Метаданные

- Файлы метаданных
- Языковые ноды
- Ресурсные ноды
- Ассеты
- Списки

Файлы метаданных

Игра получает информацию о моде из следующих файлов:

1. `meta.yaml`
2. `resources.yaml`

`meta.yaml` отвечает за информацию о моде во внутриигровом мод-селекторе;
`resources.yaml` — за всё остальное — это основной файл, который вам придётся редактировать чаще всего.

Оба файла должны быть расположены в корневой директории мода, содержимое файлов должно иметь корректный YAML-синтаксис и верно указанные параметры (об этом будет далее).

YAML достаточно простой и интуитивный, но если у вас возникли сложности, можете воспользоваться специальным ресурсом для валидации YAML-синтаксиса.

> В YAML нельзя использовать табуляцию — используйте пробелы.

Языковые ноды

Некоторые параметры могут иметь переводы на несколько языков, такие параметры далее будут называться **языковыми нодами**.

Значением такого параметра может быть просто текст:

```
title: Название мода
```

В таком случае `Название мода` будет отображаться на русском для всех языков.

Чтобы локализировать параметр для разных языков, нужно указать код языка и текст:

```
title:
    ru: Название мода
    en: Mod title
    common: Текст, который будет выводиться для любого другого языка*, перевод для которого не указан
```

Поддерживаемые языковые коды:

- `ru` — русский
- `en` — английский

* В игре на данный момент всего два языка — русский и английский.

Ресурсные ноды

Большая часть параметров в `resources.yaml` определяет какие-либо внешние ресурсы, используемые в игре: фоны, звуки, спрайты персонажей и их имена, цвета, глобальные переменные и прочее. Такие параметры для удобства будут называться **ресурсными нодами**.

Каждому такому параметру должен быть назначен уникальный ID — имя, на которое вы будете ссылаться при написании сценария для использования данного ресурса.

У таких имён есть ограничения на используемые символы, а именно: **допускаются только латинские (английские) символы, цифры и нижнее подчёркивание.** Использовать кириллицу, пробелы или какие-то либо другие символы нельзя.

Пример определения фона с ID **`bus_stop`** внутри `resources.yaml`:

```
# resources.yaml
---
  bg:
    bus_stop: bg/bus_stop.jpg
```

Строчка **`bus_stop: bg/bus_stop.jpg`** *— ресурсный нод, содержащий путь к ассету.*

Пример определения цвета с ID **`dark_pink`**:

```
# resources.yaml
---
  colors:
    dark_pink: #b00b69
```

ID регистронезависимы

Ассеты

Пути ассетов

Различные ноды часто принимают в качестве аргумента путь к какому-либо ассету (файлу).

Все пути к ассетам, загружаемым из директории мода, должны быть относительными, т.е. должны указываться без пути к корневой папке мода.

Например, у нас есть обложка мода — **cover.jpg**, которая лежит в папке images. Корректный путь к ней будет выглядеть так:
`images/cover.jpg`

Если же обложка лежит в корневой директории, достаточно просто указать имя файла:
`cover.jpg`

> Если ваш мод находится внутри **.scenario**-архива, учитывайте регистр при указании путей. Также регистр важен на **UNIX**-системах (Linux, macOS, Android и т.п.).
>
> Если ваш файл называется **Cover.jpg**, а в пути вы указали **cover.jpg** — будет выдана ошибка.

Виды ассетов

Всего существует 3 основных вида ассетов, их можно загружать из директории мода и из Unity-каталогов:

- `texture` — графический ассет
- `text` — текстовый ассет
- `sound` — звуковой ассет

Также существует ещё 2 вида, которые можно загружать только из Unity-каталогов:

- `prefab` — Unity-префаб
- `sprite` — Unity-спрайт

У каждого параметра, принимающего ассет(ы) в качестве аргумента, будут указаны виды ассетов, которые он принимает.

Списки

Большая часть основных параметров — это списки, т.е., ноды, которые содержат в себе наборы других однотипных нодов.

Например:

```
# resources.yaml
---
  bg:
    bus_stop: bg/bus_stop.jpg
    ext_road_day: bg/ext_road_day.jpg

  sound:
    blow_with_the_fires: sound/blow_with_the_fires.ogg
    no_tresspassing: sound/no_tresspassing.ogg
```

Параметр **bg** *и* **sound** *в данном примере — списки.*