

idc16

Imagination Developers Connection

Advanced Topics for PowerVR Ray Tracing



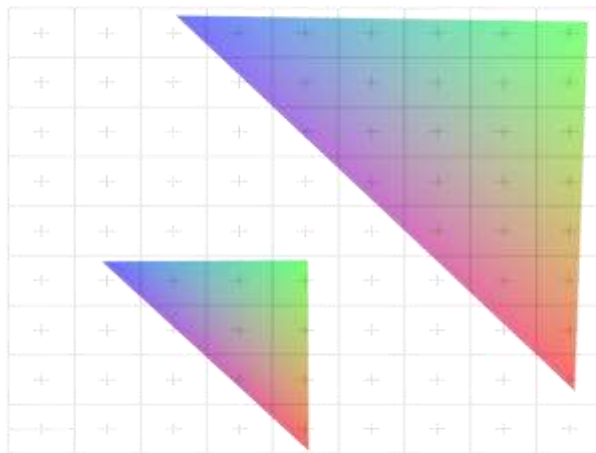


Gareth Morgan
Luke Peterson



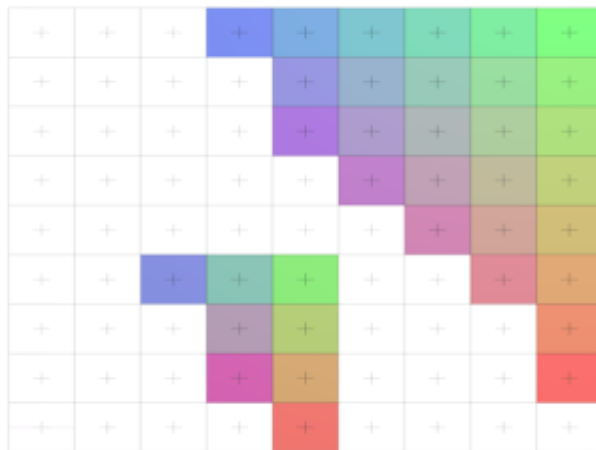
Anti-aliasing with a ray tracer

Similar to traditional rasterization - but a bit different

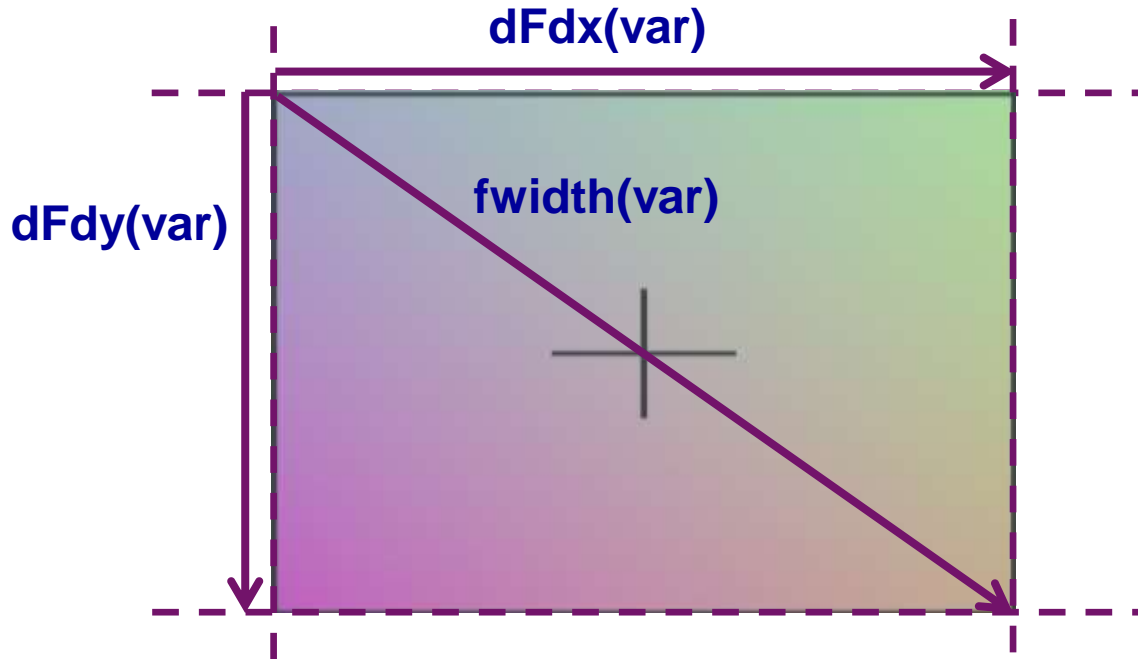


A pixel is not a point sample, but an integral

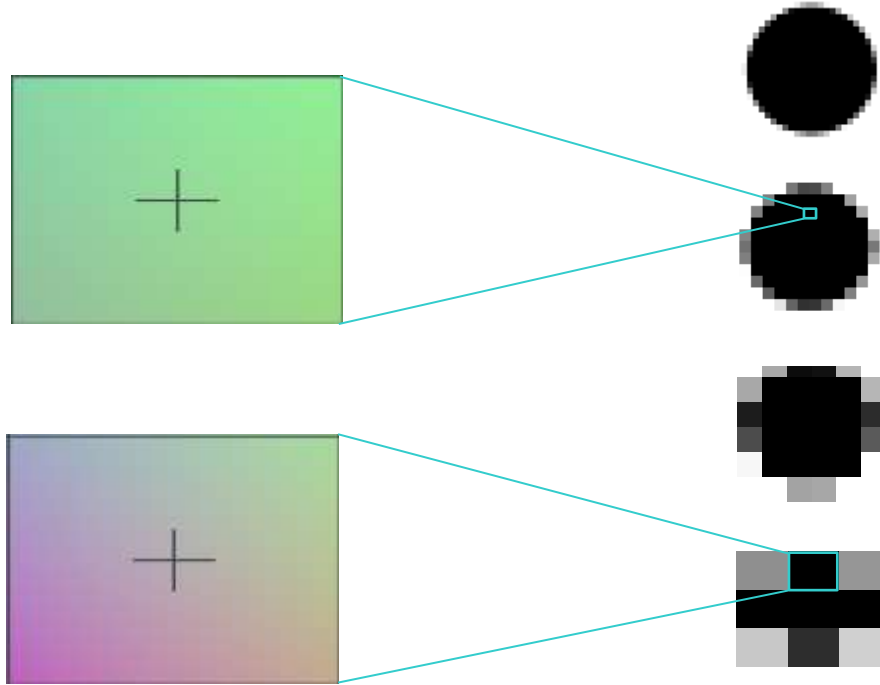
Aliasing occurs when scene features have higher spatial frequency than pixel res



Pixel awareness with derivative functions

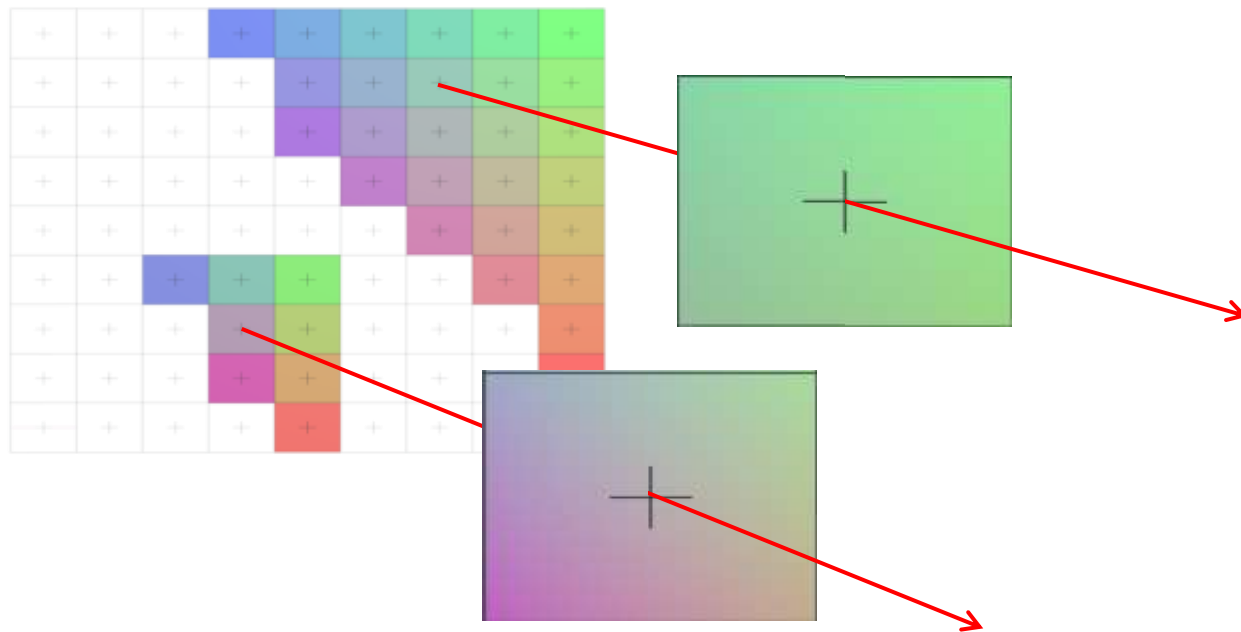


Mip Mapping automatically uses derivatives

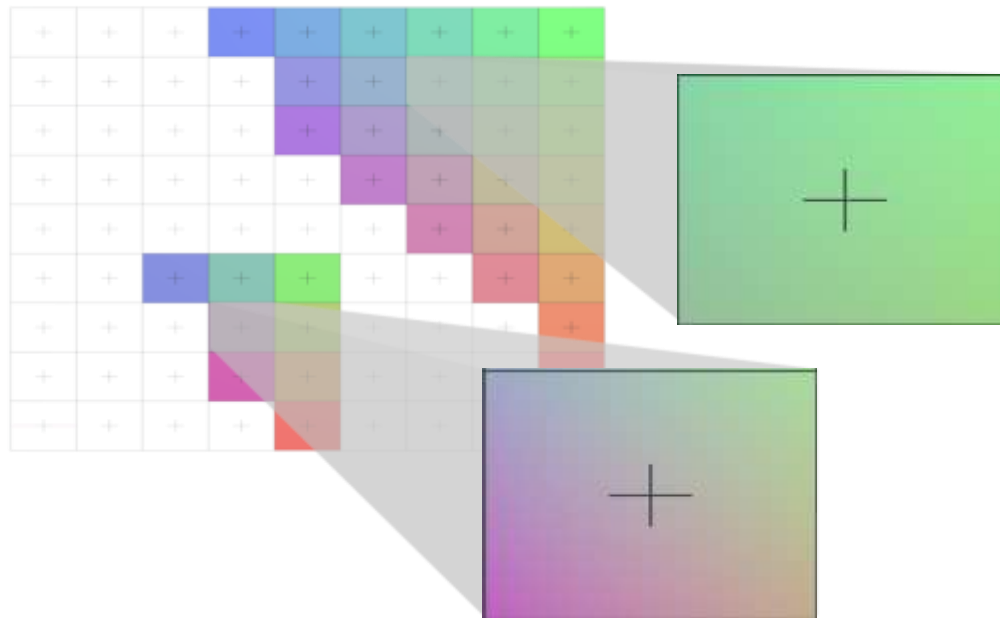


But a ray *is* a point sample, by definition

Similar to traditional rasterization - but a bit different

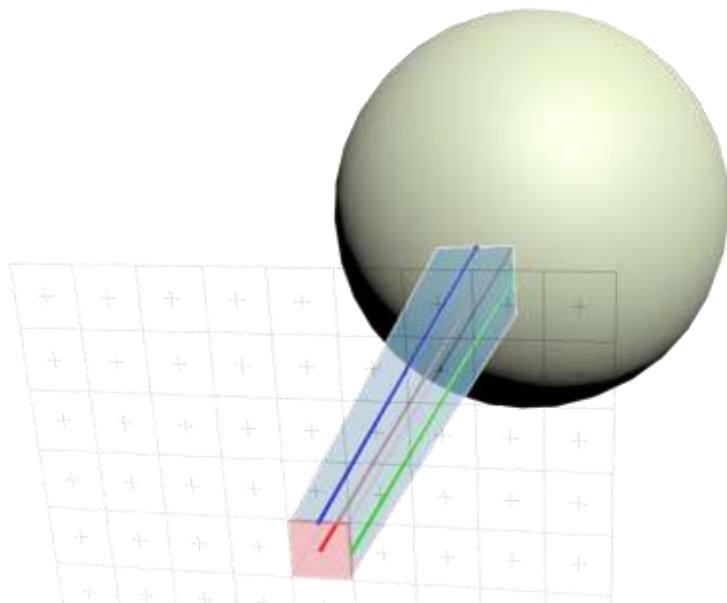


A prism is a better idea



Ray Differentials are the solution

- **Ray Differentials involve tracking values for Δx and Δy**
 - The original work was done by Homan Igehy, SigGraph '99



Selective Derivative Tracking



- This is not a released part of the spec yet – It may change
- Ray attributes declared with the *derivatives* keyword contain values for Δx and Δy tracks, in addition to the main track

```
Layout(binding=0) raytype PrimaryRay {  
    derivatives vec3 gl_Origin;  
    derivatives vec3 gl_Direction;  
};
```

- Set the tracks in a qualified variable

```
setTracksIMG( out, mainVal, dxVal, dyVal );  
val = splatTracksIMG( tracks );
```

- Samplers, dFdx, dFdy, and fwidth operate on qualified values

```
vec4 color = texture( txt, uv);
```

Emitting a ray with differentials



```
void main () {
```

```
    derivatives vec3 outRayDirection;  
    derivatives vec2 frameCoordDXDY;
```

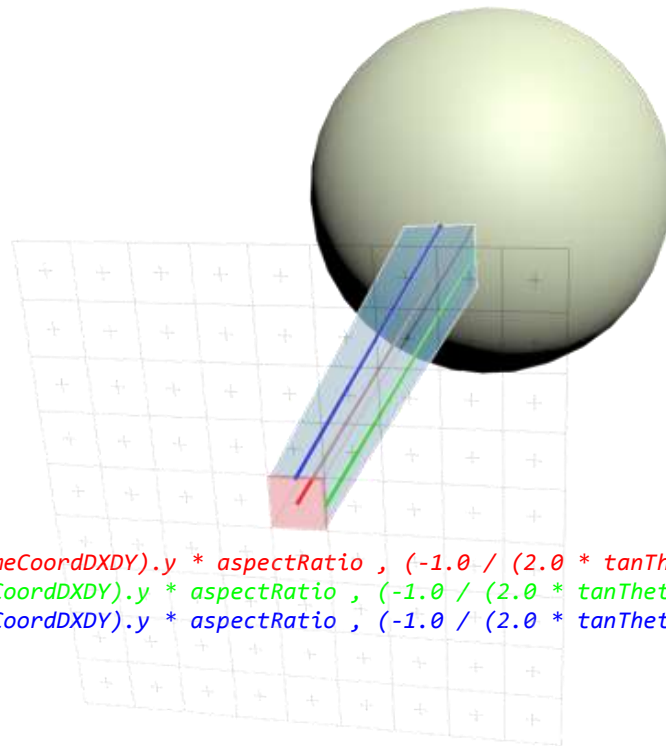
```
    vec2 frameCoordOffsetDX = gl_FrameCoordIMG + ivec2(1, 0);  
    vec2 frameCoordOffsetDY = gl_FrameCoordIMG + ivec2(0, 1);
```

```
    setTracksIMG(  
        frameCoordDXDY,  
        vec2(gl_FrameCoordIMG)/RenderSize.xy - vec2(0.5),  
        vec2(frameCoordOffsetDX)/RenderSize.xy - vec2(0.5),  
        vec2(frameCoordOffsetDY)/RenderSize.xy - vec2(0.5));
```

```
    setTracksIMG(  
        outRayDirection,  
        vec3( getMainTrackIMG(frameCoordDXDY).x, getMainTrackIMG(frameCoordDXDY).y * aspectRatio , (-1.0 / (2.0 * tanTheta)) ),  
        vec3( getDxTrackIMG(frameCoordDXDY).x, getDxTrackIMG(frameCoordDXDY).y * aspectRatio , (-1.0 / (2.0 * tanTheta)) ),  
        vec3( getDyTrackIMG(frameCoordDXDY).x, getDyTrackIMG(frameCoordDXDY).y * aspectRatio , (-1.0 / (2.0 * tanTheta)) ));
```

```
    primRay.gl_OriginIMG = sPlatTrackIMG(cameraPosition);  
    primRay.gl_DirectionIMG = outRayDirection;  
    emitRayIMG(primRay);
```

```
}
```



Using Derivate Values



- **Must be propagated to other variables in the shader**

```
derivatives vec3 hitCoordDXDY;
```

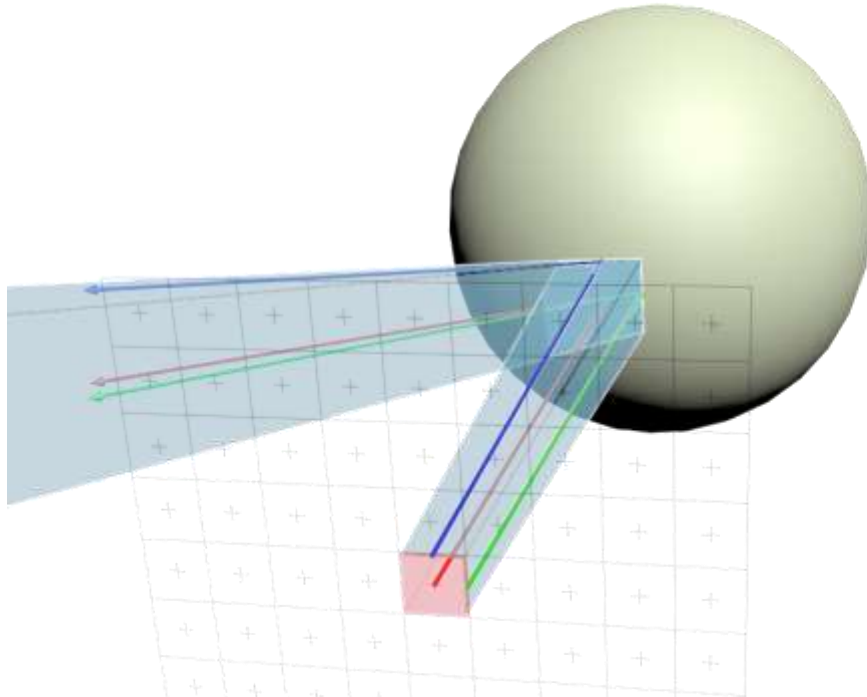
```
derivatives vec2 uv;
```

- **Overloaded interpolation functions assign to qualified values**

```
baryCoordsDXDY = coordAtRayHitDXDYIMG( gl_in[0].gl_Position.xyz,  
gl_in[1].gl_Position.xyz, gl_in[2].gl_Position.xyz, inputRay.gl_Origin,  
inputRay.gl_Direction);
```

```
uv = interpolateIMG(baryCoordsDXDY, VtxData[0].uv, VtxData[1].uv, VtxData[2].uv);
```

Differential values can be propagated



Secondary ray with differentials

```
reflectionRay.gl_OriginIMG = hitCoordDXDY ;  
reflectionRay.gl_DirectionIMG = reflect(primRay.gl_DirectionIMG, hitCoordDXDY );  
emitRayIMG(reflectionRay);
```



Handling dynamic geometry



- **Build component groups**

```
glBuildComponentGroupIMG(0, group, numComponents, components);
```

- **Merge component groups**

```
glMergeComponentGroupIMG(0, group, numGroups, groups);
```

- **Component groups bound to scene arrays**

```
sceneArray = glCreateSceneArrayIMG();
```

```
glBindSceneArrayComponentGroupIMG(sceneArray, sceneIdx, group);
```

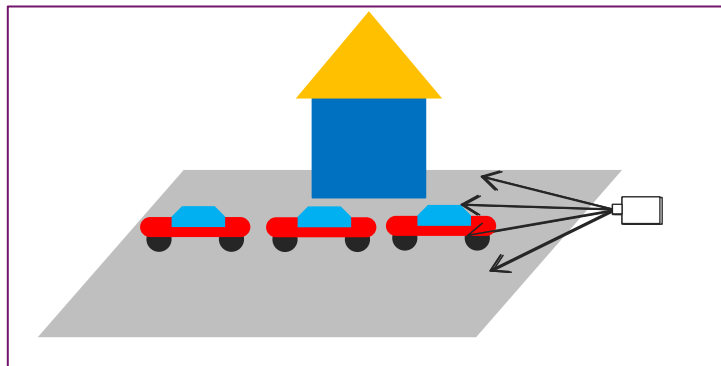
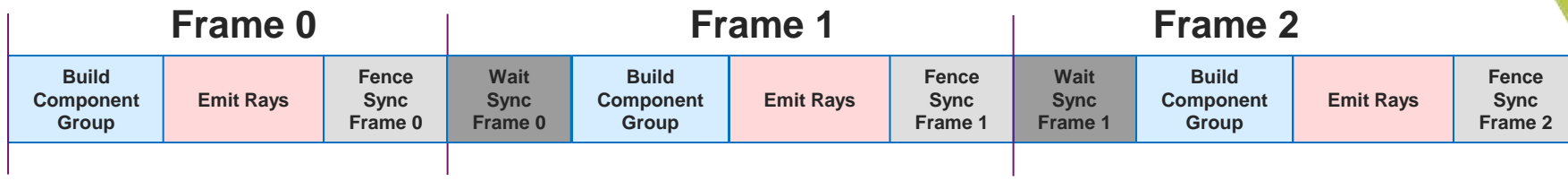
- **Synchronization**

```
sync = glFenceSync(GL_SYNC_RTU_COMMANDS_COMPLETE_IMG, 0);
```

```
glWaitSync(sync, 0, GL_TIMEOUT_IGNORED);
```

Simple approach

Single component group

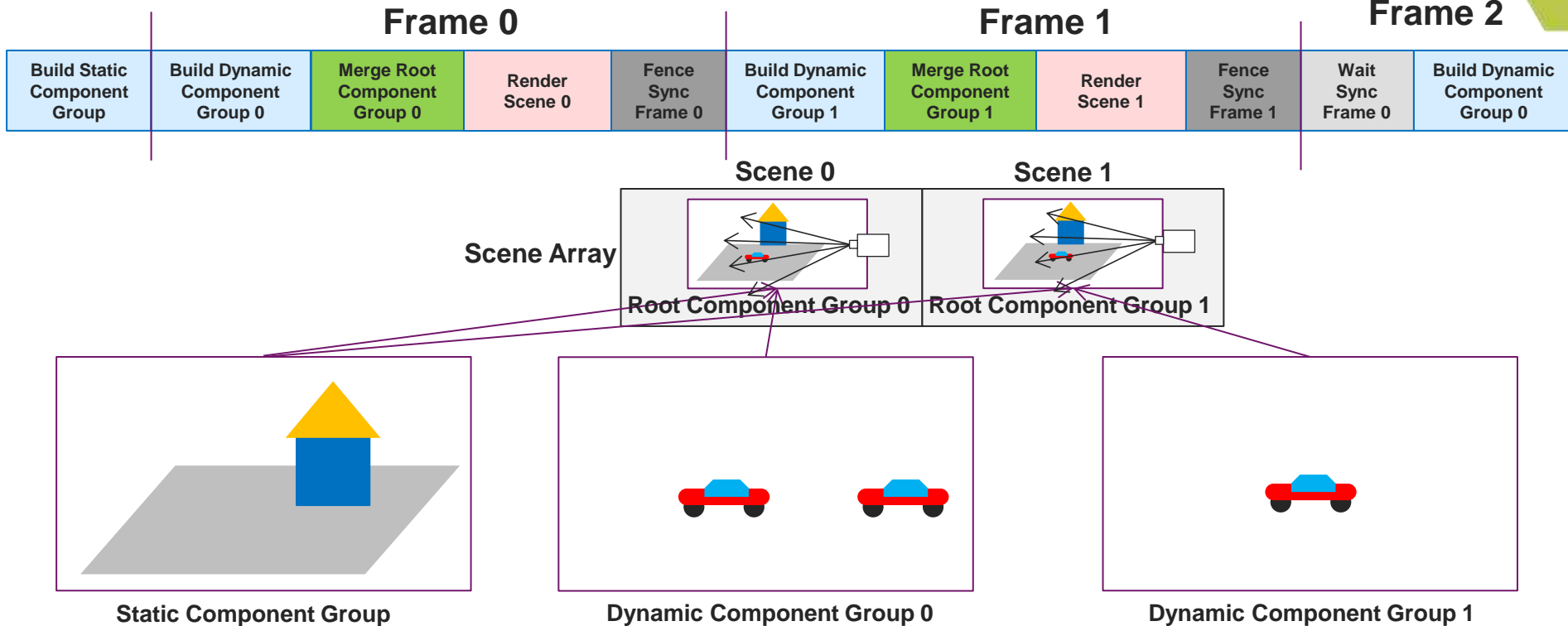


Component Group



Use merging for better performance

Divide scene based on update frequency



Questions?



Get in touch:

- gareth.morgan@imgtec.com
[@griffin1977](https://twitter.com/griffin1977)

Please come and visit us at booth
#1902 in the South Hall to see our
demos and to collect your very own
Vulkan™ Gnome t-shirt!





Imagination

www.imgtec.com/idc