

Jefferson Technical Design

Chapter 14: Resources

August 18, 1997

Jefferson is based on plug-in simulation objects, which are implemented as resource file “packages.” As much of the game as possible is data-driven, to provide for this extensibility.

There is also an integration of technologies which bring their own resources, formats, exporters, and management schemes (3D meshes, character animations, etc.)

A considerable amount of the information called for in the outline of this chapter can be found in the *Objects* and *People* chapters.

In this section, we’ll also provide a class-hierarchy diagram of the run-time resource file classes for the core game software.

1. Resource Files

1.1 Generalities

1.1.1 IFF

1.1.2 Macintosh

1.1.3 Windows (any?)

1.1.4 Serialized C++ Objects

1.1.5 Compression

1.2 Resource Types

1.2.1 Sprites

Z buffer, flip, compression, palettes

Groups

- 1.2.2 Palettes**
- 1.2.3 Meshes**
- 1.2.4 Animation**
- 1.2.5 Trees**
- 1.2.6 Sound Effects**
- 1.2.7 Music**
- 1.2.8 Framework UI**
- 1.2.9 (Localized, wide) UI Strings**
 - 1.2.9.1 Menu choices (!)**
 - 1.2.9.2 Control panel labels/tips**
 - 1.2.9.3 Messages**
 - 1.2.9.4 In-game text**
- 1.2.10 Development-time Strings**
- 1.2.11 ...**
- 1.3 *Global (Stock) Resources***
- 1.4 *Object Resources***
 - 1.4.1 Format**
 - 1.4.2 Version**
 - 1.4.3 Resource Chunks**
- 1.5 *Saved Houses and Neighborhoods***
 - 1.5.1 Format**
 - 1.5.2 Version**
 - 1.5.3 Add-Ins**
 - 1.5.4 Instance Data and State**
 - 1.5.4.1 World**
 - 1.5.4.2 Object Properties**
 - 1.5.4.3 Simulation State and Stack**

1.5.4.4 Animation State

1.5.4.5 Custom Characters and Families

1.5.4.6 Thumbnail Views

1.5.4.7 House Summaries (for use as Neighbors)

2. Resource Management

2.1 CD Map

2.1.1 Naming Conventions

2.1.2 Localization

2.2 File Formats

2.3 Installation

2.3.1 Game

2.3.2 Add-Ins

2.3.3 Uninstallation

2.4 Registration and Inventory (Add-Ins)

2.5 Run-Time

2.5.1 Naming Scope/Search

Object, Local, Global ... (my bad guess)

2.5.2 ResFile

2.5.3 Managers/Factories

2.5.4 Resolve

2.5.5 Standard C++ File IO

2.5.6 OS Files

2.5.7 Caching and Paging

2.5.8 On-the-Fly Editing

2.5.8.1 File Locking/Unlocking

2.5.8.2 Change Notification

2.5.8.3 “Re-Initialization”

2.6 Localization

Pops up in several places (CD, Run-time, Install, ...). Centralized discussion?

Procedural aspects (time, money, culture...?)