Hi, my name is Kenny Magnusson. I am a senior artist at dice, right now working on Battlefield 3. before I worked on the frostbite team as technical lighting atist

I have worded at dice for almost 15 years now and been involved with most of the games created at dice.

Lighting both in film and computer games has always been a passion of mine.

Lighting in computer games are especially interesting because we let the player take control of the camera and basically look at whatever he ore she wants.

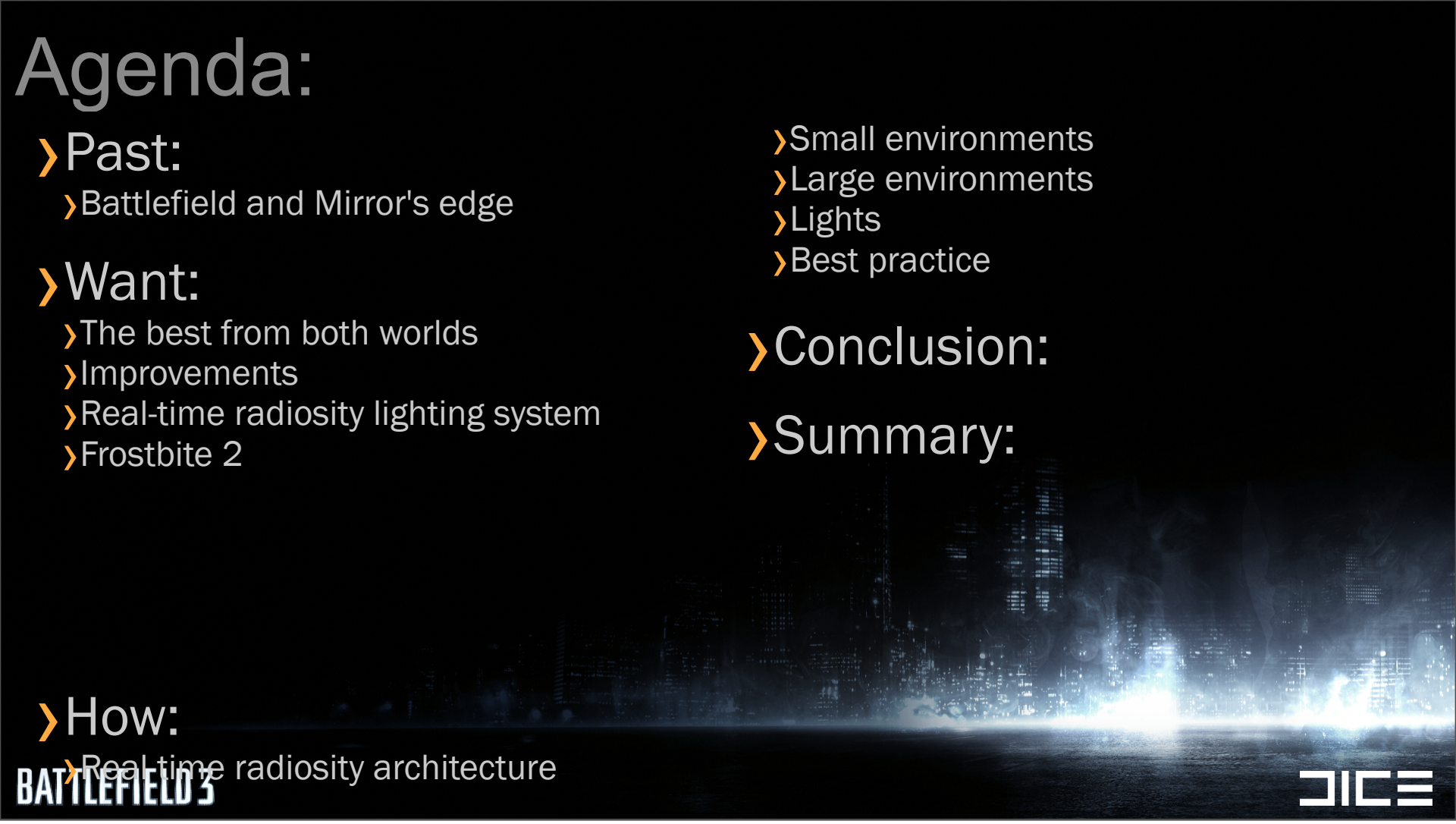
I am going to talk to you to day about the new lighting system in the Frostbite 2 engine and how it enables us to stretch the boundaries of lighting in Battlefield 3 with its highly dynamic, varied and destructible environments.

Agenda:

I will talk about our past, what we wanted for battlefield 3, how we got it and The conclusion n last I'll summarize.

Agenda:

- Past:
  - Battlefield and Mirror's edge
- Want:
  - The best from both worlds
  - Improvements
  - Real-time radiosity lighting system
  - Frostbite 2
- How:
  - Real-time radiosity architecture
    - Small environments
    - Large environments
    - Lights
    - Best practice
- Conclusion:
- Summary:



I will talk about our past, what we wanted for battlefield 3, how we got it and The conclusion n last I'll summarize.

PAST:

Battlefield Bad Company
Frostbite 1

Mirror's Edge
Unreal Engine 3

BATTLEFIELD 3

DICE

So our past : our past is Battlefield bad company using Frostbite 1 engine and Mirrors edge with Unreal engine 3

PAST:

Battlefield Bad Company
Frostbite 1

- Forward-rendering
- No streaming
- Hemispherical sky/ground light + directional sun light
  - 3 point lighting
- 100% Dynamic shadows, cascaded shadowmaps
  - 3 slices, 1024x1024
- Single (!) point light per object
- No Global Illumination
- No Screen Space Ambient Occlusion
- Static sky occlusion maps for buildings & bigger objects
- Static sky occlusion volumes for indoor environments that dynamic objects sampled a single value from

Mirror's Edge
Unreal Engine 3

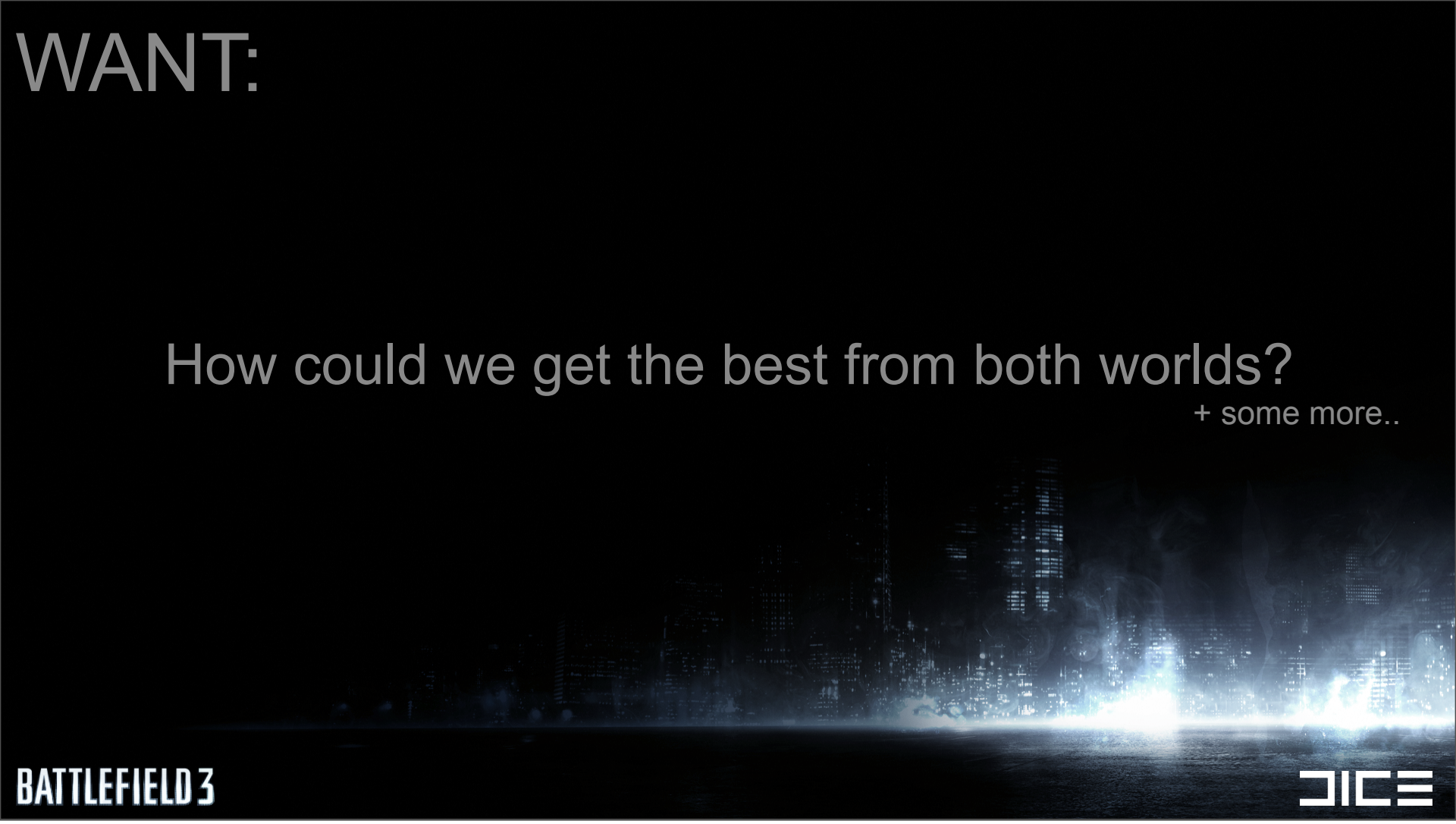
- Forward-rendering
- Streaming
- Static world / static lighting
  - Illuminate Labs Beast Offline Generation
  - Global Illumination (bounce, emissive, translucent)
  - Stored as directional lightmaps (Radiosity Normal Maps)
- Dynamic objects (i.e. Faith)
  - Pre-calculated light probes form basis for relighting dynamic objects
  - Combine most relevant probes to form SH basis for lighting
  - Extract major light axis as "directional light source"
  - Use conventional shadow mapping from extracted directional

These two engines are very different

Frostbite 1 was made to have vast outdoor landscapes and a fully destructible world.
It had 100 dynamic shadows and 3 point lighting system.
No global illumination.
We bake Static occlusion maps for bigger objects.

Mirrors edge Didn't need any destruction in the same scale as Battlefield and there main focus was a city landscape.
So static baked lightmaps worked fine.
We used Unreal Engine 3 with Illuminate Labs Beast Offline Lightmap baking for this project.

WANT:


How could we get the best from both worlds?
+ some more..
BATTLEFIELD 3
DICE

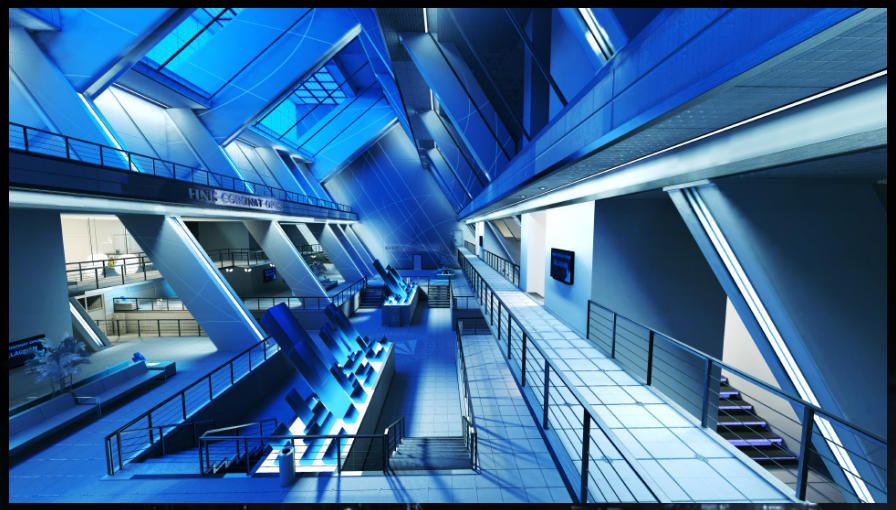
WANT:

- GI In a highly dynamic and destructible environment

This is the big one right. How to get it looking like Mirrors edge while also being a destructible world.

WANT:

- Indoor



BATTLEFIELD 3

DICE

We wanted the indoor lighting to look as good as Mirrors Edge while having a fully destructible indoor environment. Battlefield BC 2 indoor lighting did not look that good.

WANT:

- Outdoor

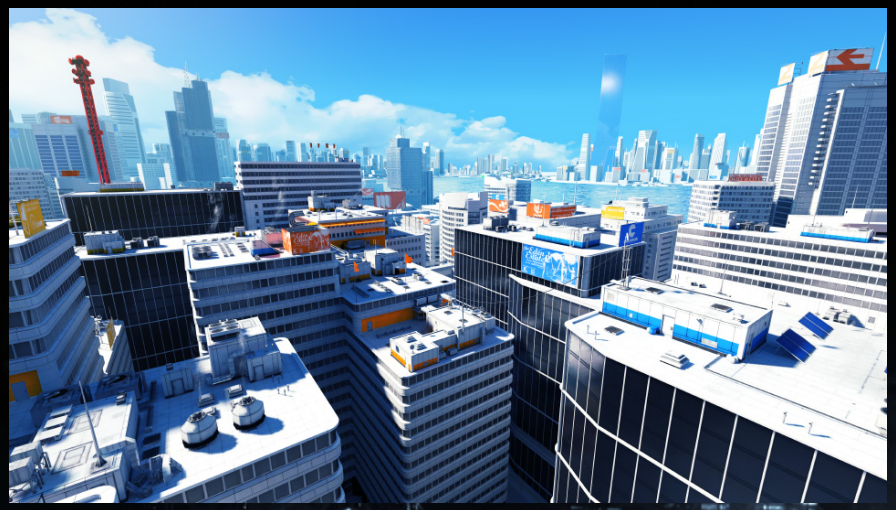
BATTLEFIELD 3

DICE

We wanted both beautiful landscape and a stunning city vista in one level.

WANT:

- Large environments with reflective surfaces



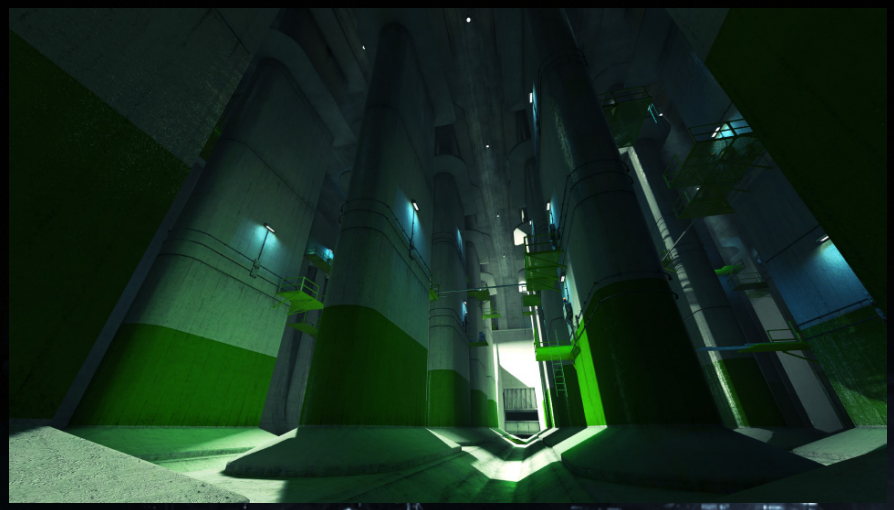
BATTLEFIELD 3

DICE

Reflective surfaces is very important to give life to all materials, ME did this very good but Frostbite 1 has always struggled with this. One big problem is when your moving from a outdoor environment in to a indoor environment.

WANT:

- Lights



With lights and destruction comes one big problem.

We want to be able to destroy lights.

ME had static baked lightmaps, so did the last BC2 level

And therefore we can not destroy light because the light is baked in to the texture.

To be able to have a very Dynamic and Destructibal world, we needed some more improvements. You cant have static baked light maps in a destructible world, we needed a another solution.

WANT:

- Improvements
  - Deferred rendering

BATTLEFIELD 3

DICE

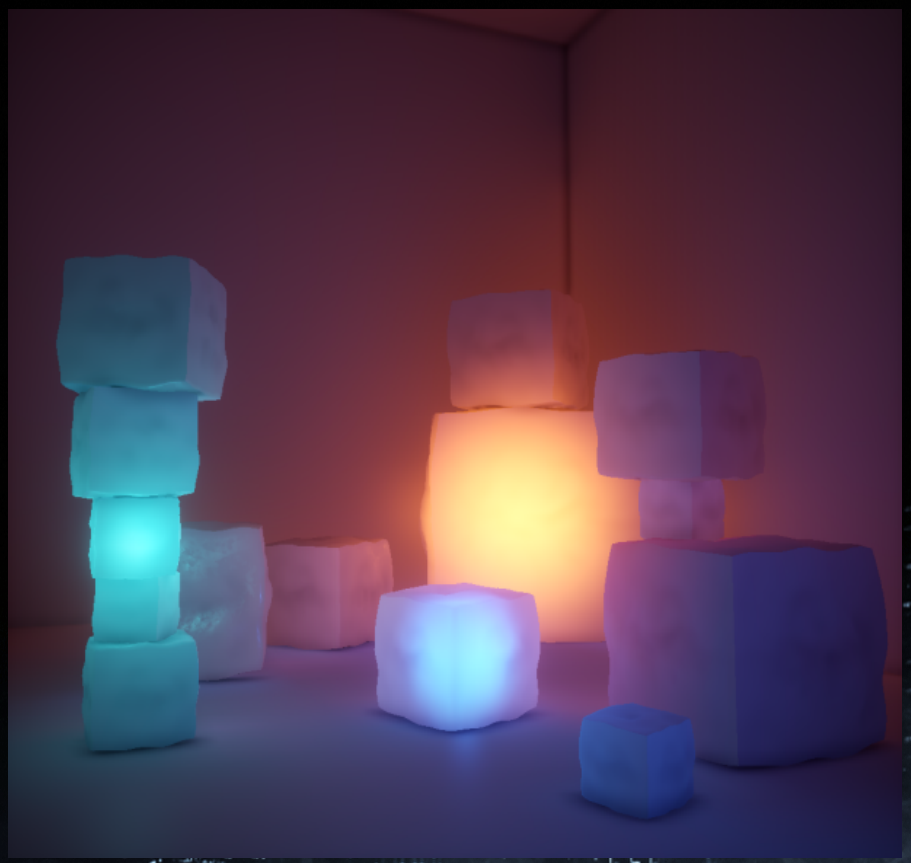
If we want lights in the world, we need to be able to destroy them to. Right…

WANT:

- Improvements
  - Deferred rendering
  - More lighting models

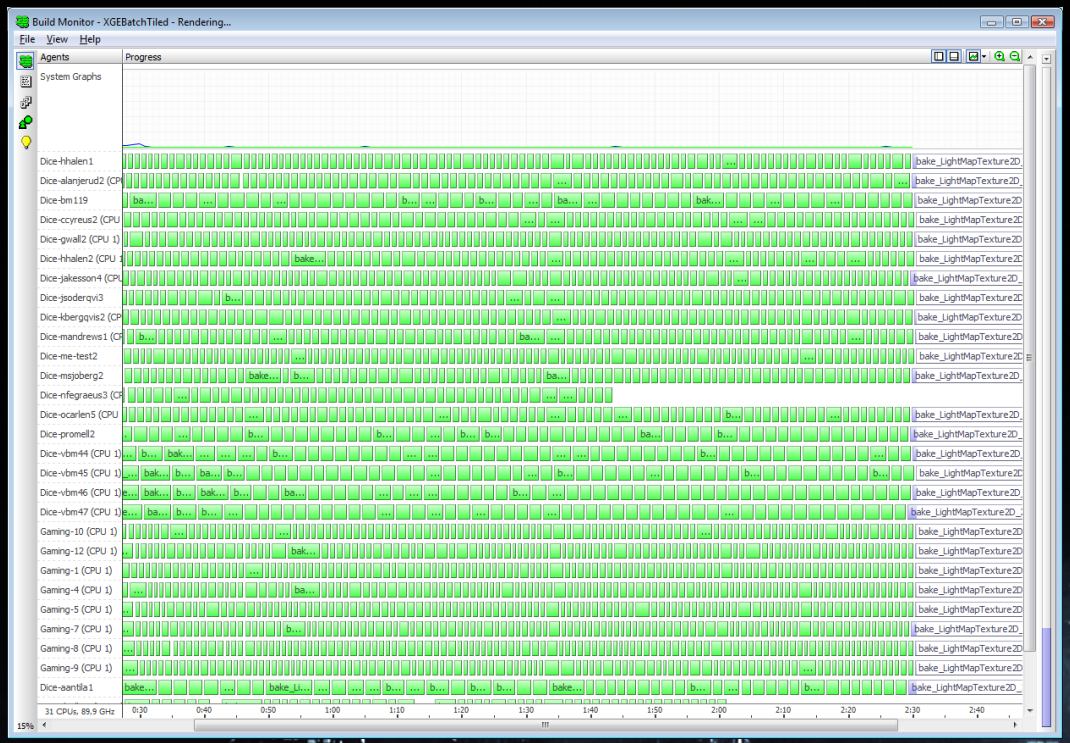
WANT:

- Improvements
  - Deferred rendering
  - More lighting models
  - Translucency



WANT:

- Improvements
  - Deferred rendering
  - More lighting models
  - Translucency
  - Faster workflows



BATTLEFIELD 3

DICE

recognize this?

Renderfarm baking lightmaps, slow!

WANT:

- Improvements
  - Deferred rendering
  - More lighting models
  - Translucency
  - Faster workflows
  - Simultaneous

BATTLEFIELD 3

DICE

Here everybody is standing in line to work on the game. To be able to work simultaneous with art, design and sound on one level would be great.

Usually we have a lot off different gates when we produce games, first design, then art and so on.

WANT:

- Improvements
  - Deferred rendering
  - More lighting models
  - Translucency
  - Faster workflows
  - Simultaneous
  - Particle lighting

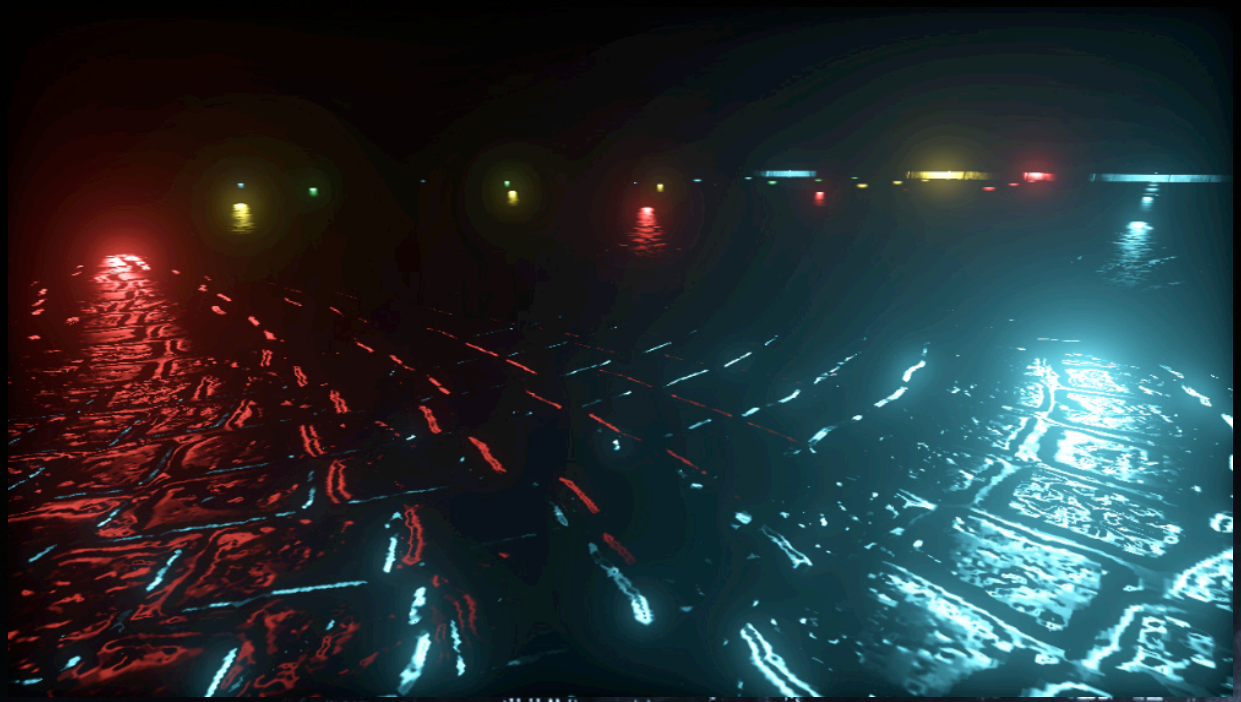
BATTLEFIELD 3

DICE

We have always had problems making particles look like they fit in to the world. They are sometimes to dark or too bright or the wrong color.

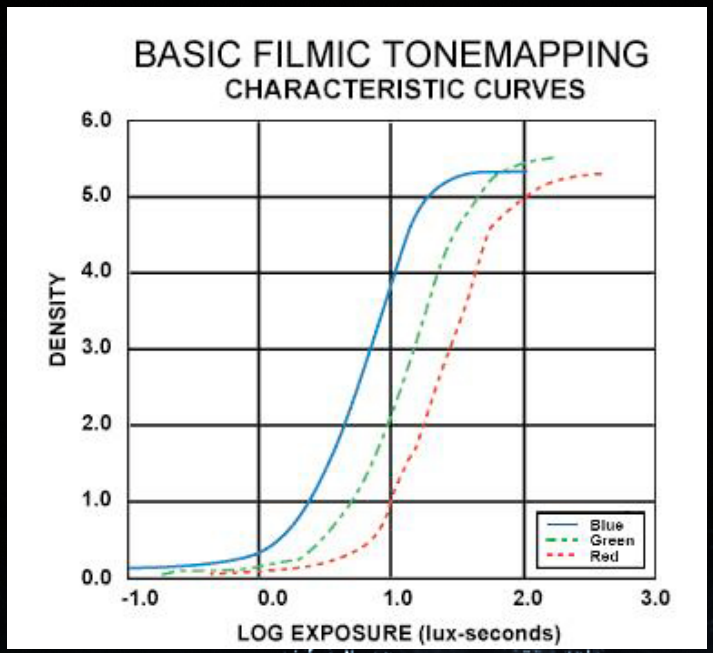
WANT:

- Improvements
  - Deferred rendering
  - More lighting models
  - Translucency
  - Faster workflows
  - Simultaneous
  - Particle lighting
  - Bloom



WANT:

- Improvements
  - Deferred rendering
  - More lighting models
  - Translucency
  - Faster workflows
  - Simultaneous
  - Particle lighting
  - Bloom
  - Filmic Tonemapping



WANT:

- Real-time radiosity lighting system
  - Enlighten
    - Geomerics
      - Cambridge, UK based. Spun out of Cambridge University
    - Collaborated over many years
    - Works for all platforms
      - PLAYSTATION®3,XBox 360™, PC for Windows.

Implementing enlighten in frostbite

WANT:

- Real-time radiosity lighting system
  - Enlighten
    - Geomerics
      - Cambridge, UK based. Spun out of Cambridge University
    - Collaborated over many years
    - Works for all platforms
      - PLAYSTATION®3,XBox 360™, PC for Windows.

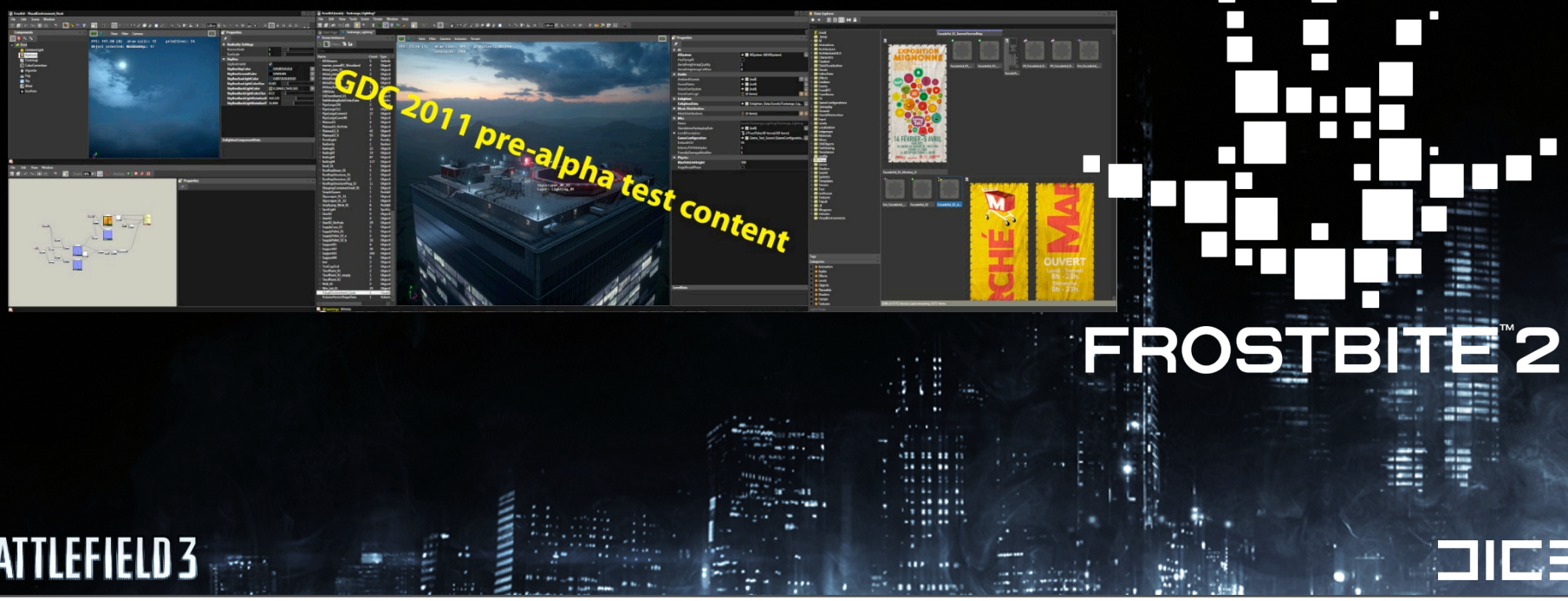
BATTLEFIELD 3

DICE

Implementing enlighten in frostbite

WANT:

- Frostbite 2



Last but not least editor workflows have always been a problem for us, especially when it comes to the consoles.

either reflected, or absorbed and partially re-emitted.

Surfaces that mostly reflect light appear shiny and we say they are highly specular.

Surfaces that absorb and then re-emit tend to do so in all directions, so they appear matte and look the same regardless of what angle you look at them from. Mostly matte surfaces, such as dry concrete or paper, are said to be highly diffuse.

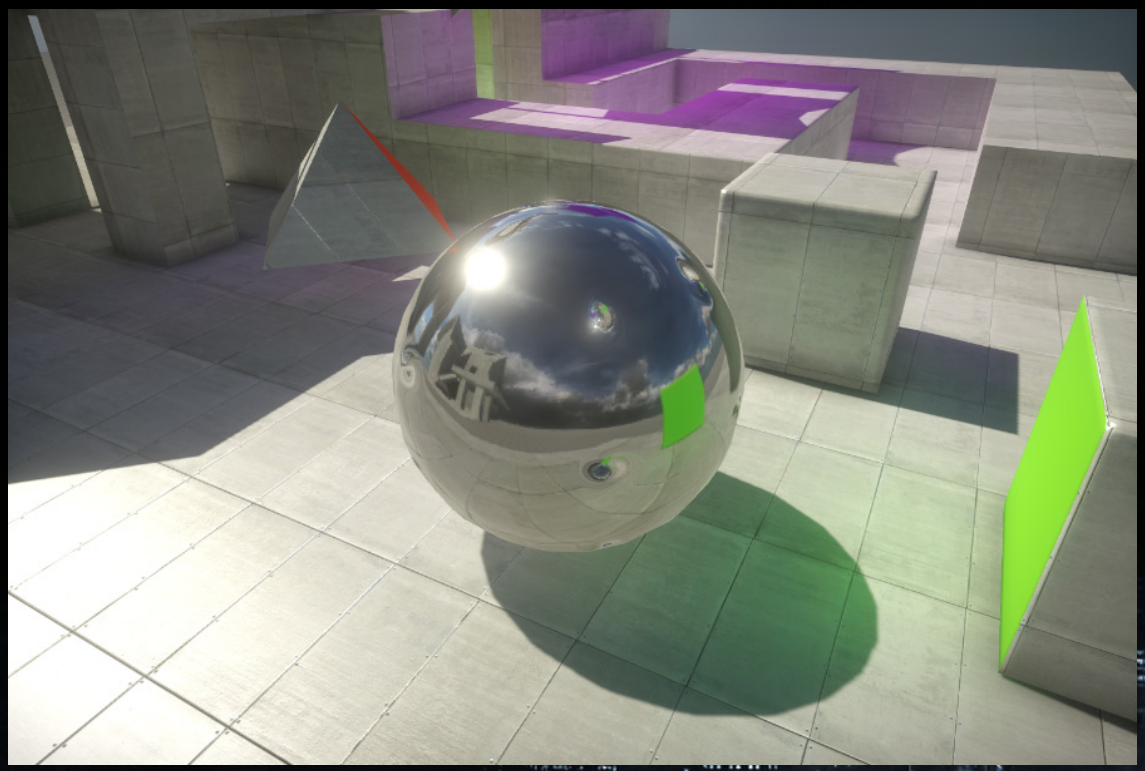
Both specular reflections and diffuse re-emission bounce light back into the world.

The reflected light could then strike another surface, where another portion of it may be bounced, and this continues until all light has been fully absorbed or reflected out of the environment.

This light bouncing effect is typically ignored in most game lighting models because it massively increases the complexity of the computations.

HOW:

Specular



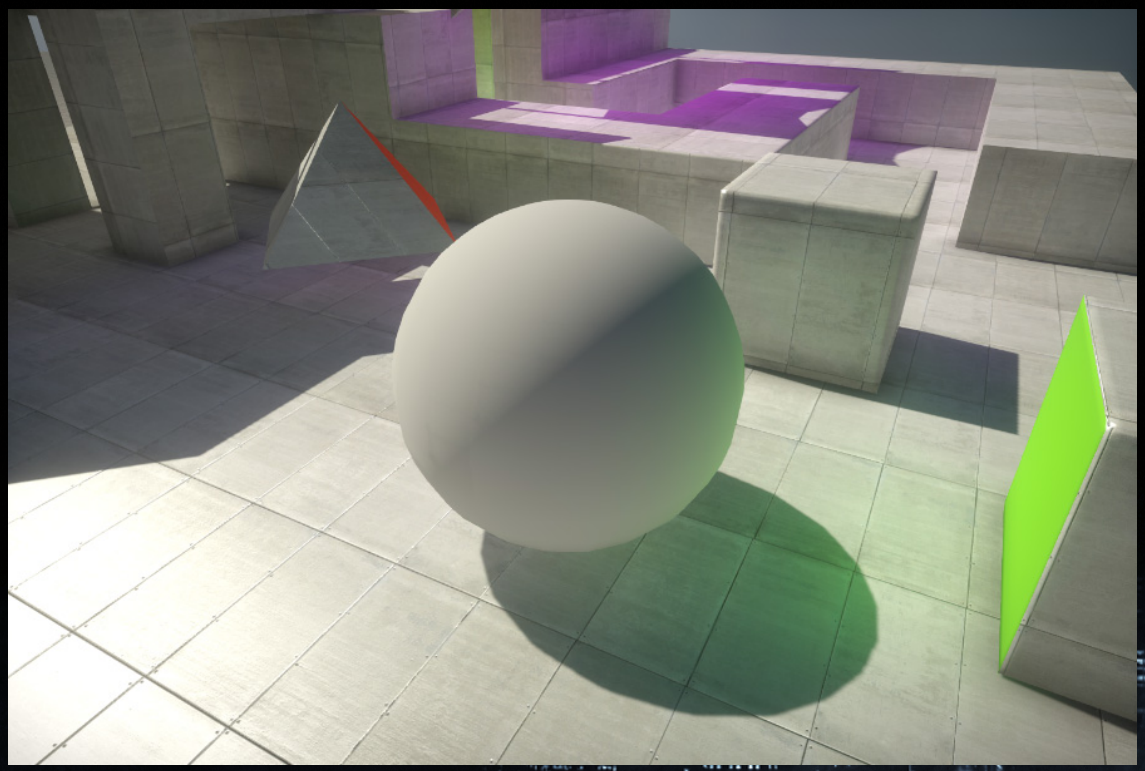
BATTLEFIELD 3

either reflected, or absorbed and partially re-emitted.

Surfaces that mostly reflect light appear shiny and we say they are highly specular.

HOW:

Diffuse



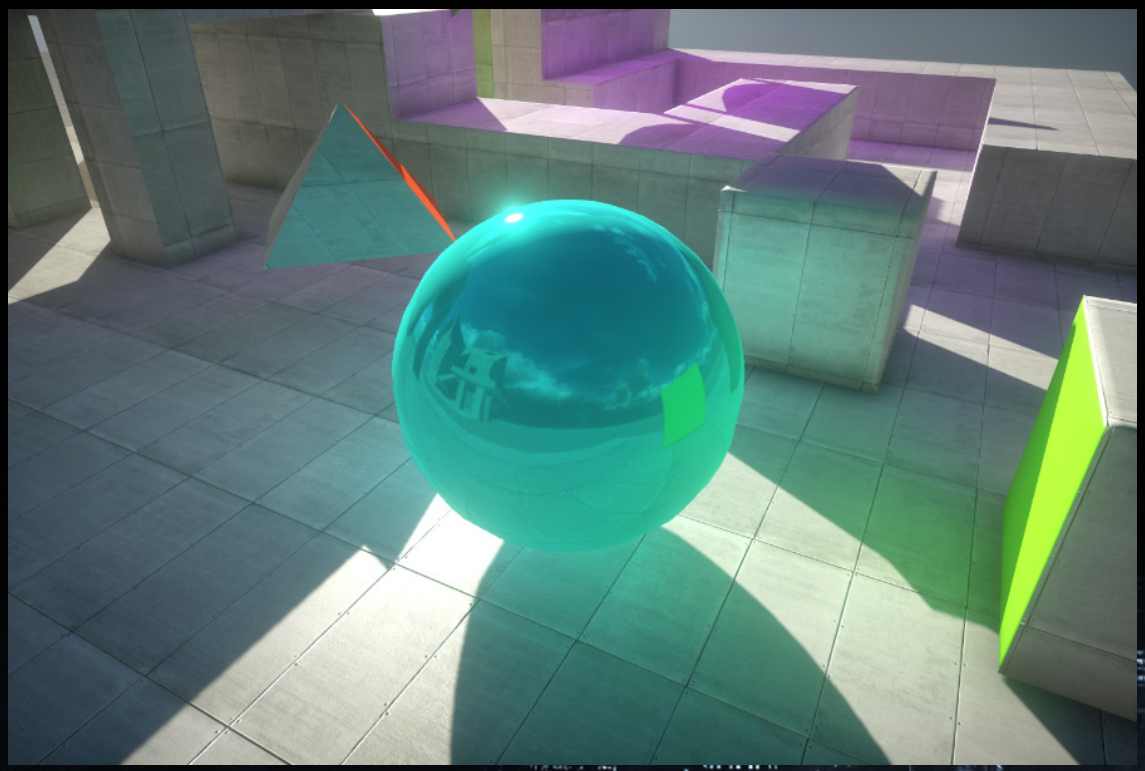
BATTLEFIELD 3

DICE

Surfaces that absorb and then re-emit tend to do so in all directions, so they appear matte and look the same regardless of what angle you look at them from. Mostly matte surfaces, such as dry concrete or paper, are said to be highly diffuse.

HOW:

Specular bounce



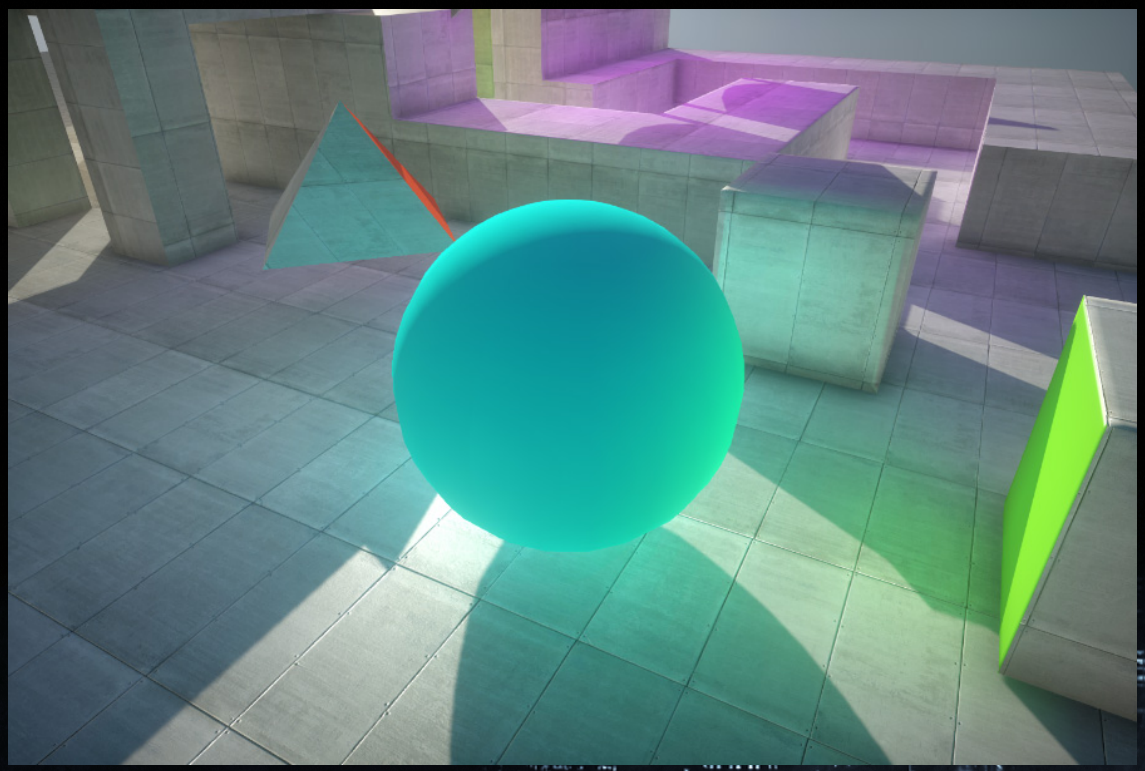
BATTLEFIELD 3

DICE

Both specular reflections and diffuse re-emission bounce light back into the world.

HOW:

Diffuse bounce



BATTLEFIELD 3

DICE

The reflected light could then strike another surface, where another portion of it may be bounced, and this continues until all light has been fully absorbed or reflected out of the environment.

This light bouncing effect is typically ignored in most game lighting models because it massively increases the complexity of the computations.

There are two main approaches to simulating global illumination effects in games right now. Artists can either resort to pre-baked lightmaps or can manually place lights in the scene to model light reflection. Both approaches to lighting have a number of failings:

HOW:

Pre-baked lightmaps

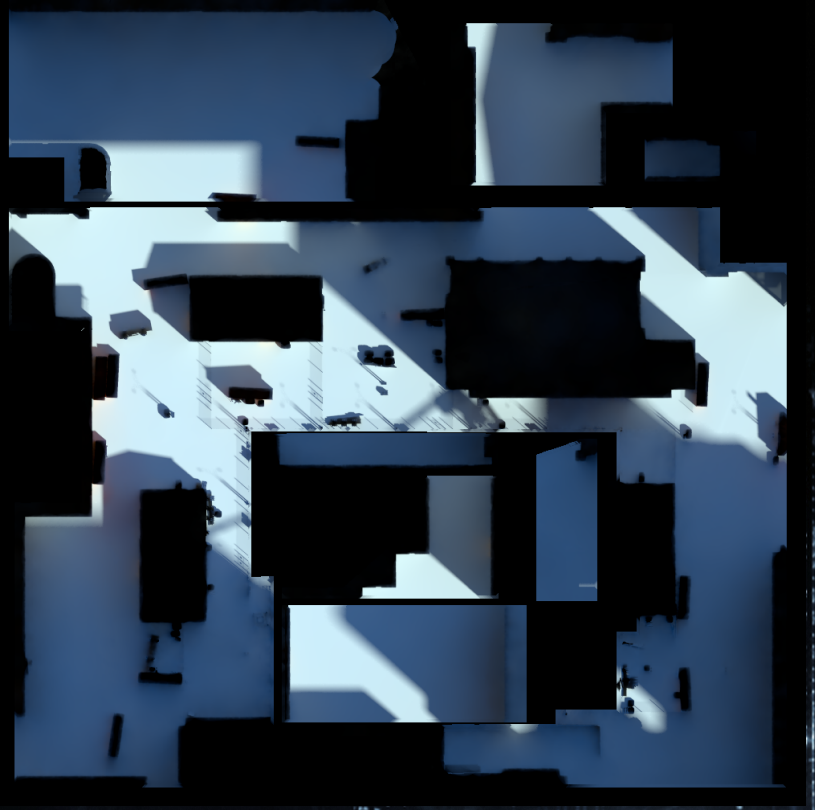
BATTLEFIELD 3

DICE

· Pre-baking lightmaps can produce high quality results, but severely limits the dynamic aspects of game lighting. time consuming, limiting artists' ability to iterate on a scene.

HOW:

Typical lightmap



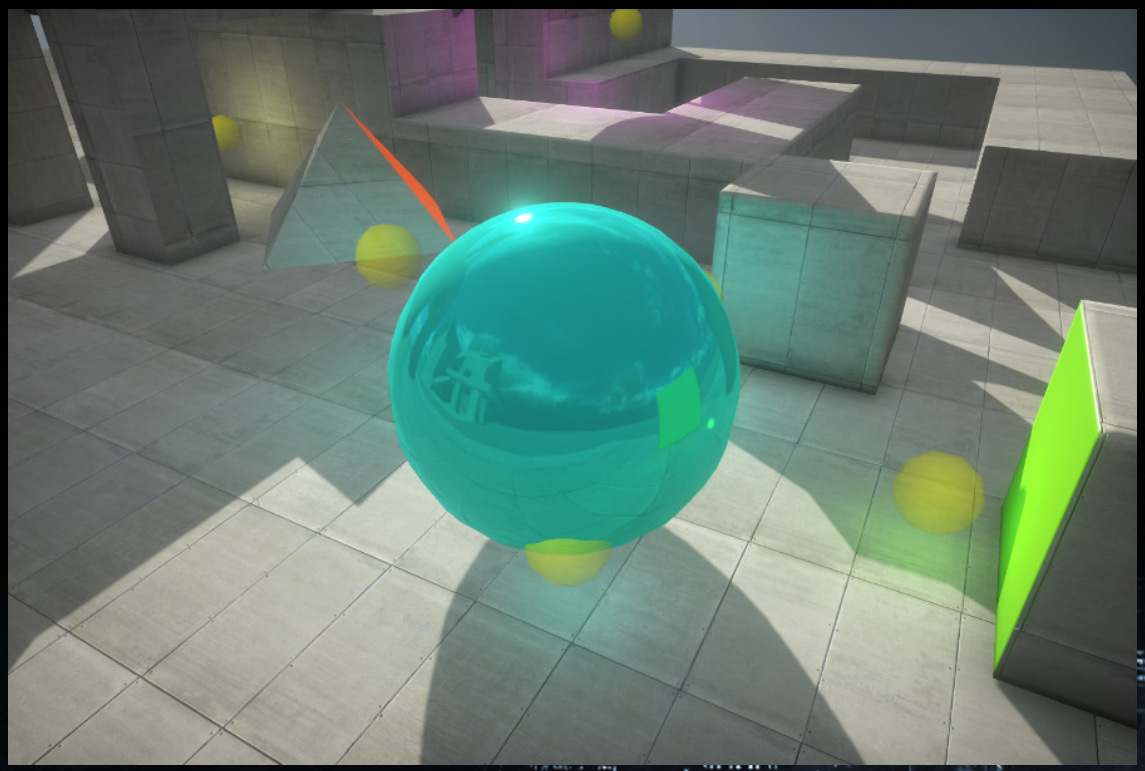
BATTLEFIELD 3

DICE

· Pre-baking lightmaps can produce high quality results, but severely limits the dynamic aspects of game lighting. time consuming, limiting artists' ability to iterate on a scene.

HOW:

Placing direct lights



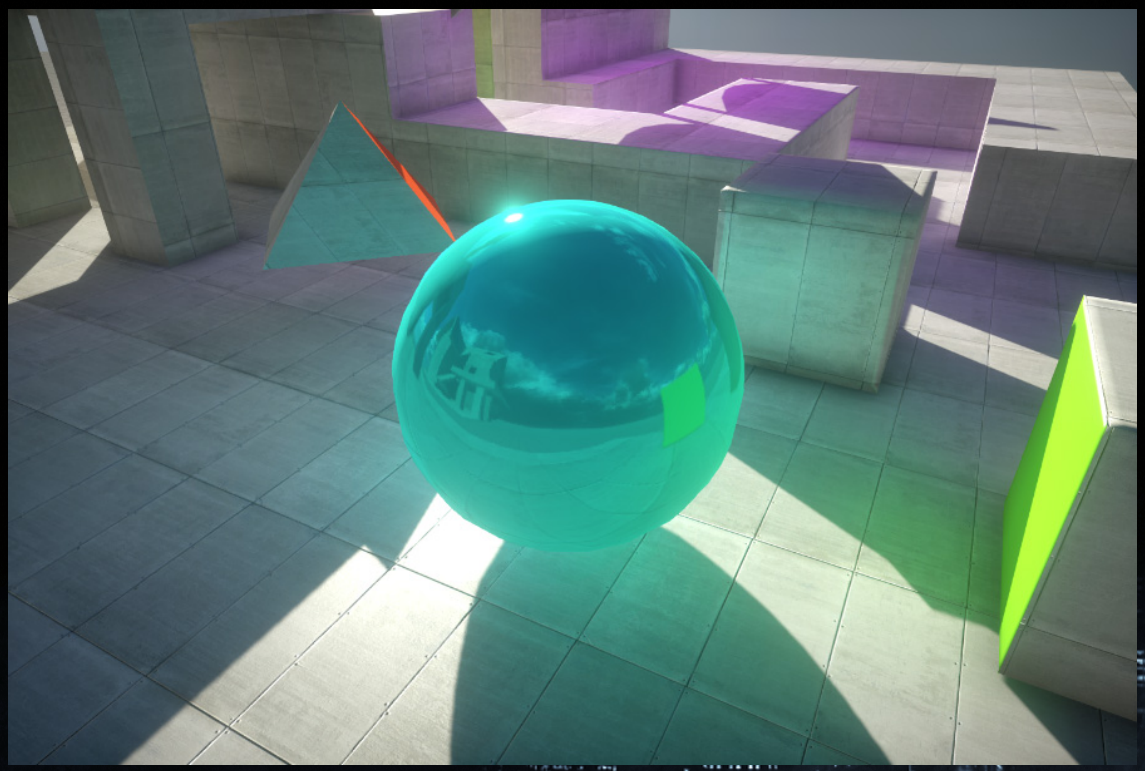
BATTLEFIELD 3

DICE

·Simulating global illumination by placing direct lights does give an artist control, but can be very difficult to use in practice. affect performance.

HOW:

Radiosity



BATTLEFIELD 3

DICE

·Simulating global illumination by placing direct lights does give an artist control, but can be very difficult to use in practice. affect performance.

HOW:


GDC 2011 pre-alpha test content
AL SHATE AL BAHABI
NATIONAL PAINTS
AL MUSALA STAR BUILDING MATERIALS
SHATE AL BAHABI

HOW:

BATTLEFIELD 3

DICE

Enlighten dynamically computes indirect lighting within your scene.

Real-time radiosity really provides full artistic control over the bounced light to allow a fully dynamic workflow.

This greatly reduces the time required to light a scene by artists, and opens up the possibility of completely changing the feel and mood of your scenes dynamically.

The Real-time radiosity output textures store the color and intensity of the indirect lighting as well as the major directions of light.

This information can be used in your shaders to compute indirect specular reflections and normal-mapped surfaces.

We need to split the lighting system in to two part and this is mainly because of memory consumption but also performance wile lighting. Simplifying the geometry is necessary for Enlighten to be able to Handle all the radiosity data.

HOW:

BATTLEFIELD 3

DICE

Enlighten dynamically computes indirect lighting within your scene.

Real-time radiosity really provides full artistic control over the bounced light to allow a fully dynamic workflow.

This greatly reduces the time required to light a scene by artists, and opens up the possibility of completely changing the feel and mood of your scenes dynamically.

The Real-time radiosity output textures store the color and intensity of the indirect lighting as well as the major directions of light.

This information can be used in your shaders to compute indirect specular reflections and normal-mapped surfaces.

We need to split the lighting system in to two part and this is mainly because of memory consumption but also performance wile lighting. Simplifying the geometry is necessary for Enlighten to be able to Handle all the radiosity data.

HOW:

The fist one is The Lightmap lit geometry and the second one is the Lightprobe lit geometry.

Lightmap lit geometry is parameterized and lit with dynamic lightmaps. This geometry can bounce light, and typically represents large and inanimate objects that don't move.

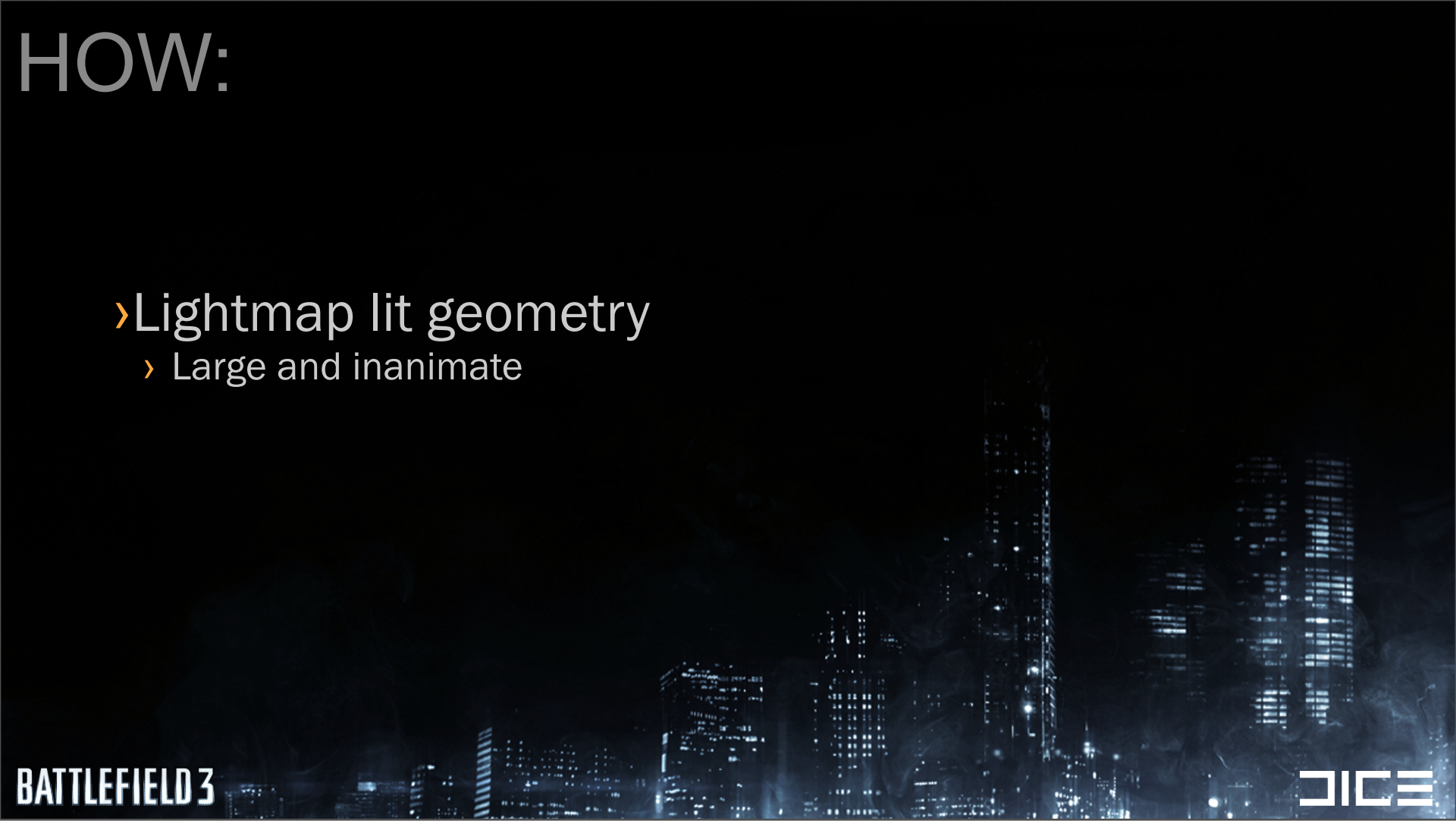
Lightprobe lit geometry can only receive bounce light by sampling light probes. I will get back to what lightprobes are in a minute. They are typically the character ore a dynamic object but can also be a static geometry that don't affect the lighting too much.

HOW:

The fist one is The Lightmap lit geometry and the second one is the Lightprobe lit geometry.

Lightmap lit geometry is parameterized and lit with dynamic lightmaps. This geometry can bounce light, and typically represents large and inanimate objects that don't move.

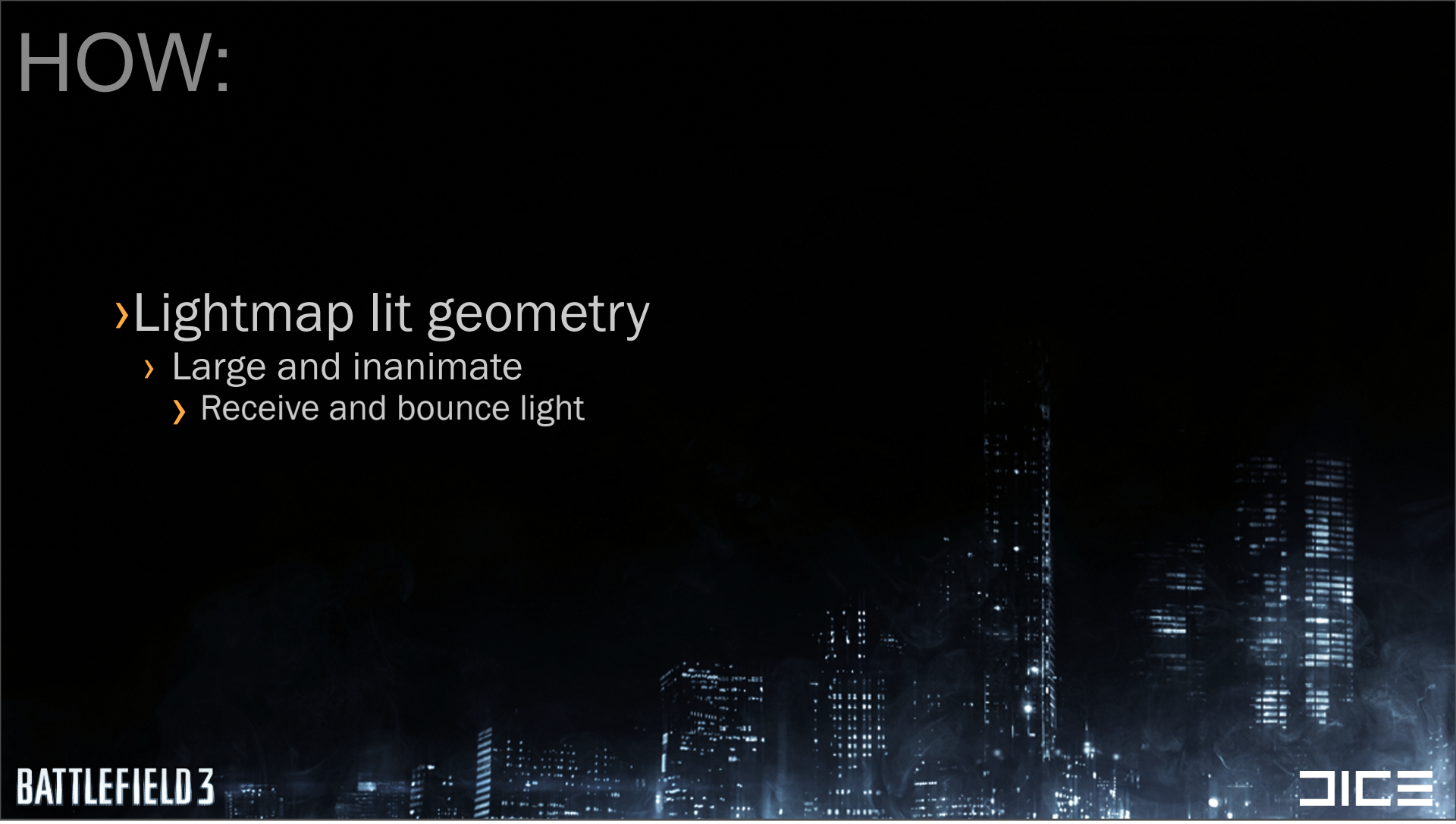
Lightprobe lit geometry can only receive bounce light by sampling light probes. I will get back to what lightprobes are in a minute. They are typically the character ore a dynamic object but can also be a static geometry that don't affect the lighting too much.



The fist one is The Lightmap lit geometry and the second one is the Lightprobe lit geometry.

Lightmap lit geometry is parameterized and lit with dynamic lightmaps. This geometry can bounce light, and typically represents large and inanimate objects that don't move.

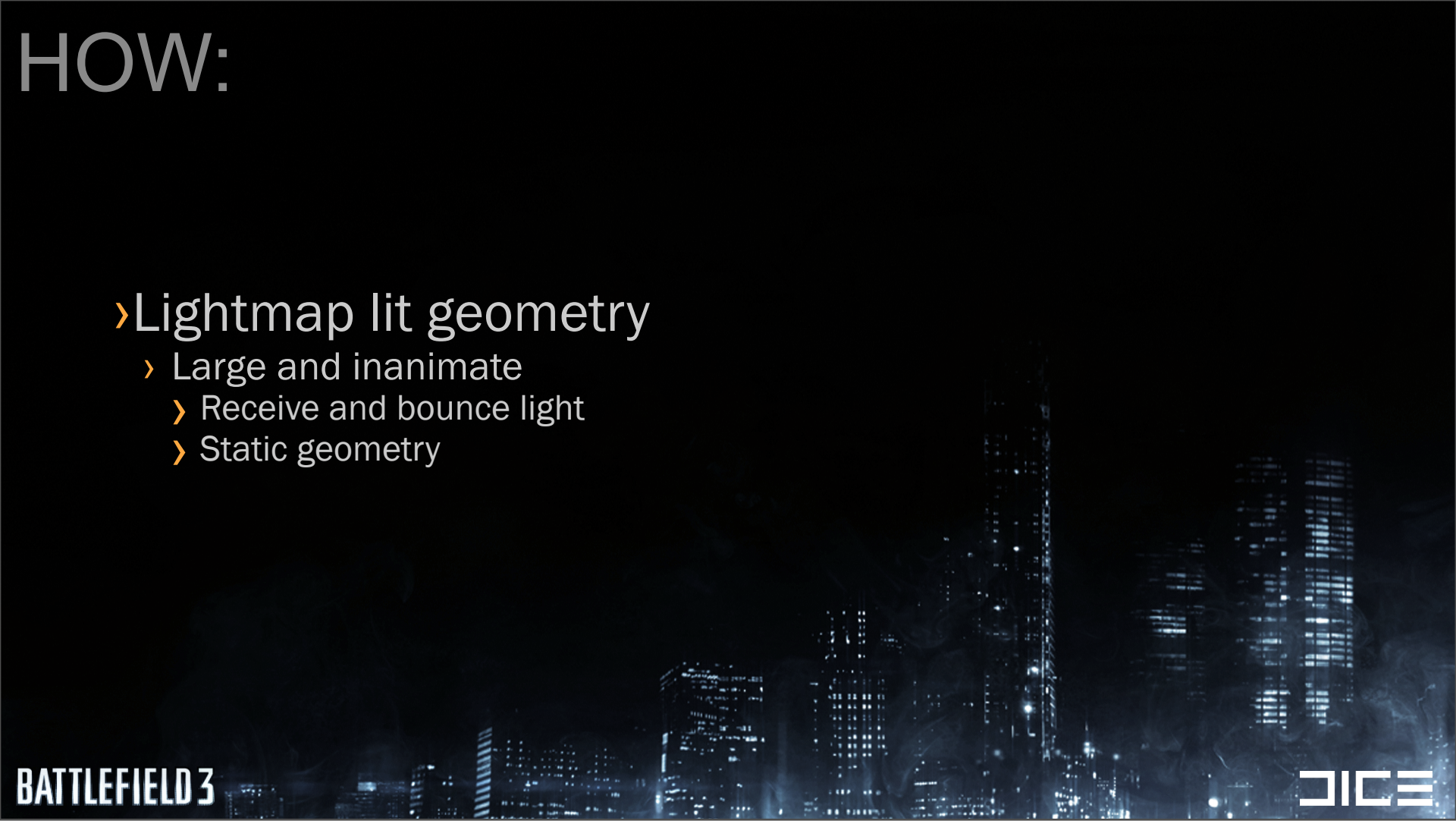
Lightprobe lit geometry can only receive bounce light by sampling light probes. I will get back to what lightprobes are in a minute. They are typically the character ore a dynamic object but can also be a static geometry that don't affect the lighting too much.



The fist one is The Lightmap lit geometry and the second one is the Lightprobe lit geometry.

Lightmap lit geometry is parameterized and lit with dynamic lightmaps. This geometry can bounce light, and typically represents large and inanimate objects that don't move.

Lightprobe lit geometry can only receive bounce light by sampling light probes. I will get back to what lightprobes are in a minute. They are typically the character ore a dynamic object but can also be a static geometry that don't affect the lighting too much.



The fist one is The Lightmap lit geometry and the second one is the Lightprobe lit geometry.

Lightmap lit geometry is parameterized and lit with dynamic lightmaps. This geometry can bounce light, and typically represents large and inanimate objects that don't move.

Lightprobe lit geometry can only receive bounce light by sampling light probes. I will get back to what lightprobes are in a minute. They are typically the character ore a dynamic object but can also be a static geometry that don't affect the lighting too much.

The fist one is The Lightmap lit geometry and the second one is the Lightprobe lit geometry.

Lightmap lit geometry is parameterized and lit with dynamic lightmaps. This geometry can bounce light, and typically represents large and inanimate objects that don't move.

Lightprobe lit geometry can only receive bounce light by sampling light probes. I will get back to what lightprobes are in a minute. They are typically the character ore a dynamic object but can also be a static geometry that don't affect the lighting too much.

The fist one is The Lightmap lit geometry and the second one is the Lightprobe lit geometry.

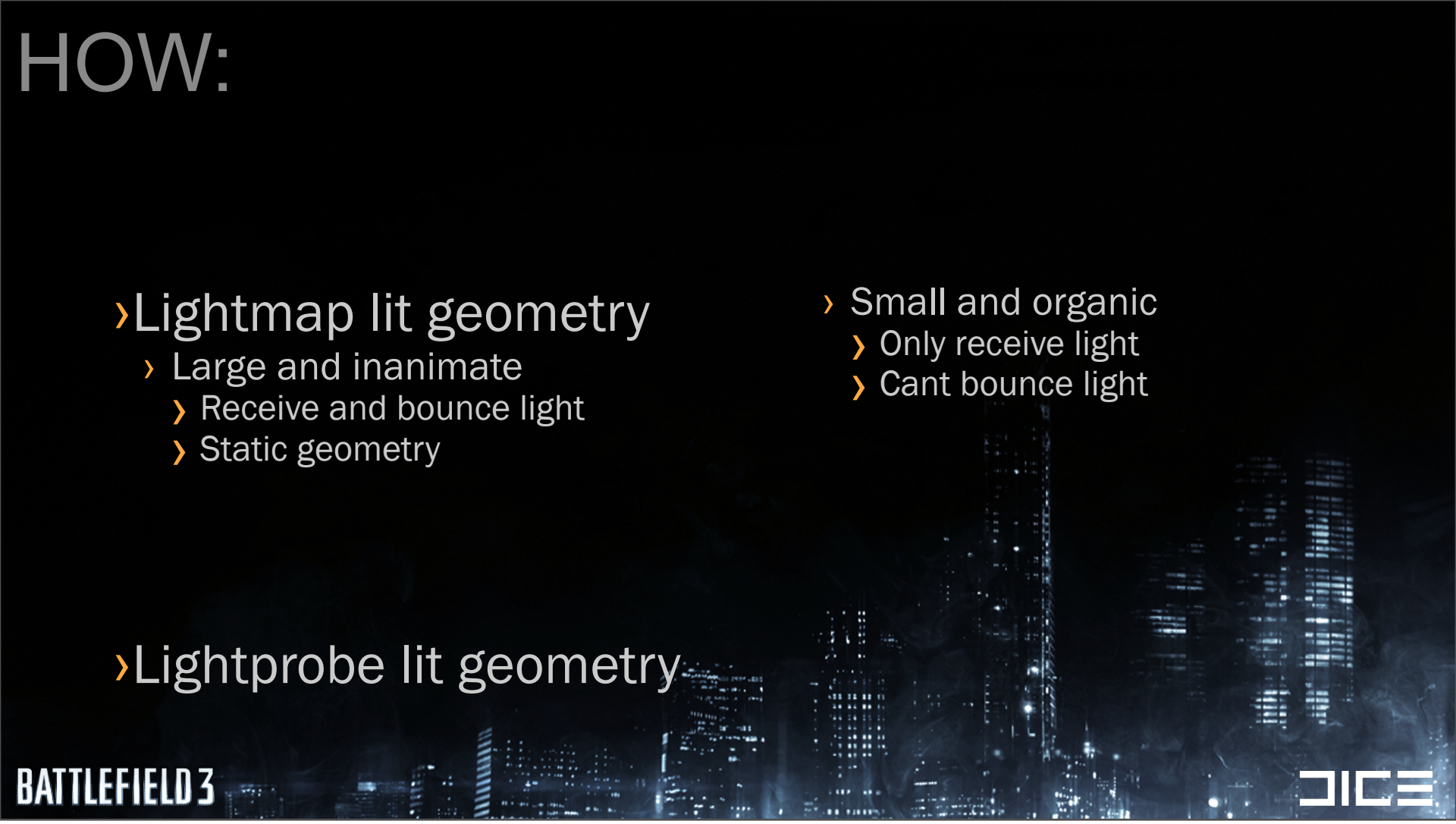
Lightmap lit geometry is parameterized and lit with dynamic lightmaps. This geometry can bounce light, and typically represents large and inanimate objects that don't move.

Lightprobe lit geometry can only receive bounce light by sampling light probes. I will get back to what lightprobes are in a minute. They are typically the character ore a dynamic object but can also be a static geometry that don't affect the lighting too much.

The fist one is The Lightmap lit geometry and the second one is the Lightprobe lit geometry.

Lightmap lit geometry is parameterized and lit with dynamic lightmaps. This geometry can bounce light, and typically represents large and inanimate objects that don't move.

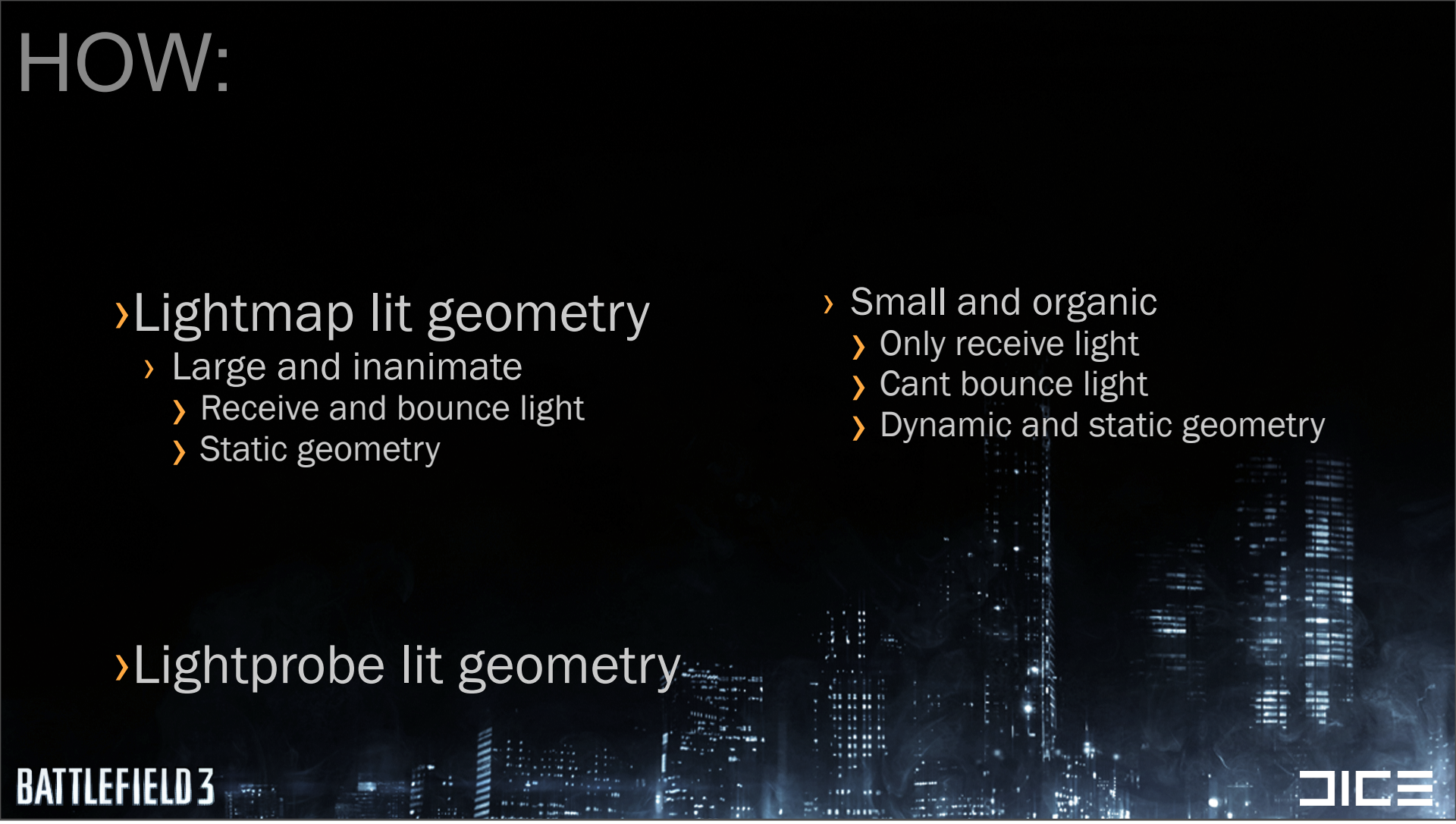
Lightprobe lit geometry can only receive bounce light by sampling light probes. I will get back to what lightprobes are in a minute. They are typically the character ore a dynamic object but can also be a static geometry that don't affect the lighting too much.



The fist one is The Lightmap lit geometry and the second one is the Lightprobe lit geometry.

Lightmap lit geometry is parameterized and lit with dynamic lightmaps. This geometry can bounce light, and typically represents large and inanimate objects that don't move.

Lightprobe lit geometry can only receive bounce light by sampling light probes. I will get back to what lightprobes are in a minute. They are typically the character ore a dynamic object but can also be a static geometry that don't affect the lighting too much.



The fist one is The Lightmap lit geometry and the second one is the Lightprobe lit geometry.

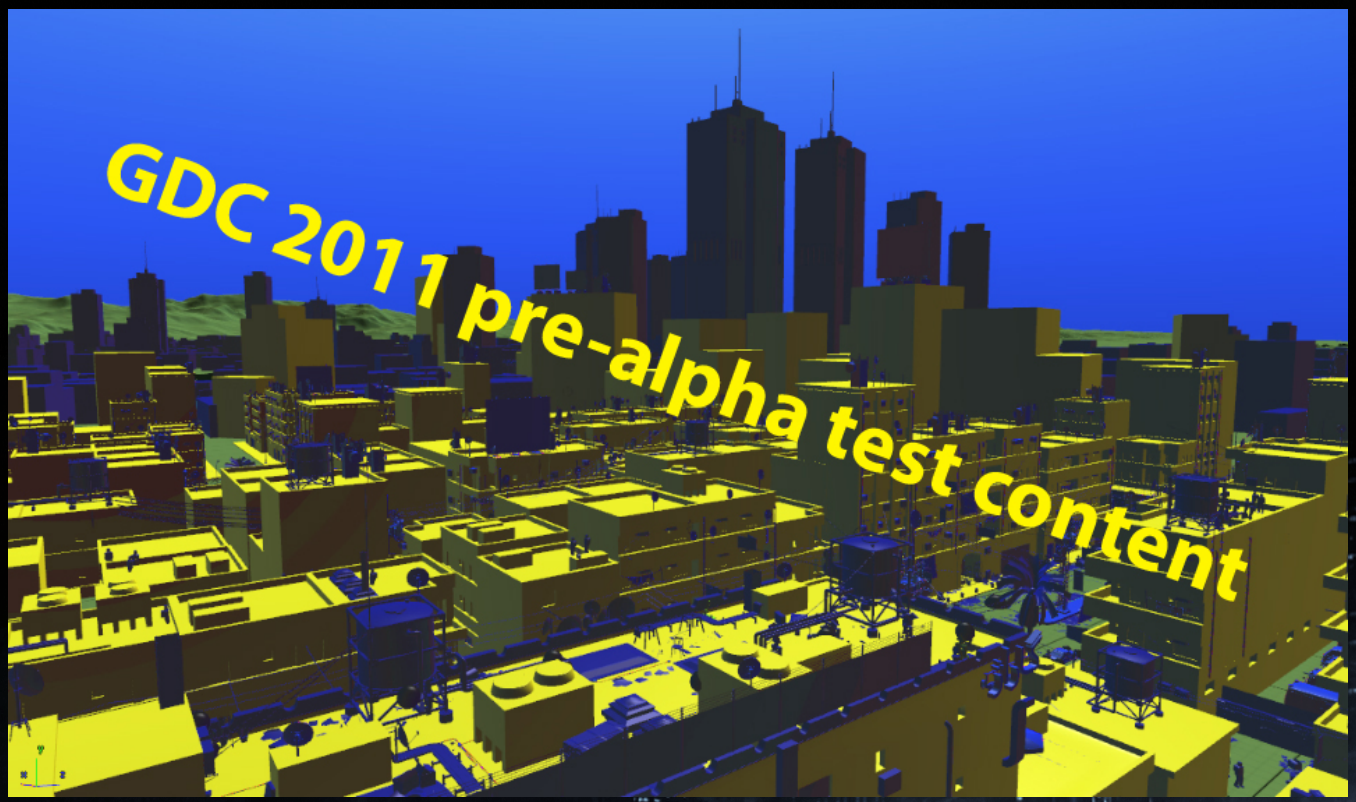
Lightmap lit geometry is parameterized and lit with dynamic lightmaps. This geometry can bounce light, and typically represents large and inanimate objects that don't move.

Lightprobe lit geometry can only receive bounce light by sampling light probes. I will get back to what lightprobes are in a minute. They are typically the character ore a dynamic object but can also be a static geometry that don't affect the lighting too much.

HOW:

In the mesh classification picture you will soon se, you can see how we clearly define Lightmap lit geometry and Lightprobe lit geometry

HOW:



BATTLEFIELD 3

DICE

: static in yellow and dynamic in blue. The underlying geometry (Lightmap lit geometry) is used to bounce light, which then is transferred to all objects (Lightprobe lit geometryin) the scene.

HOW:

BATTLEFIELD 3

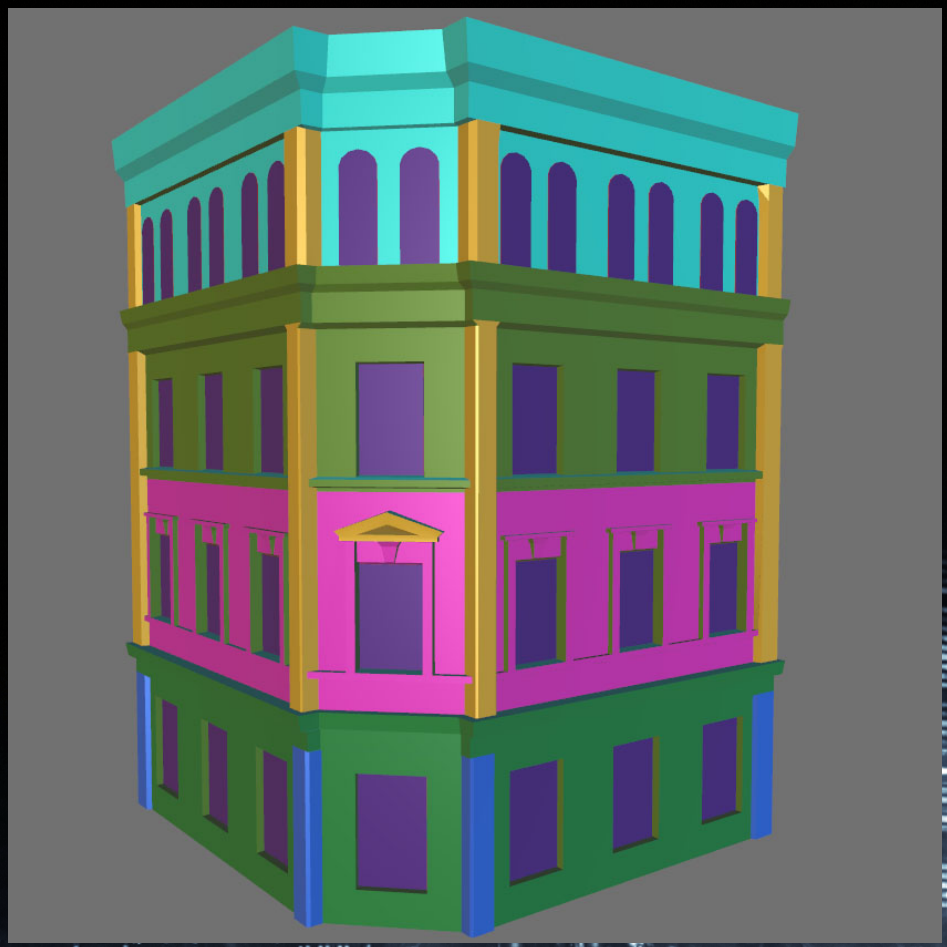
DICE

HOW:

- Lightmap lit geometry

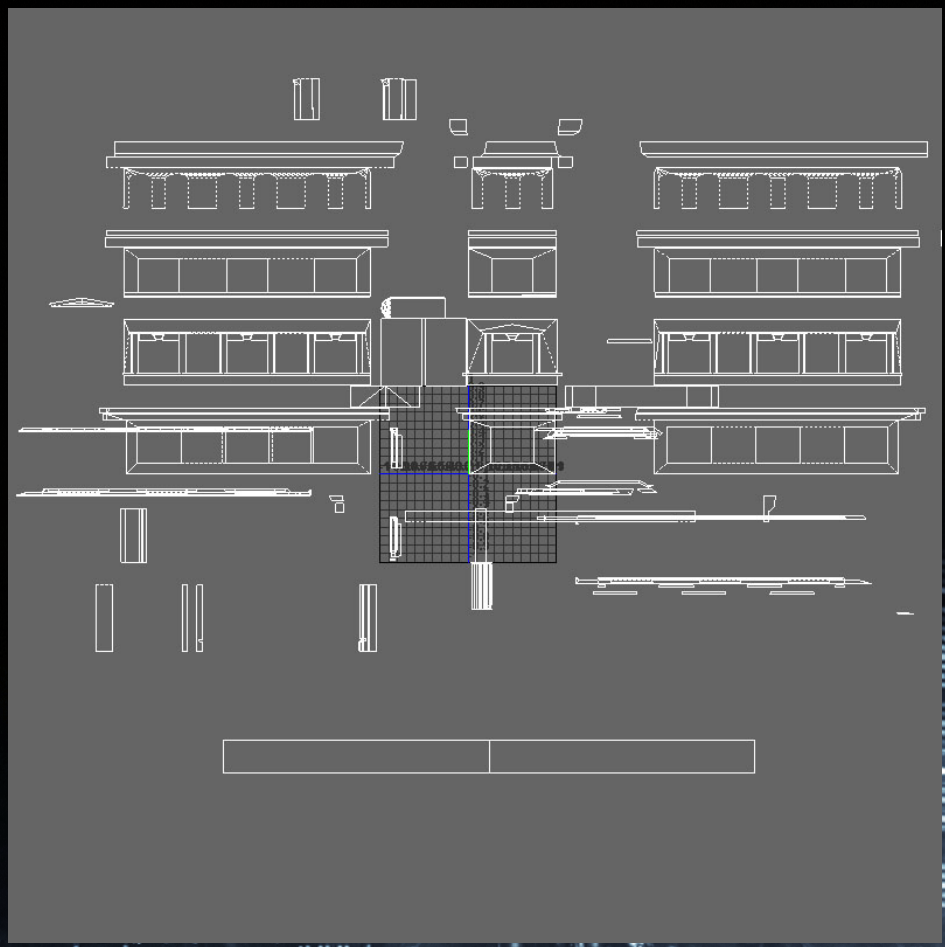
HOW:

- Lightmap lit geometry
  - Detail geometry



HOW:

- Lightmap lit geometry
  - Detail geometry
    - UVs generated by projection



BATTLEFIELD 3

DICE

This means that a new UV set is created automatically

HOW:

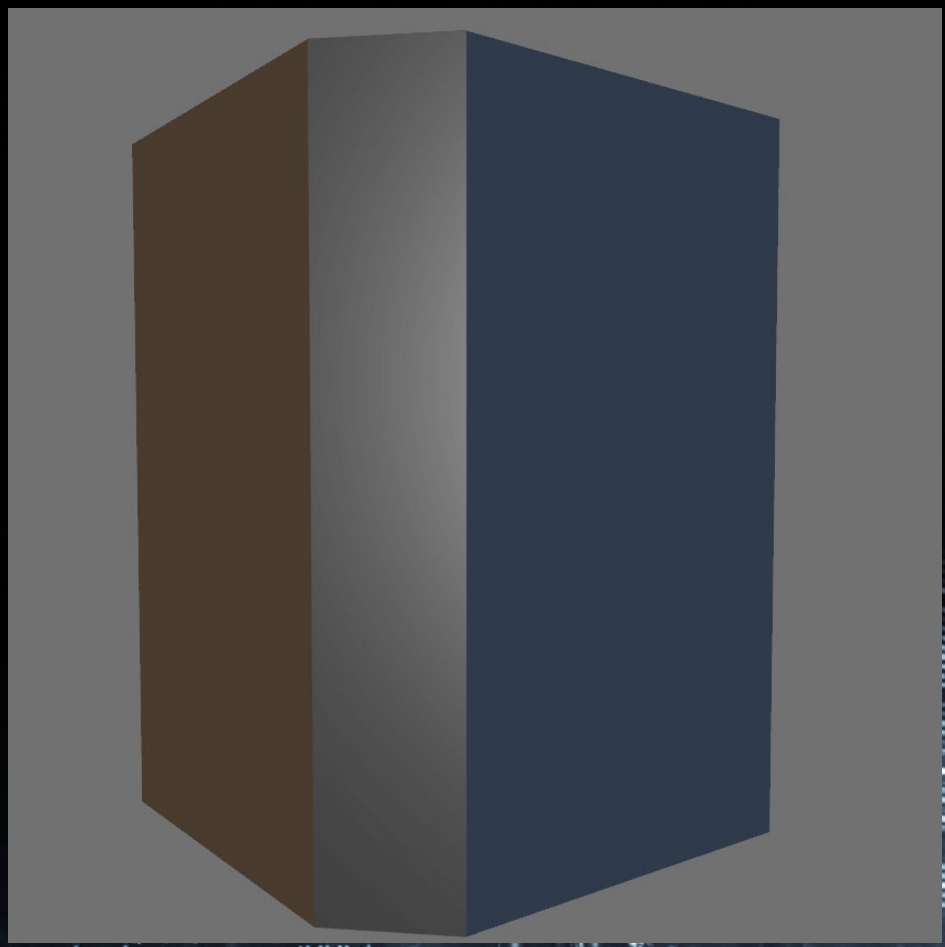
- Lightmap lit geometry
  - Detail geometry
    - UVs generated by projection
    - No additional lighting data

HOW:

- Lightmap lit geometry
  - Detail geometry
    - UVs generated by projection
    - No additional lighting data
    - “Off-axis” lighting comes from directional data in lightmap

HOW:

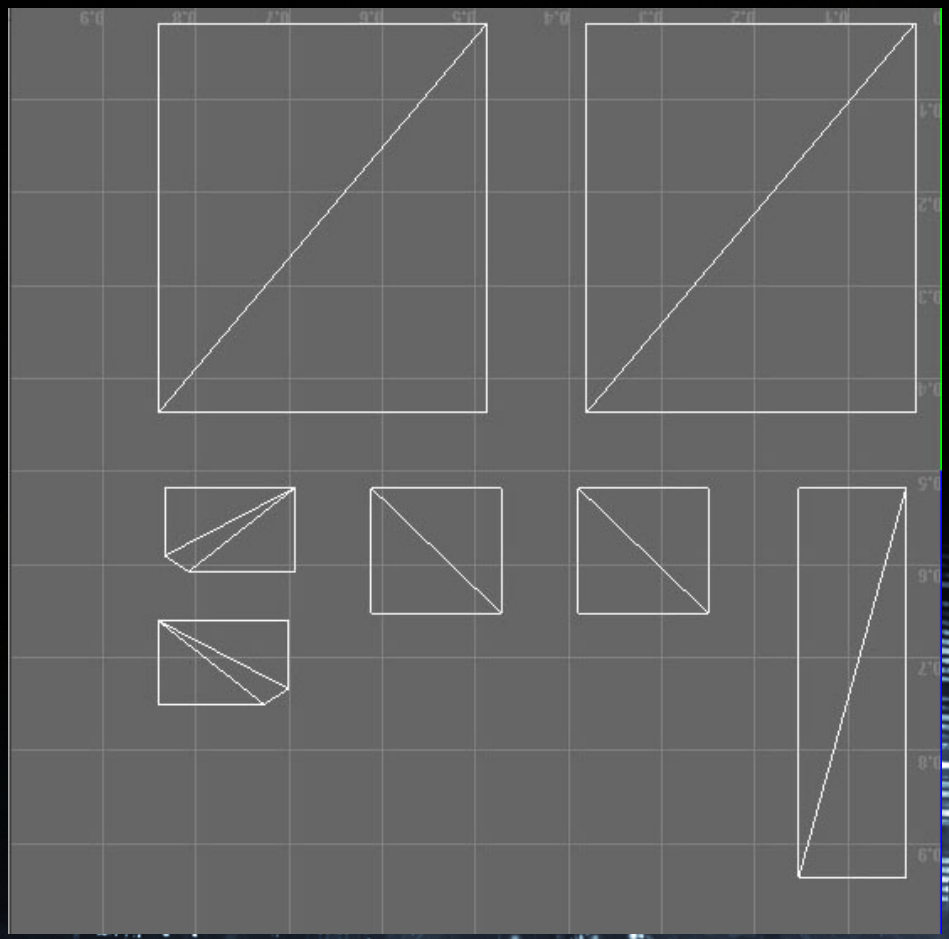
- Lightmap lit geometry
  - Detail geometry
    - UVs generated by projection
    - No additional lighting data
    - “Off-axis” lighting comes from directional data in lightmap
  - Target geometry



We need to make simpler geometry for the Lighting system to be able to handle all the radiosity data

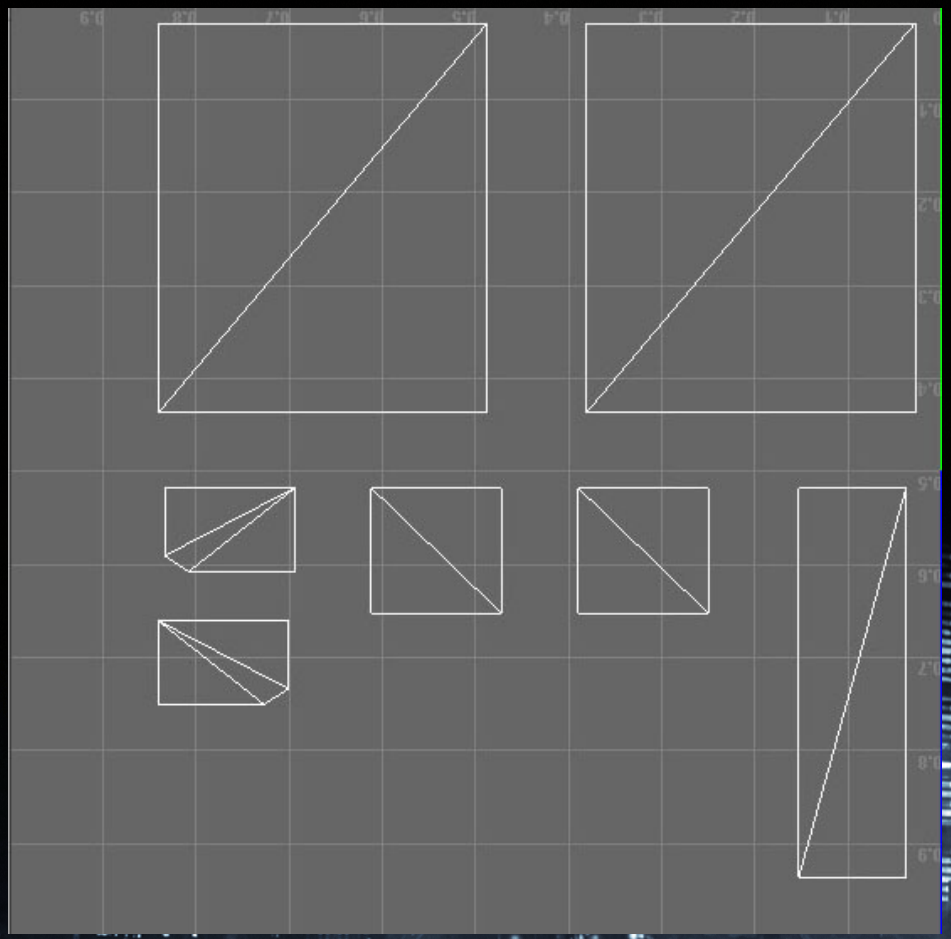
HOW:

- Lightmap lit geometry
  - Detail geometry
    - UVs generated by projection
    - No additional lighting data
    - “Off-axis” lighting comes from directional data in lightmap
  - Target geometry
    - Has simple UV surface area



HOW:

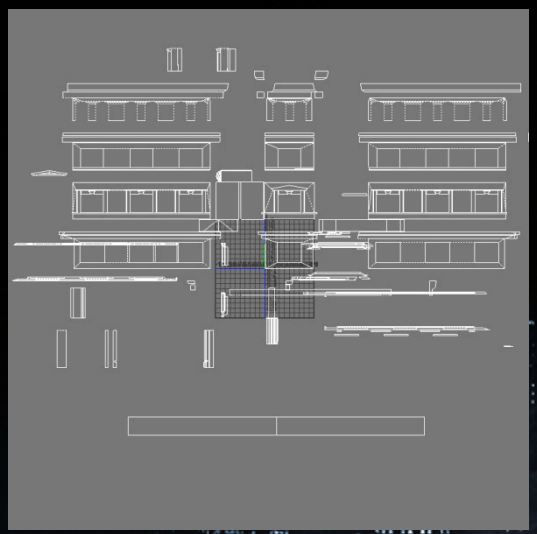
- Lightmap lit geometry
  - Detail geometry
    - UVs generated by projection
    - No additional lighting data
    - “Off-axis” lighting comes from directional data in lightmap
  - Target geometry
    - Has simple UV surface area
    - Poly count is not important
    - Various authoring options



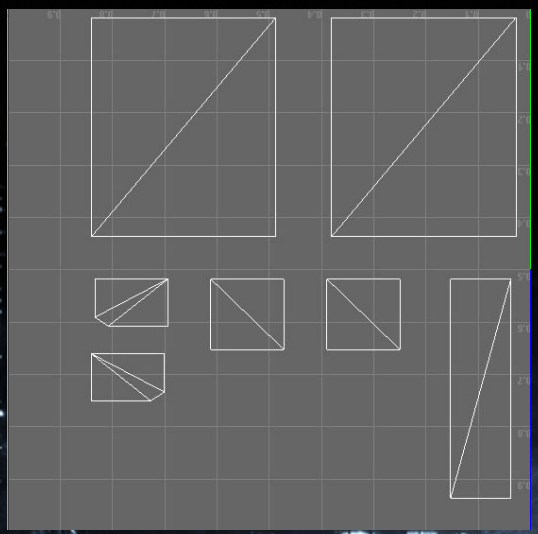
HOW:

- Lightmap lit geometry
  - Authoring of geometry
    - UVs

Detail geometry UV



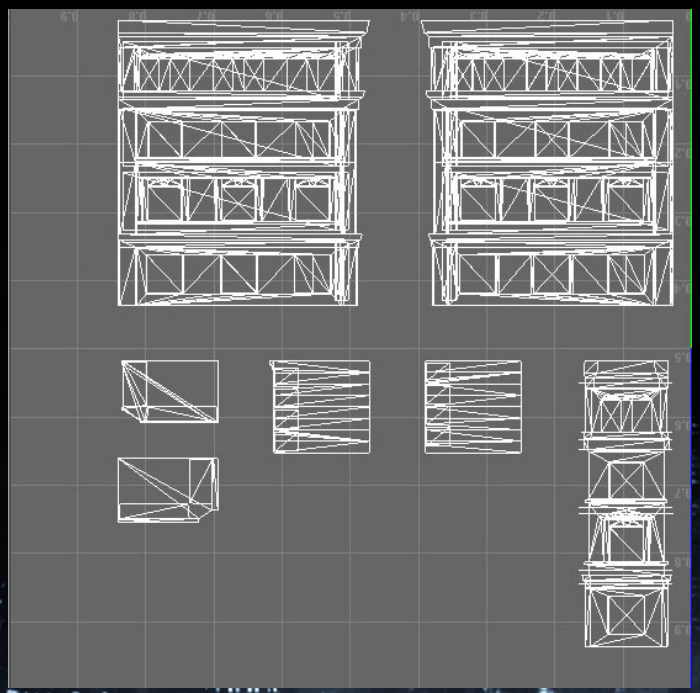
Target geometry UV



HOW:

- Lightmap lit geometry
  - Authoring of geometry
    - UVs
    - Surface transfer

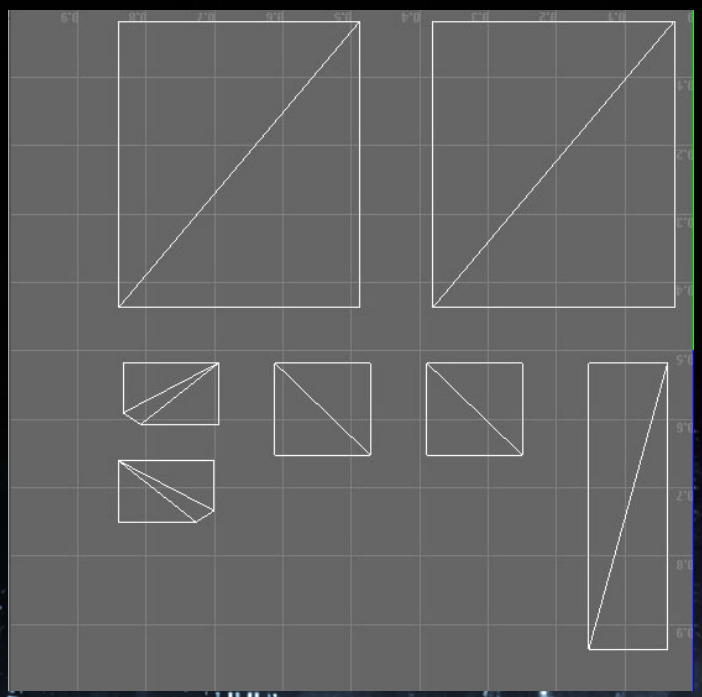
Detail geometry UV
Transferred to the
Target geometry UV



HOW:

- Lightmap lit geometry
  - Authoring of geometry
    - UVs
    - Surface transfer

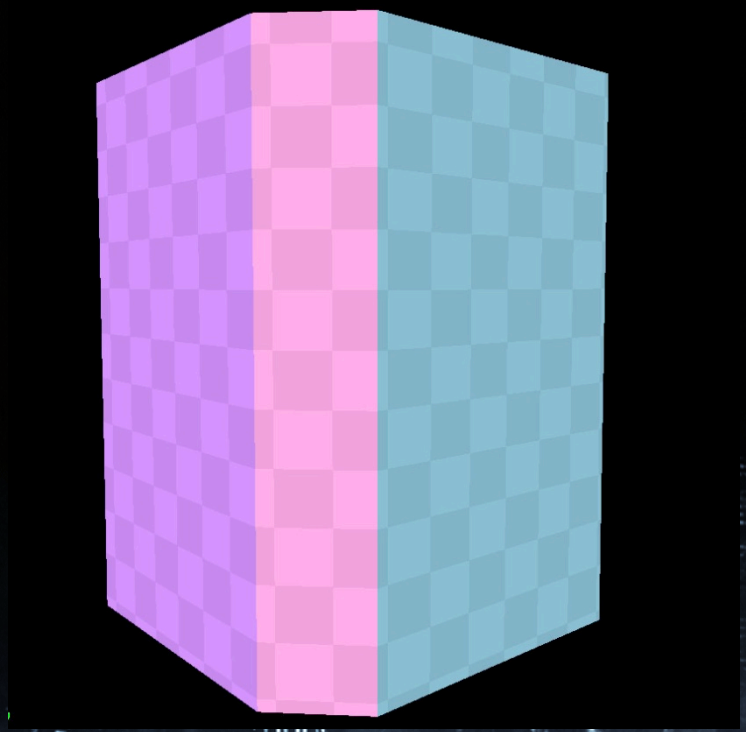
Detail geometry UV
Transferred to the
Target geometry UV



HOW:

- Lightmap lit geometry
  - Authoring of geometry
    - UVs
    - Surface transfer

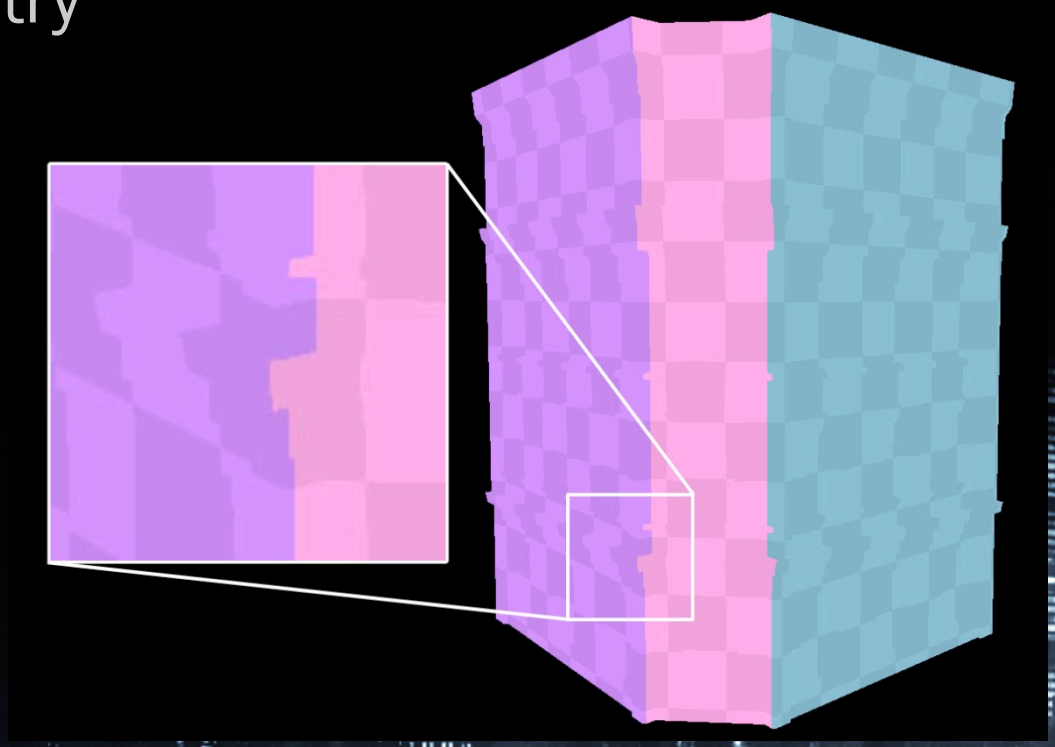
Target geometry lightmap



HOW:

- Lightmap lit geometry
  - Authoring of geometry
    - UVs
    - Surface transfer

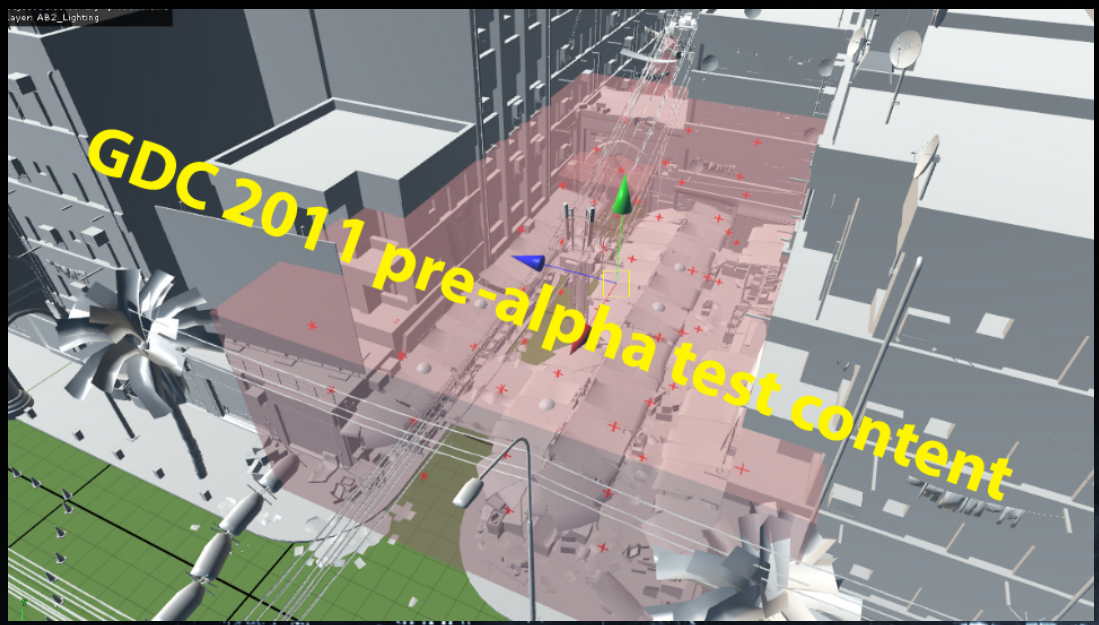
Target geometry lightmap
Transferred to the
Detail geometry



This uv projection is done when we import an object into the Editor.

HOW:

- Lightprobe lit geometry
  - Lightprobe volumes



We need to store lighting information for the dynamic, small and the very complex (organic) objects in different way.
Therefore, we place light probes in the world that store the global illumination lighting.
The objects that read these light probes can only receive light, not bounce or generate additional light.
The probes are stored in a grid:

HOW:

- Lightprobe lit geometry
  - Lightprobe volumes

in this picture their positions are represented as spheres. In each of these positions we store the lighting using a spherical harmonic representation, which means we basically have the incoming lighting in all directions. As dynamic objects move through this grid, they will pick up the correct lighting.

HOW:

- Lightprobe lit geometry
  - Lightprobe volumes

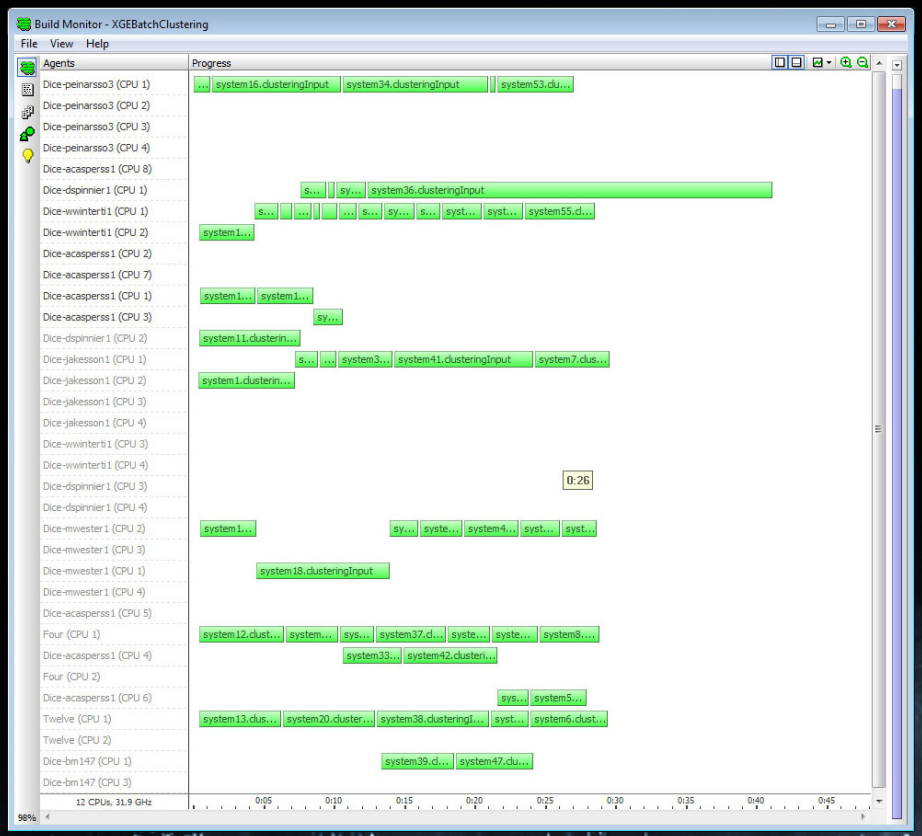
BATTLEFIELD 3

DICE

in this picture their positions are represented as spheres. In each of these positions we store the lighting using a spherical harmonic representation, which means we basically have the incoming lighting in all directions. As dynamic objects move through this grid, they will pick up the correct lighting.

HOW:

- Precompute
  - Analyze the geometry



Before we can run the game with dynamic radiosity we have to do a precompute pass in the pipeline.

In this step we analyze the geometry in the scene and generate data which we need in the game to generate the dynamic lighting.

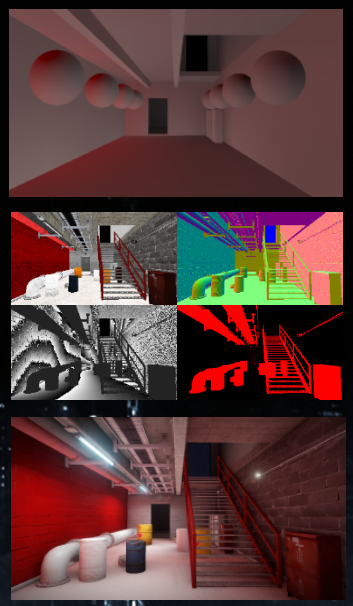
This step can be time consuming. For the scene you've seen here it takes about 20 minutes.

But we only have to redo this step when we modify the static geometry in the scene, not when we change the lighting or the surface colors.

HOW:

- Runtime lighting pipeline

1. Radiosity pass (CPU)
   - Update indirect lightmaps and lightprobes
2. Geometry pass (GPU)
   - Store indirect lighting in separate G-buffer
3. Light pass (GPU)
   - Render deferred light sources
   - Add indirect lighting from G-buffer



BATTLEFIELD 3

DICE

The complete render pipeline looks like this. First, we update lightmaps & light probes on the CPU.

Next, we run the geometry pass, where we add bounce light from the lightmaps and lightprobes to a separate g-buffer.

Finally, in the light pass, we first render all deferred lights. Then We add the indirect lighting from the g-buffer.

HOW:

- Small environments
  - Sky visibility

Sky visibility is a term we use to say how much of the sky each pixel can see.

We use it to blend between a dynamic outdoor environment map and a static indoor environment map to avoid the problem of a strong blue sky reflection on objects inside buildings as well as to smoothly transition between indoor and outdoor environments.

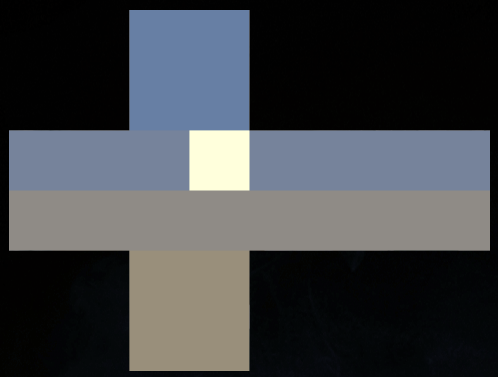
HOW:

- Small environments
  - Sky visibility
  - Environment maps

The indoor environment map gets its color and intensity from the dynamic radiosity that we calculate and this is combined with a black & white albedo envmap that modulates it and creates view-dependent motion in the surface.

HOW:

- Large environments
  - Skylight



BATTLEFIELD 3

DICE

We need to make a spherical harmonic representation of the sky. This is used for lighting big outdoor environments. The small Envmap cross you see to the left of every picture is the spherical harmonic debug view.

HOW:

- Large environments
  - Skylight

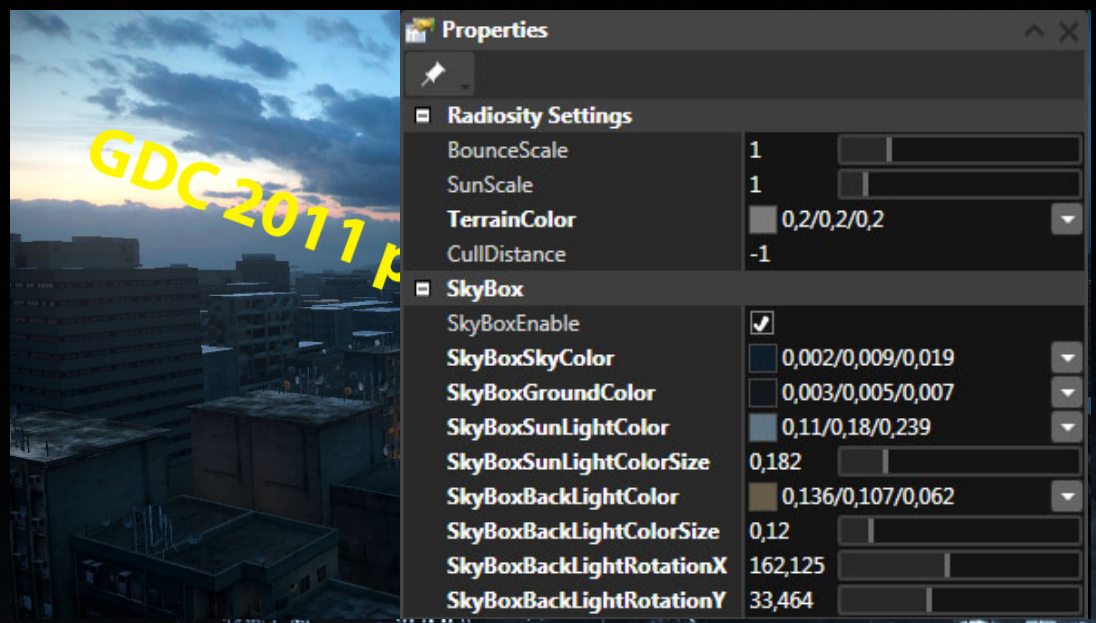
BATTLEFIELD 3

DICE

We need to make a spherical harmonic representation of the sky. This is used for lighting big outdoor environments. The small Envmap cross you see in the middle of every picture is the spherical harmonic debug view.

HOW:

- Large environments
  - Skylight



BATTLEFIELD 3

DICE

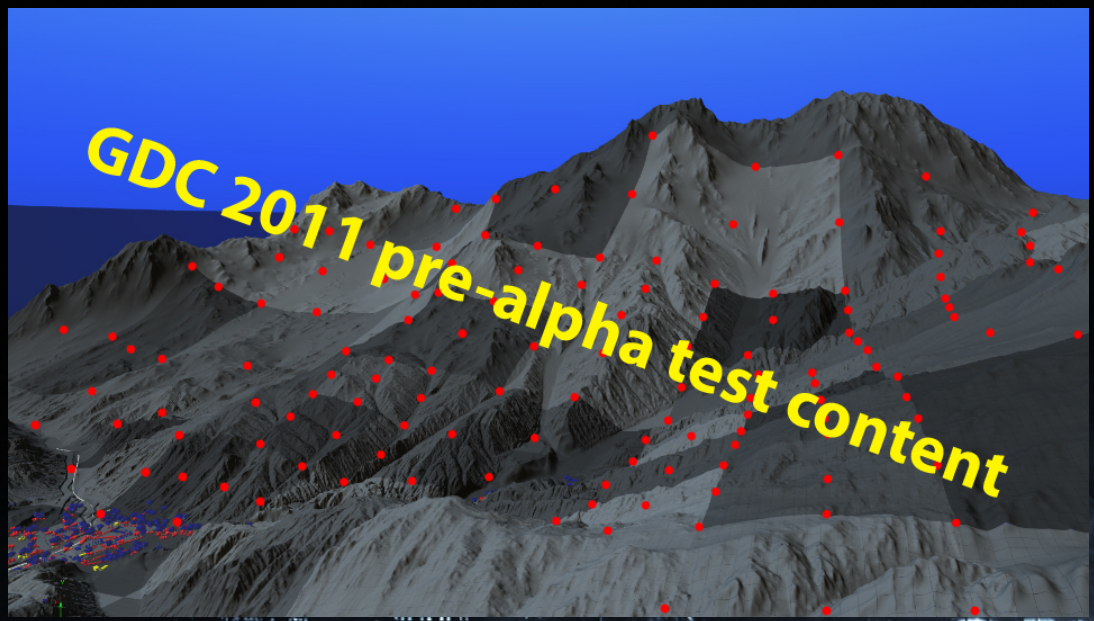
We need to make a spherical harmonic representation of the sky. This is used for lighting big outdoor environments. The small Envmap cross you see in the middle of every picture is the spherical harmonic debug view.

HOW:

- Large environments
  - Skylight
  - Terrain

HOW:

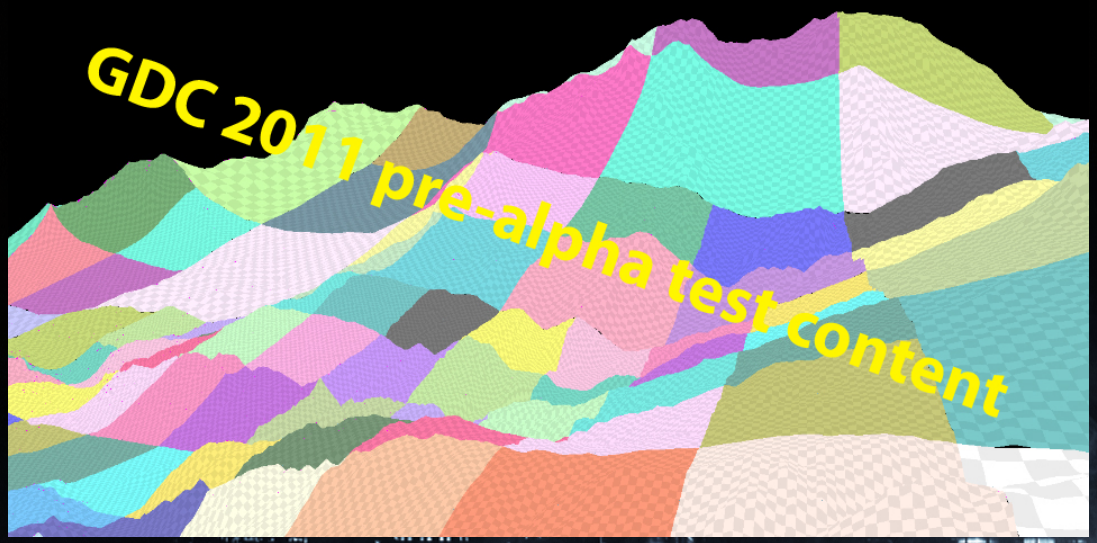
- Large environments
  - Skylight
  - Terrain



Every red point you see in the picture is a light probe, it's getting snapped to the height field of the terrain. The resolution of the light probe grid depends on the terrain patch size.

HOW:

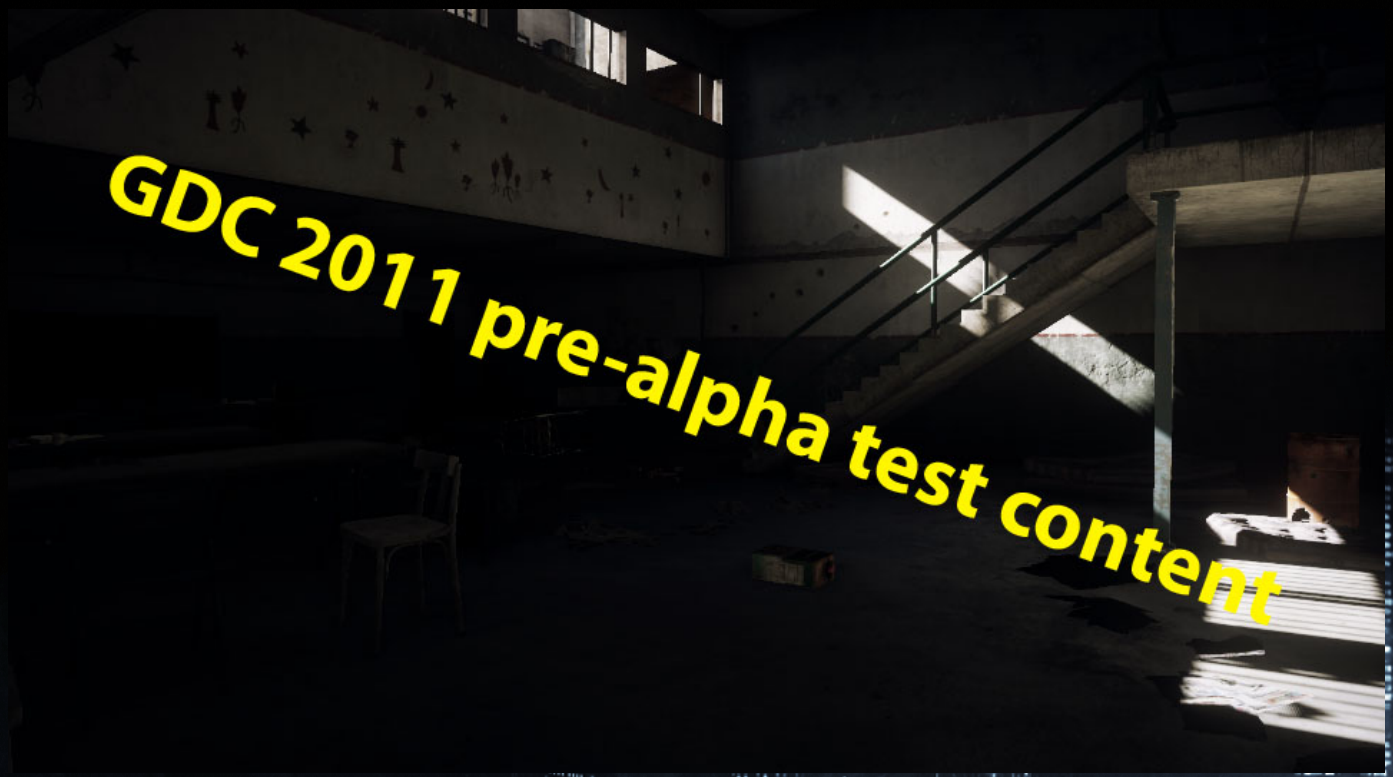
- Large environments
  - Skylight
  - Terrain



It’s the same for the lightmaps for the terrain, it also deepens on the terrain patch size..

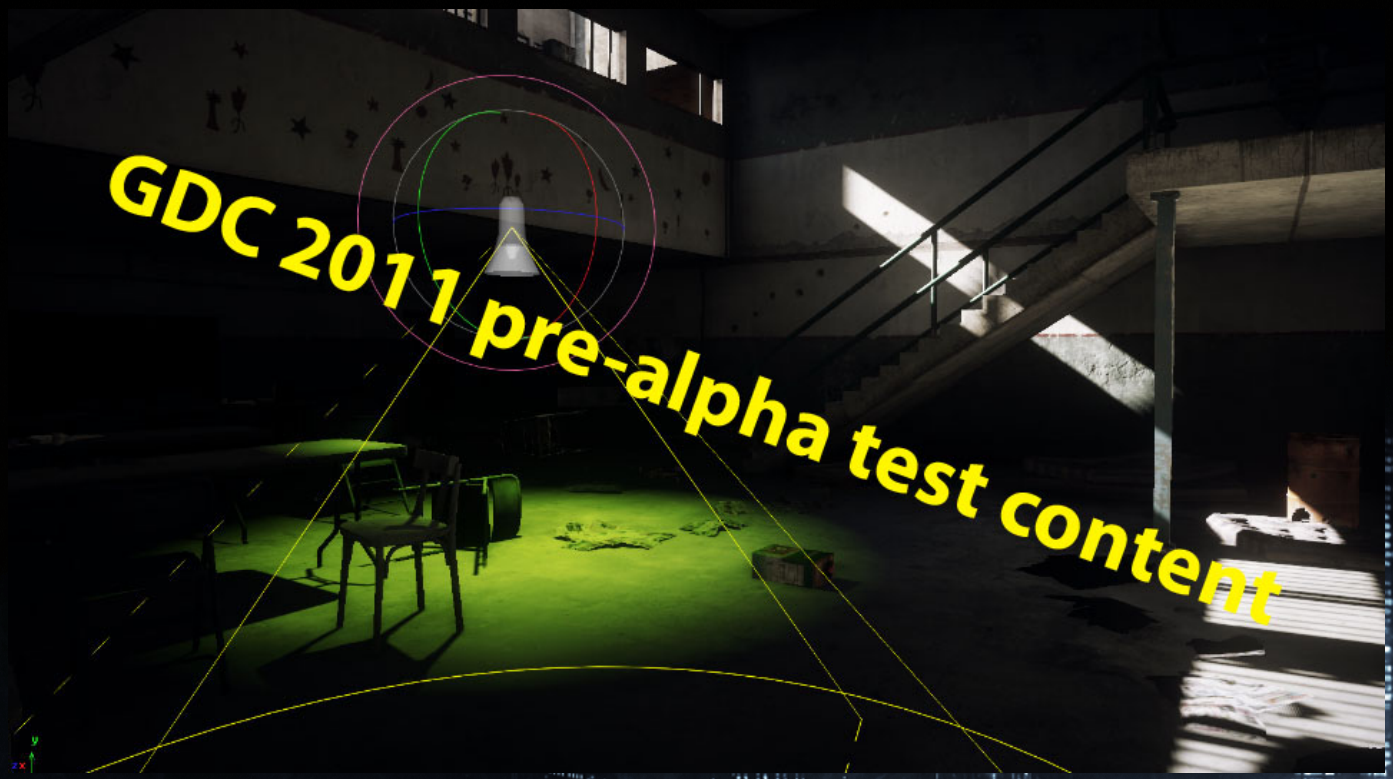
HOW:

- Light



HOW:

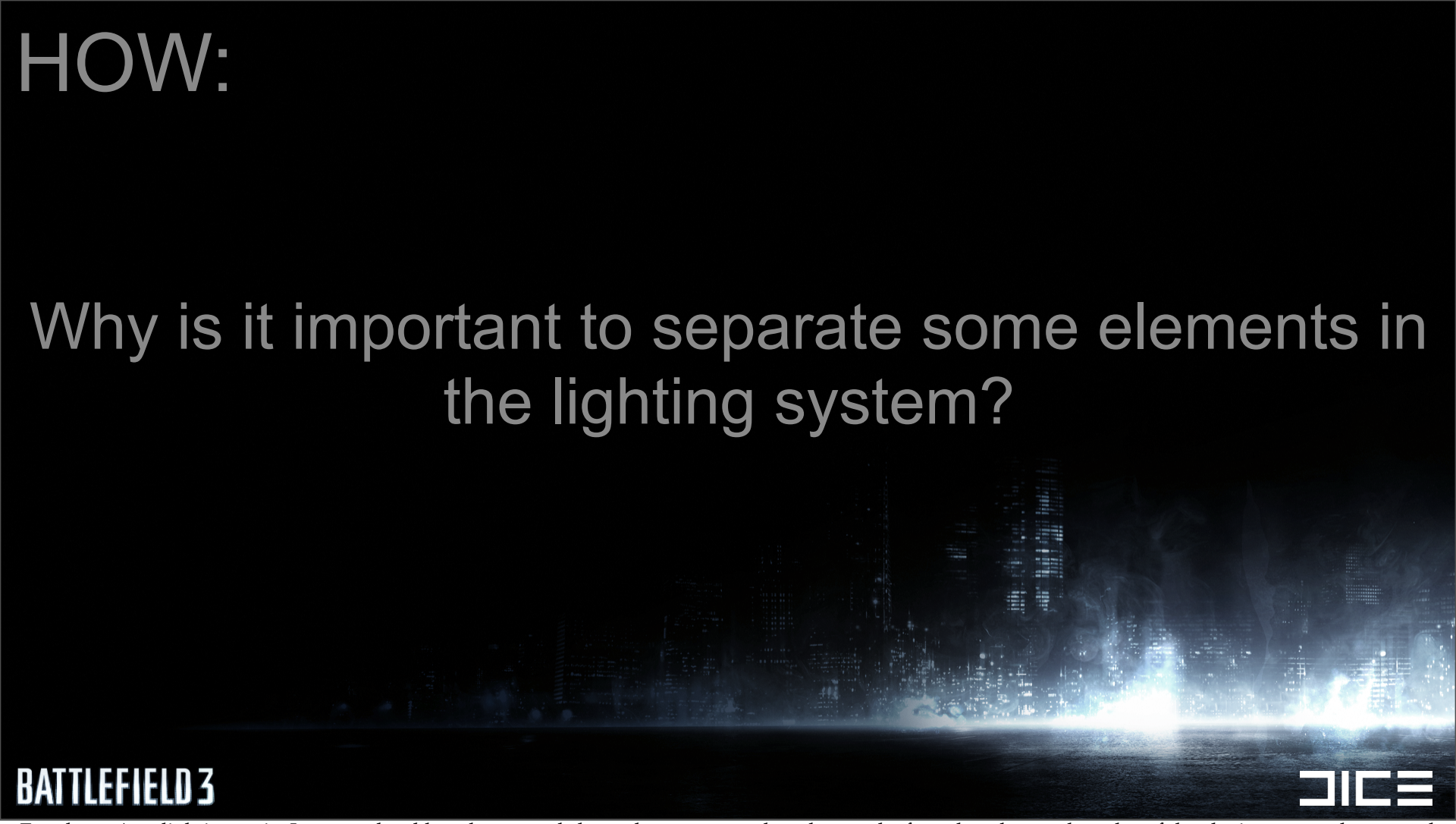
- Light



HOW:

- Light

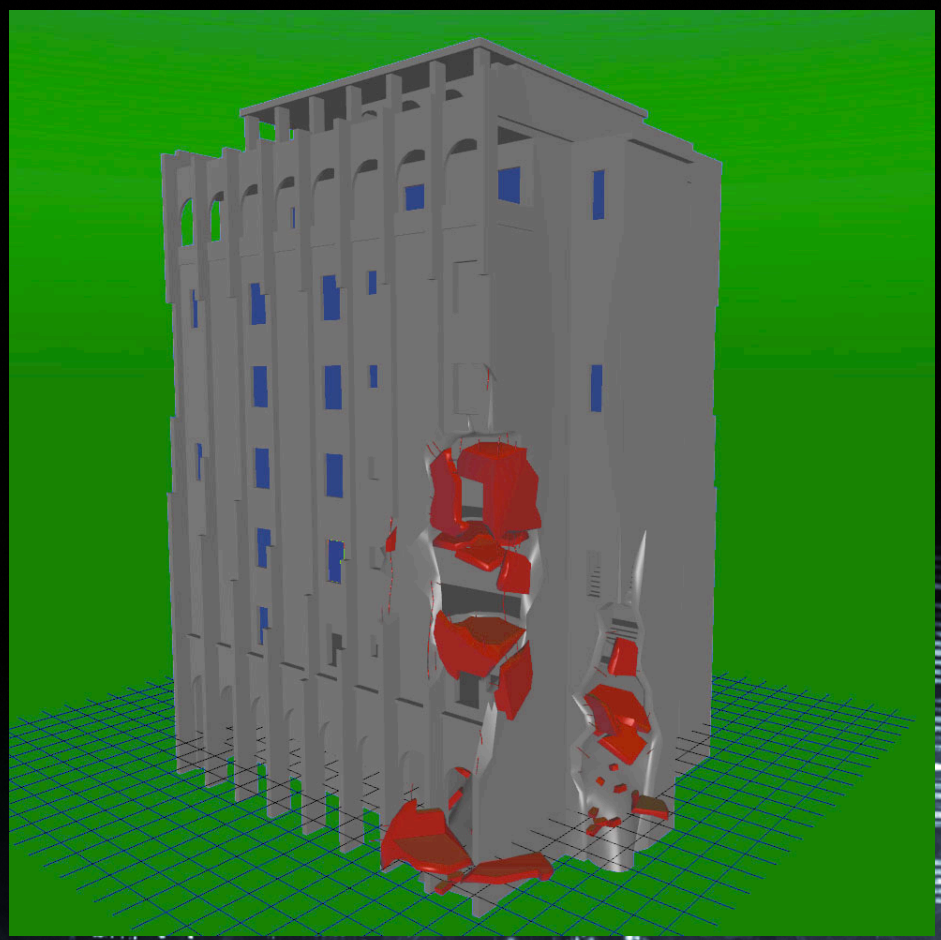
HOW:



Freedom : As a lighting artist I want to be able to have a red sky and green ground, and not to be forced to change the color of the sky just to get the ground brighter. You need to be careful what systems interact with each other in a lighting system.

HOW:

- Best practice
  - Complex geometry



BATTLEFIELD 3

DICE

Complex geometry : Working with light probes: for complex objects (i.e. house), some parts can be taken out of the geometry and imported as lightprobe lit geometry to optimize authoring time. Gray is lightmap lit geometry, red are lightprobe lit geometry and the blue are transparent surface and don't need to be lit by enlighten

Static radiosity maps : You can use this system as a fast way of baking out light maps / radiosity maps in this case (no shados).

Real-time : The real-time part on Enlighten can be used in smaller contained areas.

Streaming : Streaming in and out baked radiosity maps works fine depending on the sizes of the streaming zones

Time lapse : Time-lapse works only with dynamic enlighten enabled.

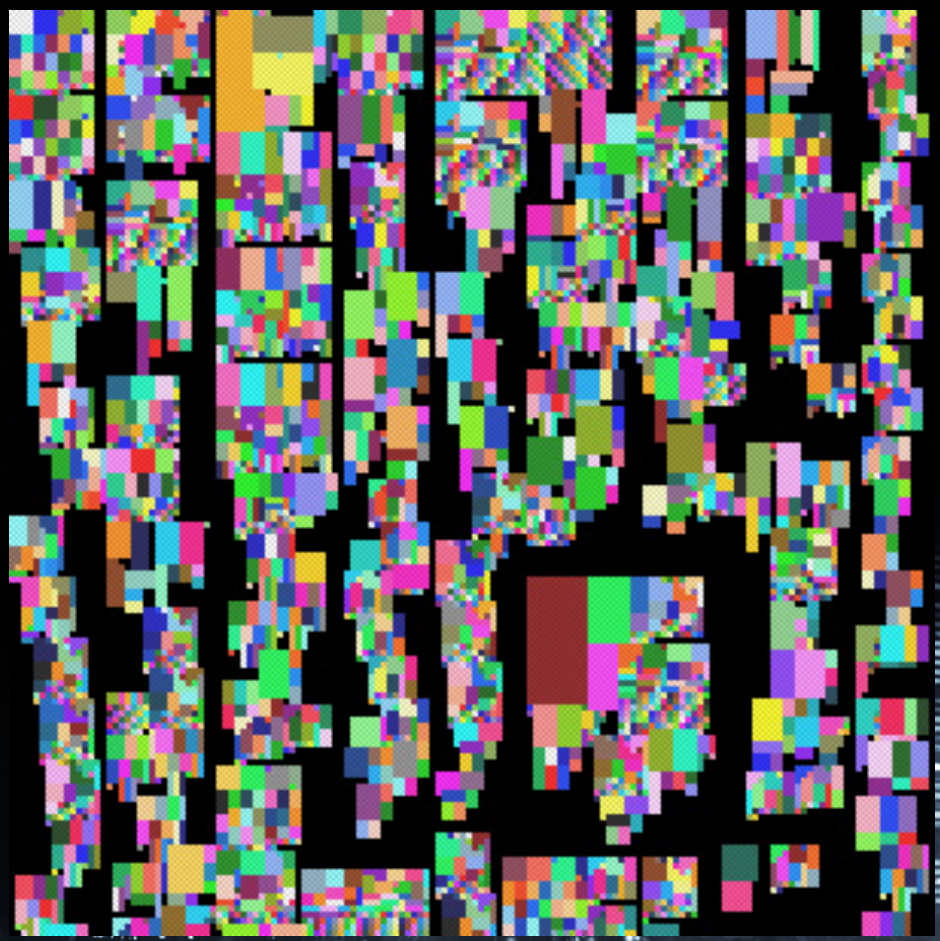
Range : Be careful if you have too much range in the lighting because of the data stored in the light probes are packed.

Color Grading : Looking how film is post processed, color grading should be applied last in the chain.

Filmic tone mapping : . There are a couple of filmic tonemapping formulas out there, use the one that suit the art direction of your game.

HOW:

- Best practice
  - Complex geometry
  - Static radiosity maps



BATTLEFIELD 3

DICE

Complex geometry : Working with light probes: for complex objects (i.e. house), some parts can be taken out of the geometry and imported as lightprobe lit geometry to optimize authoring time. Gray is lightmap lit geometry, red are lightprobe lit geometry and the blue are transparent surface and don't need to be lit by enlighten

Static radiosity maps : You can use this system as a fast way of baking out light maps / radiosity maps in this case (no shados).

Real-time : The real-time part on Enlighten can be used in smaller contained areas.

Streaming : Streaming in and out baked radiosity maps works fine depending on the sizes of the streaming zones

Time lapse : Time-lapse works only with dynamic enlighten enabled.

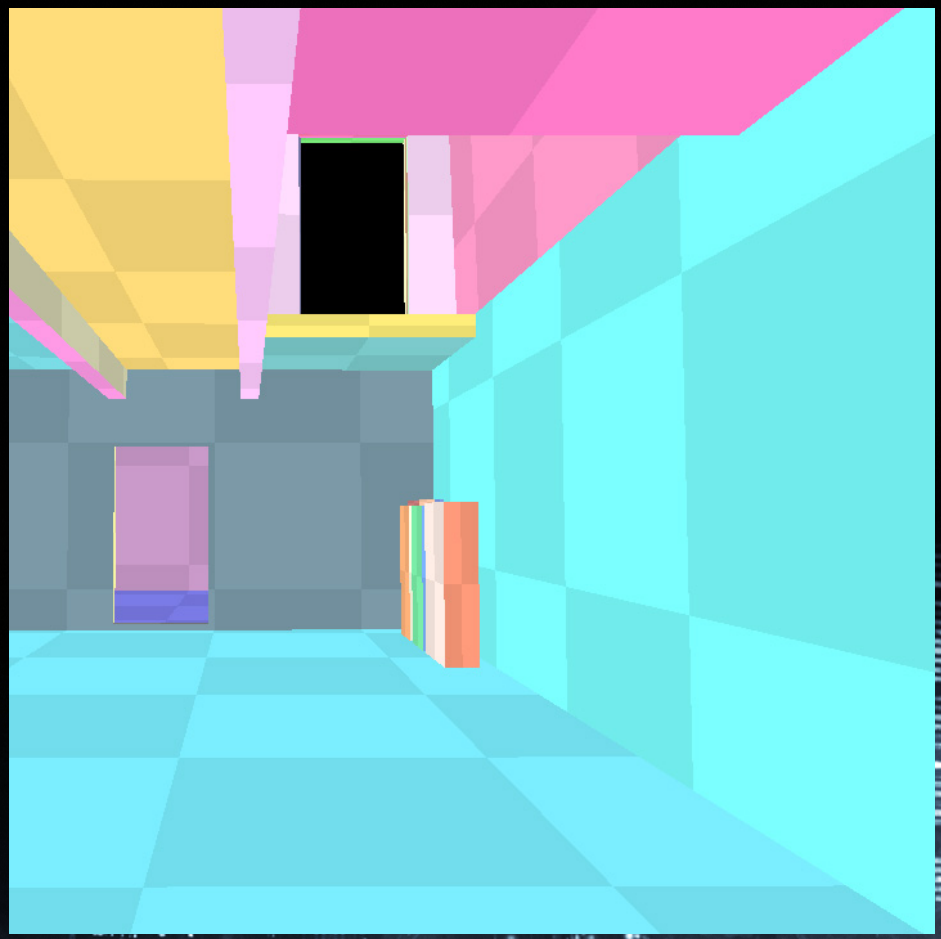
Range : Be careful if you have too much range in the lighting because of the data stored in the light probes are packed.

Color Grading : Looking how film is post processed, color grading should be applied last in the chain.

Filmic tone mapping : . There are a couple of filmic tonemapping formulas out there, use the one that suit the art direction of your game.

HOW:

- Best practice
  - Complex geometry
  - Static radiosity maps
  - Real-time



BATTLEFIELD 3

DICE

Complex geometry : Working with light probes: for complex objects (i.e. house), some parts can be taken out of the geometry and imported as lightprobe lit geometry to optimize authoring time. Gray is lightmap lit geometry, red are lightprobe lit geometry and the blue are transparent surface and don't need to be lit by enlighten

Static radiosity maps : You can use this system as a fast way of baking out light maps / radiosity maps in this case (no shados).

Real-time : The real-time part on Enlighten can be used in smaller contained areas.

Streaming : Streaming in and out baked radiosity maps works fine depending on the sizes of the streaming zones

Time lapse : Time-lapse works only with dynamic enlighten enabled.

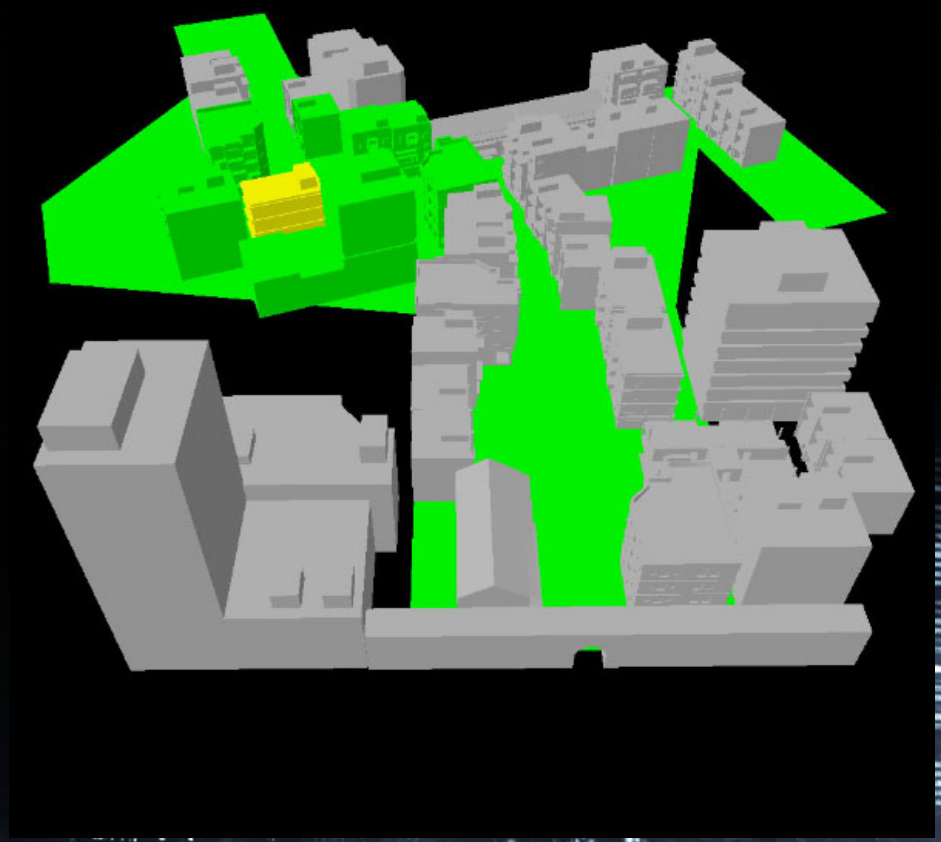
Range : Be careful if you have too much range in the lighting because of the data stored in the light probes are packed.

Color Grading : Looking how film is post processed, color grading should be applied last in the chain.

Filmic tone mapping : . There are a couple of filmic tonemapping formulas out there, use the one that suit the art direction of your game.

HOW:

- Best practice
  - Complex geometry
  - Static radiosity maps
  - Real-time
  - Streaming



BATTLEFIELD 3

DICE

Complex geometry : Working with light probes: for complex objects (i.e. house), some parts can be taken out of the geometry and imported as lightprobe lit geometry to optimize authoring time. Gray is lightmap lit geometry, red are lightprobe lit geometry and the blue are transparent surface and don't need to be lit by enlighten

Static radiosity maps : You can use this system as a fast way of baking out light maps / radiosity maps in this case (no shados).

Real-time : The real-time part on Enlighten can be used in smaller contained areas.

Streaming : Streaming in and out baked radiosity maps works fine depending on the sizes of the streaming zones

Time lapse : Time-lapse works only with dynamic enlighten enabled.

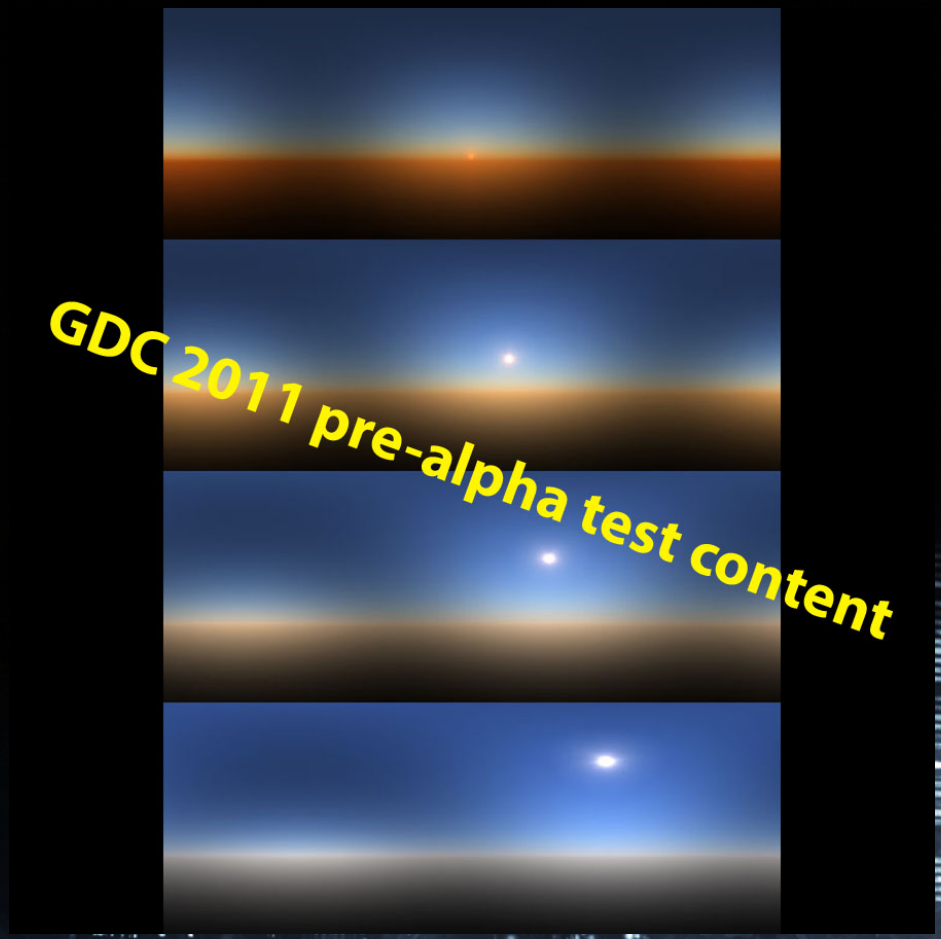
Range : Be careful if you have too much range in the lighting because of the data stored in the light probes are packed.

Color Grading : Looking how film is post processed, color grading should be applied last in the chain.

Filmic tone mapping : . There are a couple of filmic tonemapping formulas out there, use the one that suit the art direction of your game.

HOW:

- Best practice
  - Complex geometry
  - Static radiosity maps
  - Real-time
  - Streaming
  - Time laps



BATTLEFIELD 3

DICE

Complex geometry : Working with light probes: for complex objects (i.e. house), some parts can be taken out of the geometry and imported as lightprobe lit geometry to optimize authoring time. Gray is lightmap lit geometry, red are lightprobe lit geometry and the blue are transparent surface and don't need to be lit by enlighten

Static radiosity maps : You can use this system as a fast way of baking out light maps / radiosity maps in this case (no shados).

Real-time : The real-time part on Enlighten can be used in smaller contained areas.

Streaming : Streaming in and out baked radiosity maps works fine depending on the sizes of the streaming zones

Time lapse : Time-lapse works only with dynamic enlighten enabled.

Range : Be careful if you have too much range in the lighting because of the data stored in the light probes are packed.

Color Grading : Looking how film is post processed, color grading should be applied last in the chain.

Filmic tone mapping : . There are a couple of filmic tonemapping formulas out there, use the one that suit the art direction of your game.

HOW:

- Best practice
  - Complex geometry
  - Static radiosity maps
  - Real-time
  - Streaming
  - Time laps
  - Range

BATTLEFIELD 3

DICE

Complex geometry : Working with light probes: for complex objects (i.e. house), some parts can be taken out of the geometry and imported as lightprobe lit geometry to optimize authoring time. Gray is lightmap lit geometry, red are lightprobe lit geometry and the blue are transparent surface and don't need to be lit by enlighten

Static radiosity maps : You can use this system as a fast way of baking out light maps / radiosity maps in this case (no shados).

Real-time : The real-time part on Enlighten can be used in smaller contained areas.

Streaming : Streaming in and out baked radiosity maps works fine depending on the sizes of the streaming zones

Time lapse : Time-lapse works only with dynamic enlighten enabled.

Range : Be careful if you have too much range in the lighting because of the data stored in the light probes are packed.

Color Grading : Looking how film is post processed, color grading should be applied last in the chain.

Filmic tone mapping : . There are a couple of filmic tonemapping formulas out there, use the one that suit the art direction of your game.

HOW:

- Best practice
  - Complex geometry
  - Static radiosity maps
  - Real-time
  - Streaming
  - Time laps
  - Range
  - Color Grading
    - ON

HOW:

- Best practice
  - Complex geometry
  - Static radiosity maps
  - Real-time
  - Streaming
  - Time laps
  - Range
  - Color Grading
    - OFF

HOW:

- Best practice
  - Complex geometry
  - Static radiosity maps
  - Real-time
  - Streaming
  - Time laps
  - Range
  - Color Grading
  - Filmic tone mapping
    - ON

HOW:

- Best practice
  - Complex geometry
  - Static radiosity maps
  - Real-time
  - Streaming
  - Time laps
  - Range
  - Color Grading
  - Filmic tone mapping
    - OFF

CONCLUSION:

CONCLUSION:

The advantages

Time

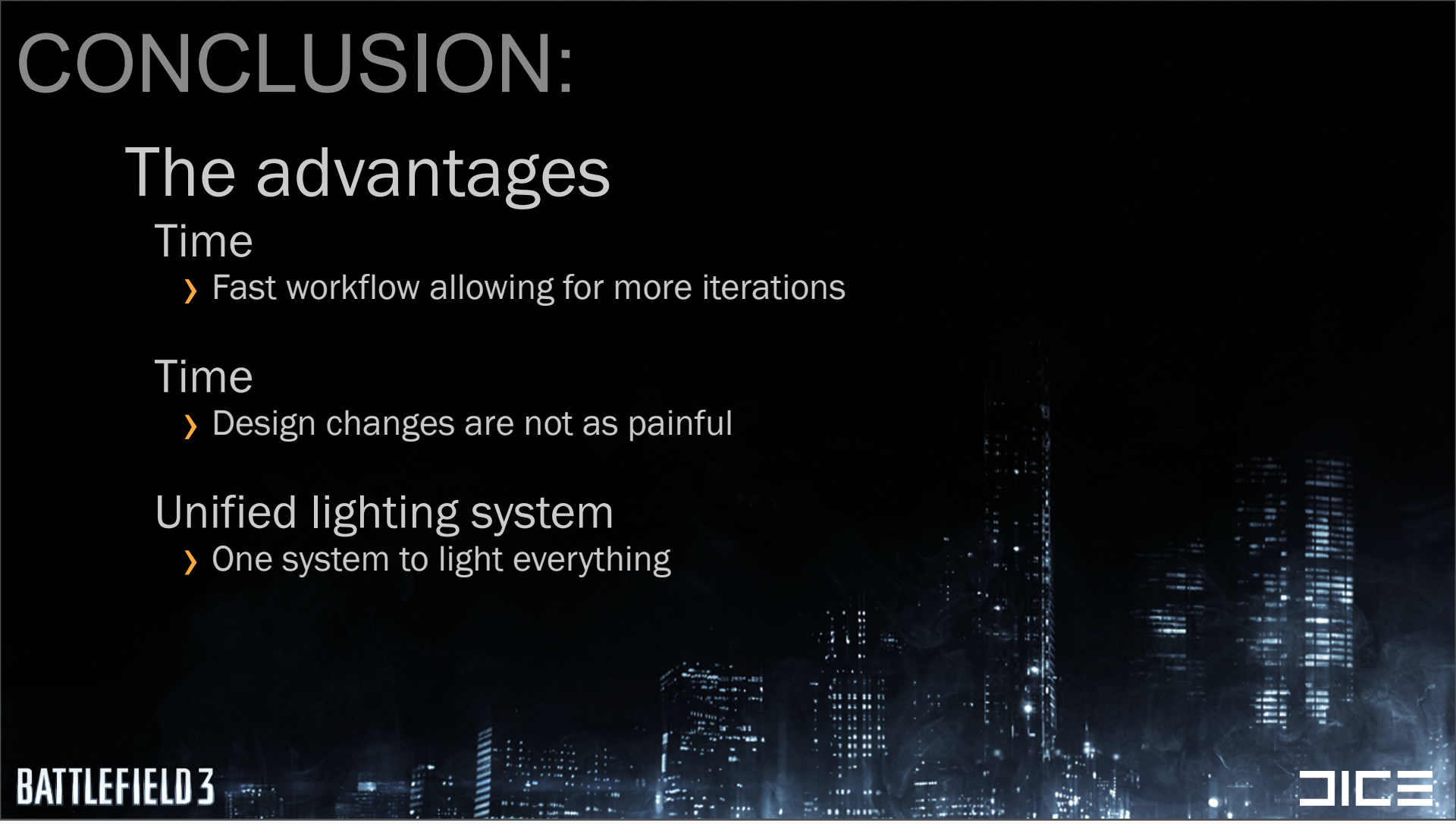
- Fast workflow allowing for more iterations

Time

- Design changes are not as painful

Unified lighting system

- One system to light everything



CONCLUSION:

CONCLUSION:

The disadvantages

Memory

- The radiosity maps uses ‘some’ memory

Performance

- Dependent on light probe density
- Size and amount of pointlights

Authoring of geometry

- It takes time
- Grasp the concept

Precompute

- Can take time depending on size of level

SUMMARY:

SUMMARY:

Workflow

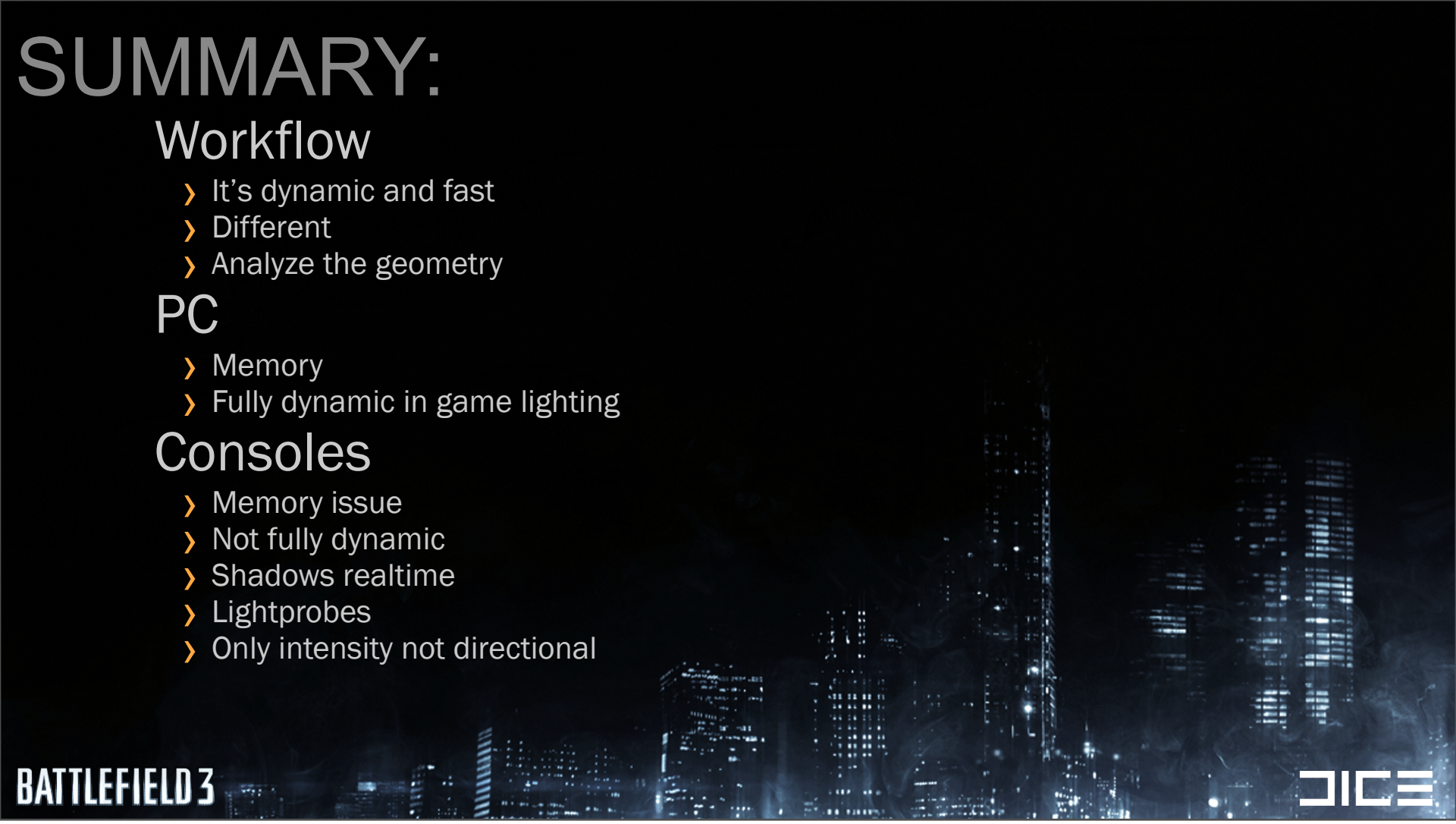
- It's dynamic and fast
- Different
- Analyze the geometry

PC

- Memory
- Fully dynamic in game lighting

Consoles

- Memory issue
- Not fully dynamic
- Shadows realtime
- Lightprobes
- Only intensity not directional



I want to thank the following:

Questions?

Email: Kenny@dice.se
WWW.dice.se
MSN: Louie50028@hotmail.com

Battlefield 3 & Frostbite 2 talks at GDC'11:

Mon 1:45	*DX11 Rendering in Battlefield 3*	Johan Andersson
Wed 10:30	*SPU-based Deferred Shading in Battlefield 3 for PlayStation 3*	Christina Coffin
Wed 3:00	*Culling the Battlefield: Data Oriented Design in Practice*	Daniel Collin
Thu 1:30	*Lighting You Up in Battlefield 3*	Kenny Magnusson
Fri 4:05	*Approximating Translucency for a Fast, Cheap & Convincing Subsurface Scattering Look*	Colin Barré-Brisebois

GDC

For more DICE talks: http://publications.dice.se

DICE