The vertical prototype serves as a template for all other instructional units, establishing consistent formatting and navigation standards for the entire course.

There are many possibilities of formatting and navigation standards. For example, the designer might specify:

- Level one headings are 18 point, bold, Arial font

- Embedded links are 12 point, plain, Arial font

- Pages contain a link back to home page and/or

- Assignment pages have links back to Overview pages