orxANIMSET structure

Summary

Animation Set

```
[AnimationSetTemplate]
Frequency     = <float>
KeepInCache   = <bool>
StartAnim     = Animation
StartAnimList = Animation1 # ... # AnimationN
Prefix        = <string>
Digits        = <int>
Animation     = <string>
Animation     = <string>
Animation     = empty
Animation->   = {.!+-}Animation1 # ... # {.!+-}AnimationN
```

Animation and Animation<N>

```
[AnimationTemplate]
Texture       = path/to/ImageFile.ext
TextureOrigin = <vector>
TextureSize   = <vector>
Text          = TextTemplate
Direction     = left | right | down | up # left | right | down | up
FrameSize     = <vector>
KeyDuration   = <float>

[AnimationTemplate<N>] ; eg: AnimationTemplate0003
KeyEvent      = <string>
TextureOrigin = <vector>
TextureSize   = <vector>
FrameIndex    = <int>
String        = "string to display"
KeyDuration   = <float>
```

Details

Animation set

This `AnimationSet` section will become the parent of any `Animation` section used by it. So they will become common properties to all the animations or animation frames. Properties such as `FrameSize`, `Texture`, `KeyDuration` or `Direction` can be defined

here.

Here's a list of the available properties for an animation set:

- `Frequency`: Relative frequency to which the animations will be played. Defaults to 1.0, which means the animation will be played with the timing defined in the config.
- `KeepInCache`: If true, the animation set will always stay in cache, even if not currently used by any objects. Can save time but costs memory. Defaults to false.
- `StartAnim`: Entry point of the animation graph, first animation to be played by default. If this StartAnim is defined, StartAnimList will be ignored.
- `StartAnimList`: Multiple entry points for disjointed animation graphs. The first animation in the list will be played by default. Ignored if StartAnim is defined.
- `Prefix`: Prefix that will used in front of all animation names and frames, ie: 0001. Defaults to empty string.
- `Digits`: How many digits will be used for the animation frame suffixes. Defaults to 0 (ie. as many digits as needed, without any prefixed 0).
- `Animation`: The first value, optional, defines a name for the animation config template to use, defaults to the key itself. The second value, optional, specifies how many frames should be created: >0 ⇒ maximum frames to be loaded, -1 ⇒ as many frames as can fit inside the whole texture, 0 ⇒ as many frames as are defined in config, defaults to -1.
- `Animation`: In this mode, separate texture files are expected for all the frames. The first value defines the animation base file name, the second one defines the file extension.
- `Animation`: In this mode, an empty animation (with no keys) will be created.
- `Animation→`: Defines all the possible transitions (ie. links) coming from Animation. The optional prefixes are properties for the link: . ⇒ immediate, ! ⇒ clear target, + ⇒ high priority, - ⇒ low priority. Defaults to looping onto itself if not defined.

Animation

NB: Do not use manual inheritance at the section level as it will be overridden at runtime; it's parent will then become the animation set section.

Here's a list of the available properties for an animation:

- `Texture`: path/to/ImageFile.ext See GraphicTemplate for more details on this property. If provided, Text property will be ignored.
- `TextureOrigin`: Defines the origin of the texture from which to get the frames. Defaults to top left corner, ie. (0, 0, 0). Unused in the case of Text animation.
- `TextureSize`: Defines the size of the texture from which to get the frames. Defaults to the actual physical size of the texture. Unused in the case of Text animation.
- `Text`: Text animations are supported provided that a Text property is defined at the animation level (here) and no Texture is defined. In this case, the animation should be defined with a value of 0 in the animation set (ie. retrieve frame data from config).
- `Direction`: Defines how the frames are read inside the texture. First value defines the

direction of the rows, second one the direction of the columns. Defaults to right # down (ie. western writing direction). Examples: left|right|down|up # left|right|down|up
- `FrameSize`: Default size of all the frames of this animation. Unused in the case of Text animation. If not defined in the case of a single frame animation, defaults to the same value as TextureSize.
- `KeyDuration`: Default key duration for all the frames of this animation.

Animation<N>

For example: Animation0004 or Animation0005 etc

> NB: Do not use manual inheritance at the section level as it will be overridden at runtime: its parent will then become the animation section.

Here's a list of the available properties for an animation frame override:

- `KeyEvent`: Optional. Defines a custom event that will be sent when this key is played. The first value is the event's name, the second one, the event's numerical value, is optional and defaults to 0.0.
- `TextureOrigin`: Optional. Overrides the computed origin of the frame. Only define it for manual override. If this is defined, FrameIndex will be ignored.
- `TextureSize`: Optional. Overrides the default size of the frame (the FrameSize attribute in the animation/Animation Set). Only define it for manual override.
- `FrameIndex`: Optional. Overrides the current frame's texture origin based on the animation's FrameSize and Direction. Only define it for manual override. Ignored if TextureOrigin is defined for this frame.
- `String`: Optional. Only used for Text animations.
- `KeyDuration`: Optional. Overrides the default key duration of the frame.
- *Any other Graphic property like Pivot or Flip can be used here.*

Examples:

See: Animation Walk-through.

General things to note on the Animation System

You can listen to the `orxEVENT_TYPE_ANIM` events if you still want to be notified when transitions occur.

Between two animations, the calculated path will take the highest priority links at each step and if two links with the same priority are found, it will then take the one that leads to the shortest path (ie. the least amount of links needed to reach destination).

NB: If you don't feel at ease with this graph system, you can still define all your animations separately and then do all the transitions manually every time the current animation has been completely played (listening to the `orxANIM_EVENT_STOP` event, for

example).

Latest config settings for the Development Version

We endeavor to keep the config properties on this page up to date as often as possible. For up to the minute config information for the latest version of Orx, check the most recent published at:

CreationTemplate.ini and

SettingsTemplate.ini

Additionally these files can be found under your orx source tree in the `orx/code/bin` folder.

From:
https://www.orx-project.org/wiki/ - **Orx Learning**

Permanent link:
https://www.orx-project.org/wiki/en/orx/config/settings_structure/orxanim?rev=1695191733

Last update: **2025/09/30 17:26 (14 months ago)**