

Формат ARC/DIR и утилита armdir_tool

Полная техническая документация

1 февраля 2026 г.

Содержание

1 Введение	3
2 Общая концепция	3
3 Архитектура системы	3
4 Последовательность выполнения программы	4
5 Интерфейс командной строки	4
5.1 Синтаксис	4
5.2 Режимы	4
5.2.1 EXTRACT	4
5.2.2 PACK	5
5.2.3 PACK_BIN	5
5.3 Примеры использования	5
6 Обнаружение и сортировка файлов	5
7 Структура бинарного формата	5
7.1 Файл DIR	6
7.1.1 Заголовок	6
7.1.2 Запись индекса	6
7.2 Файл ARC	6
7.3 Правила выравнивания	6
7.3.1 Выравнивание строк пути	6
7.3.2 Выравнивание данных файлов	6
8 Алгоритм упаковки	7
9 Алгоритм извлечения	7
10 Обработка путей	7
11 Обработка ошибок	7
12 Детерминированность	8

1 Введение

Настоящий документ представляет систематизированное техническое описание парного архивного формата `.arc/.dir` и утилиты `arcdir_tool`, предназначеннной для извлечения, упаковки и управления архивными данными. Описаны синтаксис командной строки, алгоритмы обработки данных, структура бинарных файлов, правила выравнивания и практические примеры применения.

Утилита реализована на языке C, обеспечивая:

- Высокую производительность благодаря прямому доступу к файлам;
- Кроссплатформенную совместимость (Windows и POSIX-системы);
- Детерминированное поведение при одинаковых входных данных.

2 Общая концепция

Формат архива состоит из двух взаимосвязанных файлов:

- `.arc` — последовательная конкатенация сырых данных файлов;
- `.dir` — индекс с метаданными, включающими смещения, размеры и пути файлов.

Файл `.dir` функционирует как структурированный индекс для `.arc`. Каждая запись сопоставляет путь к файлу с его смещением и размером в ARC-файле.

3 Архитектура системы

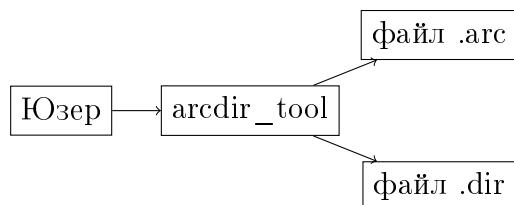


Таблица 1: Общая архитектура утилиты `arcdir_tool`

4 Последовательность выполнения программы

Процесс выполнения утилиты `arcdir_tool` описывается как линейная последовательность действий:

1. Разбор аргументов командной строки и идентификация режима работы: EXTRACT, PACK или PACK_BIN.
2. Режим EXTRACT:
 - Открытие файлов ARC и DIR в режиме чтения;
 - Последовательное чтение записей из файла DIR;
 - Применение optionalных правил перенаправления путей;
 - Запись данных файлов в соответствующие выходные директории.
3. Режимы PACK и PACK_BIN:
 - Рекурсивное сканирование указанных директорий и файлов;
 - Применение фильтра расширений (.bin для режима PACK_BIN);
 - Лексикографическая сортировка всех обнаруженных путей;
 - Запись индекса DIR и последовательное размещение данных файлов в ARC;
 - Применение дополнений для выравнивания строк пути (кратность 4 байтам) и данных файлов (кратность 32 байтам).
4. Закрытие всех файлов и корректное завершение работы.

5 Интерфейс командной строки

5.1 Синтаксис

```
arcdir_tool EXTRACT <archive.arc> <index.dir> [<src> <dst>]...
arcdir_tool PACK <archive.arc> <index.dir> [path...]
arcdir_tool PACK_BIN <archive.arc> <index.dir> [path...]
```

5.2 Режимы

5.2.1 EXTRACT

- Извлечение файлов из существующего архива .arc/.dir.
- При отсутствии пар <src> <dst> извлекаются все файлы по исходным путям.
- При указании пар извлекаются только соответствующие файлы и записываются в целевые директории.

5.2.2 PACK

- Создание новой пары архивных файлов `.arc/.dir`.
- Аргументы могут быть как файлами, так и директориями;
- Директории обрабатываются рекурсивно;
- Все обнаруженные файлы включаются в алфавитном порядке.

5.2.3 PACK_BIN

- Идентичен режиму PACK, но рекурсивно найденные файлы фильтруются по расширению `.bin`;
- Явно указанные файлы включаются независимо от расширения.

5.3 Примеры использования

```
# Extract all files
arcdir_tool EXTRACT data.arc data.dir

# Extract a single file to a custom path
arcdir_tool EXTRACT data.arc data.dir a/b.bin out.bin

# Pack a directory recursively
arcdir_tool PACK data.arc data.dir assets/

# Pack only .bin files from directories
arcdir_tool PACK_BIN data.arc data.dir assets/
```

6 Обнаружение и сортировка файлов

Процедура упаковки файлов включает следующие этапы:

1. Анализ каждого аргумента командной строки;
2. Рекурсивное сканирование директорий;
3. Непосредственное добавление файлов;
4. Применение опционального фильтра по расширению;
5. Лексикографическая сортировка результирующего списка файлов.

7 Структура бинарного формата

Все числовые поля представлены в формате big-endian.

7.1 Файл DIR

7.1.1 Заголовок

Смещение	Размер	Описание
0x00	4 байта	Общий размер файла DIR (u32, big-endian)
0x04	4 байта	Количество записей (u32, big-endian)

Таблица 2: Структура заголовка файла DIR

7.1.2 Запись индекса

Каждая запись имеет структуру:

Поле	Размер	Описание
Offset	4 байта	Смещение начала данных файла в ARC (u32, big-endian)
Size	4 байта	Размер данных файла в байтах (u32, big-endian)
PathLen	4 байта	Длина строки пути с учётом выравнивания (u32, big-endian)
Path	PathLen байт	Нуль-терминированная строка пути, выровненная до 4 байт

Таблица 3: Структура записи файла DIR

7.2 Файл ARC

- Последовательная запись сырых данных файлов;
- Добавление выравнивающего дополнения после каждого файла для обеспечения кратности 32 байтам;
- Дополнение заполняется байтами 0xCC.

7.3 Правила выравнивания

7.3.1 Выравнивание строк пути

```
pad = (~len - 1) & (4 - 1);
total_len = len + pad;
```

7.3.2 Выравнивание данных файлов

```
pad = (~data_size - 1) & (32 - 1);
```

8 Алгоритм упаковки

1. Открытие ARC и DIR в режиме записи;
2. Резервирование 8 байт в DIR для заголовка;
3. Для каждого отсортированного файла:
 - Считывание содержимого файла в оперативную память;
 - Формирование записи DIR: смещение в ARC, размер файла, длина пути с выравниванием, строка пути с выравниванием;
 - Последовательная запись данных файла в ARC;
 - Применение выравнивающего дополнения до кратности 32 байтам с использованием 0xCC.
4. Запись общего размера файла DIR и количества записей в начале файла.

9 Алгоритм извлечения

1. Открытие ARC и DIR в режиме чтения;
2. Чтение заголовка DIR (общий размер и количество записей);
3. Для каждой записи:
 - Чтение смещения, размера и длины пути;
 - Чтение строки пути и удаление выравнивающего дополнения;
 - Применение опциональных правил перенаправления;
 - Переход к смещению в ARC;
 - Чтение данных файла и запись в целевую директорию.

10 Обработка путей

- Пути представлены как байтовые строки, совместимые с кодировкой Shift-JIS;
- Разделители директорий нормализуются к символу '/' при упаковке;
- Таблицы перенаправления применяются опционально во время извлечения.

11 Обработка ошибок

- Чтение фиксированного количества байт; обнаружение EOF вызывает ошибку;
- Размер заголовка DIR используется для проверки корректности;
- Ошибки ввода/вывода приводят к немедленному завершению работы программы.

12 Детерминированность

Для идентичных входных данных:

- Структура записей DIR и размещение данных в ARC воспроизводимы;
- Лексикографическая сортировка обеспечивает детерминированность;
- Фиксированные правила выравнивания сохраняют корректное смещение данных.

13 Заключение

Формат ARC/DIR представляет собой индексированный контейнер данных:

- `.arc` содержит последовательные блоки данных;
- `.dir` содержит метаданные, сопоставляющие пути файлов со смещениями и размерами;
- Используются целые числа в формате big-endian и строгие правила выравнивания;
- Утилита `arcdir_tool` обеспечивает полное управление упаковкой и извлечением архивов с детерминированным результатом.