

# Chapter 56 - Why This Port Is Different

---

This part studies an open-source decompilation-derived porting tree and original Intuition Engine integration work as an IE systems case study. It does not reproduce ROM assets, require the reader to build the port, or teach commercial-game asset extraction.

The case study is useful because it is not a whole-console preservation layer and not a simple profile. It does not try to rebuild another console inside Intuition Engine. It treats Intuition Engine as the target machine.

The game core runs as an M68K programme with an MC68020-class address space and FPU arithmetic. Transform and lighting work is handed to an IE coprocessor service. The checked tree starts that service as an IE64 worker. Voodoo draws the triangles. Native IE audio voices play samples and notes. The file device supplies packed data and saves.

That is the important shape:

```
M68K game core
  |
  | shared RAM requests
  v
IE64 TnL worker ----> transformed vertices
  |
  v
Voodoo command stream and texture upload
  |
  v
Composited IE frame

Sequence and note logic ----> IE SFX voices ----> IE mixer
Input FIFO and File I/O ----> platform contracts ----> game state
```

The source tree contains planning notes for other coprocessor choices. The guide follows the checked implementation: the transform service used by the current integration is IE64.

## 56.1 Not Another Console In A Box

---

A preserved console boundary keeps the old machine intact. This port breaks that boundary on purpose. The old game logic is kept where it is useful, but the heavy platform work is moved onto Intuition Engine devices.

The result is a normal IE programme with an unusual history:

- M68K runs the main game loop.
- The FPU is used for game-side floating-point work.
- Shared RAM carries coprocessor requests.
- Voodoo receives already translated drawing work.
- IE audio engines play the final voices.
- File I/O supplies assets, save data, and test-visible records.

The reader should notice what is missing. There is no reader path that requires a ROM, an extraction step, or an external build. This chapter uses the port as an engineering example only.

## 56.2 Why This Belongs In The Guide

---

Earlier chapters teach the parts separately: M68K, coprocessors, Voodoo, audio, File I/O, input, IE Mon, and IE Script. A large port uses them together.

The case study shows three rules that small examples cannot show as well:

1. Keep control logic close to the game.
2. Move throughput work to hardware-shaped IE services.
3. Measure the actual frame instead of guessing where time went.

## 56.3 The General IE Lesson

---

Intuition Engine rewards ports that choose a new machine boundary. Do not recreate a whole old system when a native IE card can do the job better. Keep the programme's behaviour, but let the backplane, CPUs, Voodoo, audio engines, and file device be the machine it runs on.