Game Production TimeLine

John Laird

University of Michigan

Game Production Timeline

- Inspiration (1 month)
  - Results in game treatment/concept paper
- Conceptualization (3-5 months)
  - Results in design summary/design document
  - Story boards and prototype
- Technical Architecture (2 months)
  - Determine the technical details
  - Results in master technical specification/production document
- Tool Building (4 months)
- Assembly (12 months)
- Levels (4 months)
- Review, Testing, … (3 months)
- Total (about 2 years)

Three Stages of Design Documentation

- Design Treatment / Concept Paper
  - Is this idea worth development?
- Design Summary / Design Document
  - Blueprint for the game
  - Game script/storyboard or prototype
  - Possible markets
  - Basis for in depth evaluation
- Design Specification / Production Document
  - Used to write the code and manage product

Design Treatment (Project Proposal)

- Purpose:
  - Basis if worth investing any serious effort
  - Sales tool
- Many sections that summarize game:
  - The “high concept” of game
  - Basic design choices: game genre, …
  - Features that make game exceptional
  - Game Story: plot, characters: no more than 1 page
  - Game play: what does the player do?
  - Technical development: platform, development environment, …
  - Expected audience
  - Estimated cost [development time]
  - Risk analysis
- Keep it short and simple: 1-2 pages
  - Does not have all plot twists -- just main ideas
- May include a mini-prototype of key ideas

Design Treatment:
Game Story

It’s 2025, 5 years after a major nuclear war has destroyed many things. Australia has been divided into three segments, a penal colony, the badlands – a swath of land made up of villages of policed mutants, and other crazed beings, and the free zone where a group of outcasts are reestablishing democracy. It is up to you, the player, to escape and make it to the free zone. You have only martial arts expertise to help you. You must run a foot race to the free zone to lead a revolution.

Design Treatment: *Game Play and Look*

The game features 3D graphics which scroll toward the player (the movement somewhat like a raycaster like Doom except the movement is 180 degrees not 360. Imagine “Auto Racing Doom”). The player is represented by a third person polygon figure (like Virtua Fighter) controlled with a joystick. The faster you run and avoid obstacles, the better chance you have to survive. Too slow a pace allows your chasers to catch up to you.

Design Treatment:
Game Play and Look 2

The player runs through a continuous stream of towns, forest, brush and other settings attempting to maintain speed jumping onto, over and around, while mutants and security guards and more enemies attack them. At your disposal are weaponry you pickup, water and food for health and a series of “Virtua Fighter” like martial arts moves including flips and summersaults.

The trick is to use the flips and rolls to jump over things, on to building rooftops (in towns) and to use kicks, puches and weaponry to take out oncoming enemies. …

Design Treatment: Technical

This game will feature our new 3D engine, coupled with animated 3D backgrounds. There are roughly 50 scenes we are constructing, each with several view; these scenes will be rendered with 3D Studio Maxx. We will use a PC development platform and later port to the XBox. The development language will be C++, using several APIs developed in and out of house. Overall, we expect a development time of roughly 10-16 months.

Design Document

- Purpose
  - Exhaustively details *everything* that will happen in the game
    - Cheaper to evaluate and change than a computer implementation
  - Basis for estimating cost and tracking progress
  - Allows others to start designing implementation
  - Is the game “bible” - over 100 pages
- Created by the Game Designer
  - Usually not same as the game developer
- If it isn’t in this, it might not be in the game
  - Depends on development philosophy

Design Document

- More formal and complete than treatment
  - What does the player do?
  - What is the interface.
  - What is the plot.
- Details! Leave nothing to the imagination.
  - What are all the levels?
  - What are all the characters?
  - How can they interact?
  - What can you do to them?
  - Art bible – what will be the look of the game
- Development plan/schedule
  - What is going to be done
  - When is it going to be done
  - Who is going to do it

Design Document Sections

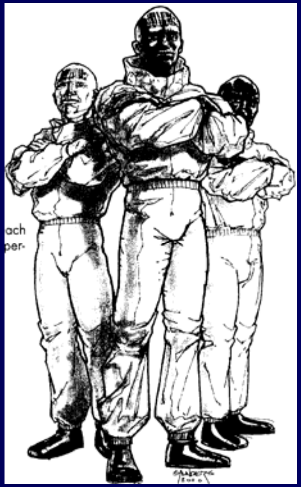
- Executive Summary
  - Similar to Design Treatment
- Product Specification
- Game Specification
- Art Specifications

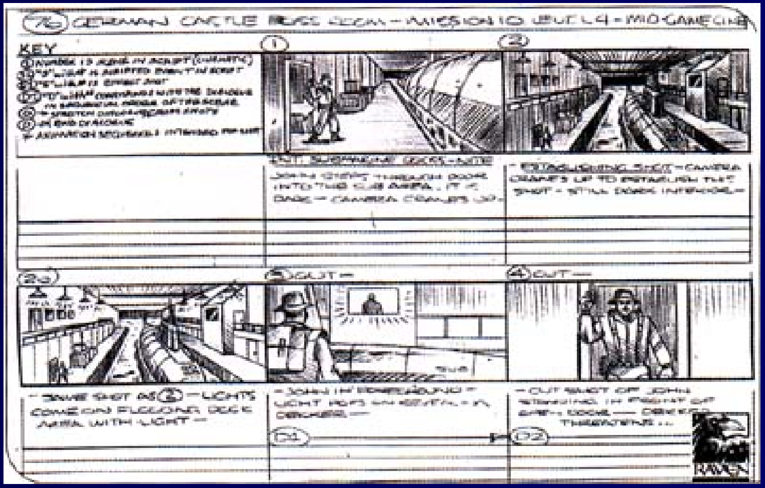
Design Document: Product Specification

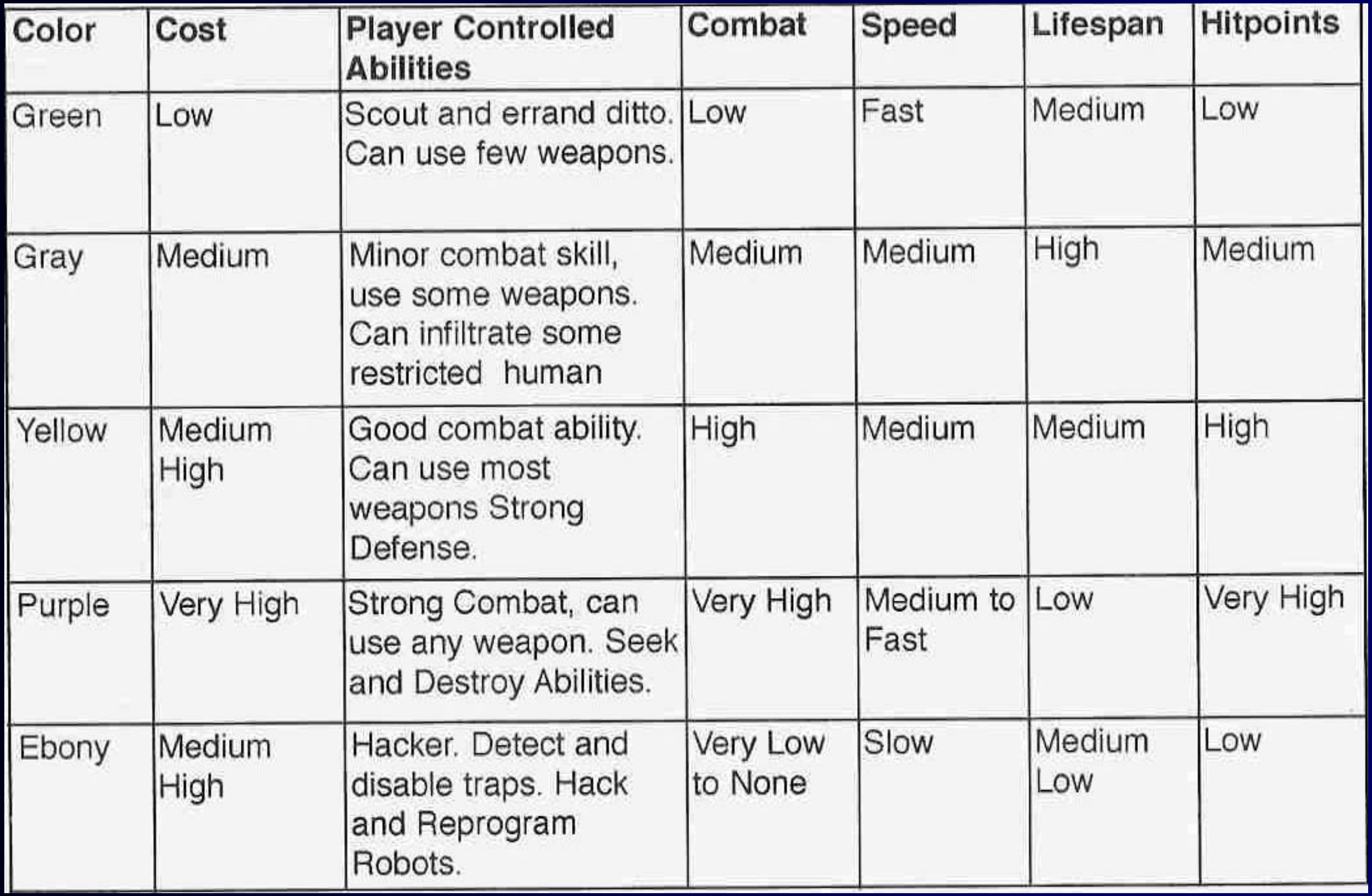
- Production team:
  - Members and experience
- Target audience
- Time: game play, shelf life
- Target platform
- Production tools:
  - Microsoft project, excel, PhotoShop, ...
- Schedule with milestones and deliverables
- Etc.
  - Localization: Languages, art work, ...
  - Packaging and Documentation

Design Document: Game Specification

- What is it like to play the game?
- Interface: Mockup
- Summary of the story line.
  - Major Goals: Final accomplishments.
  - Smaller Goals: Intermediate accomplishments.
- Storyboards
  - Prototype artwork of what screens will look like
- Character Bibles
  - Bio and profile of each character.
- Flowcharting
  - What are the choice points and how they connect

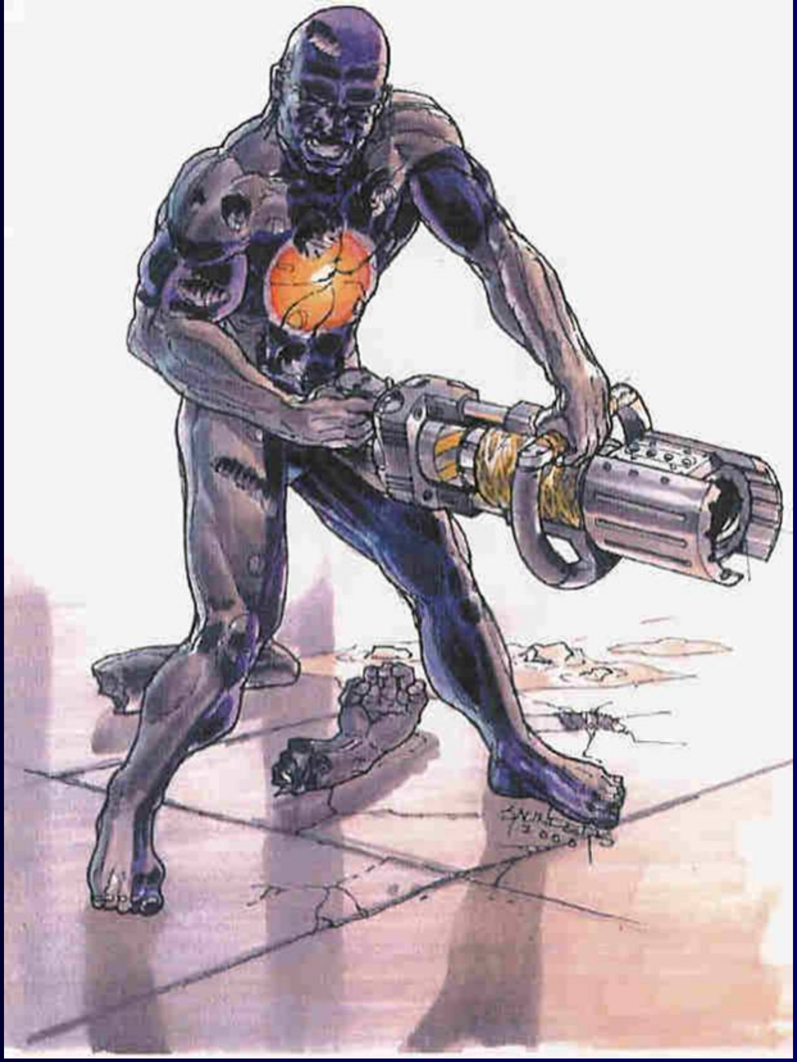



GERMAN CASTLE ... – MISSION 10 LEVEL 4 – MID-GAME CIN
KEY
1
2
JOHN STEPS THROUGH DOOR INTO THE SUB AREA. IT IS DARK – CAMERA CRANES UP.
ESTABLISHING SHOT – CAMERA CRANES UP TO ESTABLISH THE SHOT – STILL DOES INTERIOR –
2a
3 CUT –
4 CUT –
SAME SHOT AS 2 – LIGHTS COME ON FLOODING DECK AREA WITH LIGHT –
JOHN IN FOREGROUND –
CUT SHOT OF JOHN STANDING IN FRONT OF OPEN DOOR — THREATENS ...
RAVEN



Color	Cost	Player Controlled Abilities	Combat	Speed	Lifespan	Hitpoints
Green	Low	Scout and errand ditto. Can use few weapons.	Low	Fast	Medium	Low
Gray	Medium	Minor combat skill, use some weapons. Can infiltrate some restricted human	Medium	Medium	High	Medium
Yellow	Medium High	Good combat ability. Can use most weapons Strong Defense.	High	Medium	Medium	High
Purple	Very High	Strong Combat, can use any weapon. Seek and Destroy Abilities.	Very High	Medium to Fast	Low	Very High
Ebony	Medium High	Hacker. Detect and disable traps. Hack and Reprogram Robots.	Very Low to None	Slow	Medium Low	Low

Example Art



Design Document:
Outline of a Level

- Name of the Section/Level/Scene
- Physical and audio appearance
- Background - sketch if possible
- Foreground objects and characters
  - What actions can they perform?
  - Which objects are animated?
  - Sound effects
- Character scripts
- Transitions

Level Design

- Consumer testing (not game designers)
  - Where do they get bored?
    - “Basically, I fell and I died. Instantly. The game consisted only of levels where I wanted to give up before I’d reached the end.”
  - Where did they die?
  - How much time there was between challenges
  - 20 minutes to finish a level.
    - Sometime move the player along if having trouble
      - Double the number of checkpoints, easier problems
  - Not hard to make level more difficult
    - Hard to make level easier

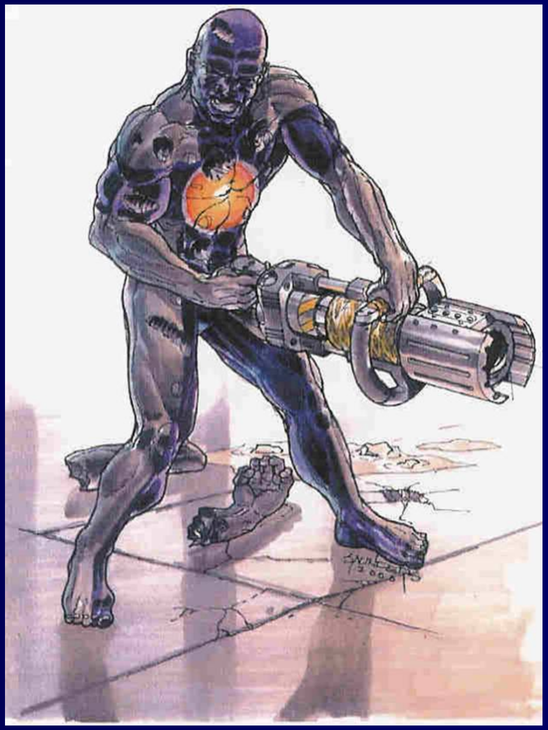
Sometimes need a prototype

- To show some technology is possible
- To illustrate interface and game play

- Use existing tools
  - Macromedia Director for fast animations
  - 3D Studio for 3D models

Design Document:
Art Specification

- Define overall style of art work
  - Example backgrounds
- Include example sketches of all main characters



Key points to a good design document

- Philosophy needs to be explicit (goals of game)
- Make it readable, well written.
  - Use tables, graphics as appropriate.
- Give priorities to the ideas so everyone knows what is important and what has been rejected
  - Indispensable, Important, If Possible, Rejected
  - Include reasons!
- Give all the details
  - Tables of what happens when X hits Y
- How will you do things: motion capture/animation

Technical Design Specification

- Purpose: Details of implementation
  - Written before you spend lots of $$’s.
  - Allows many people to work at once.
- Further details of art, music, sounds, video, etc.
  - Definition of data structures and interfaces.
  - Pseudo-code.
    - Can debug ideas before “cut bits”.
    - Provides documentation.
- Development Environment.
- Schedule with Milestones, Releases

Creating a Time Line

- When is your story going to be done?
- When is the art going to be done?
- How much time to write the code?
- What has to be done first, second, third?
  - Can things be done in parallel?
  - Can more people speed it up?
- Typical schedule is 1 to 2 years.

Development Phase

- Tool Building
  - Create 3D graphics engine, level editors, scripting language...
- Assembly
  - Develop all game features and artwork
  - Results in a playable game and tools ready for level design
- Levels
  - Create all the game levels and tweak gameplay as necessary
  - Results in finished game (alpha release)
- Review
  - Extensive play testing
  - Results in gold master (finished product)

Releases

- Alpha:
  - When all the features are in, but not all bugs are out.
- Beta:
  - Development team believes all the bugs are out.
  - No new features except ones to eliminate huge problems.
- Release:
  - No new features!
  - Everything is done.