3D GAME PROGRAMMING

presented by:

EACOMM Corporation

EACOMM CORPORATION

Agenda

- Introduction
  - **About EACOMM Corporation**
  - **Why Author in Director MX**
  - **Other authoring environments**
- Typical Game Development Methodology
- Common Concepts/Issues
  - **Cameras and camera control**
  - **Lighting**
  - **Textures**
  - **Particle Systems**
  - **Introducing Interaction**
  - **Performance Optimization**
  - **Collision Detection**
  - **Physics**
- Common Game Algorithms
  - **The Isometric Game Engine**
  - **Path Finding/Detection**
  - **Others**
- Q and A

EACOMM Corporation

Corporate Profile

EACOMM CORPORATION

EACOMM Corporation

EACOMM Corporation is a comprehensive IT solutions provider.

- EACOMM was founded in 2001 with a mission to provide low-cost IT solutions to small and medium scale enterprises in the country.
- EACOMM has two offices:
  - 12th Floor IBM Plaza Building, Eastwood City Cyberpark, Libis, Quezon City
  - Suite 304, Eagle Star Condominium, Loyola Heights, Quezon City
- EACOMM has four main divisions: Multimedia/Web, Software, Training, and Embedded Systems

EACOMM CORPORATION

EACOMM Corporation

- **Multimedia and Web Division:**
  - Web Development
    - Static Website Development
    - Interactive Website Development
    - E-commerce Website Development
  - Multimedia Authoring and Development
    - CD-ROM Authoring
    - Information Kiosks
    - Game Programming
- **Software Development Division**
  - Financial Planning System
  - Custom Software Solutions

EACOMM CORPORATION

EACOMM Corporation

- **Embedded Systems Division**
  - Wireless Data Transmission Systems
  - Micro-controller based design and development
  - LED/LCD Display Systems
  - Custom Hardware/Software Solutions
- **Training Division**
  - Short Courses / Workshops on multimedia and web development skills and theories
  - Web and Multimedia Developer Certification programs (Early 2004)

EACOMM CORPORATION

Why Author in Director MX?

1. WYSIWYG Development Platform
2. Easy to use and learn
3. Powerful Object Oriented pseudo programming language (Lingo)
4. Capable of directly importing a variety of file formats and types.
5. Built-in 3D Engine
6. Cross-platform Development Platform
7. Deployable On-line as a Shockwave Application
8. Fast development time

Why (Not to) Author in Director MX?

Disadvantages:

- Produces slower running applications than coding in C/C++
- Requires relatively high system requirements for end-user
- For very big, ambitious projects, Director and Lingo just won't cut it.
- Requires a "Made by Macromedia" logo on the delivered software.

Other Authoring Options

Darkbasic (www.darkbasic.com) - Programming environment catered specifically to game development. Darkbasic is a derivative of the BASIC programming language. Cheap and supposedly easy to learn.

CrystalSpace (crystal.sourceforge.net) – Opensource 3D engine and development platform written in C++. Available for Linux, Windows and Mac. Knowledge of C/C++ required.

Check out www.sourceforge.net and/or www.gamedev.net for more information regarding 3D engines, and authoring environments.

Game Development Methodology

- Planning and Storyboarding
  - Designing a Non-linear story.
  - Challenge and Reward
  - Designing the characters, and their motivations
  - ***Developing a Marketing Plan***
  - Determine target audience and target platform(s)
- Conceptualization, layout and design
  - Gradually progress from napkin drawings to 3D Models
  - Screen and Scene designs
  - Character models
- Initial Programming
  - Develop an "Alpha" version to test your various designs and plans
  - Get feedback and opinions as early as possible
  - Test and Debug!!!

Game Development Methodology

- Media Collection and Integration
  - Populate your game with sounds, animations, videos and others.
- Beta Version Development
  - Develop your working prototype for Beta Release.
  - Continuously test the software and game concept on a variety of audiences.
  - Cyclical process, test->program->test->program, until you are confident of your work.
  - Don't be afraid to go back to step one. Blizzard Philosophy: Better to not release a game than release a lousy one.
- Publication and Distribution
  - CD-ROM Duplication
  - Package Design
  - "Discouraging" Piracy
  - Distribute yourself or get a distributor?

Common Concepts/Issues in 3D Game Programming

Common Game Algorithms

THE ISOMETRIC (TILE-BASED) ENGINE:

- "God" View.
- Used extensively in games such as: Most Real Time Strategy Games (C&C, Starcraft, Warcraft II), Some Role Playing Games (Baldur's Gate, Fallout, etc.), and simulations (The Sims, SimCity 2000, etc.)
- Creates a "False" 3D environment or a mixture of 3D and 2D elements.
- Much faster engine than a "true" 3D engine. (Lower system requirements)
- All elements can be dynamically created. (e.g. random maps)
- Can create a level of detail and realism not easily realizable in true 3D games.

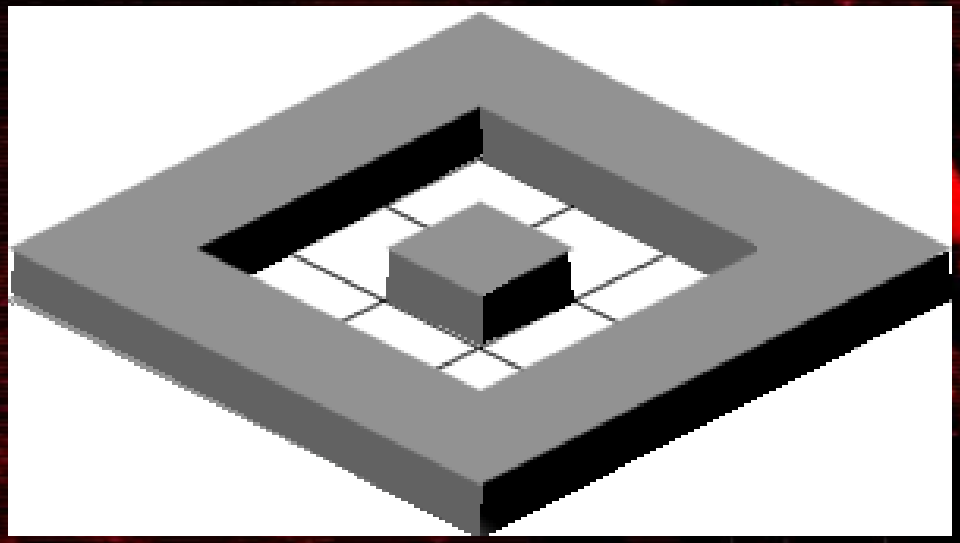
Common Game Algorithms

THE ISOMETRIC (TILE-BASED) ENGINE:

- Essential Elements:
  - The Map Array: A 2-dimensional array of integers that represents the map appearing on screen.

    A 5x5 Map Array:
    - (1,1,1,1,1),(1,0,0,0,1),(1,0,1,0,1),(1,0,0,0,1),(1,1,1,1,1)
    - Where 1 represents a wall, this maparray will render to the image below:

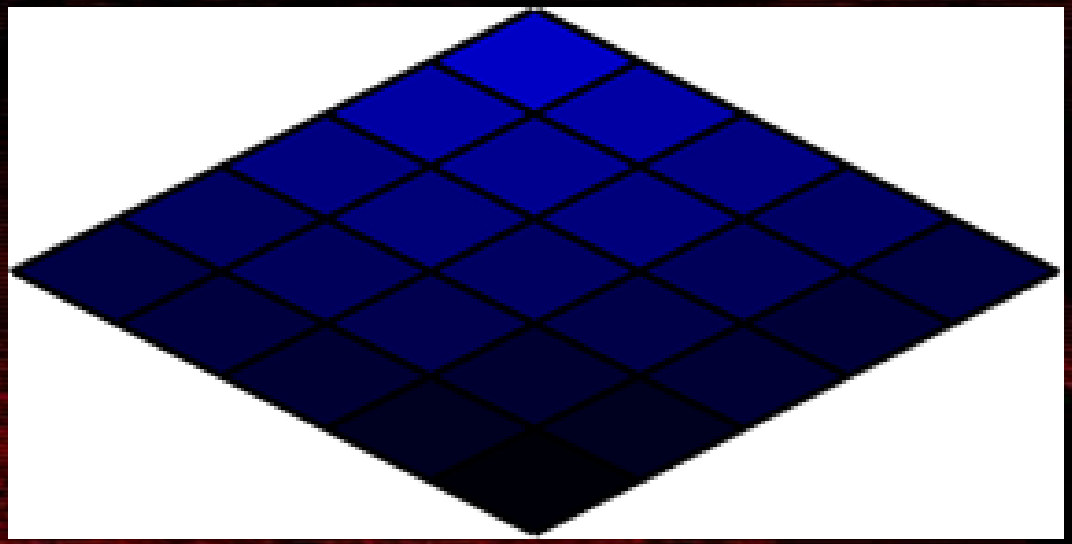
1	1	1	1	1
1	0	0	0	1
1	0	1	0	1
1	0	0	0	1
1	1	1	1	1



Common Game Algorithms

THE ISOMETRIC (TILE-BASED) ENGINE:

- Essential Elements:
  - The Color Map: The color map is used to detect mouse clicks on the isometric map. Each map tile is represented by a known, unique color.
  - A 5x5 Color Map Array:



Common Game Algorithms

THE ISOMETRIC (TILE-BASED) ENGINE:

- Essential Elements:
  - Animation
    - ✓ To show animation in an isometric engine, “image sequences” are used. Eight main animation sequences are created, representing 8 points of the compass.

Common Game Algorithms

THE ISOMETRIC (TILE-BASED) ENGINE:

- Essential Elements:
  - **Collision/Object Detection in an Isometric Engine.**
    - ✓ **Each object is assigned an integer that represents it in the map array. (e.g. 1=wall, 2=Non-player character, 3=house, etc.)**
    - ✓ **Since everything lies on *at least* one tile, the program need only identify the clicked/target object by checking the map array.**
    - ✓ **Thus an object such as a wall can be programmed to be unpassable or another object can initiate a new action.**
    - ✓ **Sample Collision/Detection Algorithm:**
      1. **Wait for click/target selection.**
      2. **Identify click/target from map array**
      3. **Case:**
         1. **If target=1 then it is a wall. Disallow character to walk over it.**
         2. **If target=2 then it is a NPC. Initiate dialogue routine.**
         3. **If target=3 then it is a house. Load house interior.**
         4. **Etc.**
      4. **And so on…**

Common Game Algorithms

PATH DETECTION:

- ✓ Path detection is a very useful algorithm that can be seen in most RTS, RPG, and 3D games.
- ✓ A commonly used path detection algorithm is the A* Algorithm or A-Star.
- ✓ Takes advantage of the grid-nature (2D-map array) of game maps. It can be used for both 2D and 3D games.
  - Since there is a grid with grid coordinates, A* takes advantage of this knowledge to provide a fast determination of the path from origin to target.

Common Game Algorithms

PATH DETECTION:

- ✓ A* generates "estimates" on how "costly" each step towards the target is going to take. It then chooses the "least costly" method to reach the target.
- ✓ A* is ideal for games in that it can be "tweaked" so that it does not necessarily obtain the "perfect" path to a target.
- ✓ A* can also take into account "terrain effects" defined by the map array. (e.g. walking through water or forest areas can be defined as slower than walking on a clear path or road)
- ✓ A* is a fairly complex algorithm. Fortunately, there are examples and source codes available in the Net.

Q and A

EACOMM CORPORATION