

Doomsday Engine - Bug #1906

Light decorations not changing on animated textures

2014-11-20 22:10 - vermil

Status: Closed	Start date: 2014-11-20
Priority: Normal	% Done: 100%
Assignee: danij	
Category: Regression	
Target version:	
Description	
Light decorations on animated textures do not update as the texture animates; the dynamic lights of the first stage of the animation are simply displayed throughout the entire animation.	
This is actually a long long standing issue in Dday 1.9.x; I thought I reported back it when I noticed it long ago, but looking back I forgot to do so. My apologies.	

Associated revisions

Revision 8252f7bc - 2015-02-09 21:17 - danij

Definitions|libdoomsday: Working on interpretation of old style Decoration defs

Animation is now mostly working although there appears to be some timing issues remaining...

IssueID #1906

Revision ecc042ae - 2015-02-12 16:01 - danij

Fixed|Definitions|libdoomsday: Old style Decoration def interpretation

This kind of null stage optimization will require an object level controller so lets keep things simple for now.

IssueID #1906

History

#1 - 2014-11-21 19:49 - danij

- Tags set to Lights, Decorations, Client
- Category set to Regression
- Target version set to 49

#2 - 2014-12-06 00:22 - danij

Seemingly this is a compatibility issue with old style linked Decoration definitions vs Materials. The new style decorations defined with Material.Light appear to work correctly.

#3 - 2014-12-06 02:58 - danij

- Status changed from New to In Progress
- Assignee set to danij
- % Done changed from 0 to 10

#4 - 2014-12-06 10:42 - danij

- % Done changed from 10 to 40

#5 - 2014-12-07 12:08 - danij

- % Done changed from 40 to 60

#6 - 2015-02-07 22:41 - danij

- % Done changed from 60 to 70

#7 - 2015-02-09 12:55 - danij

- % Done changed from 70 to 80

#8 - 2015-02-09 19:01 - danij

- % Done changed from 80 to 90

#9 - 2015-02-09 23:21 - danij

With my latest fix in place, old style decoration definitions are being interpreted correctly.

However, there appears to be a general timing issue with the decorations in that they are never in sync with the texture animation. Clearly something is going wrong at animator level.

Timing test with a dead simple red > green > blue animation:

```
Decoration {
  Material = "flats:nukage1";
  Light {
    Color { 1, 0, 0 };
    Offset { 32, 32 };
    Radius = 1;
  };
};
```

```
Decoration {
  Material = "flats:nukage2";
  Light {
    Color { 0, 1, 0 };
    Offset { 32, 32 };
    Radius = 1;
  };
};
```

```
Decoration {
  Material = "flats:nukage3";
  Light {
    Color { 0, 0, 1 };
    Offset { 32, 32 };
    Radius = 1;
  };
};
```

As above using new style Material.Light defs (interpolated):

```
Material Mods "flats:nukage1" {
  Light {
    Stage { Tics = 8; Color { 1, 0, 0 }; Offset { 32, 32 }; Radius = 1; }
    Stage { Tics = 8; Color { 0, 1, 0 }; Offset { 32, 32 }; Radius = 1; }
    Stage { Tics = 8; Color { 0, 0, 1 }; Offset { 32, 32 }; Radius = 1; }
  };
};
```

```
Material Mods "flats:nukage2" {
  Light {
    Stage { Tics = 8; Color { 0, 1, 0 }; Offset { 32, 32 }; Radius = 1; }
    Stage { Tics = 8; Color { 0, 0, 1 }; Offset { 32, 32 }; Radius = 1; }
    Stage { Tics = 8; Color { 1, 0, 0 }; Offset { 32, 32 }; Radius = 1; }
  };
};
```

```
Material Mods "flats:nukage3" {
  Light {
    Stage { Tics = 8; Color { 0, 0, 1 }; Offset { 32, 32 }; Radius = 1; }
    Stage { Tics = 8; Color { 1, 0, 0 }; Offset { 32, 32 }; Radius = 1; }
    Stage { Tics = 8; Color { 0, 1, 0 }; Offset { 32, 32 }; Radius = 1; }
  };
};
```

Both methods exhibit the same timing issues.

#10 - 2015-02-10 13:49 - danij

Instrumenting MaterialAnimator reveals animation of decorations and layers are in fact in sync, down to the millisecond. Perhaps an interpolation issue?

#11 - 2015-02-10 14:31 - danij

Yes, it is indeed an interpolation issue. Specifically, the `de::lerp<VectorN>()`'s in `MaterialAnimator::Decoration::update()` seemingly return the wrong values. I wonder if this is MSVC specific?

Edit: `de::lerp<>()` is working fine with the `VectorN` types. It would seem the calculation of the interpolation point is wrong.

#12 - 2015-02-12 16:08 - danij

- *Status changed from In Progress to Closed*

- *% Done changed from 90 to 100*

#13 - 2015-06-08 10:47 - skyjake

- *Target version deleted (49)*