

X-Ray game asset converter

Материал из xrWiki

Программа converter предназначена для работы с форматами игры и редакторов (LE/AE). Реализует следующие функции:

- преобразование .ogf в .object;
- преобразование скелета из .ogf в .bones;
- преобразование всех анимаций из .ogf или .omf в .skls;
- преобразование отдельных анимаций из .ogf или .omf в .skl;
- преобразование .dm в .object;
- преобразование игровых карт в формат Level Editor;
- распаковка и упаковка игровых архивов;
- множественное преобразование текстур из .dds в .tga;
- множественное преобразование звуков из .ogg в .wav/.thm.

Работает в консоли (Start -> Run -> cmd.exe).

Содержание

- 1 Установка
- 2 Использование
 - 2.1 Преобразование .ogf в .object
 - 2.2 Преобразование скелета из .ogf в .bones
 - 2.3 Преобразование всех анимаций из .ogf или .omf в .skls
 - 2.4 Преобразование отдельных анимаций из .ogf или .omf в .skl
 - 2.5 Преобразование .dm в .object
 - 2.6 Преобразование игровых карт в формат SDK
 - 2.7 Распаковка и упаковка игровых архивов
 - 2.8 Множественное преобразование текстур из .dds в .tga
 - 2.9 Множественное преобразование звуков из .ogg в .wav/.thm

Установка

Для работы с картами и MOD SDK (параметры -level, -dds2tga, -ogg2wav) нужно задать `sdk_root` в **fsconverter.ltx** и положить этот файл в каталог с converter. Аналогично с путями к картам в секциях настроек профилей из файла converter.ini, который можно разместить ещё и в `sdk_root`.

Для работы с остальными функциями утилиты настройка не требуется.

Первый запуск после любых изменений в настройках желательно всегда делать с ключом -го (работа без записи на диск).

Использование

Преобразование .ogf в .object

Команда:

```
lconverter [-ogf] [-object] <оригинал> [-out <результат>]
```

Если ключ `-out <результат>` не задан, имя результирующего файла генерируется из исходного путём замены расширения на `.object`.

Замечания/проблемы:

- для OGF v3 (сборки 1098, 1114, 1154) не восстанавливается `bind pose` скелета (просто потому что не предусмотрена форматом), но часто её можно (и нужно) реконструировать вручную из `idle`-анимации.
- для OGF v4+ (ЧН, в том числе сборка 3120) не восстанавливаются метки в анимациях (неизвестно, как они хранятся в обновлённом `.object`). Кроме того, текущий MOD SDK (SoC) игнорирует информацию о третьей и четвёртой влияющих на вершину костях.
- нет гарантии, что у двухсторонних поверхностей правильно определится "главная" сторона, однако на практике обычно всё хорошо.

Преобразование скелета из .ogf в .bones

Команда:

```
lconverter [-ogf] -bones <оригинал> [-out <результат>]
```

Если ключ `-out <результат>` не задан, имя результирующего файла генерируется из исходного путём замены расширения на `.bones`.

Преобразование всех анимаций из .ogf или .omf в .skls

Команда:

```
lconverter [-ogf|-omf] -skls <оригинал> [-out <результат>]
```

Если ключ `-out <результат>` не задан, имя результирующего файла генерируется из исходного путём замены расширения на `.skls`.

Преобразование отдельных анимаций из .ogf или .omf в .skl

Команда:

```
|-----|
|iconverter [-ogf|-omf] -skl <имя_анимации> <оригинал> [-out <результат>]|
|-----|
```

Если ключ `-out <результат>` не задан, имя результирующего файла генерируется из исходного путём замены расширения на `.skl` и добавления суффикса с именем анимации.

Чтобы преобразовать *все* анимации из `.omf` в отдельные `.skl`, используйте ключ `-skl all` (при этом не обрабатывается случай, когда в `.ogf/.omf` присутствует анимация с именем **all**).

Замечания/проблемы:

- Нельзя задать анимацию, в имени которой используются пробелы.

Преобразование `.dm` в `.object`

Команда:

```
|-----|
|iconverter [-dm] [-object] <оригинал> [-out <результат>]|
|-----|
```

Если ключ `-out <результат>` не задан, имя результирующего файла генерируется из исходного путём замены расширения на `.object`.

Заметка

Информация о границах масштабирования детальной модели теряется, однако можно её посмотреть с помощью ключа `-info` (вместо `-object`).

Преобразование игровых карт в формат SDK

Команда:

```
|-----|
|iconverter -level <номер_сборки>:<имя_карты> [-out <имя_сцены>] [-mode maya|le|le2|raw]|
|-----|
```

Примеры:

```
|-----|
|iconverter -level default:l01_escape -out l01_escape -mode le|
|-----|
```

```
|-----|
|iconverter -level 1844:military_game -out military_game -mode le|
|-----|
```

Ключ `-mode` задаёт режим работы:

- `maya` — декомпилировать только геометрию карты.
- `le` — декомпилировать в формат LE.
- `le2` — декомпилировать в формат LE (для `l12_stancia_2`).
- `raw` — декомпилировать в формат LE (отладочный режим).

Ключ `-with_lods` обрабатывается в зависимости от значения `-mode`. Если он задан совместно с `-mode maya`, программа выдаст ещё и MU-модели. С остальными режимами его наличие заставляет converter разбивать LOD-текстуры, что необходимо для полной пересборки.

Отладочные параметры: `-dbgcfm`, `-dbgmrg`.

Замечания/проблемы:

- Геометрия карт из 1098, 1114, 1154 часто содержит ошибки, на которых валится современный xrLC. По крайней мере часть из них диагностируется при декомпилировании и сохраняется в стандартном формате LE в файле **`logconverter_<имя_сцены>.err`**
- Деревья из ранних сборок (1098-1154) нельзя использовать с современными шейдерами для крупной растительности. Следует использовать современные MU-модели.
- Геометрия карт из сборок 2215-3312 (ТЧ) не содержит ошибок вроде нулевых треугольников или потяжек текстур, однако в редких случаях они могут появиться при пересборке с помощью xrLC на этапе оптимизации. Наблюдается, например, в шахтах лифтов на `l11_pripyat`.
- Геометрия карт из сборки 3456 и выше (ЧН) наоборот, иногда содержит ошибки (с точки зрения xrLC из MOD SDK 0.4).
- Неправильно назначается `compiler shader` для собранных в режиме `draft` карт.
- Абажуры ламп, тайники — фиксированная геометрия, то есть изменения будут видны только после полной пересборки карты.
- Декали (`wallmark.part`) восстанавливаются только в случае наличия избыточного `level.wallmarks`.
- Не обрабатывается случай с пересекающимися областями звукового окружения (`level.snd_env/sound_env.part`). В оригинальной игре карт с таким свойством нет.
- У моделей детальных объектов форсируется флажок `two-sided` для нормальной отрисовки в LE.
- Сетка навигации ИИ (`level.ai/ai_map.part`) некоторых карт (например, `l08_yantar`) утоплена в геометрии. Вопрос о причине открыт.

Распаковка и упаковка игровых архивов

Команда для распаковки:

```
┌-----┐  
|converter -unpack <архив> [-11xx|-2215|-2945|-2947ru|-2947|-xdb] [-flt <маска>]|  
└-----┘
```

Ключ `-flt` служит для извлечения файлов по маске. Примеры:

```
┌-----┐  
|converter -unpack -xdb gamedata.db0 -flt .ogf|  
└-----┘
```

```
┌-----┐  
|converter -unpack -2215 gamedata.xp0 -flt \levels\|  
└-----┘
```

Команда для упаковки:

```
┌-----┐  
|converter -pack <папка> [-2947ru|-2947ww|-xdb|-xdb_ud <файл>]|  
└-----┘
```

Заметка

Используйте ключ `-xdb` для распаковки архивов Чистого Неба и Зова Припяти.

Множественное преобразование текстур из .dds в .tga

Команда:

```
|-----|  
|converter -dds2tga [-with_solid] [-with_bump]|  
|-----|
```

Утилита читает все .thm файлы в *\$textures\$* и воссоздаёт .tga из .dds в *\$game_textures\$*. Уже существующие .tga файлы не перезаписываются. Ключ `-with_solid` предписывает обрабатывать ещё и непрозрачные текстуры, что нужно при пересборке с `-gi`.

Множественное преобразование звуков из .ogg в .wav/.thm

Команда:

```
|-----|  
|converter -ogg2wav|  
|-----|
```

Утилита воссоздаёт .wav/.thm файлы в *\$sounds\$*, используя *.ogg из *\$game_sounds\$*. Уже существующие .wav/.thm не перезаписываются.

Источник — «https://xray-engine.org/index.php?title=X-Ray_game_asset_converter&oldid=915»

Категории:

X-Ray game asset tools

Архиваторы

-
- Страница изменена 4 января 2019 в 19:14.
 - К этой странице обращались 11 907 раз.
 - Содержимое доступно по лицензии GNU Free Documentation License 1.3 или более поздняя (если не указано иное).

