

Chapter 1

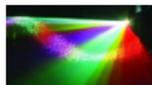
Background

The Design and Implementation of Multimedia Software

David Bernstein

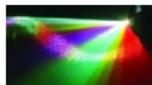
Jones and Bartlett Publishers

www.jbpub.com



About this Chapter

- The **title** of this book will attract a lot of attention because an enormous number of people are interested in multimedia, and many of them are interested in using multimedia to software design and create multimedia content.
- The **book** will only be of interest people who want to design multimedia software.
- This chapter is about the difference between the two.



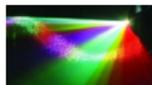
Definitions

Definition

Software Product Design is the process of specifying software product features, capabilities and interfaces to satisfy client needs and desires.

Definition

Software Engineering Design is the process of specifying programs and subsystems, and their constituent parts and workings, to meet software product specifications.



An Example - A Role-Playnig Game

- Product Design:

- How the user controls play

- When games can be saved

- When and how ((e.g., visual or auditory) scores are presented

- How the game transitions between levels

- ⋮

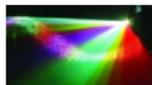
- Engineering Design:

- How the player's location is managed

- How artifacts (e.g., weapons, treasure) are managed (e.g., carried by the player, state changes)

- How health/power/etc.... levels are stored and changed

- ⋮



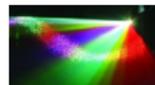
Borrowed Terms

Definition

A *system* is a set of entities (including their attributes) and the relationships between them.

An Observation

Every system can be defined in many ways. Deciding which is best involves deciding that some things are important and others can be ignored, a process that is known as *abstraction*.



Borrowed Terms

Definition

A *system* is a set of entities (including their attributes) and the relationships between them.

An Observation

Every system can be defined in many ways. Deciding which is best involves deciding that some things are important and others can be ignored, a process that is known as *abstraction*.



More Borrowed Terms

Definition

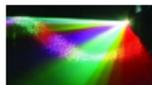
The *environment/context/domain* is the set of all other systems.

Definition

A *subsystem* is a part of the whole system that, in and of itself, display a richness of interrelationships.

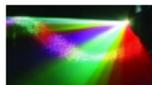
Definition

Elements/atoms are the smallest parts of the system .



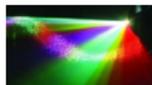
The Activities

- The identification of goals, objectives and constraints;
- The generation of alternatives;
- The evaluation of alternatives; and
- The selection of an alternative.



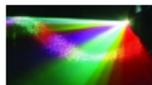
Alternatives to Object-Oriented Design

- *Data Driven Design* first identifies the important data and then identifies the processes that manipulate these data.
- *Function Driven Design* first identifies the important processes and then identifies the data that are manipulated by these processes.



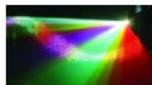
Characteristics of Object-Oriented Design

- Responsibility Driven:
Object-oriented engineering involves thinking about real-world entities/concepts and their responsibilities.
- Neither Data Driven Nor Function Driven:
Object-oriented engineering design involves the consideration of data and functions simultaneously.



The Process

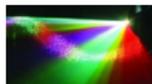
1. Identify important concepts (like physical objects, places organizations, events, records, containers, roles of people, descriptions, specifications, rules, and processes)
2. Identify the important associations between concepts (like part-of, contained-in, description-of, member-of, uses/manages/controls, communicates-with, related-to, owned-by, and near/far-from/above/below/etc)
3. Combine the attributes and behaviors that define a concept into a class



The Process (cont.)

Definition

The process of combining the attributes and behaviors that define a concept into a program unit is called *encapsulation*.



Defining Sets

Extension

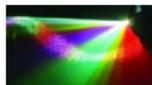
A list of all of the elements in the set. So, for example, one can define the set S as follows:

$$S = \{1, 2, 3, 4\} \quad (1)$$

Intension

A description of the elements in the set. So, for example, one can define the set T as follows:

$$T = \{x : 0 < x < 5, x \text{ is an integer}\} \quad (2)$$



Set Theory and Object-Oriented Design

- A *class* is an intensive definition of a set
- An *object* (of class C) is a member of the set (defined by C) (and is said to be an *instance* of a class)



Static and Non-Static Members

- A *static* attribute (or class variable) is an attribute of the set.
- A *non-static* attribute (or instance variable) is an attribute of each member of the set.



Semiotics – The Study of Signs

Definition

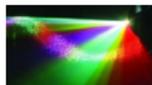
A *sign* is anything that someone interprets as signifying something.

Definition

A *token* is a sign that represents by way of its particular place in time and space.

Definition

A *type* is a sign that represents a class or set.



Signs

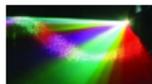
- A *signifier* is the form that the sign takes (which is something that can be sensed using one of the five traditional senses).
- The *signified* is the concept that the sign represents.



The Roots

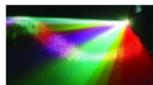
multi is from the Latin word *multus* which means “numerous”;
and

media is from the Latin word *medium* which means “center”.



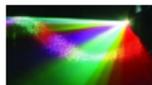
Uses of the Term “Medium”

- Physics – a sequence of interacting particles (e.g., air is a medium through which sound is transmitted)
- Biology – the substance in which an organism lives (e.g., agar in a Petri dish)
- Chemistry – a substance used for filtering (e.g., filter paper)
- Communications – a means of mass communication (e.g., radio and television are both media)
- Computer Engineering – a device/object on which data are stored (e.g., a magnetic disk)
- Art – both the materials used in a technique (e.g., oils versus water colors) and solvents (e.g., paint thinner)



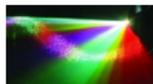
Towards a Definition

- The word “medium” seems to be used in the sense of the words “intermediary” or “means”.
- Thus, a reasonable definition of the word “multimedia” is “involving multiple methods, means or intermediaries”.



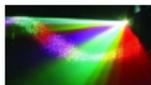
Common Terms

- Text
- Graphics
- Images
- Animation
- Video
- Sounds
- Music
- Audio



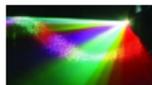
Problems with the Common Terms

- Some Are Sense-Specific and Some Are Not:
“Images” have a visual signifier, “sounds” have an auditory signifier, “text” can have either.
- Different Levels of Abstraction are Used:
“Image” refers to a type and a particular image is a token
“Text” is a type and a particular piece of text is also a type. A token doesn’t exist until the text is set in a font, size, color. . .



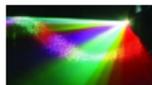
Problems with the Common Terms (cont.)

- Some Definitions are Unclear:
Are “images” and “graphics” the same?
- Some Important Distinctions are Ignored:
“Music” often means both “recorded music” and “musical score”.



Characterizing Content

- The Sense Used to Perceive It:
 - Visual
 - Auditory
- Temporal Nature of the Content:
 - Static
 - Dynamic
- Representation:
 - Sampled
 - Described



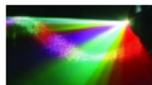
Using the Characterization

Definition

A *medium* is a means of representing, storing, transmitting, or presenting information that is perceived using a particular sense.

Definition

Multimedia software is software that uses more than one medium.



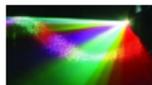
Using the Characterization

Definition

A *medium* is a means of representing, storing, transmitting, or presenting information that is perceived using a particular sense.

Definition

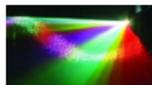
Multimedia software is software that uses more than one medium.



A Look Ahead

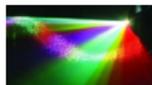
It is important to understand how auditory and visual information, are perceived, beginning with the the physics of sound and light.

To understand the physics of sound and light it is first necessary to understand waves.

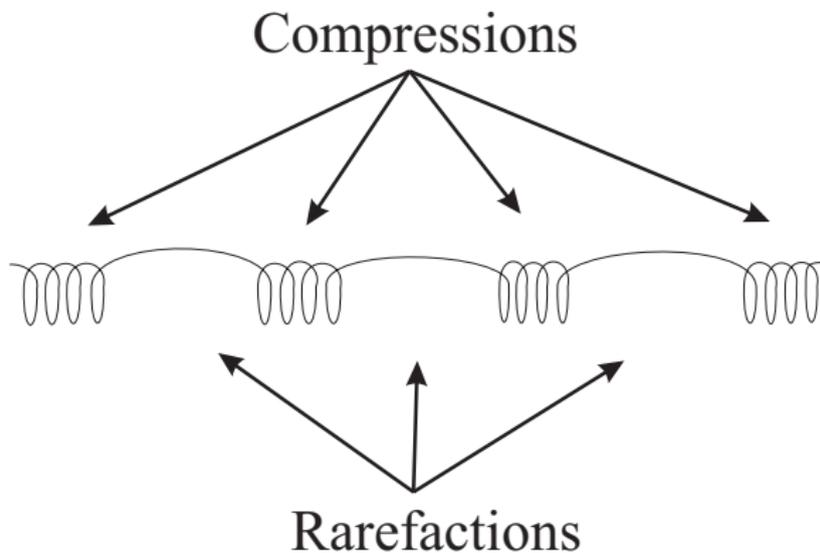


Getting Started

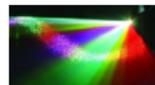
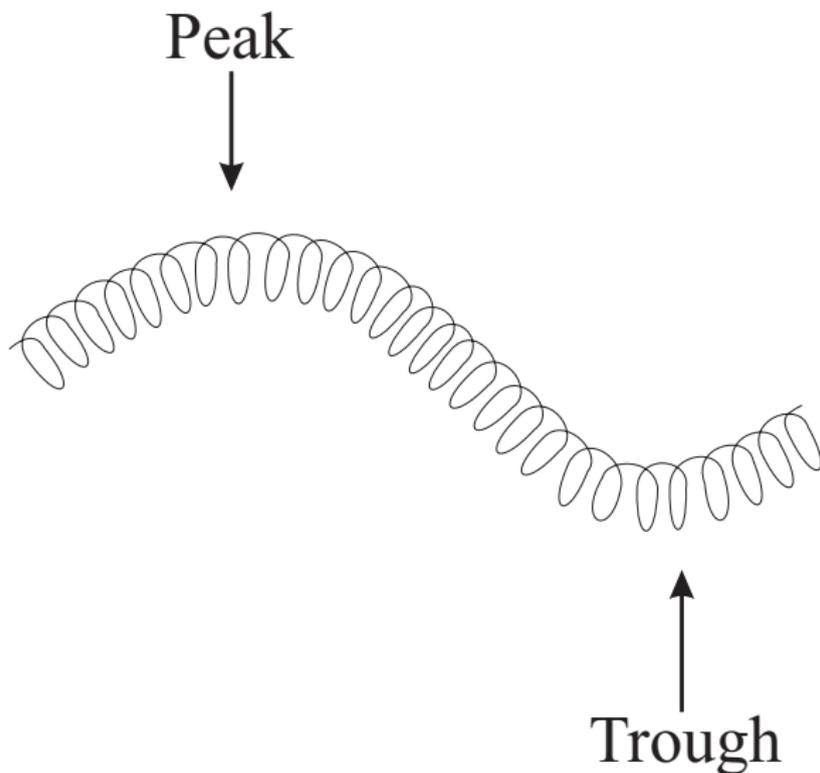
- Mechanical *Pulse*:
A single disturbance that moves through a a sequence of interacting particles (called, unfortunately, a medium).
- Mechanical *Periodic Wave*:
A periodic (i.e., evenly ‘timed/spaced’) disturbance that moves through a medium, transporting energy as it moves.



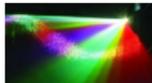
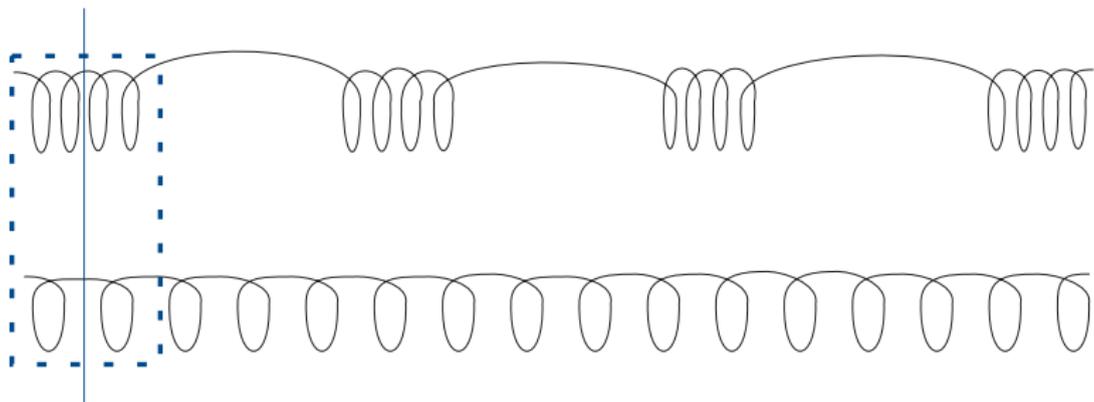
A Longitudinal Wave



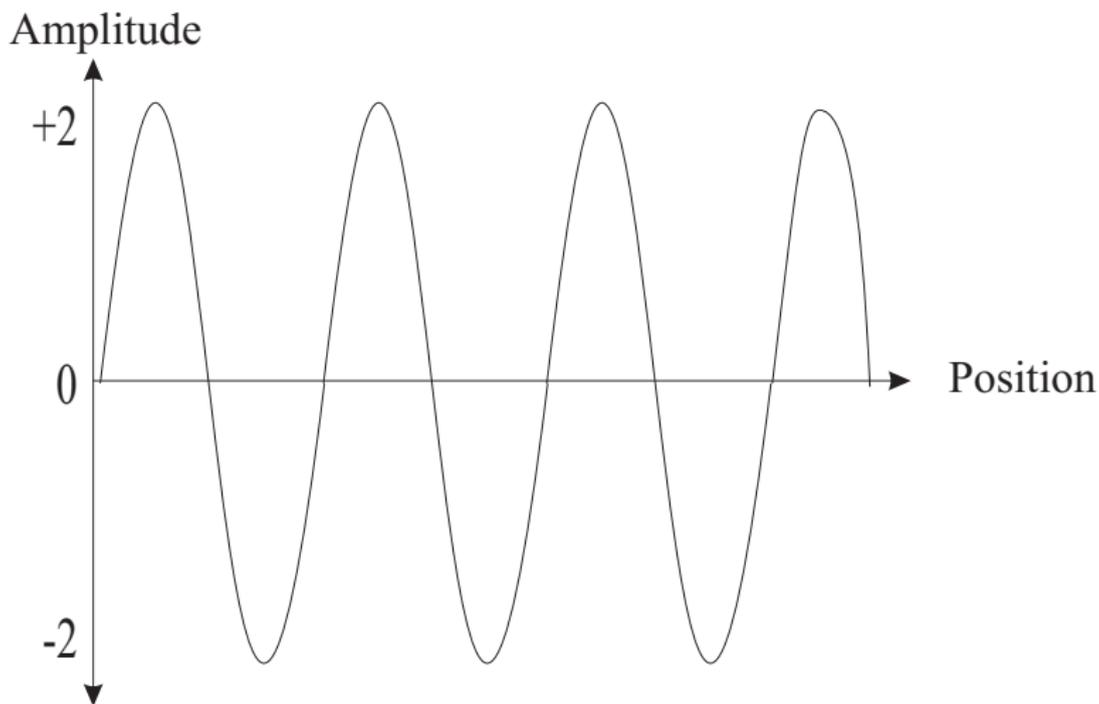
A Transverse Wave



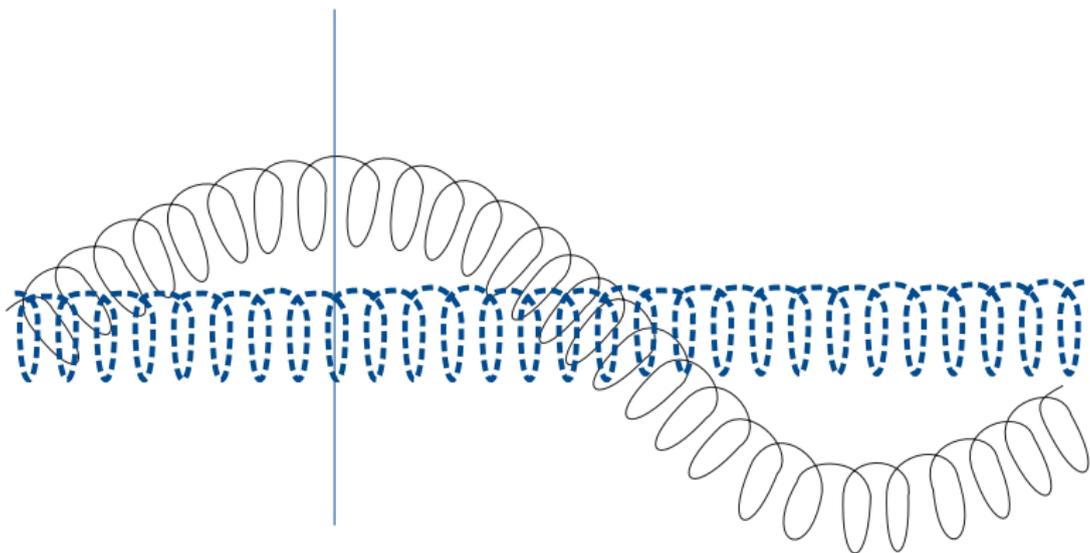
A Spring: Moving Longitudinally and at Rest



Amplitude versus Position for a Longitudinal Wave

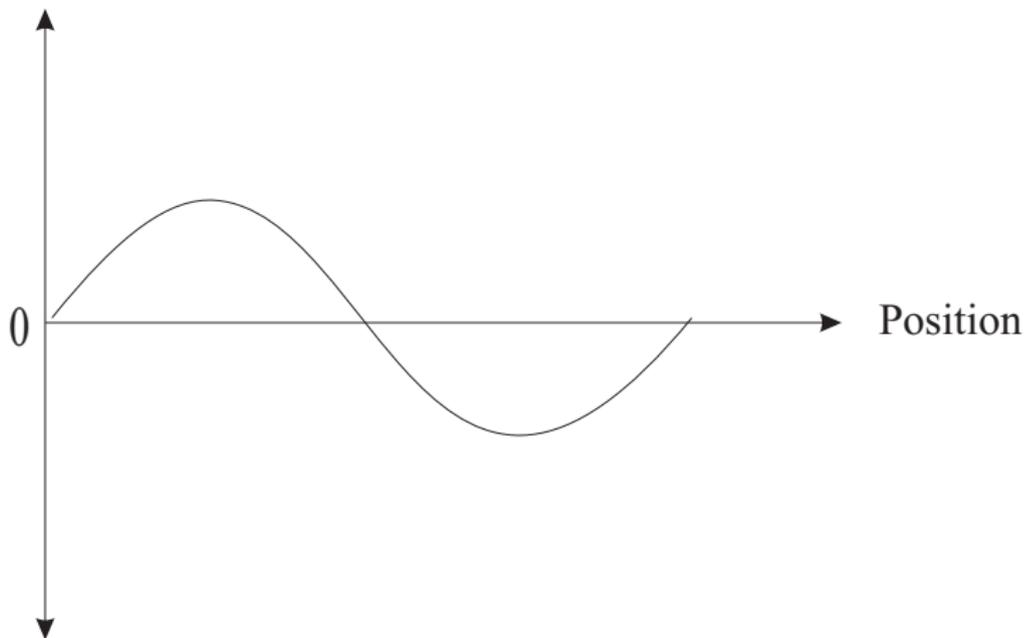


A Spring: Moving Transversely and at Rest



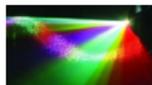
Amplitude versus Position for a Transverse Wave

Amplitude

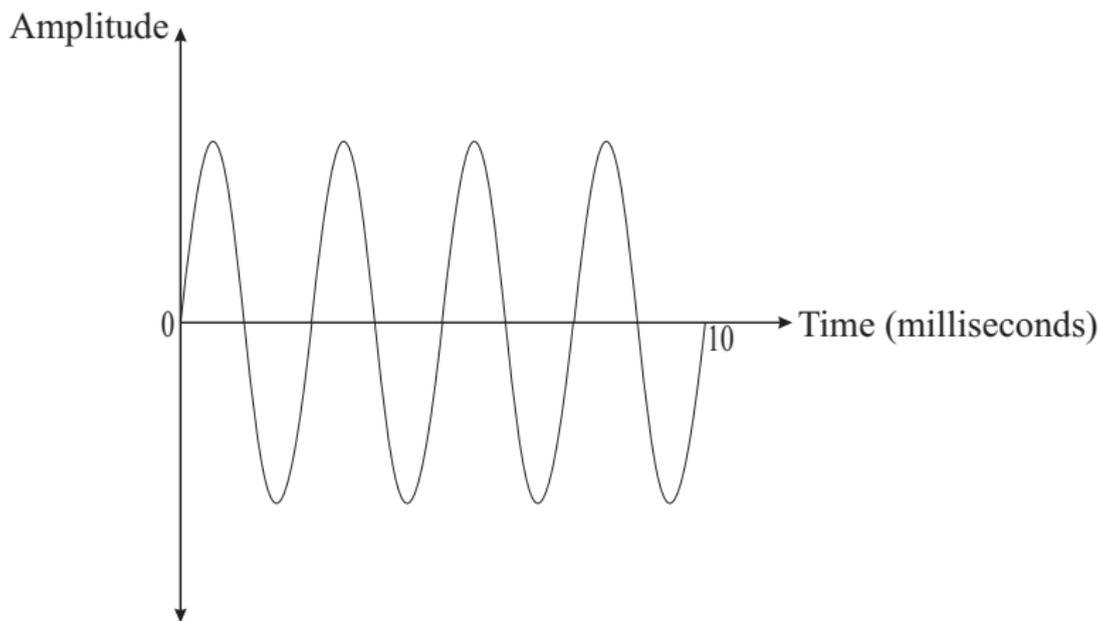


Interference

- *Principle of Superposition:*
When two waves interfere, the resulting displacement of the medium at any location is the algebraic sum of the displacements of the individual waves at that same location.
- *Constructive Interference:*
The displacements are both positive.
- *Destructive Interference:*
One displacement is positive when the other is negative.

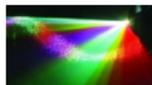


Amplitude versus Time



Terminology in the Time Domain

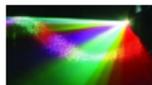
- A *cycle* is a portion of a wave from rest to crest to rest to trough to rest.
- The *period* is the time required for a cycle (measured in seconds per cycle).
- The *frequency*, denoted by f , is the reciprocal of the period and, hence, is measured in cycles per second (i.e., *hertz*).



Speed/Velocity

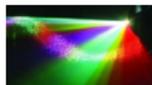
The *speed* of a wave, denoted by v , is the product of its wavelength and frequency:

$$v = \lambda f \quad (3)$$



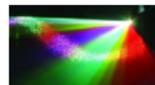
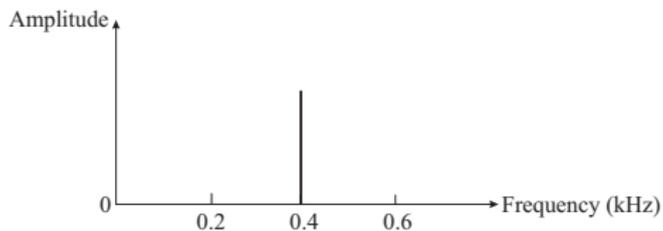
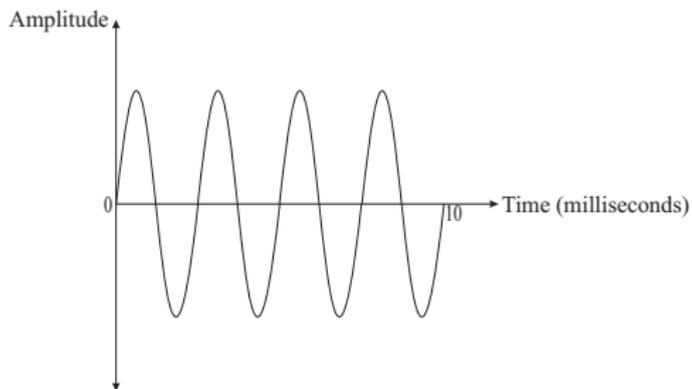
Different Spectra

- *Line Spectrum:*
The wave is strictly periodic.
- *Harmonic Spectrum:*
The wave is quasiperiodic.
- *Continuous Spectrum:*
The wave is aperiodic.



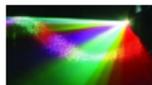
A Wave in the Time and Frequency Domains

A 400Hz Periodic Wave



Not Considered

- Software Product Design (e.g., user interface design issues)
- Multimedia Product Design (e.g., usability issues)
- Content Development (e.g., art, music, scripts)
- Delivery Media (e.g., CD/DVD Production)



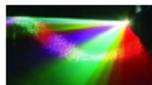
Considered: Higher Level Engineering Design

- Characterizing content based on whether it is auditory or visual, static or dynamic, and sampled or described.
- Designing software components that encapsulate different kinds of content.



Characteristics of Good Designs

1. *Adequate* - Meets all of the requirements and satisfies all of the constraints.
2. *Rugged* - Has a low probability of failure under normal conditions (i.e., *reliable*), is able to operate under a wide variety of conditions (i.e., is *robust*), and minimizes the damage that results from failure (i.e., is *safe*).
3. Easy to repair and enhance.
4. Easy to understand/document and use.



Things That Help

- Use of *Information Hiding*
- Minimization of *Coupling*
- Maximization of *Cohesion*

