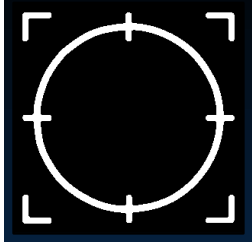
Developing for PlayStation™ - An Overview

David Coombes

SCEE

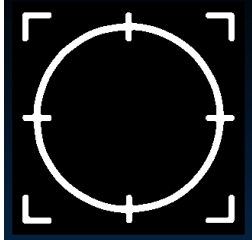
™



Developing for PlayStation™

- Content
  - Intro to Development
  - Development Platforms
  - A simple game demo (source available soon)
  - Questions

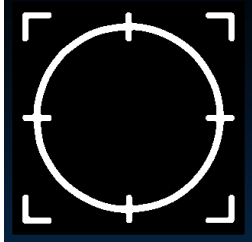
PlayStation™



Developing for PlayStation™

- Program in C using Libraries
- Complete Development Environment
- Documentation
- All the tools you need to begin
- Technical Support

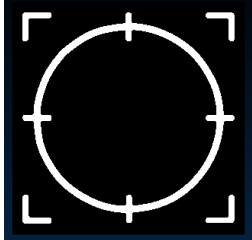
PlayStation™



How to get Support

- Use the BBS
- Email
- Fax
- Hot line
- Normal phone

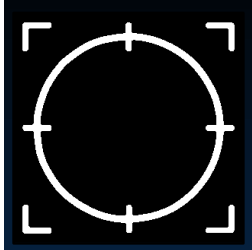
PlayStation™



Hey, lets port something!

- Playstation™ not architecturally similar to other consoles or PC
- PlayStation™ is BETTER in many things.
- Play to the machines strengths
- Don’t port the game, port the concept!

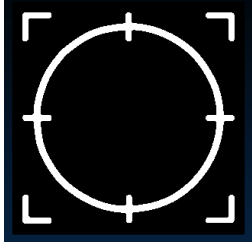
PlayStation™



Understand what the machine does best

- Learn what works well and do this!
- Don’t fight the hardware/libraries
- Then you’ll make a great game!

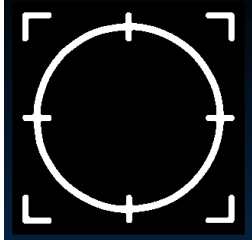
PlayStation™



Platforms

- Development Kit (PC based)
- Debugging Station
- PlayStation™

PlayStation™



What is a Debugging Station?

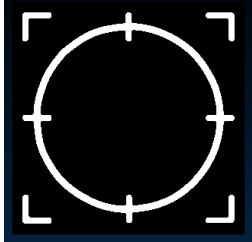
- It is blue!
- It is like a consumer PlayStation™ but with the anti-piracy mechanisms removed so it can play gold disks
- Allows you to test your code on the final machine

PlayStation™

PlayStation™ Vs Dev Kit

- Main Ram
  - PlayStation™ 2Mb
  - Debugging Station 2Mb
  - Dev Kit 8Mb
    - Non optimised code (Debugging)
    - Load data into main ram direct from PC for rapid development

PlayStation™



Development Kit

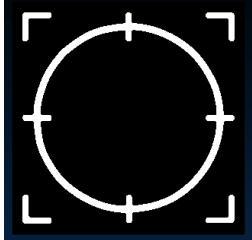
- Uses 2 full length slots
- Can read CDs with optional CD-ROM drive
- CD Emulator

PlayStation™

PlayStation™ Vs Dev Kit

- Mass Storage
  - PlayStation ™
    - CD ROM Dual Speed 300 Kps
  - Dev Kit
    - CD ROM Dual Speed 300 Kps
    - CD Emulation (Hard Drive)
    - PCFS (Read and Write)

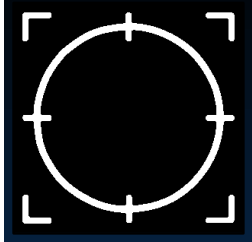
PlayStation™



A Simple Game

- Initialise the hardware
- Setup some data
- Main loop
  - graphics
  - sound
  - logic

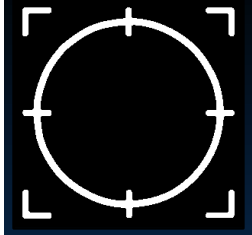
PlayStation™



Create Sound Effects

- Use Sound Artist Board and Sound Tools
- Play CD-DA or XA-ADPCM

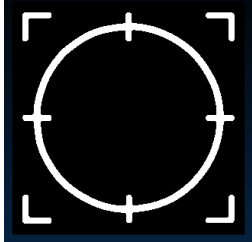
PlayStation™



Create Graphics

- Choose from a wide variety of pixel editors and modelling packages
- Use plugins and utilities to convert these to native PlayStation™ format

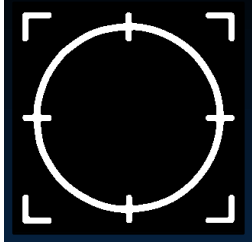
PlayStation™



A Simple Game

- Initialise PlayStation™
  - Reset Graphics system
  - Initialise drawing environment
  - Reset CD system
  - Install Pad Reading routine
  - Create and Initial game data structures

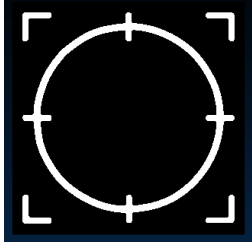
PlayStation™



A Simple Game

- Load game data
  - Load data from PC directly to main ram
  - Transfer sound data to sound ram
  - Transfer Texture data to video ram

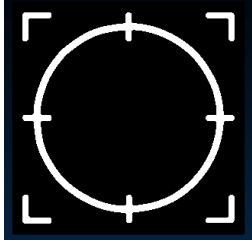
PlayStation™



A Simple Game

- Main loop
  - Read pads
  - do game logic
  - draw polys to create screen

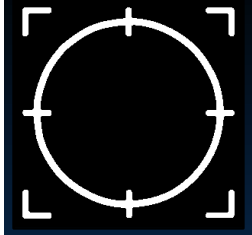
PlayStation™



A Simple Game

- Sound
  - Simple samples
  - Reverb

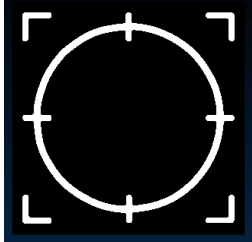
PlayStation™



A Simple Game

- Graphics
  - 320*240 16bit double buffered
  - PolyFT4 (flat shaded, textured, four sided polygons)
  - Background loaded from main ram each frame to save vram
  - Switch buffers on Vsync()
  - Built in Font

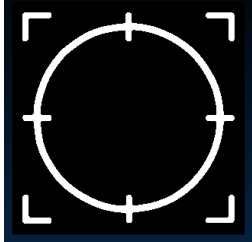
PlayStation™



A Simple Game

- 3D on PlayStation™
  - Dedicated 3D Co-processor (GTE)
  - 3D to 2D co-ordinate conversion
  - Z Sorting
  - Real time lighting/Depth cueing

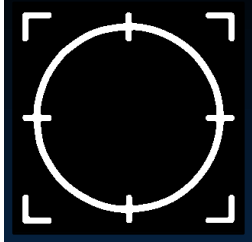
PlayStation™



Conclusion

- PlayStation™ is powerful
- PlayStation™ development is easy

PlayStation™



The end.....

- Good Luck with your Products
- Question Time

PlayStation™