

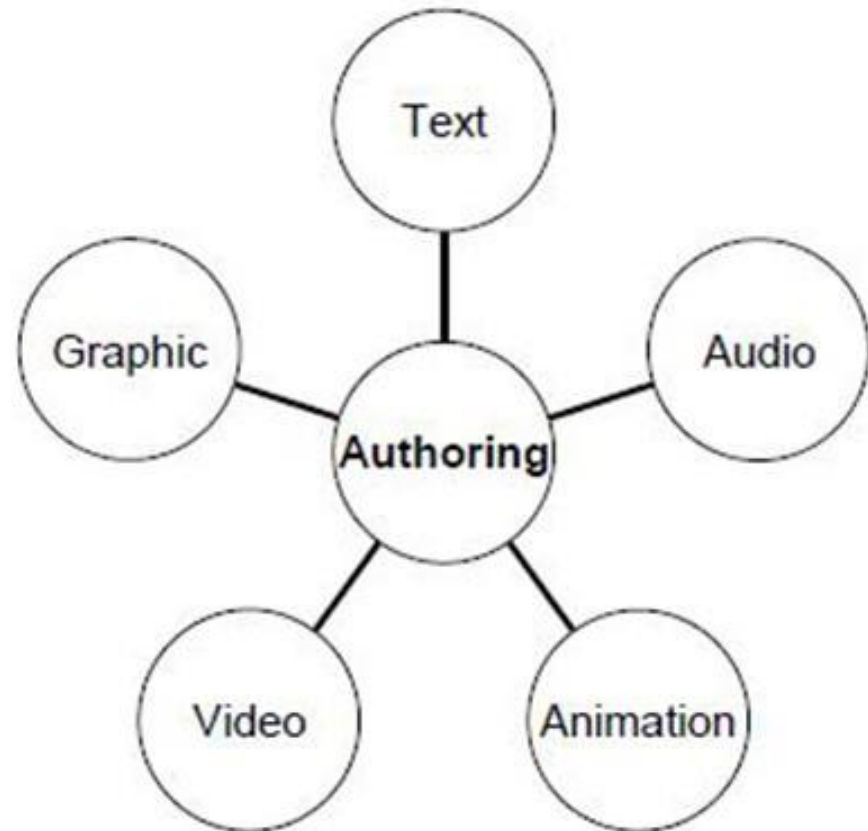
# Materi

1. Konsep dasar, pembuatan dan aplikasi multimedia
2. Organisasi pengembang multimedia
3. Perangkat pembuatan aplikasi multimedia
4. Kerangka bangun multimedia
5. Metodologi pengembangan multimedia
6. Piranti authoring multimedia
7. Pengembangan/perancangan multimedia
8. Konsep dasar toolbox
9. Pembuatan proyek, menu bar

# **PIRANTI AUTHORIZING MULTIMEDIA**

# Multimedia Authoring

- Menyediakan lingkungan terintegrasi untuk mengikat semua elemen secara bersama-sama dalam sebuah proyek Multimedia



# Komponen/Fitur Dalam Pirati Authoring

- Kebanyakan sistem authoring menyediakan fasilitas :
  - Loading dan saving aset media (graphics, sounds, videos, text, dan lain-lain)
  - Creating/editing audio dan graphics images
  - Pembuatan page/slide (layering data multimedia untuk membuat sebuah single page)
  - Efek animasi/transisi
  - Menspesifikasikan urutan aplikasi
  - Menambahkan kontrol interaktif untuk navigasi dan aksi pada saat run-time
  - Pembuatan CD-ROM atau distribusi berbasis web

# **Jenis-jenis piranti authoring**

1. Slide, Card atau Frame Based
2. Icon Based
3. The Score Model
4. Score dan Timelines
5. Scripting Language Based
6. Object-oriented Models

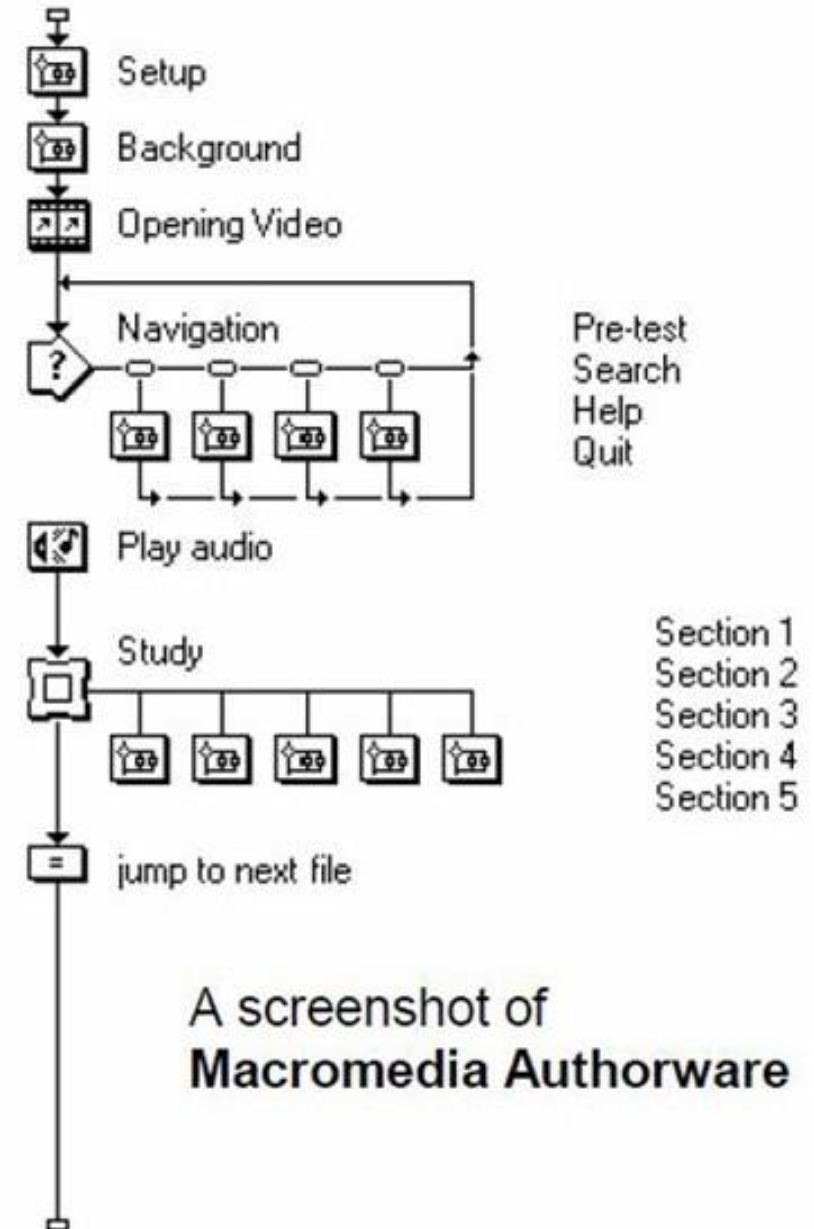
□ Beberapa sistem mengkombinasikan satu atau lebih kategori tersebut untuk membuat tools yang lebih baik

# 1. Slide, Card / Frame Based

- Pada kategori ini, elemen sistem authoring diorganisasikan menyerupai halaman buku atau tumpukan kartu
- Dalam sistem authoring ini diperbolehkan untuk menghubungkan antar halaman atau kartu ke suatu urutan yang terorganisasi
- Dapat melompat dari satu halaman ke halaman lain yang diinginkan dalam suatu pola navigasi yang terstruktur
- Sebuah halaman dapat berisi hyperlinks ke halaman lain untuk menyediakan navigasi atau halaman dapat dilihat secara sequential
- Memungkinkan adanya parameter global yang dapat di set untuk keseluruhan aplikasi
  - Contoh : background colour, default font, etc

## 2. Icon Based

- Setiap bagian direpresentasikan dengan sebuah *icon* (*symbolic picture*)
- Setiap icon memiliki tugas yang spesifik, contoh : *plays a sound*
- Semua icon-icon dihubungkan untuk membentuk suatu aplikasi lengkap
- Dapat memvisualisasikan struktur dan navigasi dari aplikasi final secara mudah



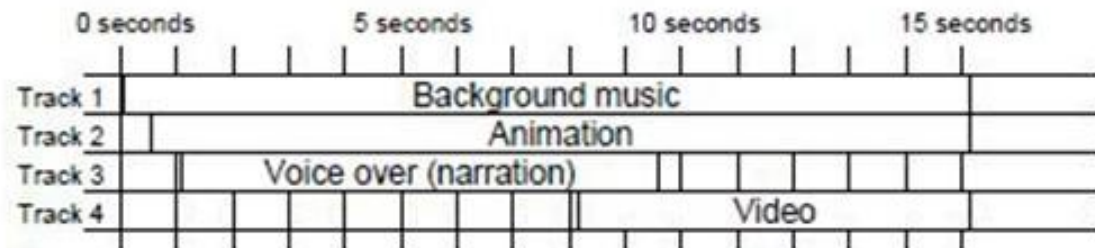
### 3. The Score Model

- Model ini mengambil analogi *musical score*
- Namun dalam hal ini bukan nada-nada yang dimainkan oleh alat musik, melainkan obyek-obyek multimedia (sounds, images, video, text) yang dimainkan pada titik (waktu) tertentu pada *score tersebut*
- Biasanya ada beberapa cara untuk menganimasikan suatu obyek (contoh : rotasi, *scaling*, *perubahan warna*) antara tiap transisi (yaitu saat satu obyek mulai dan selesai dimainkan)
- Setiap obyek multimedia hidup dalam suatu *track*
- Suatu score memiliki lebih dari satu *track*



## 4. Scores and Timelines

- *Timelines merupakan cara yang berguna untuk merepresentasikan data multimedia dalam presentasi atau aplikasi*
  - *Time direpresentasikan sepanjang x-axis*
  - *Tracks direpresentasikan sepanjang y-axis*
- Pengembang dapat memindahkan obyek-obyek ke kiri dan ke kanan untuk mengubah urutan informasi dan memanjangkan atau memendekkan



## 5. Scripting Language Based Models

- Merupakan versi pendek dari bahasa pemrograman yang lengkap
- Memiliki feature yang lebih sedikit dan karena itu lebih mudah untuk dipelajari
- Memungkinkan pengembang untuk menulis *script pendek* (program) yang dapat diasosiasikan dengan sebuah obyek multimedia

## 6. Object-Oriented Models

- Dalam sebuah *object-oriented model*, semua yang dapat diletakkan dalam suatu aplikasi dianggap sebagai obyek
- Setiap object memiliki :
  - **State** : suatu deskripsi dari properties-nya yang unik
    - Contoh → Suatu state dari graphics image akan mendeskripsikan ukurannya, colour depth-nya, nama pembuatnya
  - **Behavior** : hal-hal yang dapat dilakukan oleh suatu obyek
    - Contoh → Suatu video dapat dimainkan, dihentikan/dimainkan

## 6. Object-Oriented Models

- *Icon dan score-based models biasanya juga memiliki feature-feature object-oriented*
- Obyek-obyek memiliki sifat *self contained* dan dapat digunakan dalam proyek yang lain

# Multimedia Programming Tools

- Programming tools berbeda dengan authoring tools, dimana *programming tools lebih low level* dan pengembang yang menggunakannya harus memiliki pengetahuan tentang *hardware dan bahasa pemrograman*.
- Programming tools memungkinkan pengembang untuk mengakses banyak fungsionalitas daripada authoring tools tetapi membutuhkan banyak usaha

# Authoring vs Programming

## Programming Language

- Membutuhkan kemampuan programming
- Sulit dalam penggunaan
- Murah
- Low level program
- Program berjalan lebih cepat
- Tool untuk penggunaan lebih umum

## Authoring System

- Didesain untuk non-programmers
- Mudah digunakan
- Mahal
- High level program
- Program berjalan lebih lambat
- Memfasilitasi perancangan Multimedia & instructional