Computer Games 2011
Engineering

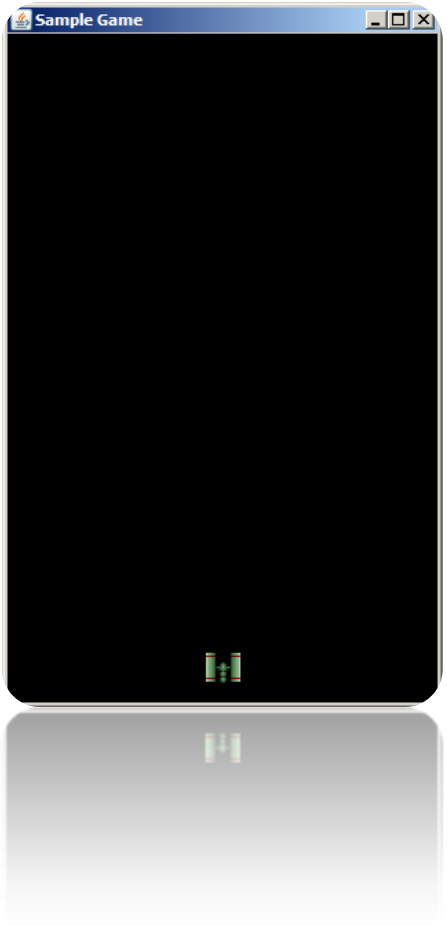
Dr. Mathias Lux
Klagenfurt University

Agenda

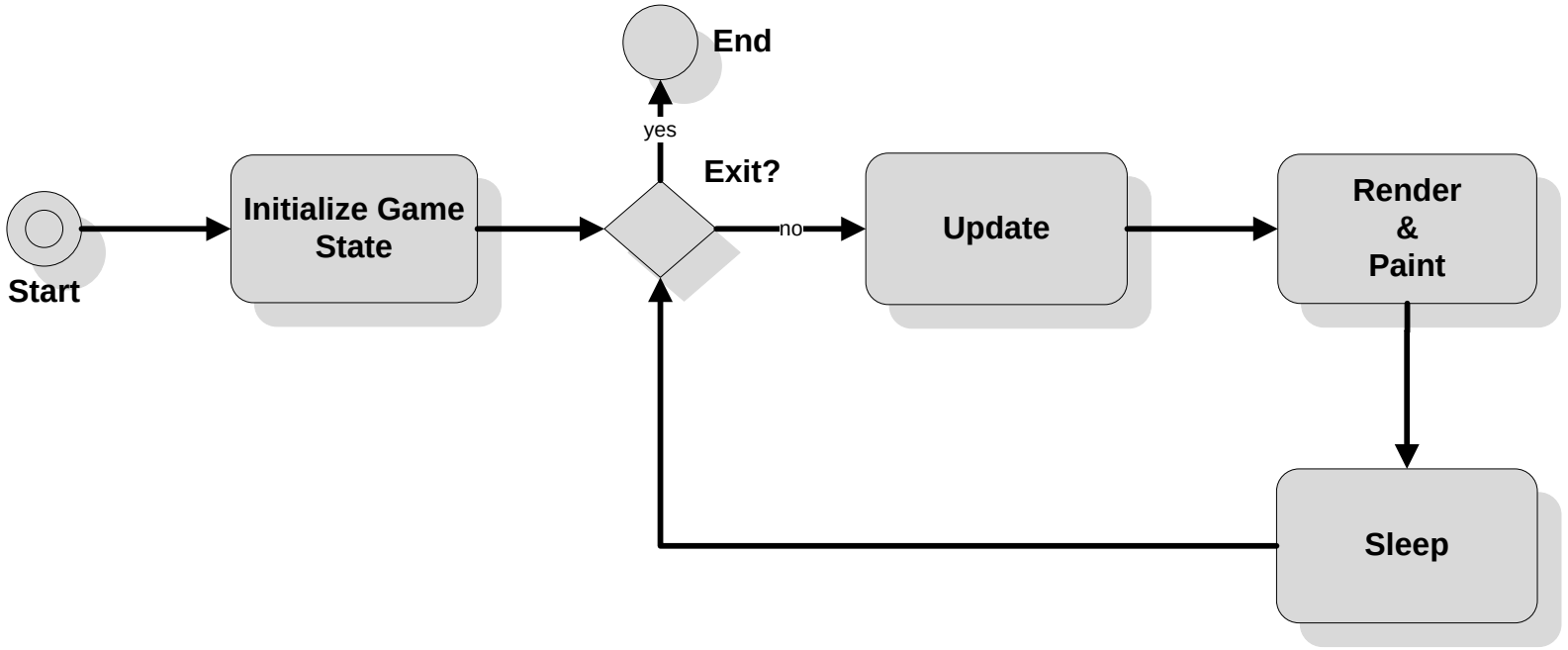
- Game Loop
- Sprites & 2.5D
- Game Engines

Example: Space Ship

- Simple Game:
  - A single space ship
  - Moving left to right
- Advanced Tasks
  - Firing rockets
  - Explosions
  - Sound & music



Game Loop


End
yes
Exit?
Start
Initialize Game State
no
Update
Render & Paint
Sleep

Game Loop

- while(user doesn't exit)
  - check for user input
  - run AI
  - move objects
  - resolve collisions
  - draw graphics
  - play sounds
- end while

Check for user input

- Get state of keys
  - e.g. is <space> key pressed
- initiate action
  - e.g. spawn rocket

Run AI

- Check current state
- Initiate action
  - spawn UFOs,
  - drop bombs,
  - change paths etc.

Move Objects

- Move objects
  - along their (changed) paths
  - matching their (changed) velocity

Collision Detection

- Check if
  - either there is a crossing in paths
  - or a double setting of pixels
- Pixel based vs. boundary based
- Runtime issues
  - Grid based, data structures etc.

Draw Graphics

- Direct engine
  - to allocate resources
  - to paint the buffer
  - then flip the buffer

Play Sounds

- Decode sounds
  - maintain storage
- Fill buffer
  - to be played
- Trigger events
  - explosions, sounds, etc.

Game Loop

- Frames per second
  - 20 or more are minimum
  - 60+ frames are optimum
  - jitter is a problem (sync to display device)
- Stereoscopic 3D needs double frame rates

Game Loop

- Parallel processing
  - Xbox has 3 cores (with HT)
  - PS3 has 8 cores
- Game loops run in parallel
  - AI loop
  - sound & painting loop
  - control loop

Agenda

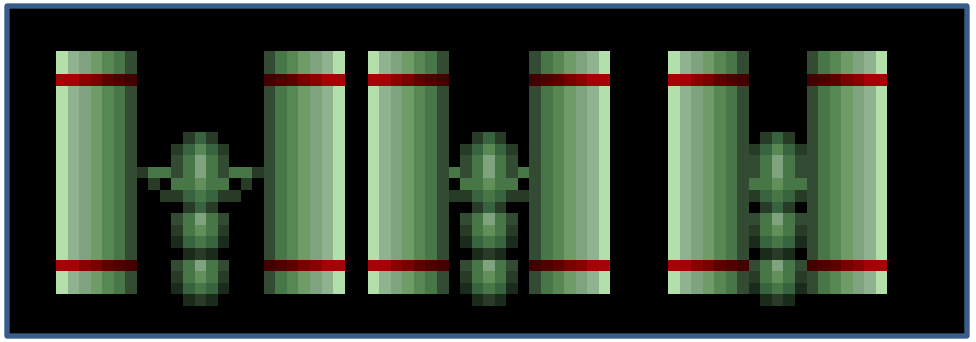
- Game Loop
- Sprites & 2.5D
- Game Engines

Sprites

- What is a sprite?
  - A (moving) object on the screen
- Resources needed
  - visuals, audio, state
- Loading and displaying
  - game loop, effects, resources needed in time

Simple Sprite Animation

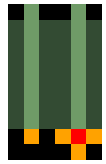
- Image strips …
  - All possible animation frames in one image
  - Cut it in initialization method
  - Display the right one in each state



Features

- Left-right movement
  - spring based physics
  - “feels more real”

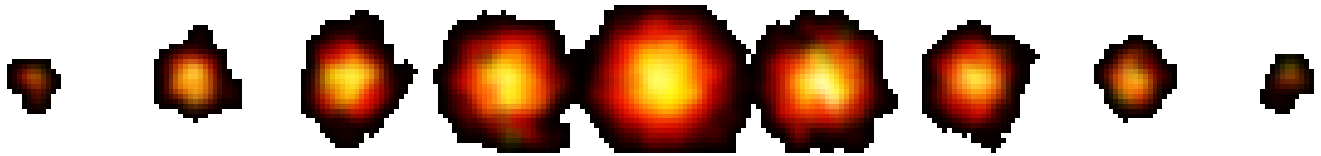
Rocket



- Another sprite
  - Only one allowed at a time
- Acceleration
  - The longer it moves the faster it gets
- Removed if out of sight
  - Sprite should be re-used (e.g. ammo)
  - Too many sprites consume too much memory
- Simple sprite with 2-frame animation

Explosion

- Rocket explodes
  - rocket is removed
  - explosion sprite is displayed
- Animation with 9 different frames
  - No alpha …
- Removed when over



Parallax Scrolling

- Common Technique for 2.5D
  - In contrast to “real 3D”
- Simulates depth with multiple layers
  - Each layer moves with different speed
- Side scrollers
  - Games moving from left to right (Mario, etc.)

Parallax Scrolling

Background layer: a starry sky.

Layer 1: a chain of mountains.

Layer 2: background vegetation.

Layer 3: foreground vegetation and path.

Source: http://en.wikipedia.org/wiki/Parallax_scrolling

Demo-Video

- California Games

Starfield Simulation

- Create 3 different layers
- Load them during startup
- Display them with wrap around
- Move them in different speeds

Starfield: Performance

- Performance issues with Java
  - Translucent images are not rendered with hardware acceleration.
  - This has to be turned on explicitly on Windows
- Better: Draw stars yourself

More 2.5D Tricks

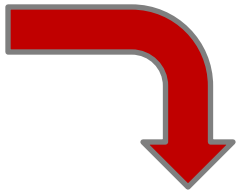
- Assume top-down view on landscape
  - Draw shadow
    - Use translucent color
    - While scrolling move and scale shadow
    - Creates illusion of uneven terrain
  - Implement jump action of sprite:
    - Move and scale shadow
    - Scale sprite

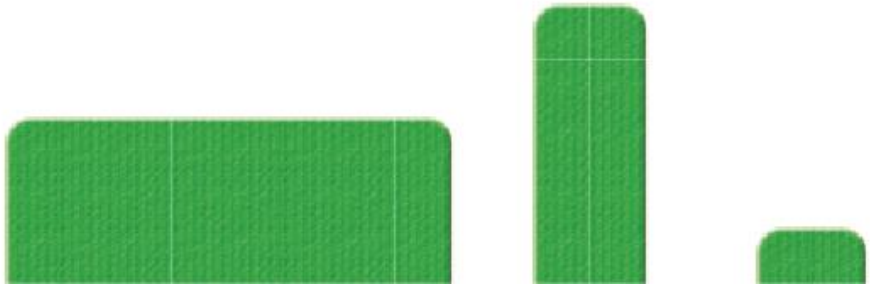
Demo

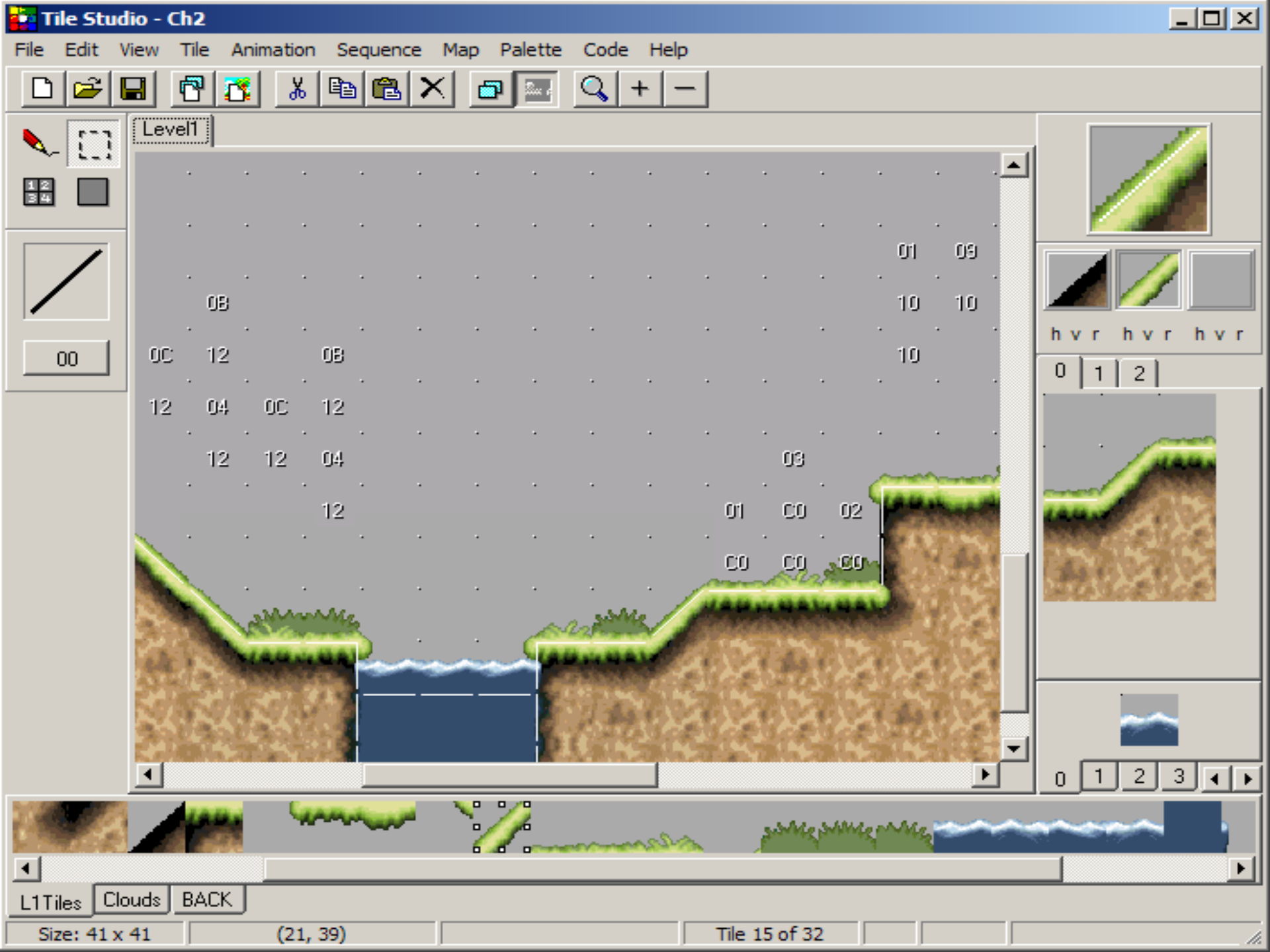
Video: 1942

Image Tiles …

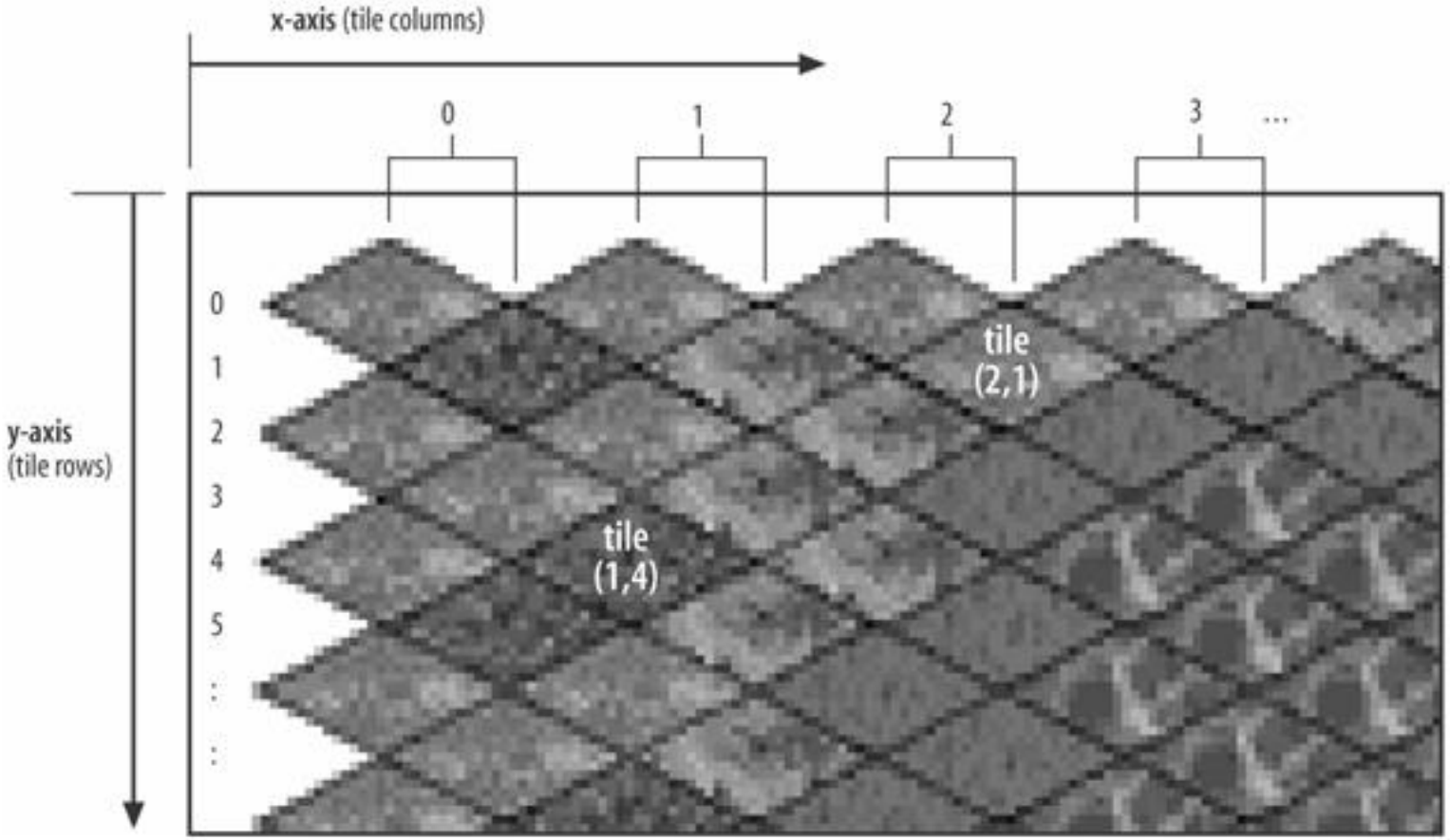
- Common technique to “create worlds”
- Add up small tiles to big picture





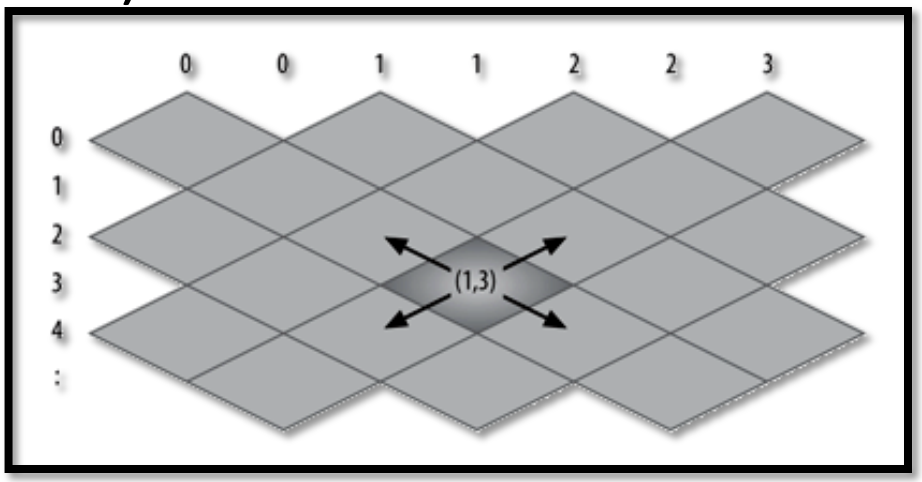

Tile Studio - Ch2
File Edit View Tile Animation Sequence Map Palette Code Help
Level1
00
h v r h v r h v r
0 1 2
0 1 2 3
L1Tiles Clouds BACK
Size: 41 x 41
(21, 39)
Tile 15 of 32

Isometric Tiles



Isometric Tile Games

- Render back to front
  - Support for sprites (trees, characters, etc.)
- Movement
  - From tile to tile (animated?)
  - World “moves”



Demo

- Diablo

Agenda

- Game Loop
- Sprites & 2.5D
- Game Engines

Microsoft XNA

- Game Engine
  - provides many routines for 2D and 3D games
- Multi-platform
  - based on .NET compact runtime
  - develop for PC, Xbox360, Zune, Windows Phone

Game1.cs

WindowsGame1.Game1

```
using System;
using System.Collections.Generic;
using System.Linq;
using Microsoft.Xna.Framework;
using Microsoft.Xna.Framework.Audio;
using Microsoft.Xna.Framework.Content;
using Microsoft.Xna.Framework.GamerServices;
using Microsoft.Xna.Framework.Graphics;
using Microsoft.Xna.Framework.Input;
using Microsoft.Xna.Framework.Media;
using Microsoft.Xna.Framework.Net;
using Microsoft.Xna.Framework.Storage;

namespace WindowsGame1
{
    /// <summary>
    /// This is the main type for your game
    /// </summary>
    public class Game1 : Microsoft.Xna.Framework.Game
    {
        GraphicsDeviceManager graphics;
        SpriteBatch spriteBatch;

        public Game1()...

        /// <summary>
        /// Allows the game to perform any initialization it needs to before starting to run.
        /// This is where it can query for any required services and load any non-graphic
        /// related content.  Calling base.Initialize will enumerate through any components
        /// and initialize them as well.
        /// </summary>
        protected override void Initialize()...
```

```
        public Game1()...

        /// <summary>
        /// Allows the game to perform any initialization it needs to before starting to run.
        /// This is where it can query for any required services and load any non-graphic
        /// related content.  Calling base.Initialize will enumerate through any components
        /// and initialize them as well.
        /// </summary>
        protected override void Initialize()...

        /// <summary>
        /// LoadContent will be called once per game and is the place to load
        /// all of your content.
        /// </summary>
        protected override void LoadContent()...

        /// <summary>
        /// UnloadContent will be called once per game and is the place to unload
        /// all content.
        /// </summary>
        protected override void UnloadContent()...

        /// <summary>
        /// Allows the game to run logic such as updating the world,
        /// checking for collisions, gathering input, and playing audio.
        /// </summary>
        /// <param name="gameTime">Provides a snapshot of timing values.</param>
        protected override void Update(GameTime gameTime)...

        /// <summary>
        /// This is called when the game should draw itself.
        /// </summary>
        /// <param name="gameTime">Provides a snapshot of timing values.</param>
        protected override void Draw(GameTime gameTime)...
    }
}
```

Microsoft XNA

- Simple tutorials online
  - http://create.msdn.com/en-us/education/gamedevelopment
- Create a game within hours

Gamebryo

- Multiplatform game engine
  - PC, PS3, XBox360, Wii
- Toolset & integration of tools
  - Physx, Scaleform, etc.
- Employed for AAA titles
  - Fallout 3, Oblivion, Civilization IV, Epic Mickey

Gamebryo

UDK & Unreal Engine

- Unreal Engine 3
  - Xbox 360, PS3 & PC
- Lots of features
  - AI, animation, lighting, editor, scripting etc.
- Employed for AAA games
  - Mass Effect 1+2, Bioshock 1+2, Batman: Arkham Asylum etc.

Source

- Valve‘s game engine
  - Counterstrike Source, TF2, Half Life 2, Portal etc.
- Available for modding
  - Criticized for „old“ toolset

source™

Ogre

- Open source game engine
  - scene graph based
  - particle system, scripting, HDR etc.
- Tool support
  - Maya, Blender etc.
- Employed for several games
  - Torchlight, …

Other Engines

- Unity
- Cocos2d
  - iOS, Python
- CryEngine
  - CryTek
- SCUMM
  - Script Creation Utility for Maniac Mansio
- Frostbyte (DICE)
- Phyre Engine

Other Tools

- FMOD
- Bink
- Havok
- Scaleform

Vielen Dank …

… für die Aufmerksamkeit