NVIDIA®

Havok FX Physics on NVIDIA GPUs

havok®

What is Effects Physics?

- Physics-based effects on a massive scale
  - 10,000s of objects
  - Rigid bodies
  - Particles
  - Fluids
  - Cloth
  - and more
- Tightly coupled with rendering
  - Physics on this scale requires simulation to be done close to the rendering

A New Class of Visual Effects

- Effects physics adds new realism to games
  - Litter and debris to add detail and realism
  - Smoke & fog that reacts when you move through it
  - Cloth and fluid that collide with objects and characters
  - Massive amounts of rubble from collapsing buildings
  - Flocking and swarming enemies

Physics-based visual effects interact with the game world, its characters, and the game player

Havok and NVIDIA

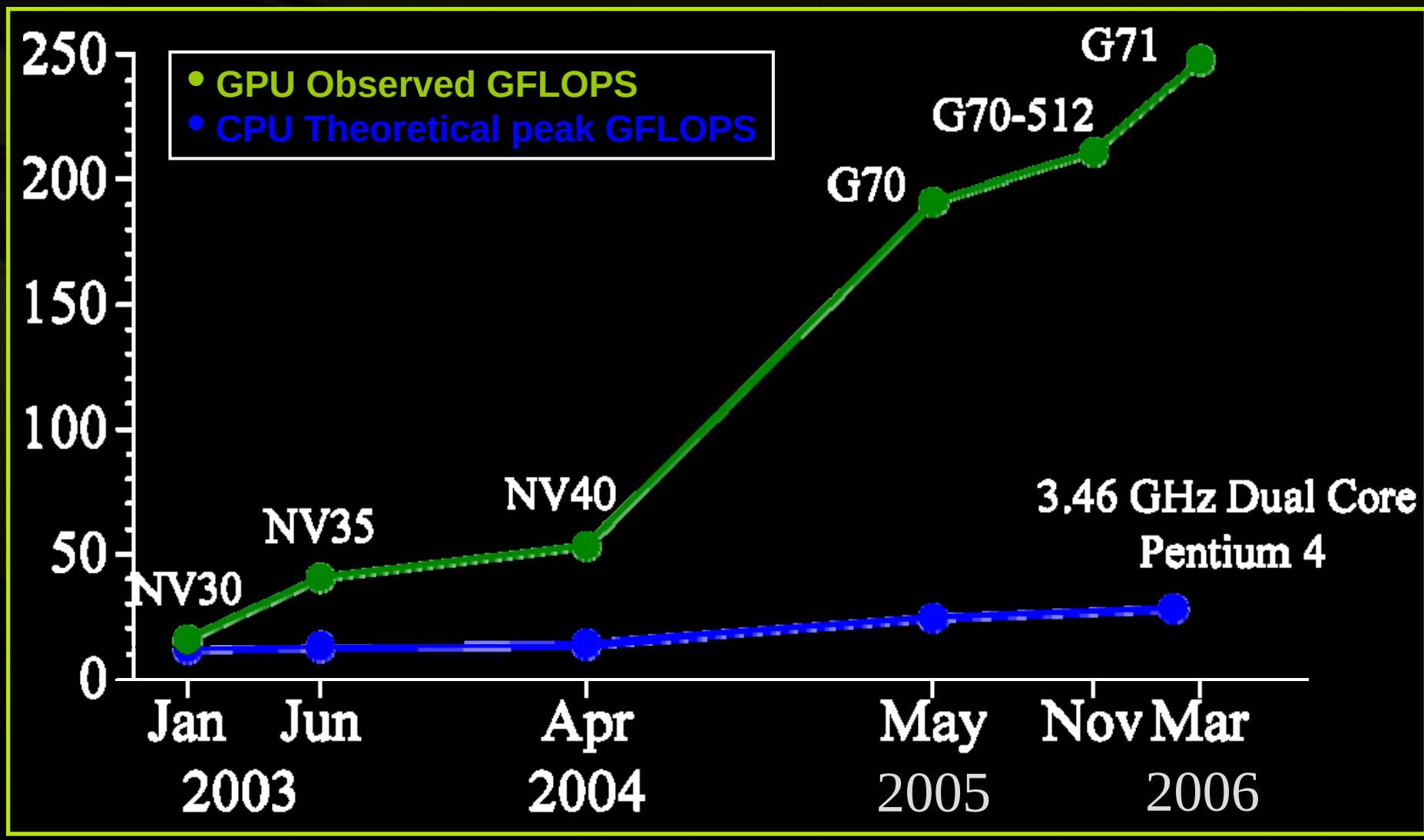
- Joint R&D project launched in 2005 to investigate physics on GPUs
- Havok FX is the world's first GPU-accelerated game physics solution
- Optimized for NVIDIA platforms

Why Physics on GPUs?

- GPU: very high data parallelism
  - G71: 24 pixel pipelines, 48 shading processors
  - 1000s of simultaneous threads
  - Very high memory bandwidth
  - SLI enables 1-4 GPUs per system
- Physics: very high data parallelism
  - 1000s of colliding objects
  - 1000s of collisions to resolve every frame
  - Requires 1000s of floating point operations per collision

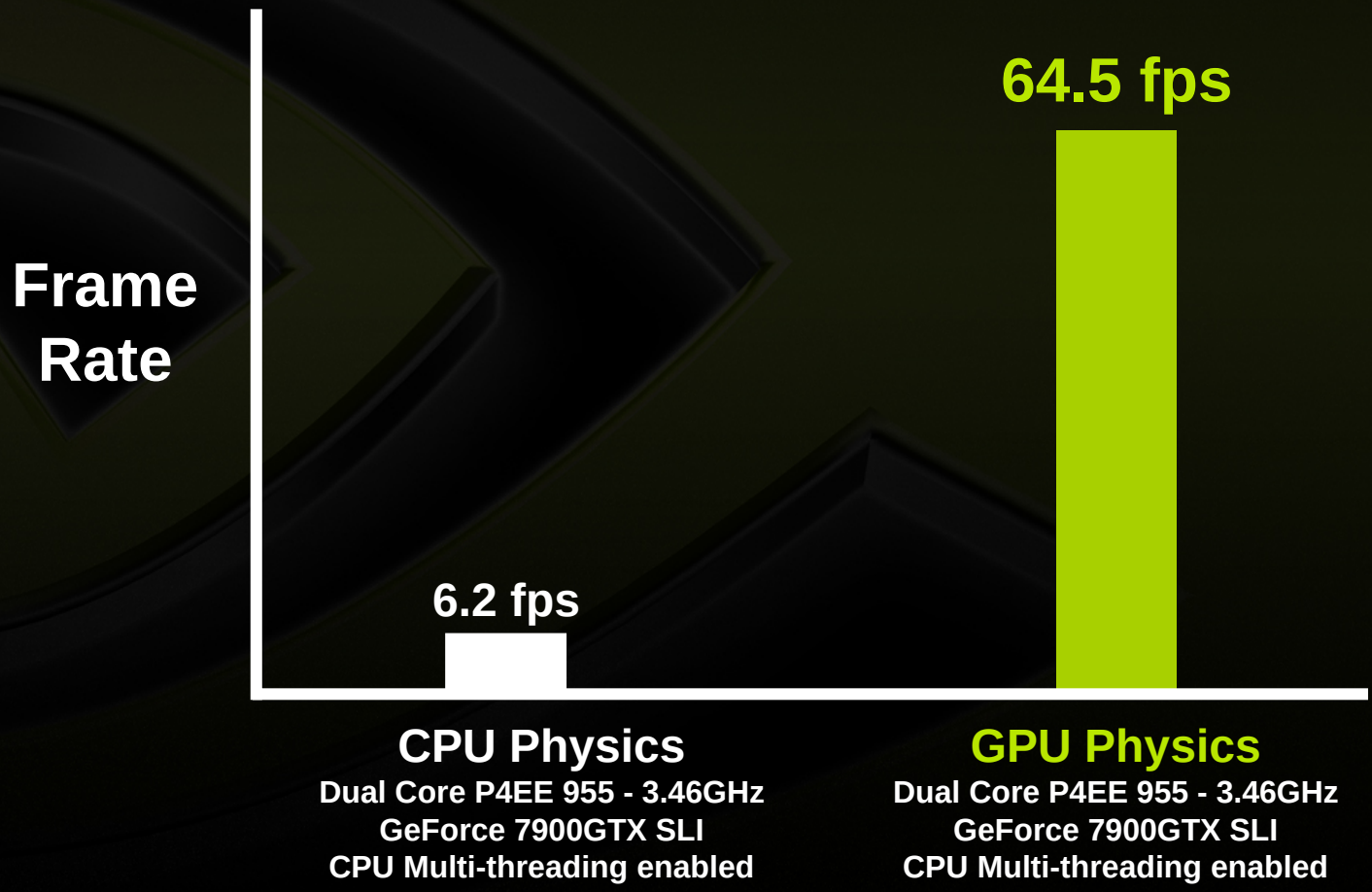
Physics is an ideal match for GPUs

NVIDIA GPU Pixel Shader GFLOPS



Dedicated Performance For Physics

Performance Measurement
15,000 Boulder Scene



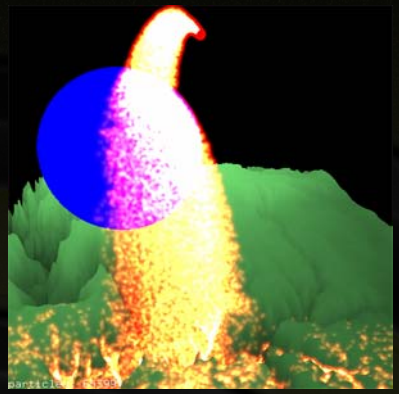
General-Purpose Computation on GPUs

- Highly parallel applications
  - Physically-based simulation
  - image processing
  - scientific computing
  - computer vision
  - computational finance
  - medical imaging
  - bioinformatics

www.gpgpu.org

- Many examples of physical simulation on GPUs

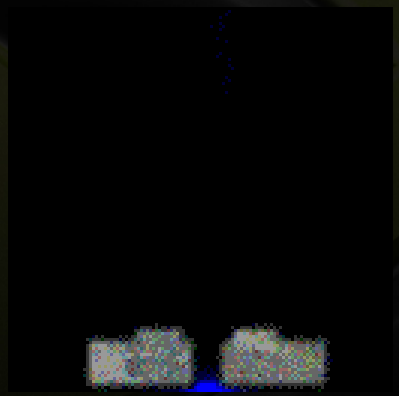
Physically-based Simulation on GPUs



Particle Systems

Jens Krüger, TU-Munich

Fluid Simulation



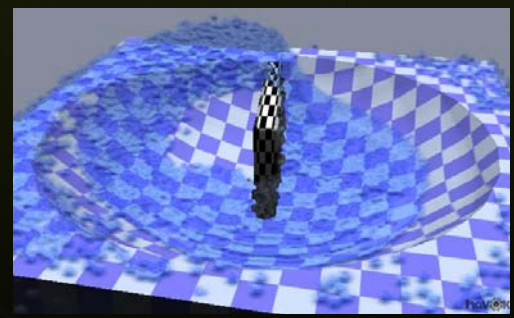
Cloth Simulation

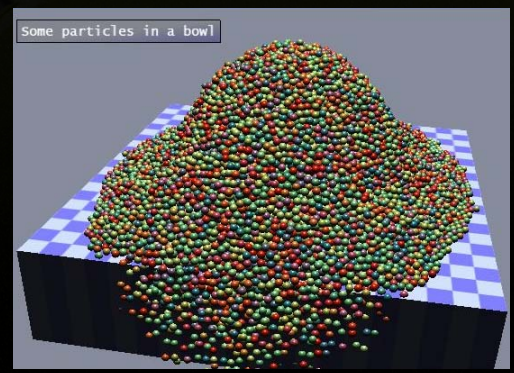
Soft-body Simulation

Doug L. James, CMU

Havok FX Features Overview

- Rigid Bodies
  - Convex collision bodies
  - Stable stacking
- Particles
  - Collisions
  - Fluid, Cloth etc.
- Lightweight Framework
- Fully integrated with Havok 4
  - Everything collides with everything else
- Integrated Toolchain
  - Max, Maya, XSI

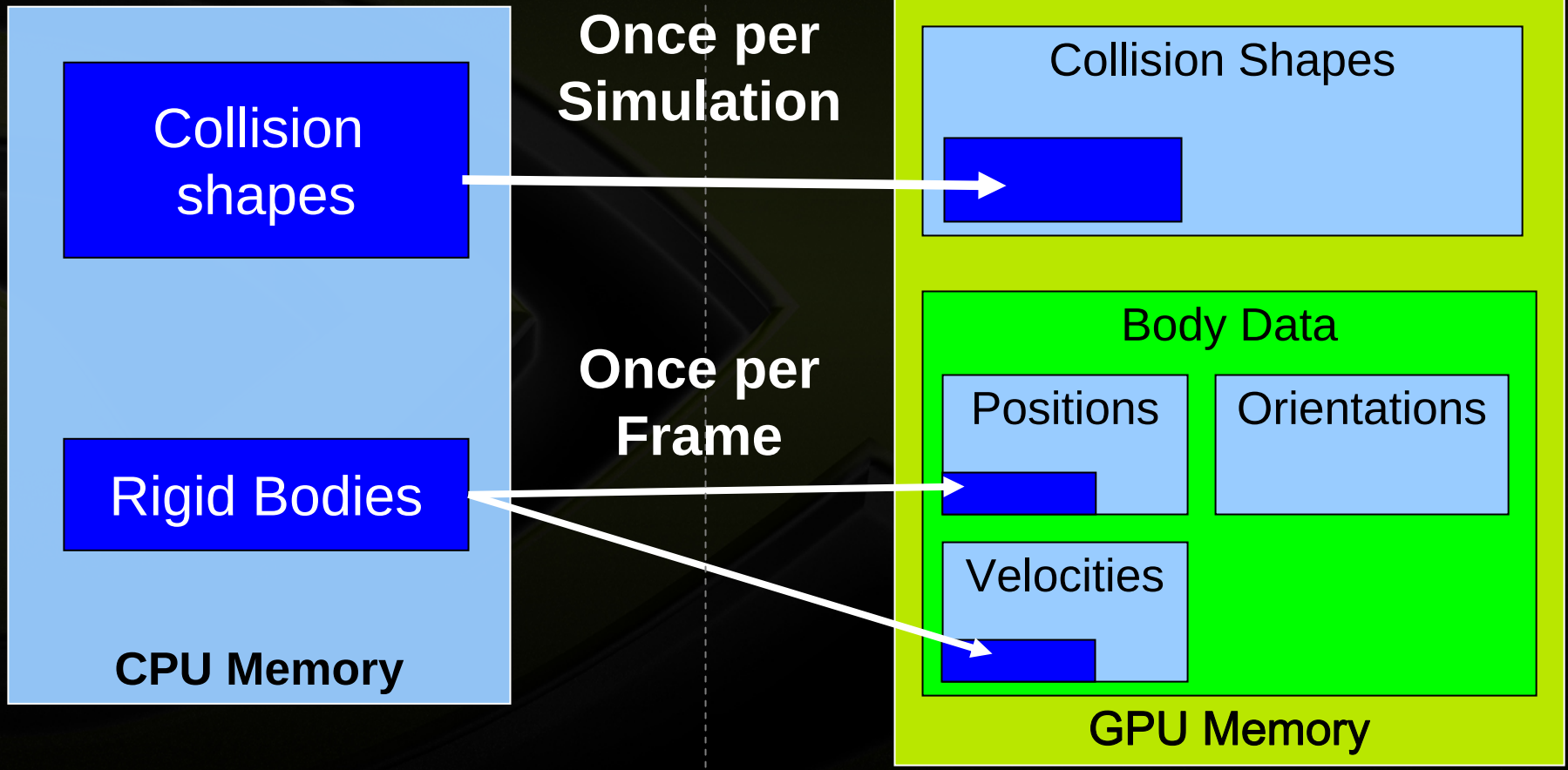




Custom Behaviors

- User-defined behaviors run *on the GPU* to modify object state
  - Custom Cg shaders implement a simple interface
- Very simple and flexible architecture
  - Read access to all data
  - Output position, orientation, linear and angular velocity
- Examples
  - Boundaries – reset/deactivate objects that exit the scene
  - Vortices
  - Attractors
  - Swarm effects

Gameplay physics interaction



Gameplay physics

FX physics

Havok FX in Hellgate: London

NVIDIA Technology for Physics

- Shader Model 3 GPUs

- SLI multi-GPU technology

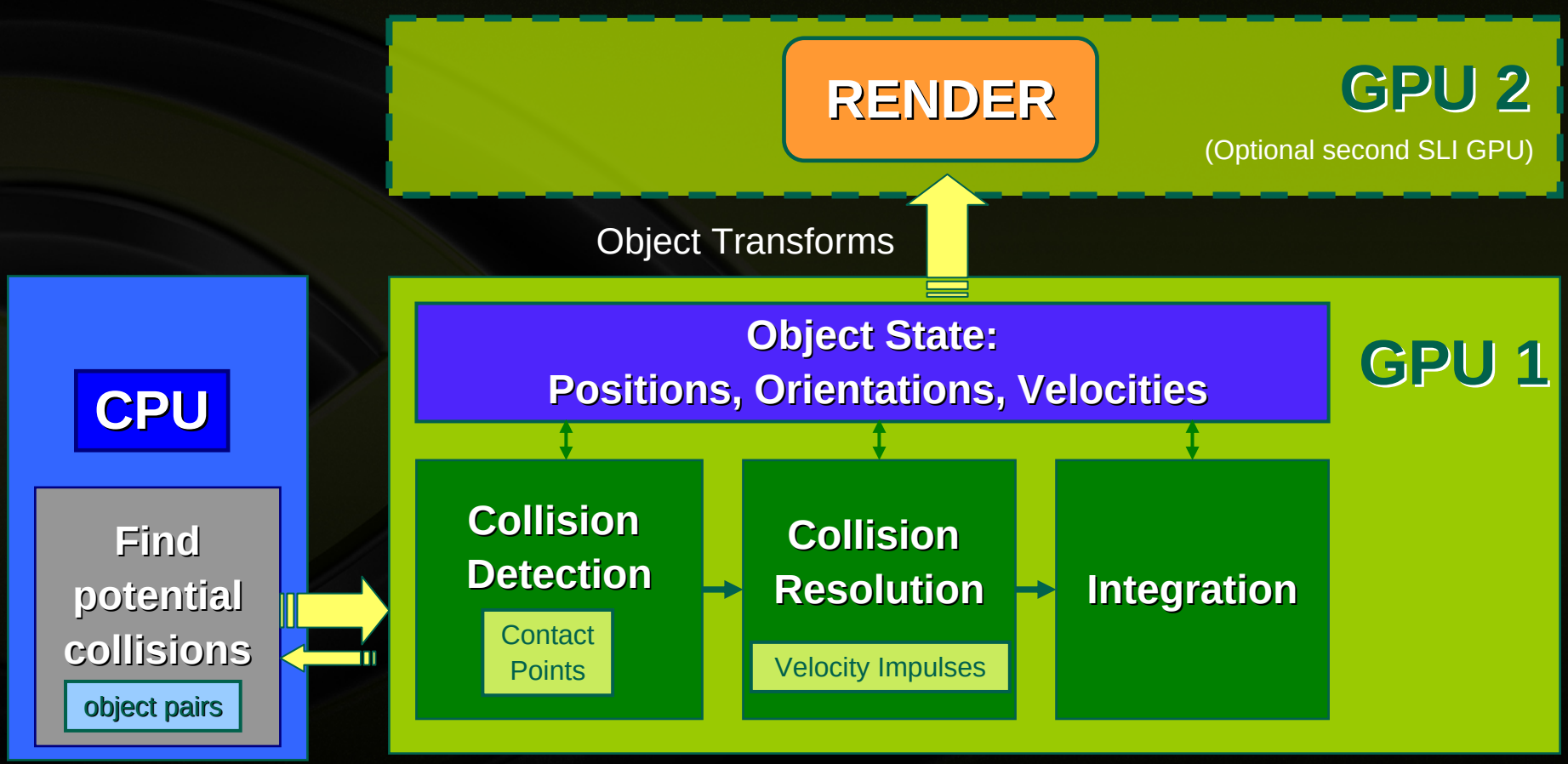
- Cg Compiler

- New driver technology for physics

NVIDIA Shader Model 3 GPUs

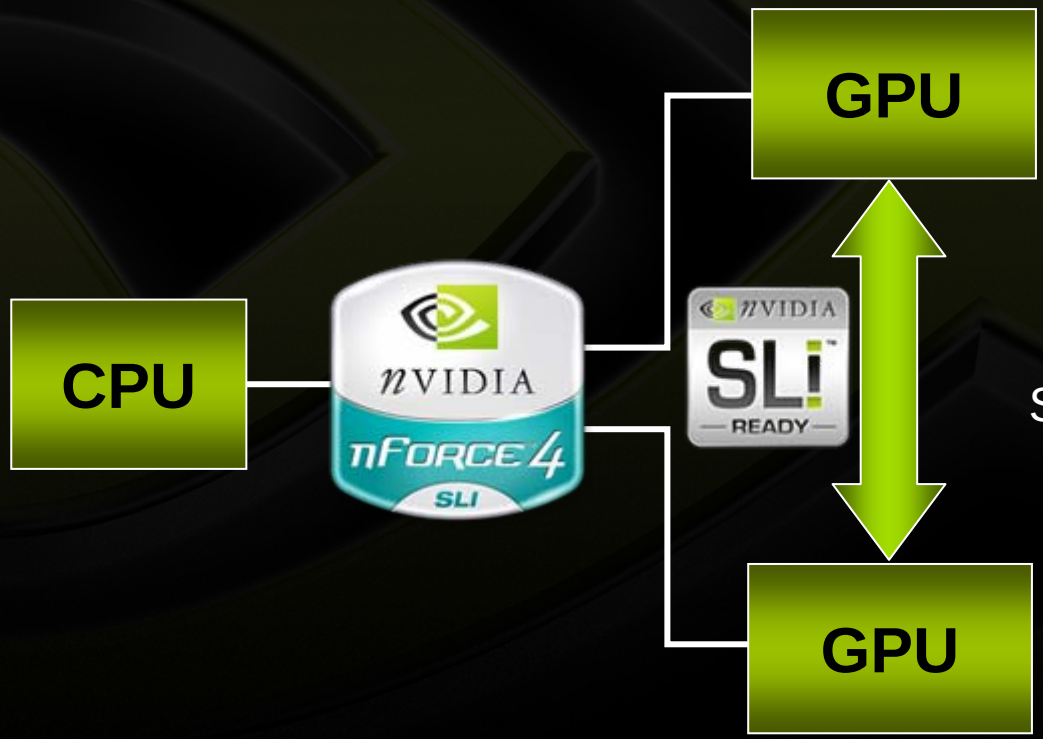
- GeForce 6 and 7 Series GPUs
- Shader Model 3 is essential for physics
  - Branching and looping in the pixel shader
- Support for long shaders
  - Physics shaders are much longer than average
  - 1000s of cycles per collision
  - Shader Model 2 instruction limits insufficient

Havok FX Physics Block Diagram



SLI™ Game Physics for Effects

- Second GPU can be used for SLI graphics or physics simulation

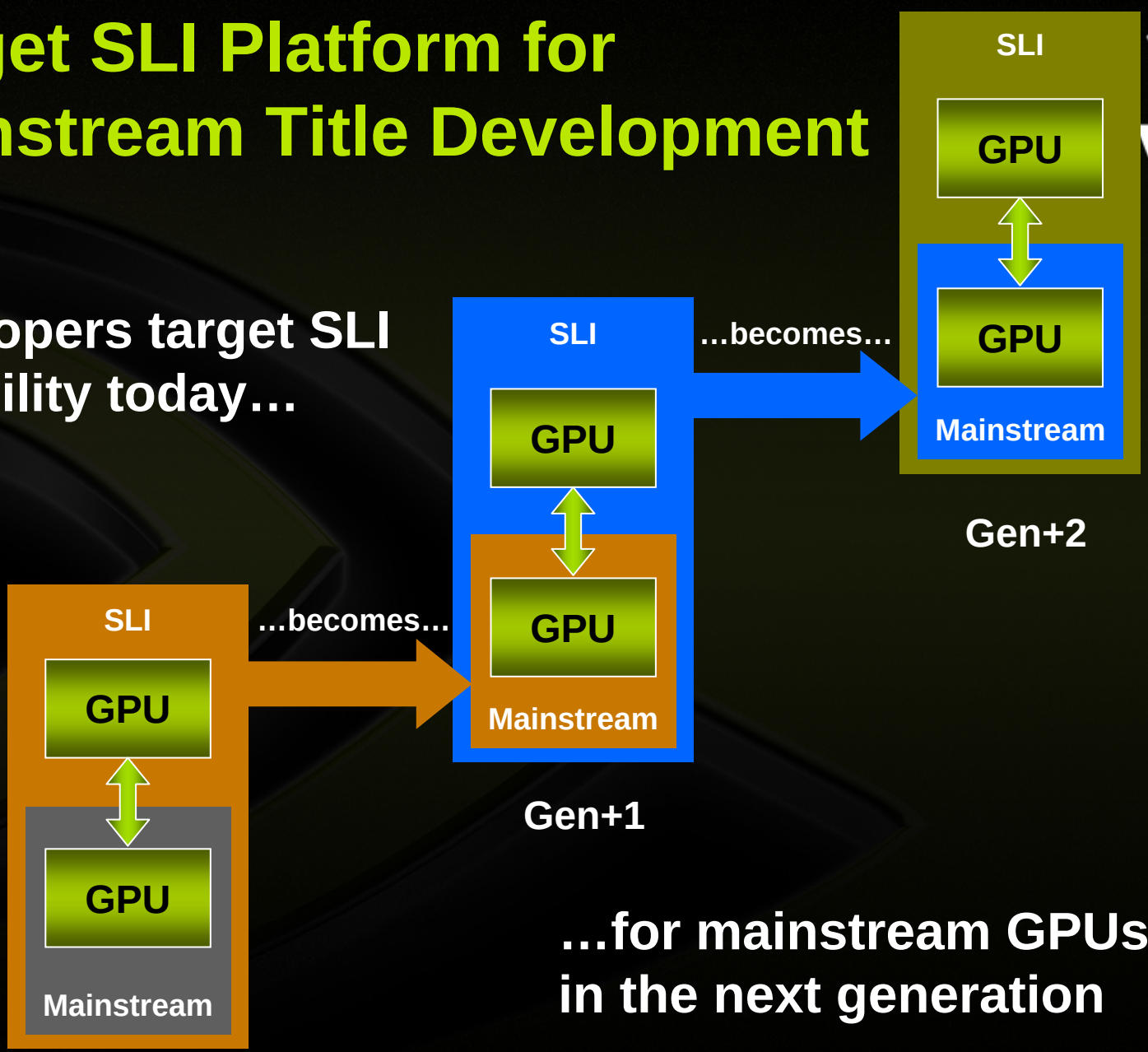


Graphics on GPU 1

SLI Graphics or Physics on GPU 2

Target SLI Platform for Mainstream Title Development

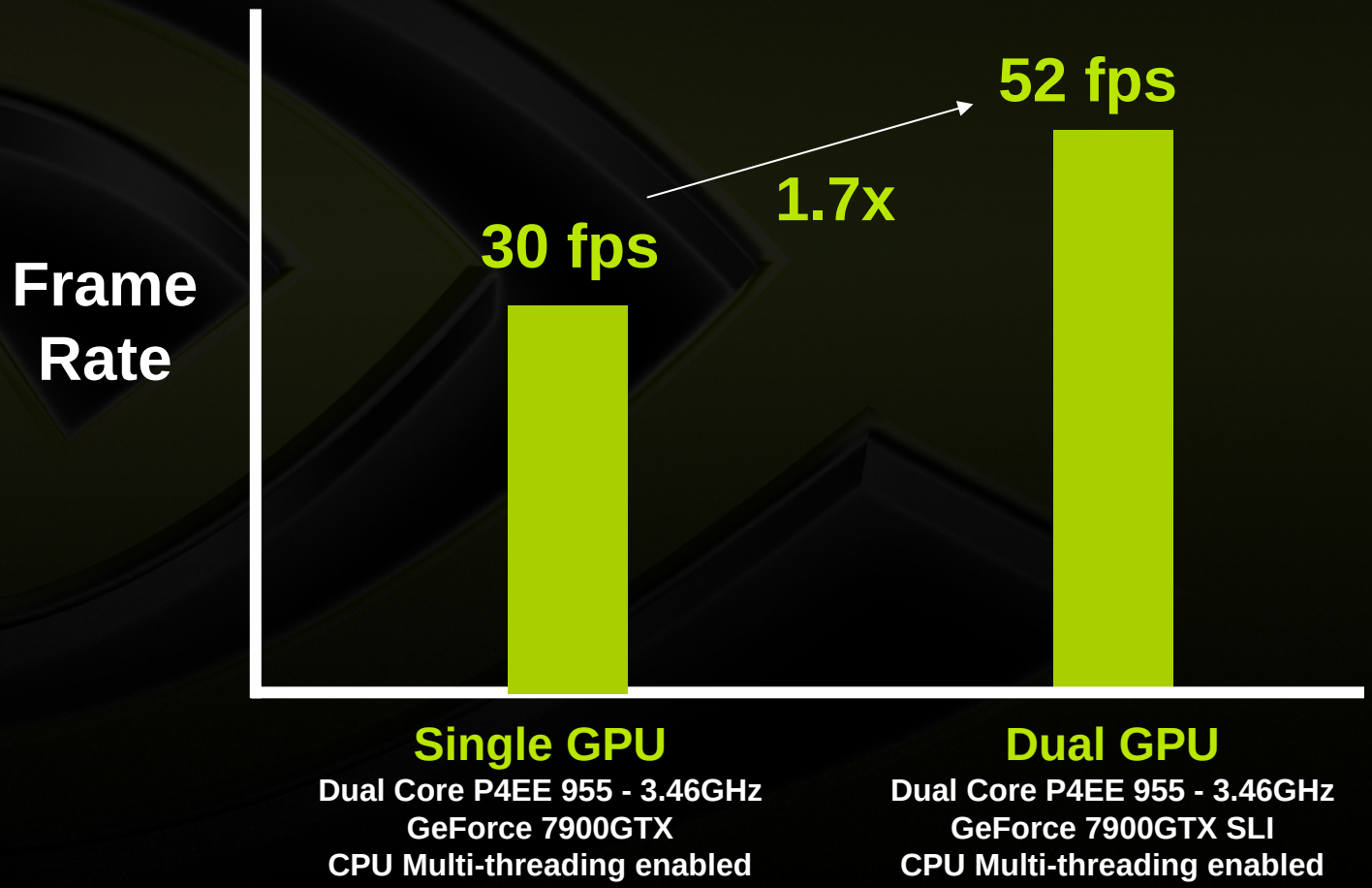
Developers target SLI capability today…



…for mainstream GPUs in the next generation

SLI Performance Scaling

Performance Measurement
15,000 Boulders with Shadows



- Familiar to developers
  - Can compile to OpenGL or Direct3D
- Supports offline compilation
  - Important for loading time
  - Debugging, performance analysis
- Cg Interfaces
  - Used extensively in implementation (C++ style code)
  - FX Behavior shaders

Rendering

- Rendering is fully controlled by application
- Havok FX returns vertex buffers with position, velocity and optional user data
  - Supports OpenGL and Direct3D
  - Fast, automatic SLI transfers
- Rigid bodies rendered using instancing
- Particles rendered as point sprites
  - Motion blur by stretching quad between previous and current position
  - Can modify particle color or size over time
  - Texture atlases for particle animation

The Future of GPU Physics

- Distributing physics across multiple GPUs
- Brittle fracture
- Advanced fluids
  - Smoothed particle hydrodynamics
  - Level set methods
- Advanced particle rendering
  - Volumetric shadowing
- Isosurface extraction for fluids
  - Using DirectX 10 geometry shader
- More objects, faster

Questions?