The future of PC Gaming:
The possibilities of DirectX 10

Bob Drebin, Fellow, ATI
Guennadi Riguer, Senior ISV Engineer, ATI

- Look stunning
  - Feature detailed characters and complex materials
  - Contain lots of effects
    - light bloom, motion blur, HDR lighting

Oblivion – No HDR   **Oblivion – HDR + 4x AA**

Image courtesy of Bethesda Softworks LLC's The Elder Scrolls IV: Oblivion

- Feel immersive

  - Feature convincing environments, vast areas to explore

Tomb Raider: Legend – Next Gen Content Off

Tomb Raider: Legend – Next Gen Content On

Image courtesy of Eidos's Tomb Raider: Legend

Today's games

- Play to more than just a Good Guy and Bad Guy
  - Can pick up and interact with particular objects
  - Can damage or destroy objects like walls or buildings

FEAR – Low Settings

FEAR – Max Settings

Image courtesy of Monolith's FEAR

Today's games

- Games today are a showcase of the genius of developers at bringing a false reality closer to real life than ever before

- We'll look at the constraints of today's GPUs and APIs that game developers are starting to hit

- DirectX 9 overhead
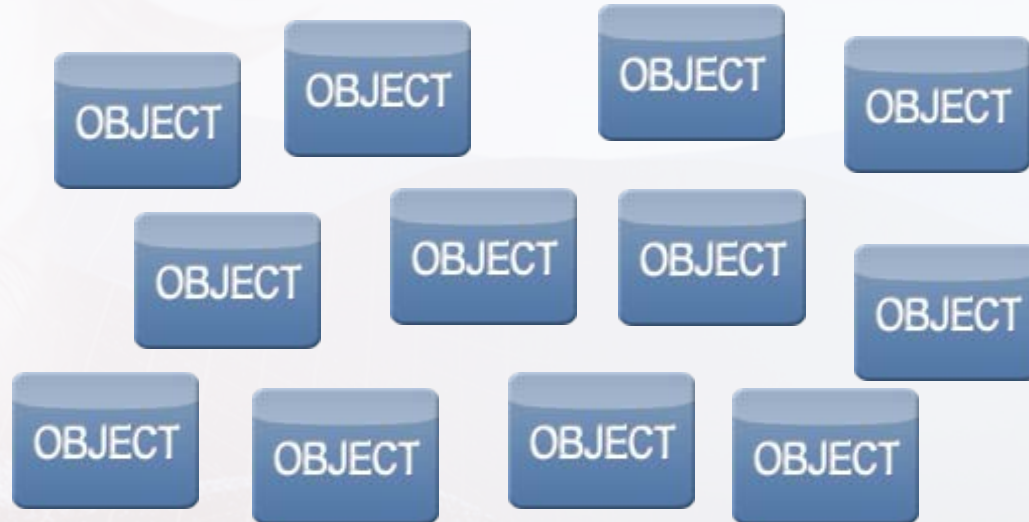
- DirectX 9 overhead

- DirectX 9 overhead

- DirectX 9 overhead

- DirectX 9 overhead

APPLICATION → API | OBJECT | OVER HEAD → DRIVER →

- DirectX 9 overhead
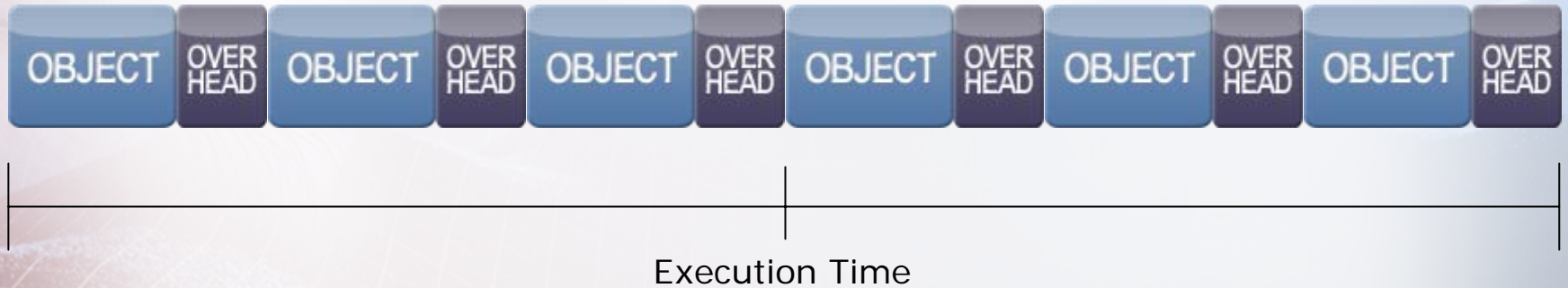
Execution Time

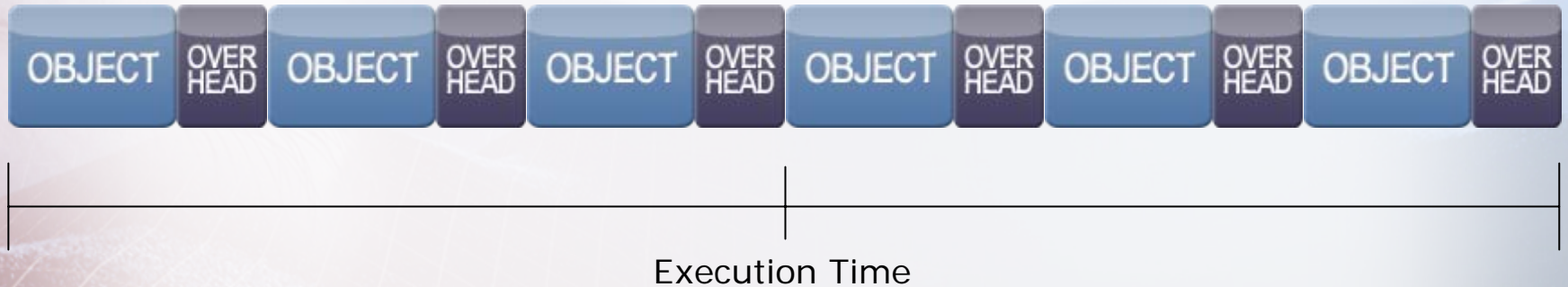More objects, more changes to those objects = Longer execution time

- DirectX 9 overhead

Execution Time

Longer execution time = Lower performance

Also known as the small batch problem

- Constraints related to computing resources

  - Limited to when you can use the GPU vs. the CPU

  - Makes a difference for large data set processing, for example, processing physics
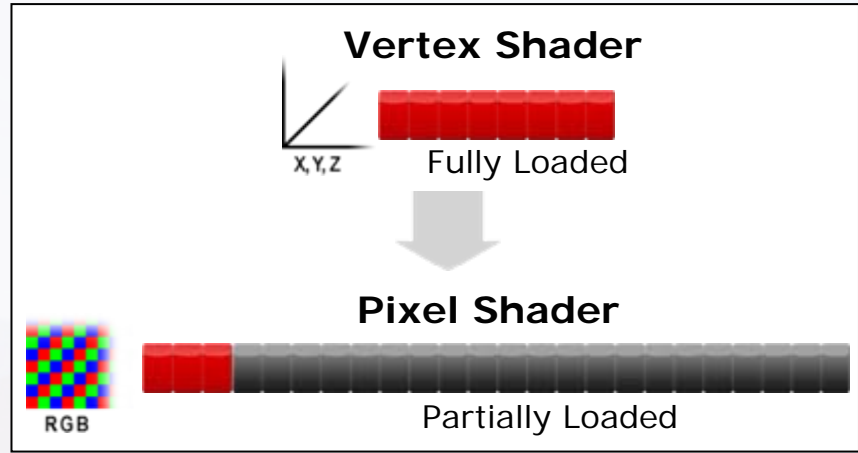
Removing constraints:
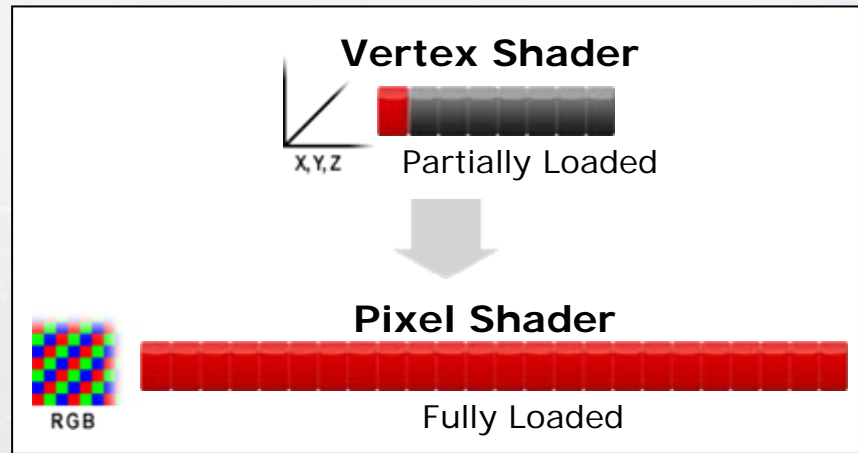# The Unified Shading Architecture

- Traditional GPU

Complex Geometry Processing

Complex Pixel Processing

Vertex Shader

Fully Loaded

Pixel Shader

Partially Loaded

Vertex Shader

Partially Loaded

Pixel Shader

Fully Loaded

- Traditional GPU extended to DirectX 10 pipeline
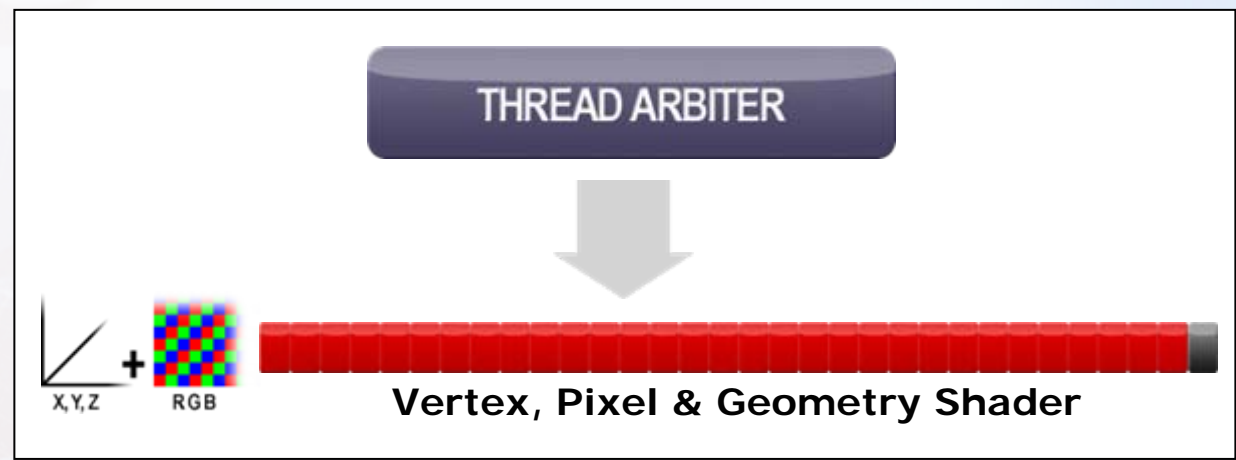

Vertex Shader

Geometry Shader
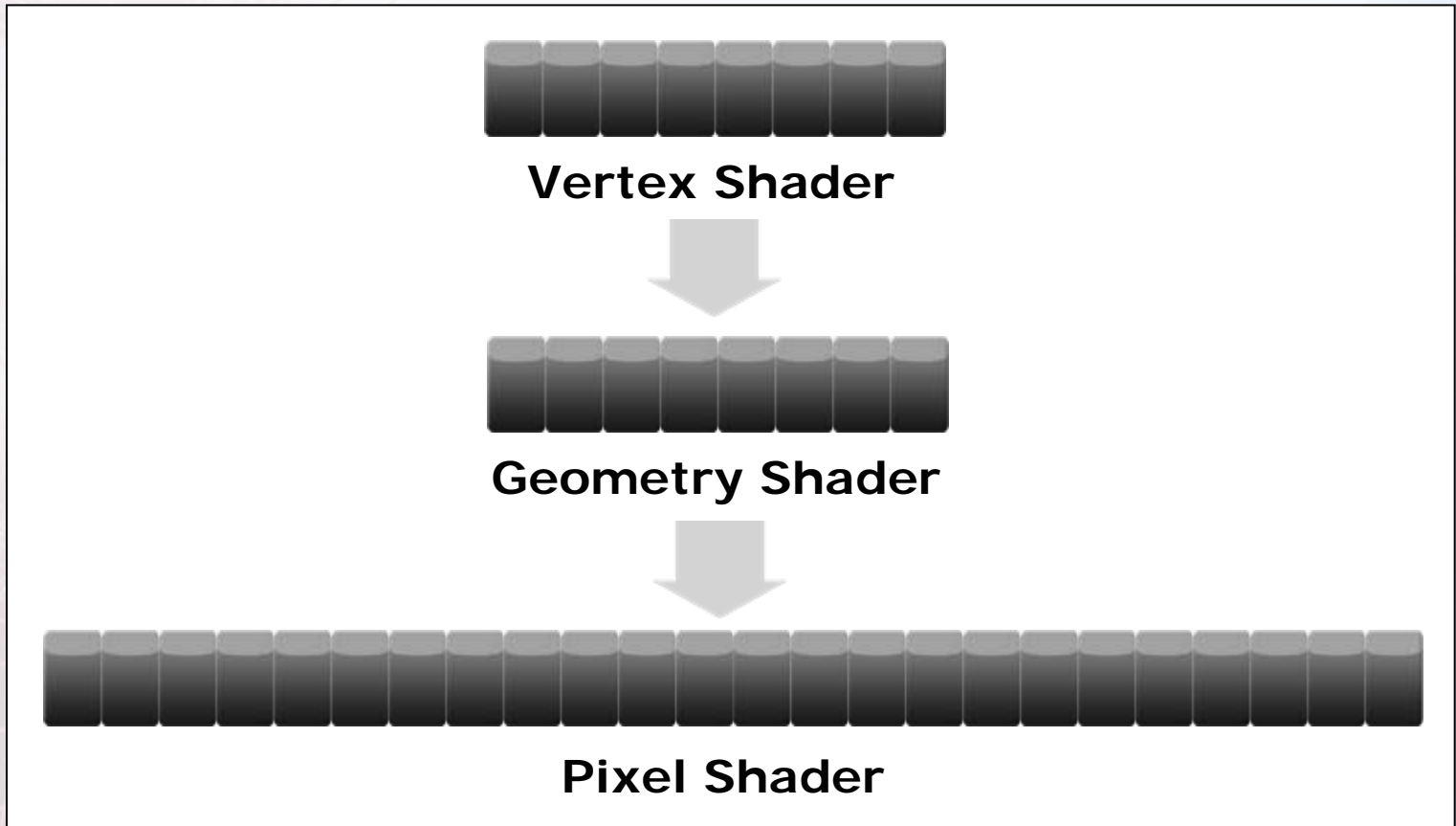
Pixel Shader

Removing constraints:
The Unified Shading Architecture

Unified Shader Architecture

**Complex Geometry
and
Pixel Processing**

THREAD ARBITER

X,Y,Z + RGB

Vertex, Pixel & Geometry Shader

Unified we stand, divided we fall

- The future of PC gaming hardware: the unified shader architecture

- Common Shader engine used to execute vertex, geometry and pixel, maximizing shader performance efficiencies

- First Unified Shading Architecture for the PC

  - ATI's DirectX 10 GPU

  - ATI's second Unified Shading Architecture
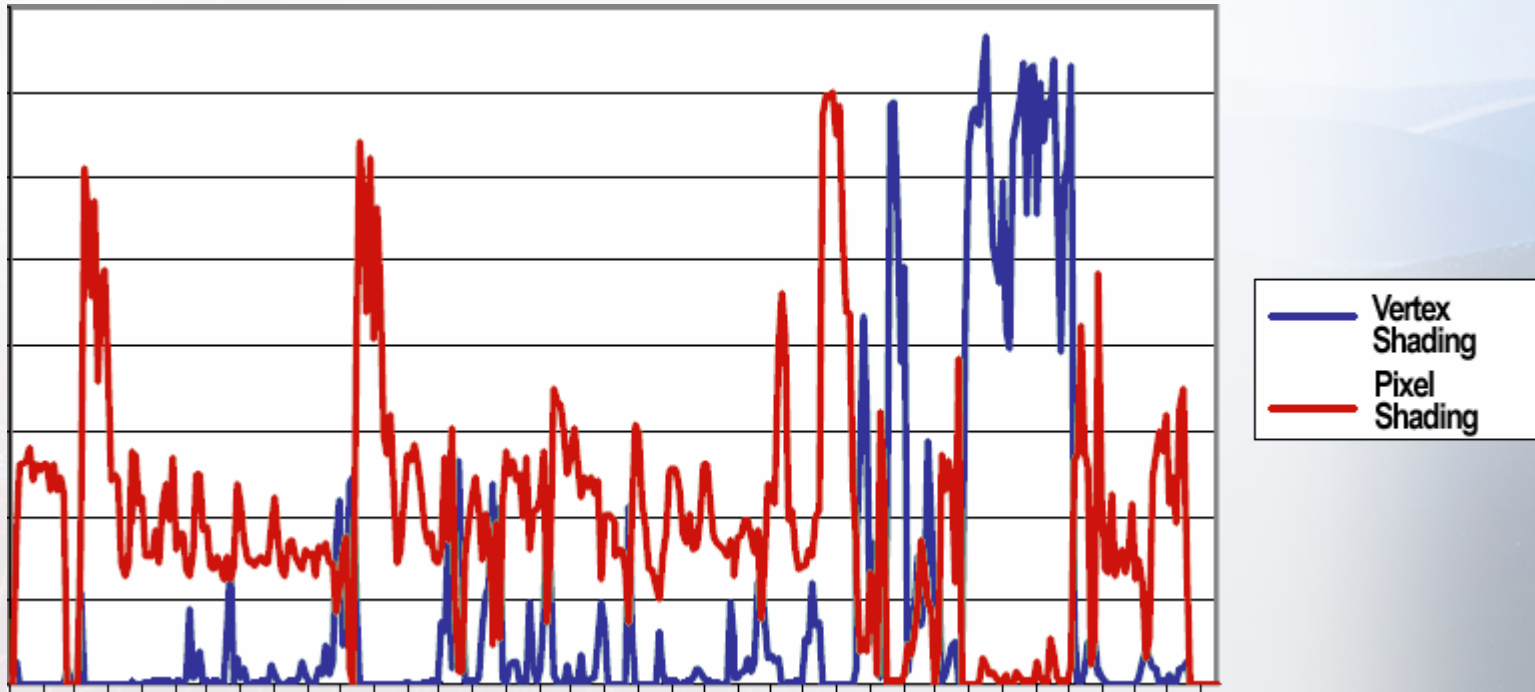
Unified we stand, divided we fall

- Why is a unified architecture so important?

  - GPU's role moving from *game rendering* to *game computing*

  - DirectX 10 was designed with unified shading in mind

    - Common specification and feature set for Vertex, Geometry and Pixel Shaders

    - Flexible and efficient memory & texture access expected in all shader types

Unified we stand, divided we fall

- Upcoming games will be both geometry and pixel compute intensive

    - In the same frame... but not at the same time!

Sample of average frame

Legend:
- Vertex Shading
- Pixel Shading
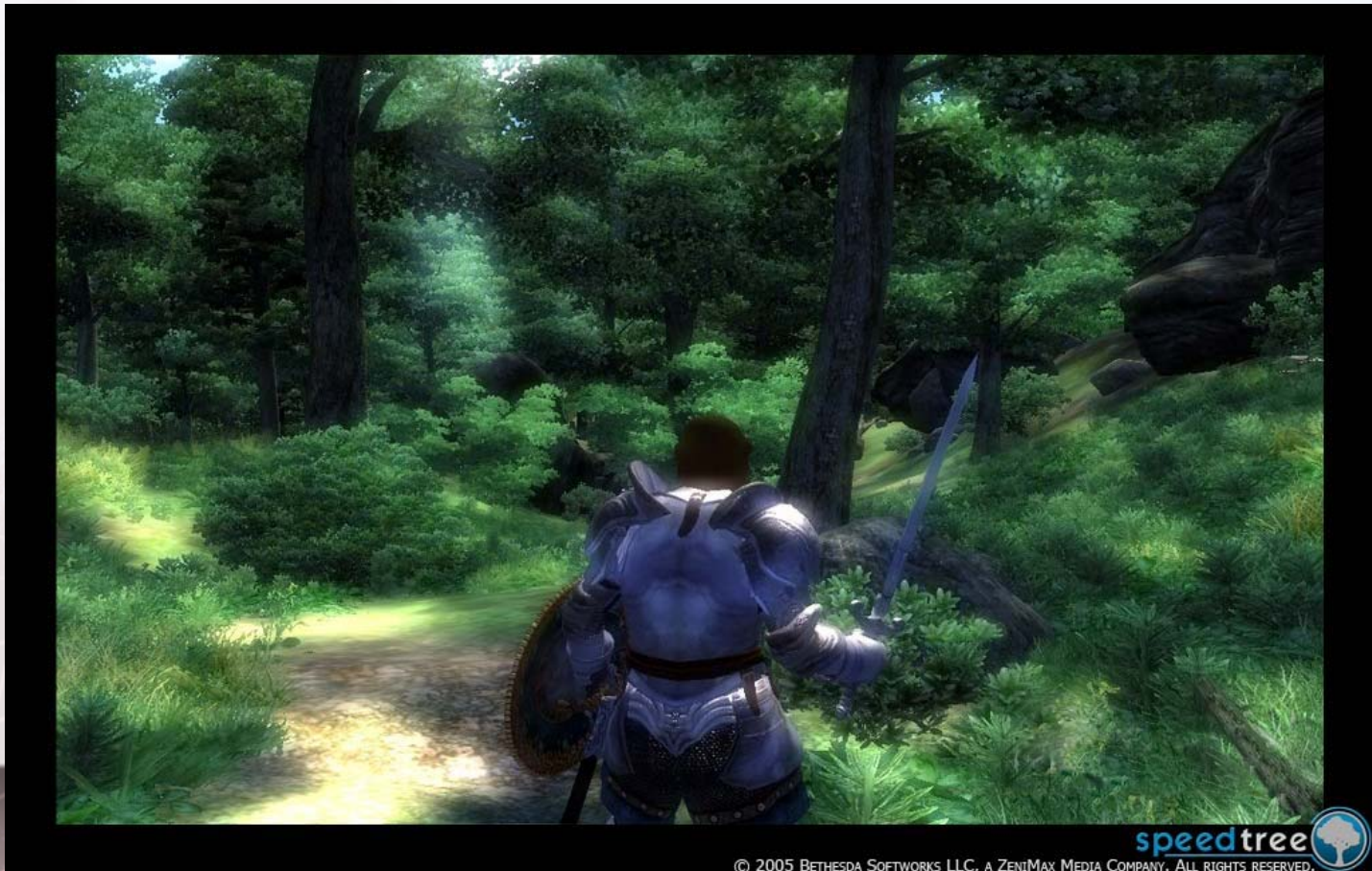
Unified we stand, divided we fall

- Upcoming games will be both geometry and pixel compute intensive
  - High geometry density with complex morph animations

speedtree

Image courtesy of Bethesda and SpeedTree

Unified we stand, divided we fall

- Upcoming games will be both geometry and pixel compute intensive
    - Significant image post-processing for HDR, depth of field, motion blur

Image courtesy of Valve Software's Day of Defeat trailer

Unified we stand, divided we fall

- Centralizing compute resources is key as more, varied tasks migrate to the GPU

Physics & other game calculations

Polygon rendering

Geometry creation

Sorting

Image post-processing

Morphing

Vertex processing

- **Games are being developed today for ATI's Unified Shading Architecture on XBox360**

  - Microsoft Xbox360 games being designed to take advantage of the unified architecture's flexibility

  - Hard to go back once you've made the switch

  - Developers can continue with the advantage of Xbox360 development, and can easily port Xbox360 games to the PC and vice versa

"The Xbox 360 is the most productive graphics development platform I have ever worked on -- clean and powerful hardware that is well documented and easy to exploit, coupled with absolutely first rate development tools."

- John Carmack, id Software

- Based on the good points of past DirectX architectures, DirectX 10 has been written from the ground up

- More ways to process, access, move data

  - Geometry processing & streaming

  - Rendering to more than just pixels

  - Enables compute beyond rendering

- Programming limits significantly raised

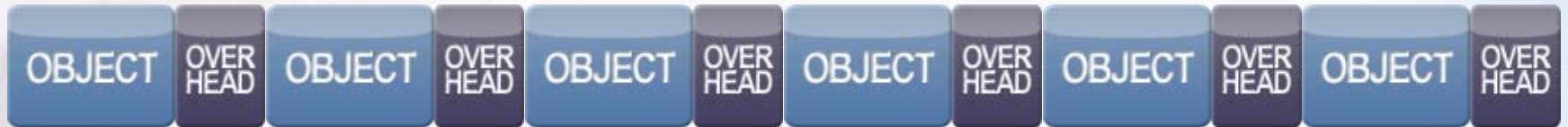 - Shader instructions, registers, constants, outputs

DirectX 10 helps remove constraints

- DirectX 10 reduces per-object state changing overhead
  - Dynamic indexing state within shader programs
    - vs heavyweight state setting outside of shader programs
  - State snapshots that can be loaded atomically
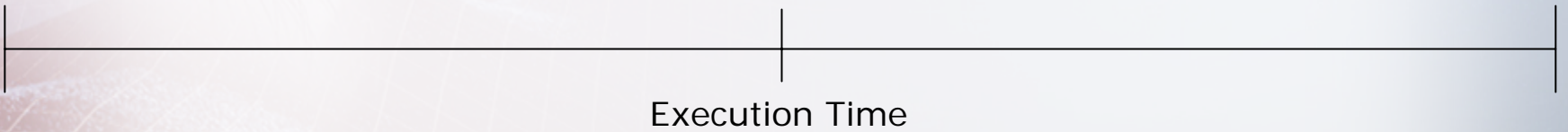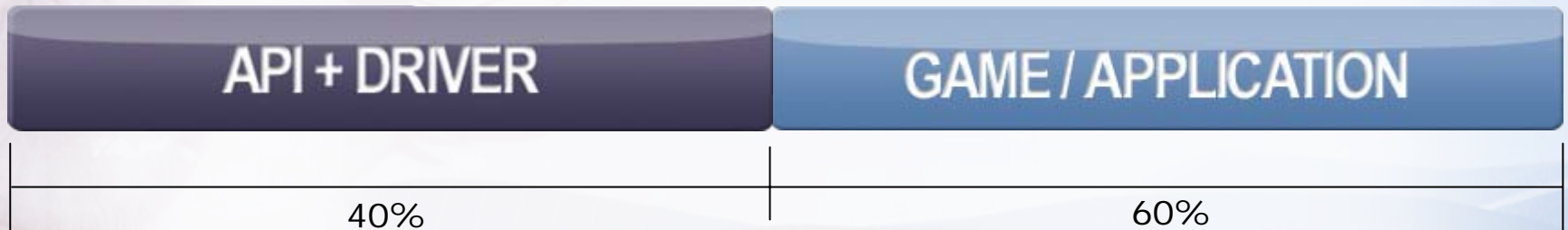   - 1 call vs many

- DirectX 9 overhead

- DirectX 10 overhead

Execution Time

What does this all mean?
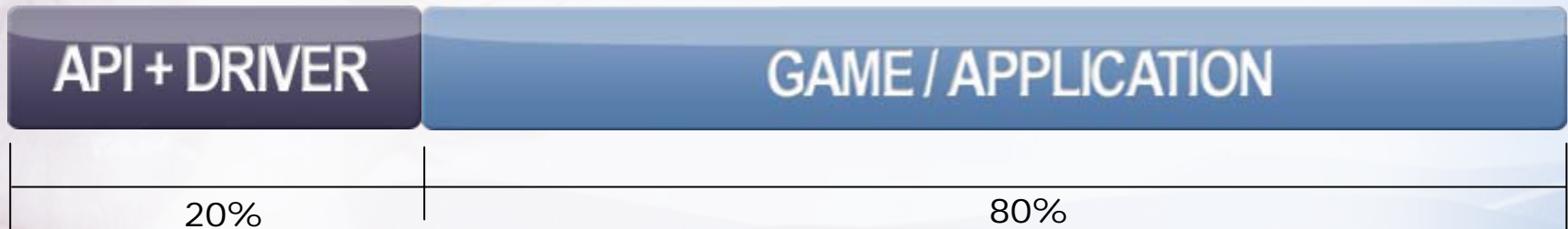
- Current execution time is balanced like this:

API + DRIVER	GAME / APPLICATION
40%	60%

What does this all mean?

- With less overhead, execution time *could* be balanced like this:

API + DRIVER	GAME / APPLICATION
20%	80%

- ## DirectX 10 will result in less overhead

  - Doesn't mean you'll see 80% of execution time dedicated to the game
  - It means that developers will be able to put **more** into their games

- DirectX 10 games on a unified shader architecture will have more...

Objects

- DirectX 10 games on a unified shader architecture will have more...

Clutter

- DirectX 10 games on a unified shader architecture will have more...

Materials

What does this all mean?

- DirectX 10 games on a unified shader architecture will have more...

Vegetation/Foliage

- DirectX 10 games on a unified shader architecture will have more...

Shadows

- DirectX 10 games on a unified shader architecture will have more...

Physics

Water, collisions, cloth, hair, particles, etc.

What does this all mean?

- DirectX 10 games on a unified shader architecture will have more...

Effects

Lighting, volumetric effects, etc.

- DirectX 10 games on a unified shader architecture will have more…

Character Animation

- DirectX 10 games on a unified shader architecture will have more...

... all scalable

- Will look…
  - … even better than today, closer to what our eyes see
  - … more photorealistic, moving toward CG

Tomorrow's games

- Will feel...
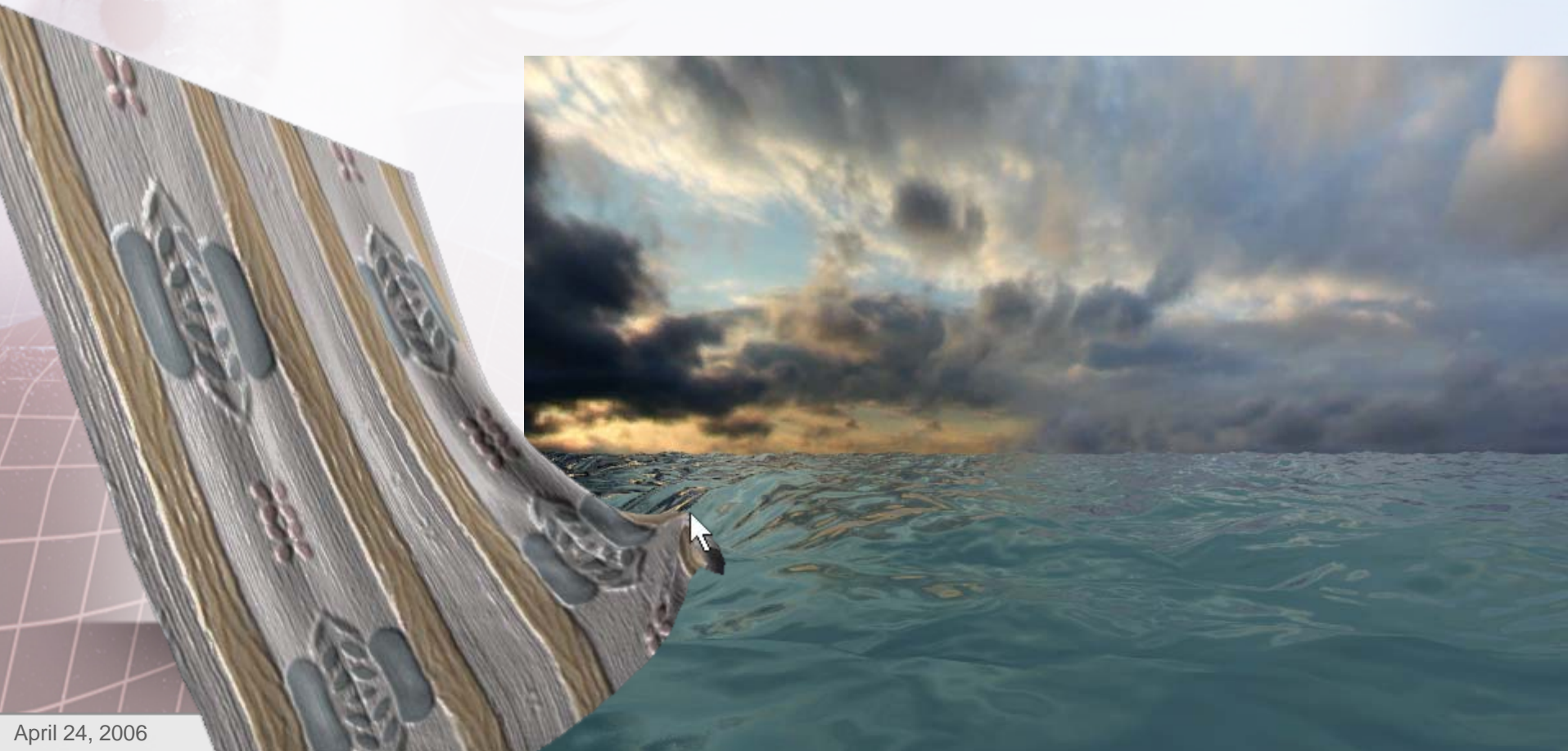   - ... richer, with detailed worlds where there's always something to see
    - ... unique and distinct

GOTHIC III © 2005 PIRANHA BYTES. ALL RIGHTS RESERVED.

Image courtesy of Piranha Bytes and SpeedTree 40

- Will play…
  - … with more opportunity for interaction
  - … more realistically as a result of physics

- These demos show off possibilities of DirectX 10 done on DirectX 9

- These aren't in-game demos, rather they're designed to showcase the programming possibilities and possible gaming concepts of tomorrow's games

Summary

- DirectX 10 combined with the unified shader architecture is the future of PC gaming
    - Moving beyond game rendering today, to game computing tomorrow
    - Removes the development constraints of today's games
    - Enables the next generation of visual effects in games
    - Is the only platform that truly maximizes compute resources
- DirectX 10 was designed with a unified shader in mind
    - Allows different kinds of shaders (vertex, geometry, pixel) to all work on the same instruction sets and access the same resources
        - Xbox360 games are already taking advantage of this model
    - No one understands the unified shader architecture better than ATI
        - ATI's unified shader for DirectX 10 represents the first unified shader ever for the PC, and the second generation of our unified shader technology
- Tomorrow's games will have enhanced:
    - Objects, clutter, materials, vegetation and foliage, shadows, effects (lighting, volumetric effects), physics (water, collision, cloth, particles, etc.), character animation