

Официальная спецификация формата A3D (rus)

- Введение
- Структура файла
 - Длина пакета
 - Null-маска
 - Короткая null-маска
 - Длинная null-маска
 - Данные сообщения
 - Представление массивов и строк
 - Версия
 - A3D2
 - A3D2AmbientLight
 - A3D2AnimationClip
 - A3D2Track
 - A3D2Box
 - A3D2CubeMap
 - A3D2Decal
 - A3D2DirectionalLight
 - A3D2Image
 - A3D2IndexBuffer
 - A3D2Joint
 - A3D2Map
 - A3D2Material
 - A3D2Mesh
 - A3D2Object
 - A3D2OmniLight
 - A3D2SpotLight
 - A3D2Sprite
 - A3D2Skin
 - A3D2JointBindTransform
 - A3D2Keyframe
 - A3DMatrix
 - A3D2Surface
 - A3D2Transform
 - A3D2VertexBuffer
 - A3D2VertexAttributes
 - ByteBuffer
 - Структура
 - Пример

Введение

A3D – формат, разработанный специалистами компании AlternativaPlatform, предназначенный для хранения трёхмерных сцен в виде бинарных файлов. Двоичное представление позволяет передавать данные без какого-либо преобразования, существенно ускоряя скорость передачи.

В данной спецификации описаны общие особенности, которые необходимо учитывать при работе с файлами в формате A3D.

Структура файла

Файл в формате A3D представляет из себя пакет. Максимальный размер данных, которые можно упаковать в пакет - 2 147 483 648 байт (в ситуации с упаковкой это ограничение накладывается после упаковки). Пакет имеет следующую структуру:



Длина пакета



L-флаг размера длины пакета (1 бит)

L	Описание
0	длина пакета содержится в оставшихся 6-ти битах флагового байта и последующем байте (всего 14 бит), вторым битом указан признак упаковки
1	длина пакета содержится в оставшихся 7-ти битах флагового байта и последующих 3-х байтах (всего 31 бит). Признак упаковки принимется равным true

Z-флаг упаковки, указывается при L=0 (1 бит)

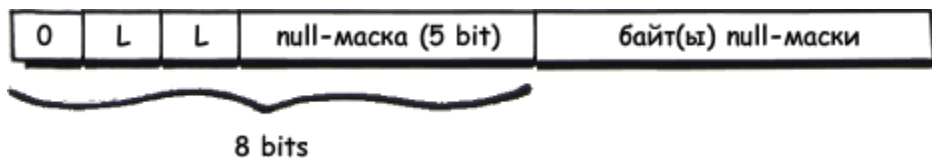
Z	Описание
0	данные пакета не упакованы
1	данные пакета упакованы (gzip)

Null-маска

Битовая маска, каждый бит которой определяет наличие или отсутствие null объектов в сообщении. Длина маски указывается в байтах.

Короткая null-маска

Кодирует null-маску до 29-ти бит. Первый бит первого байта установлен в 0.



LL-длина null-маски (2 бита)

LL	Описание
00	null-маска содержится в оставшихся 5-ти битах первого байта (всего 5 бит)
01	null-маска содержится в оставшихся 5-ти битах первого байта и последующем байте (всего 13 бит)
10	null-маска содержится в оставшихся 5-ти битах первого байта и последующих 2-х байтах (всего 21 бит)
11	null-маска содержится в оставшихся 5-ти битах первого байта и последующих 3-х байтах (всего 29 бит)

Длинная null-маска

Кодирует null-маску до 33 554 432-х бит. Первый бит первого байта установлен в 1.



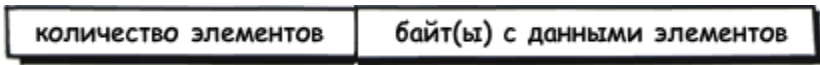
L-флаг размера длины маски (1 бит)

L	Описание
0	длина маски содержится в оставшихся 6-ти битах первого байта (всего 6 бит)
1	длина маски содержится в оставшихся 6-ти битах первого байта и двух последующих байтах (всего 22 бита)

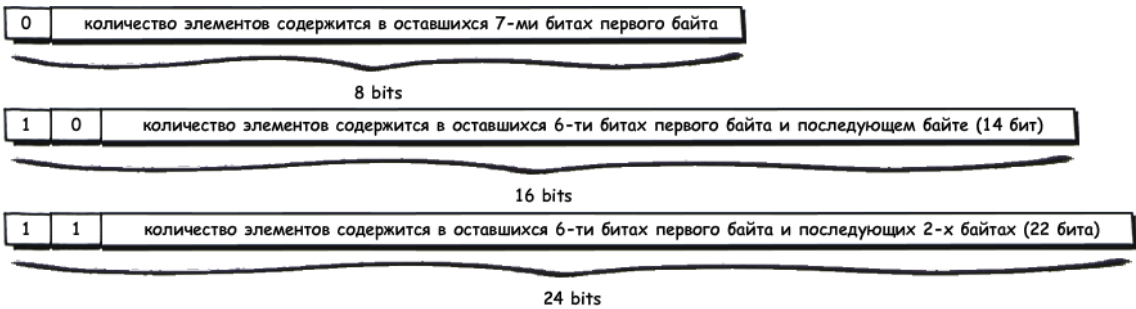
Данные сообщения

Представление массивов и строк

Строки в A3D - это массивы символов в формате UTF-8. Общая структура кодирования массивов/строк:



Максимальное количество элементов массива составляет 4 194 304, символов в строке - 2 097 152. Существует 3 варианта кодирования количества элементов:



Данные элементов хранятся в [свойствах](#).

Версия

Поля	Тип	Размер (в байтах)	Дополнительно
Major version	ushort	2	2
Minor version	ushort	2	0

A3D2

Одним из главных требований формата является соблюдение логики расположения свойств. Они указываются в порядке, указанном в следующей таблице:

Свойство	Тип
----------	-----

ambientLights	массив A3D2AmbientLight
animationClips	массив A3D2AnimationClip
animationTracks	массив A3D2Track
boxes	массив A3D2Box
cubeMaps	массив A3D2CubeMap
decals	массив A3D2Decal
directionalLights	массив A3D2DirectionalLight
images	массив A3D2Image
indexBuffers	массив A3D2IndexBuffer
joints	массив A3D2Joint
maps	массив A3D2Map
materials	массив A3D2Material
meshes	массив A3D2Mesh
objects	массив A3D2Object
omniLights	массив A3D2OmniLight
spotLights	массив A3D2SpotLight
sprites	массив A3D2Sprite
skins	массив A3D2Skin
vertexBuffers	массив A3D2VertexBuffer

Свойства, отмеченные жирным шрифтом являются обязательными и должны быть указаны. Остальные свойства опциональны. Если опциональное свойство не указывается, то в **null-маске** оно должно быть отмечено 1 (1 бит). Если опциональное свойство указывается, то в **null-маске** оно должно быть отмечено 0 (1 бит).

A3D2AmbientLight

См. также [A3D2](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
boundingBoxId	int	4	A3D2Box
color	uint	4	
id	int64	8	
intensity	float	4	
name	String		массив char(UTF-8)
parentId	int64	8	
transform	A3D2Transform		
visible	bool	1	

A3D2AnimationClip

См. также [A3D2](#), [A3D2Track](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
id	int	4	
loop	bool	1	
name	String		массив char(UTF-8)
objectIDs	массив int64		
tracks	массив int	4	

A3D2Track

См. также [A3D2](#), [A3D2AnimationClip](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
id	int	4	
keyframes	массив A3D2Keyframe		
objectName	String		массив char(UTF-8)

A3D2Box

См. также [A3D2Object](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
box	массив float		[minX, minY, minZ, maxX, maxY, maxZ]
id	int	4	

A3D2CubeMap

См. также [A3D2Material](#)

Свойство	Тип	Размер (в байтах)
backId	int	4
bottomId	int	4
frontId	int	4
id	int	4
leftId	int	4
rightId	int	4
topId	int	4

A3D2Decal

См. также [A3D2](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
boundingBoxId	int	4	A3D2Box
id	int64	8	
indexBufferId	int	4	
name	String		массив char(UTF-8)
offset	float	4	
parentId	int64	8	
surfaces	массив A3D2Surface		
transform	A3D2Transform		
vertexBuffers	массив int		
visible	bool	1	

A3D2DirectionalLight

См. также [A3D2](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
boundingBoxId	int	4	A3D2Box
color	uint	4	
id	int64	8	
intensity	float	4	
name	String		массив char(UTF-8)
parentId	int64	8	
transform	A3D2Transform		
visible	bool	1	

A3D2Image

См. также [A3D2Material](#), [A3D2Map](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
id	int	4	
url	String		массив char(UTF-8)

A3D2IndexBuffer

См. также [A3D2](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
----------	-----	-------------------	---------------

byteBuffer	массив byte		каждый индекс занимает 2 байта (little-endian)
id	int	4	
indexCount	int	4	max 524287

A3D2Joint

См. также [A3D2](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
boundingBoxId	int	4	A3D2Box
id	int64	8	
name	String		массив char(UTF-8)
parentId	int64	8	
transform	A3D2Transform		
visible	bool	1	

A3D2Map

См. также [A3D2Material](#)

Свойство	Тип	Размер (в байтах)
channel	ushort	2
id	int	4
imageId	int	4

A3D2Material

См. также [A3D2Surface](#)

Свойство	Тип	Размер (в байтах)
diffuseMapId	int	4
glossinessMapId	int	4
id	int	4
lightMapId	int	4
normalMapId	int	4
opacityMapId	int	4
reflectionCubeMapId	int	4
specularMapId	int	4

A3D2Mesh

См. также [A3D2](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
boundingBoxId	int	4	A3D2Box
id	int64	8	
indexBufferId	int	4	
name	String		массив char(UTF-8)
parentId	int64	8	
surfaces	массив A3D2Surface		
transform	A3D2Transform		
vertexBuffers	массив int		
visible	bool	1	

A3D2Object

См. также [A3D2](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
boundingBoxId	int	4	A3D2Box
id	int64	8	
name	String		массив char(UTF-8)
parentId	int64	8	
transform	A3D2Transform		
visible	bool	1	

A3D2OmniLight

См. также [A3D2](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
attenuationBegin	float	4	
attenuationEnd	float	4	
boundingBoxId	int	4	A3D2Box
color	uint	4	
id	int64	8	
intensity	float	4	
name	String		массив char(UTF-8)
parentId	int64	8	

transform	A3D2Transform		
visible	bool	1	

A3D2SpotLight

См. также [A3D2](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
attenuationBegin	float	4	
attenuationEnd	float	4	
boundingBoxId	int	4	A3D2Box
color	uint	4	
falloff	float	4	
hotspot	float	4	
id	int64	8	
intensity	float	4	
name	String		массив char(UTF-8)
parentId	int64	8	
transform	A3D2Transform		
visible	bool	1	

A3D2Sprite

См. также [A3D2](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
alwaysOnTop	bool	1	
boundingBoxId	int	4	A3D2Box
height	float	4	
id	int64	8	
materialId	int	4	
name	String		массив char(UTF-8)
originX	float	4	
originY	float	4	
parentId	int64	8	
perspectiveScale	bool	1	
rotation	float	4	
transform	A3D2Transform		
visible	bool	1	

width	float	4	
--------------	-------	---	--

A3D2Skin

См. также [A3D2](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
boundingBoxId	int	4	A3D2Box
id	int64	8	
indexBufferId	int	4	
jointBindTransforms	массив A3D2JointBindTransform		
joints	массив int64		
name	String		массив char(UTF-8)
numJoints	массив ushort		
parentId	int64	8	
surfaces	массив A3D2Surface		
transform	A3D2Transform		
vertexBuffers	массив int	4	
visible	bool	1	

каждый элемент массива numJoints - это количество костей на поверхность, а сами индексы костей хранятся в массиве joints. К примеру, если есть 2 поверхности в коже - на первой 4 кости с idшниками 0,2,3,4, а на второй 3 кости с idшниками 1,2,5. Соответственно, в векторе joints будут: [0,2,3,4,1,2,5], а в векторе numJoints [4,3]

A3D2JointBindTransform

См. также [A3D2Skin](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
bindPoseTransform	A3D2Transform		
id	int64	8	id кости

A3D2Keyframe

См. также [A3D2](#), [A3D2Track](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
time	float	4	
transform	A3D2Transform		

A3DMatrix

См. также [A3D2Transform](#)

Свойство	Тип	Размер (в байтах)
a	float	4
b	float	4
c	float	4
...		
l	float	4

A3D2Surface

См. также [A3D2Mesh](#), [A3D2Material](#)

Свойство	Тип	Размер (в байтах)
indexBegin	int	4
materialId	int	4
numTriangles	int	4

A3D2Transform

См. также [A3D2Object](#)

Свойство	Тип
matrix	A3DMatrix

A3D2VertexBuffer

См. также [A3D2](#), [A3D2VertexAttributes](#)

Свойство	Тип	Размер (в байтах)	Дополнительно
attributes	массив int		A3D2VertexAttributes
byteBuffer	ByteBuffer		little-endian float
id	int	4	
vertexCount	ushort	2	

A3D2VertexAttributes

перечисляемый тип

POSITION = 0, размер в byteBuffer (3 поля типа float - x,y,z)

NORMAL = 1, размер в byteBuffer (3 поля типа float - x,y,z)

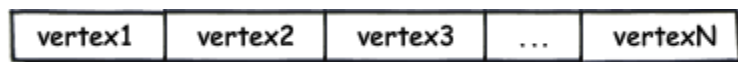
TANGENT4 = 2, размер в byteBuffer (4 поля типа float - tangent.x, tangent.y, tangent.z, binormalDirection)

JOINT = 3, размер в byteBuffer (4 поля типа float - jointA.index, jointA.weight, jointB.index, jointB.weight)

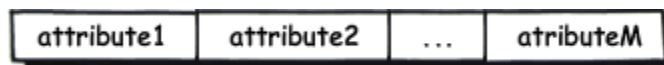
TEXCOORD = 4, размер в byteBuffer (2 поля типа float - u, v)

ByteBuffer

Структура



vertex:



каждый атрибут имеет размерность, соответствующую типу атрибута.

Порядок, тип и количество атрибутов задается массивом attributes.

Пример

3 вершины, для каждой заданы позиция, нормаль и текстурные координаты.

1 вершина:

- позиция = 10 20, 30
- нормаль = 0, 0, 1
- текстурные координаты 0, 0

2 вершина:

- позиция = 15, 20, 30
- нормаль = 0, 0, 1
- текстурные координаты 0, 1

3 вершина:

- позиция = 30, 0, 20
- нормаль = 0, 0, 1
- текстурные координаты 1, 1

```
attributes = [0,1,4]

vertexCount = 3

byteBuffer = [[10,20,30] [0,0,1] [0,0]] [[15,20,30] [0,0,1] [0, 1]] [[30,0,20] [0,0,1] [1,1]]
```

В итоге byteBuffer будет иметь вид:

10,20,30,0,0,1,0,0,15,20,30,0,0,1,0,1,30,0,20,0,0,1,1,1

каждый элемент записывается в формате little endian float