NVIDIA®

GPU Physics

Mark Harris
NVIDIA Developer Technology

Game Physics

- Enhance game experience through simulation
- Simulate objects and interactions between them
  - Rigid bodies, particles, “rag dolls”, cloth, fluids, etc.
  - Collisions, constraints, fluid forces, etc.
- State of the art in Game Physics:
  - Max ~1-2K colliding objects on current CPUs
  - Or equivalent number of other constraints

Goal: scalable game physics

- Physics-based effects on a massive scale
  - 10,000s of objects
  - Rigid bodies
  - Particles
  - Fluids
  - Cloth
  - and more
- Physics effects should scale with capability of platform
- Mostly visual effects
  - But can interact with “game play” physics too

Havok and NVIDIA

- Havok is a world leader in physics middleware
- NVIDIA launched R&D project with Havok in 2005 to investigate physics on GPUs
- Optimized for NVIDIA platforms

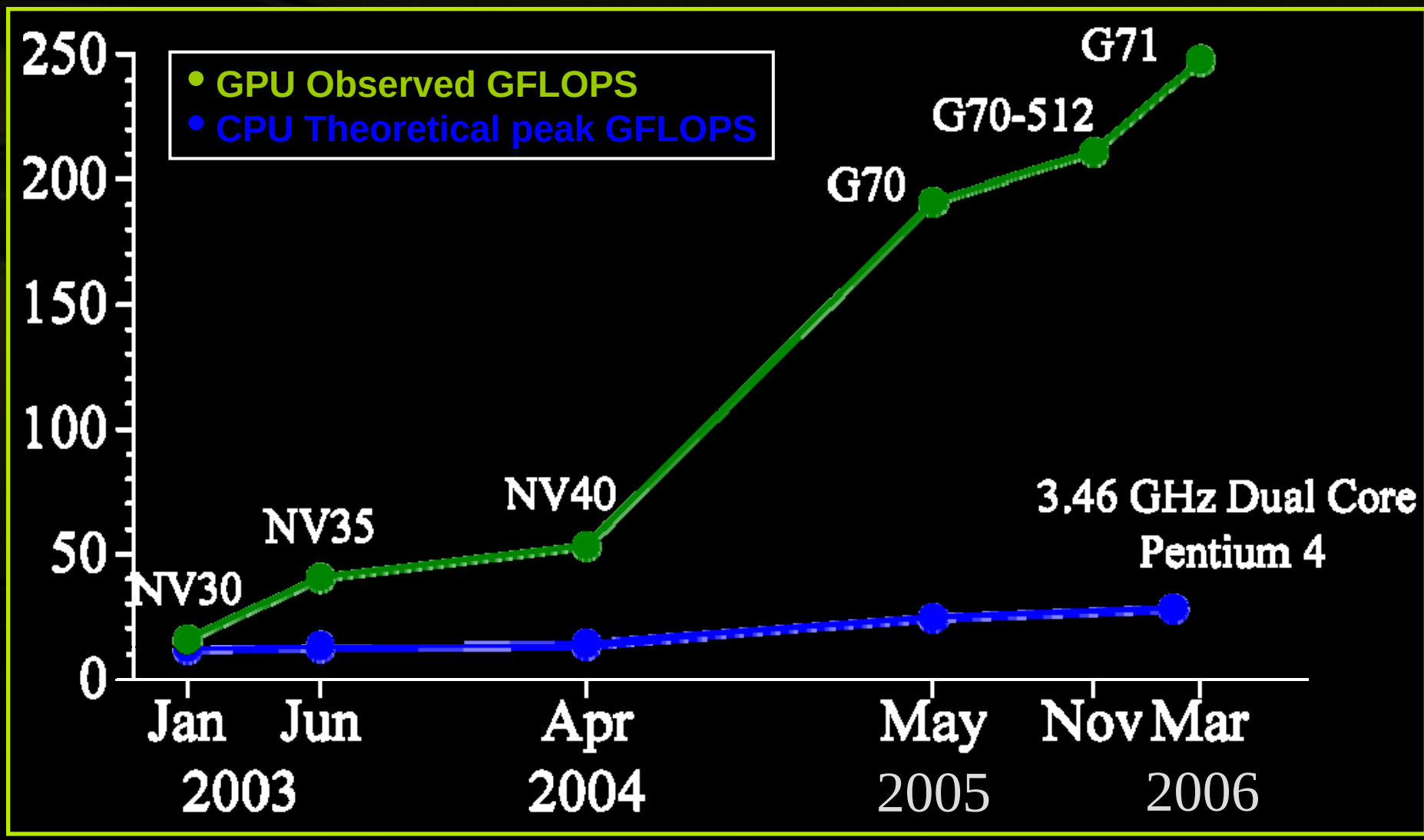
Why Physics on GPUs?

- Pixel quality has improved tremendously over the last few years
  - Material shaders, lighting and shadowing
- Still much room for improvement in physics
  - Small number of objects, limited interaction
- Many games today are CPU limited
  - Makes sense to perform simulation close to rendering

Why Physics on GPUs?

- GPU: very high data parallelism
  - G70: 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

NVIDIA GPU Pixel Shader GFLOPS

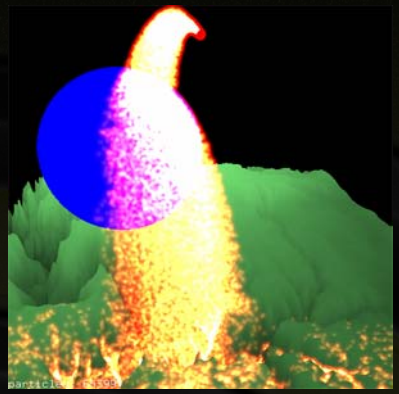


General-Purpose Computation on GPUs

- GPUs have been used to accelerate many highly parallel applications
  - Physically-based simulation
  - image processing
  - scientific computing
  - computer vision
  - computational finance
  - medical imaging
  - bioinformatics

www.gpgpu.org

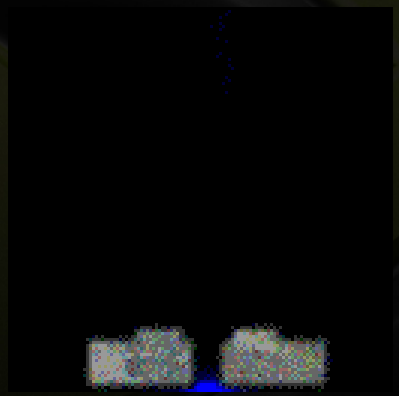
Physically-based Simulation on GPUs



Particle Systems

Jens Krüger, TU-Munich

Fluid Simulation



Cloth Simulation

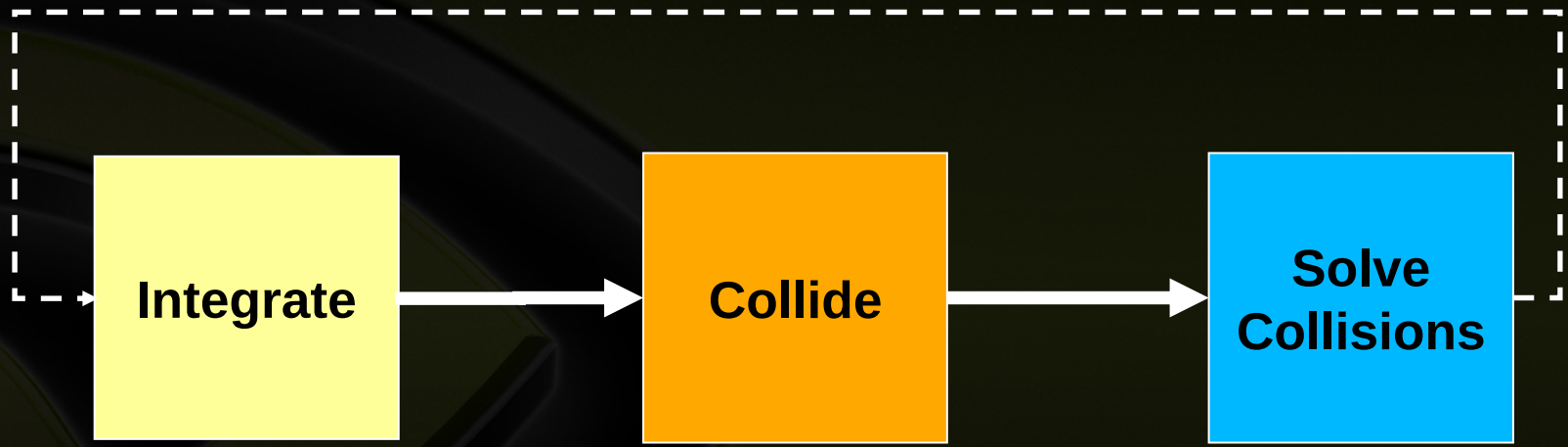
Soft-body Simulation

Doug L. James, CMU

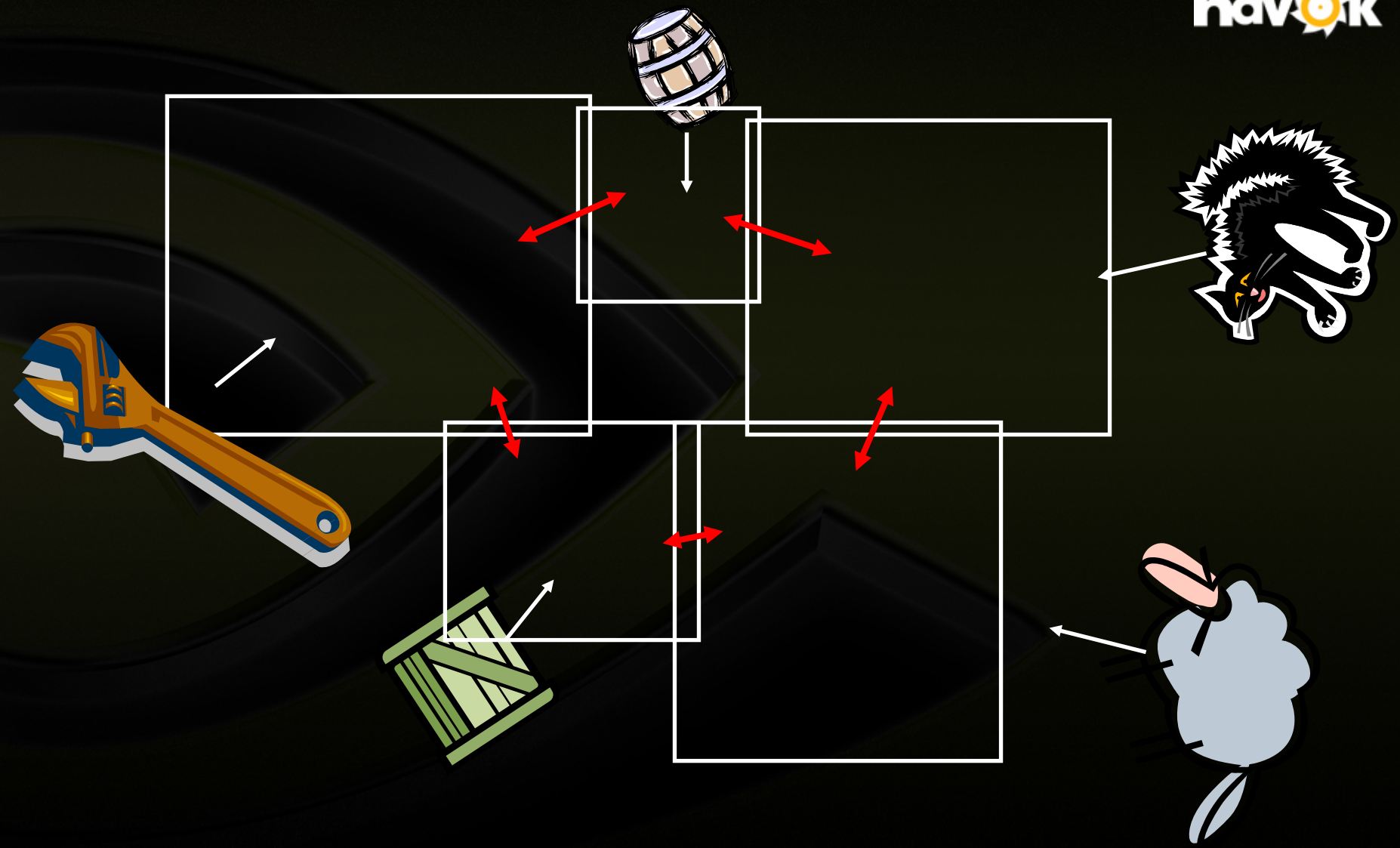
What About Rigid Body Physics?

- Fluids, particles, cloth map naturally to GPUs
  - Highly parallel, independent data
- Rigid body physics is more complicated
  - Arbitrary shapes
  - Arbitrary interactions and dependencies
  - Parallelism is harder to extract

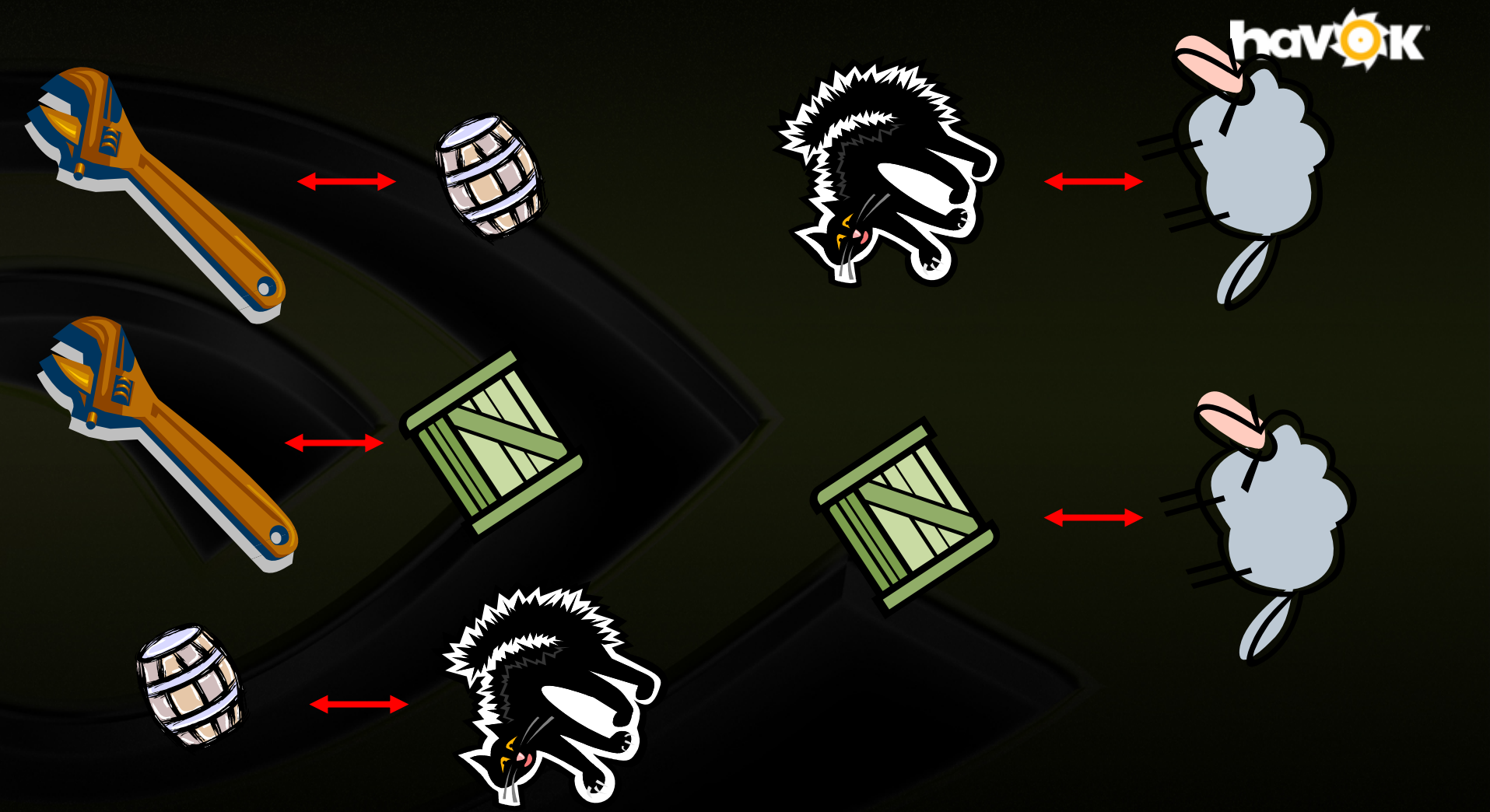
Ballistic Physics Refresher Course



Broad Phase Collision Detection
Integrate

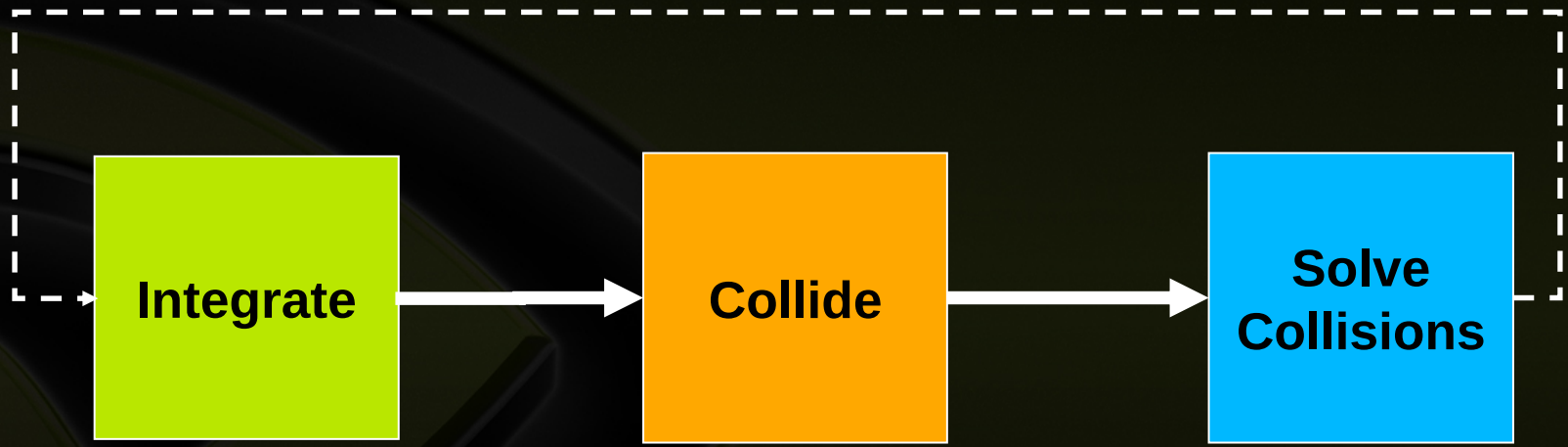


Narrow Phase Collision Detection



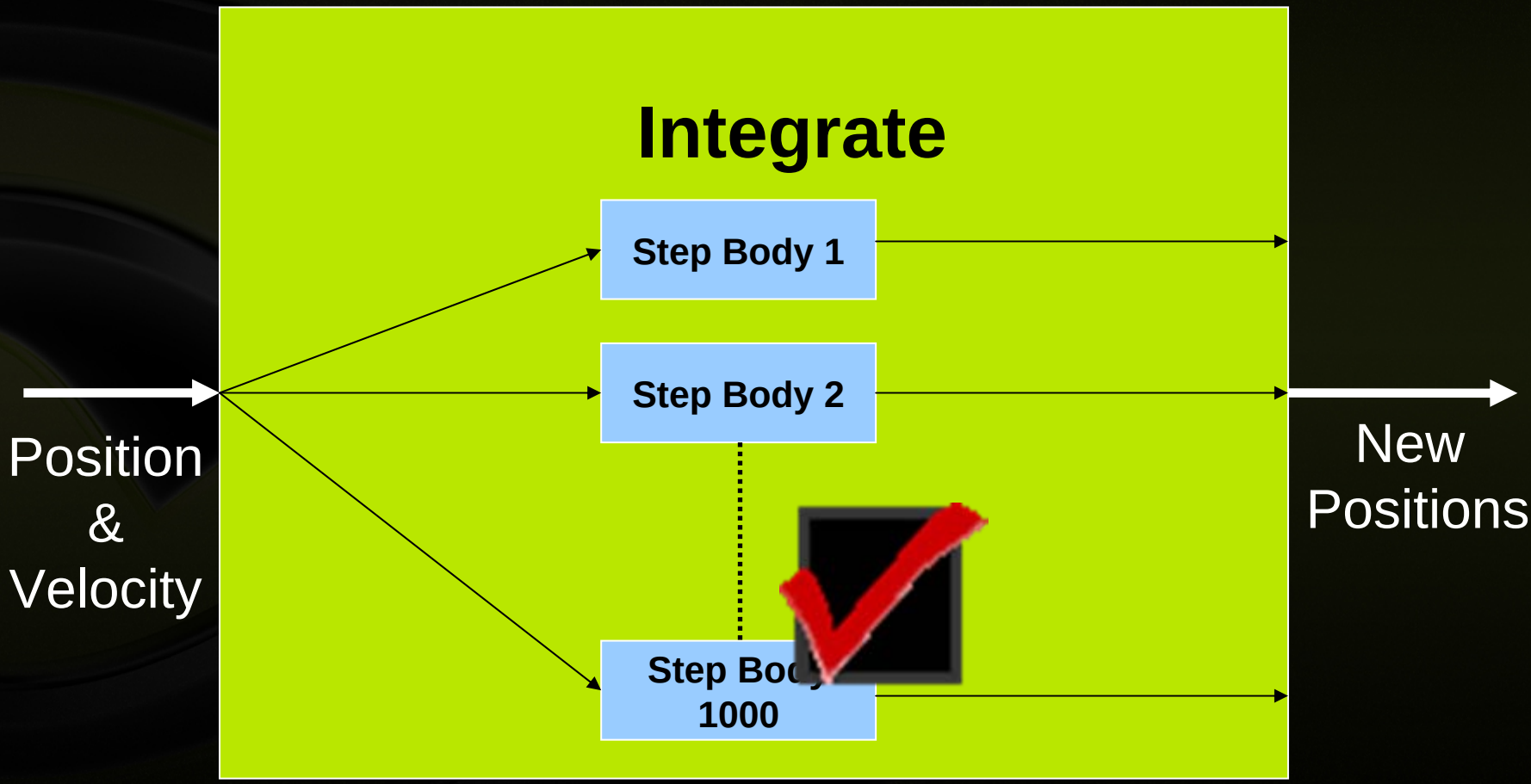
Solve collisions

Is Physics A Data Parallel Task?

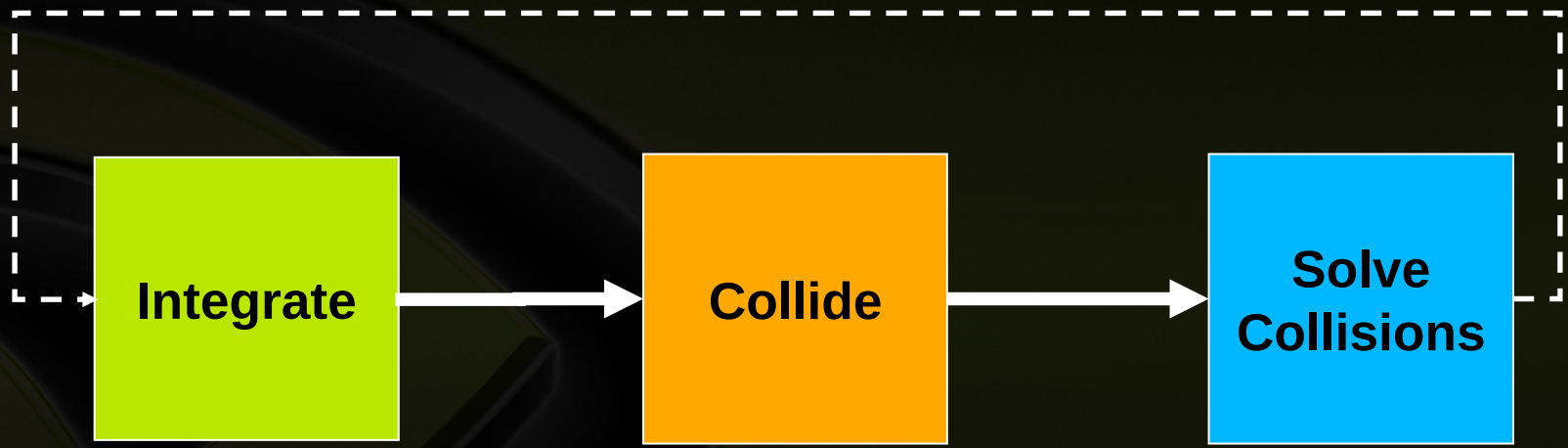


Anatomy of a clock tick

Is Physics A Data Parallel Task?

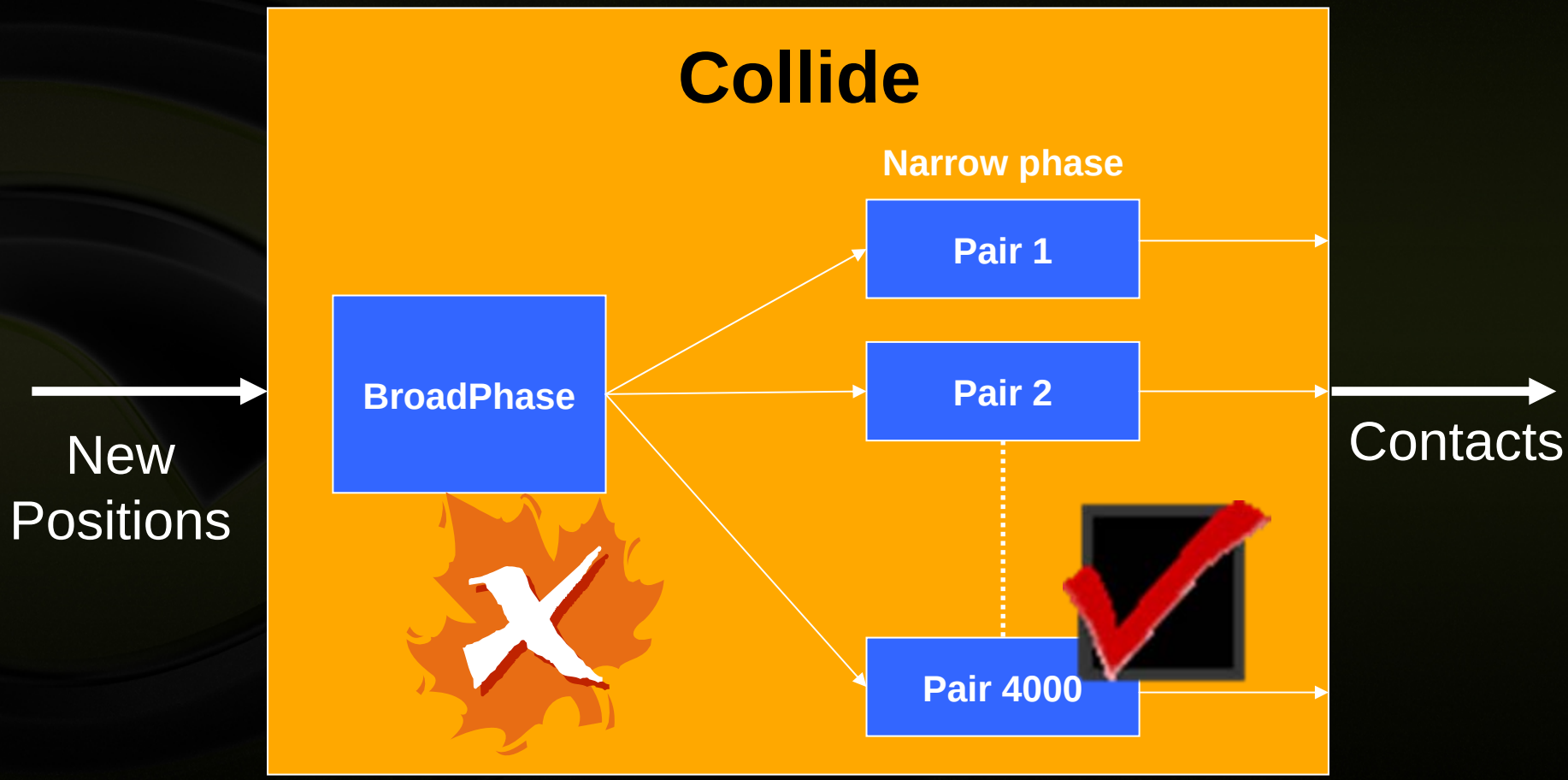


Is Physics A Data Parallel Task?

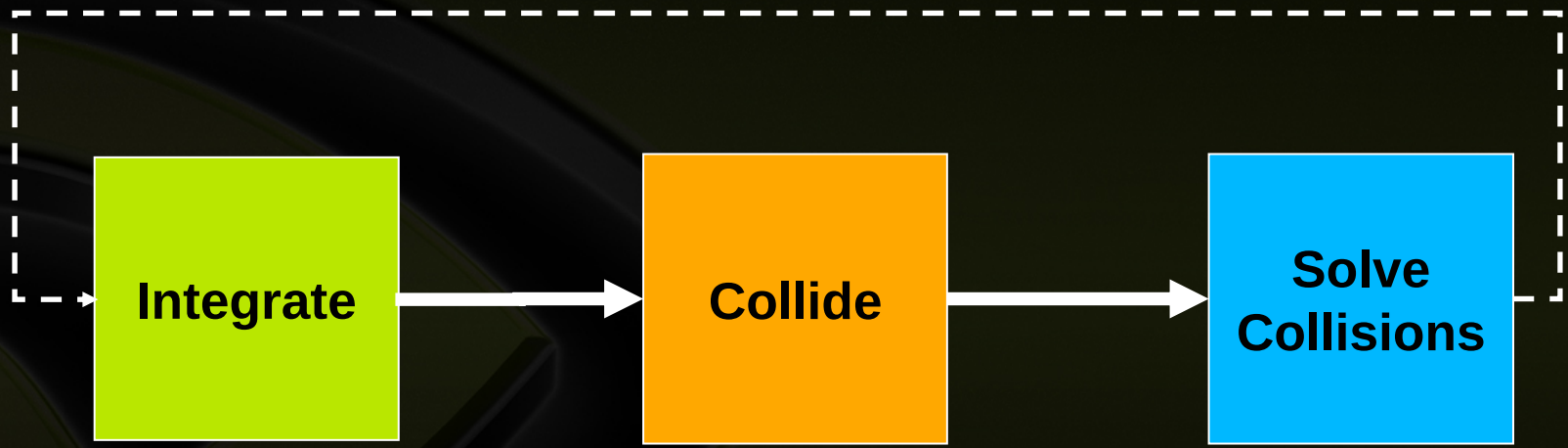


Anatomy of a clock tick

Is Physics A Data Parallel Task?

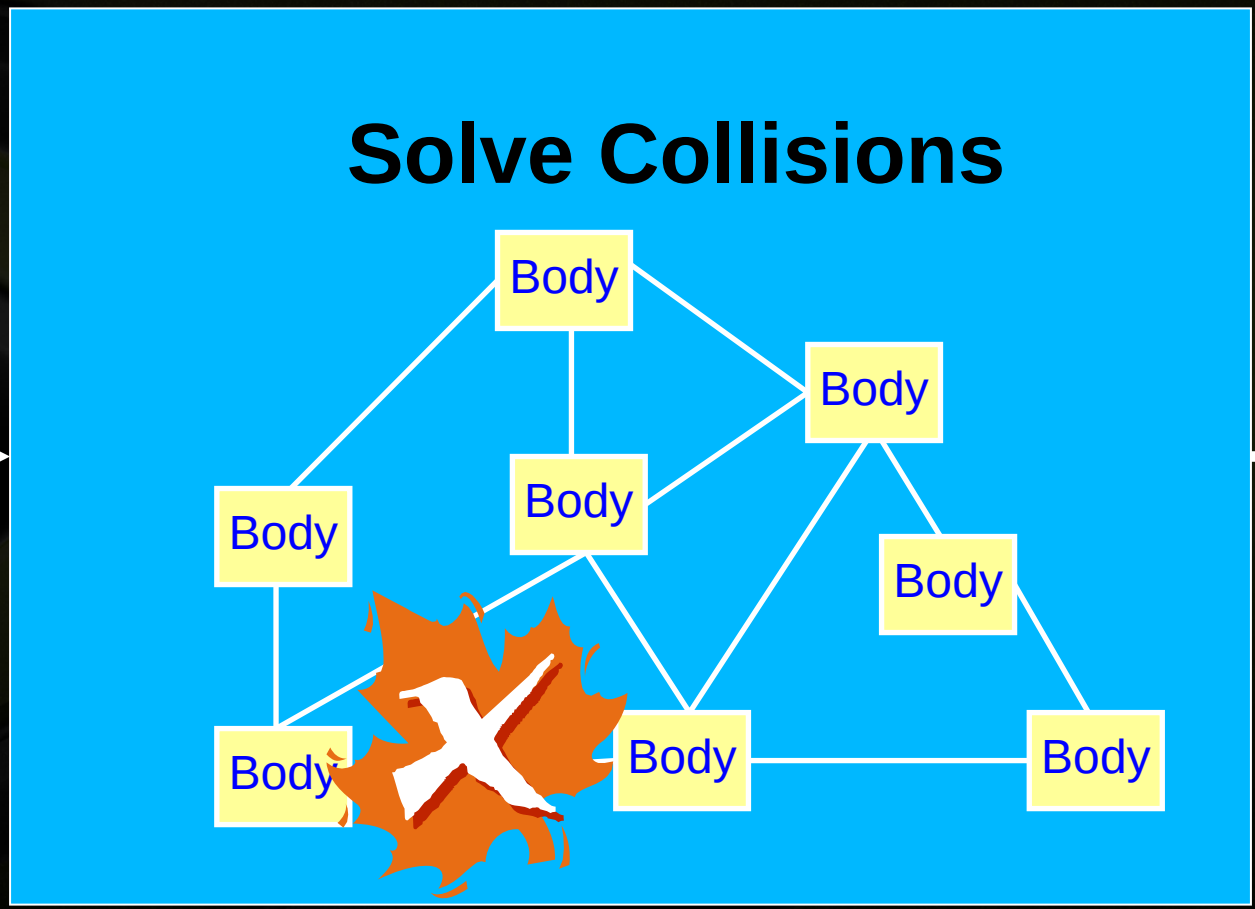


Is Physics A Data Parallel Task?



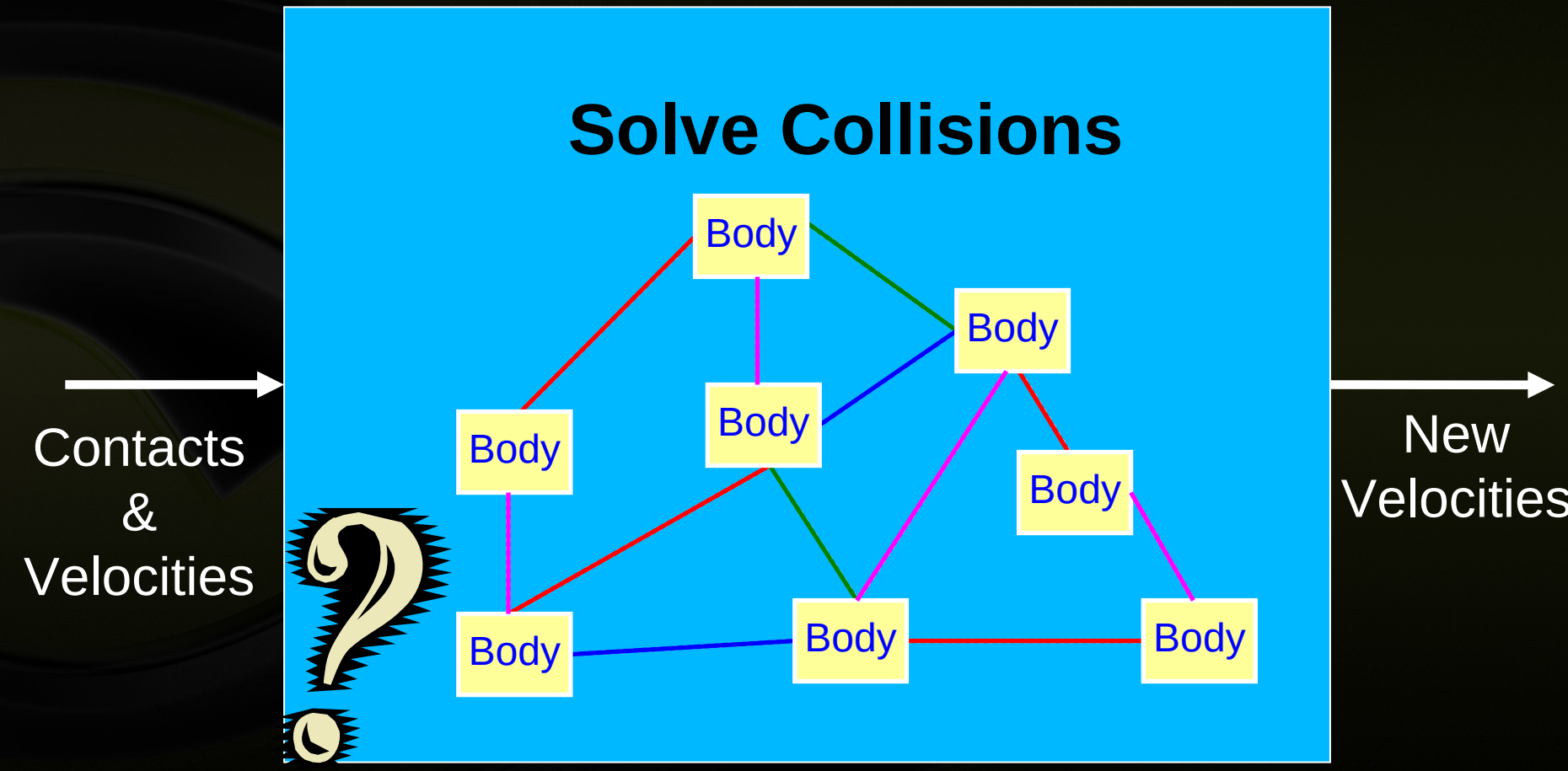
Anatomy of a clock tick

Is Game Physics A Data Parallel Task?



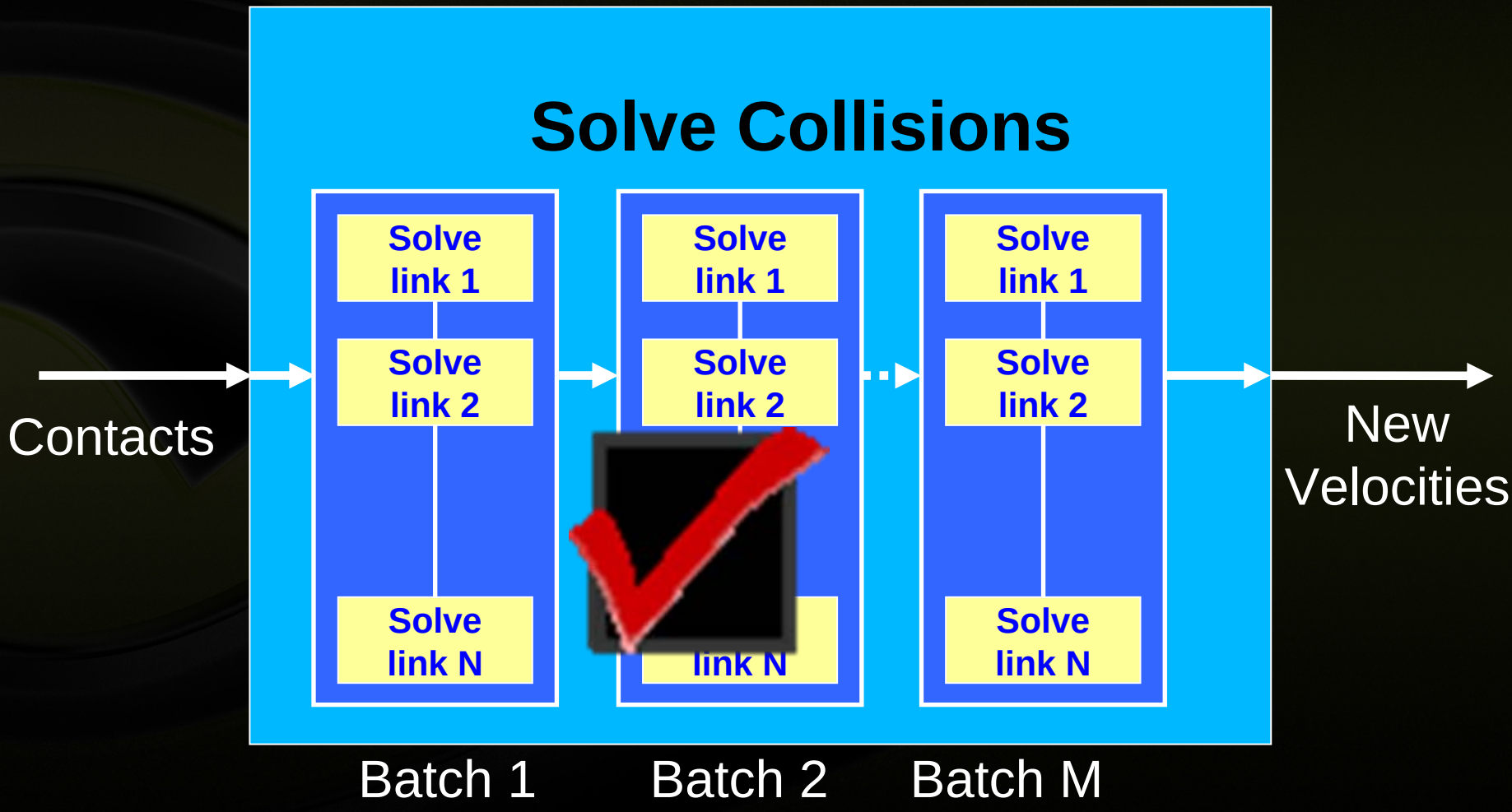
Slide courtesy of Andrew Bond, Havok

Is Game Physics A Data Parallel Task?



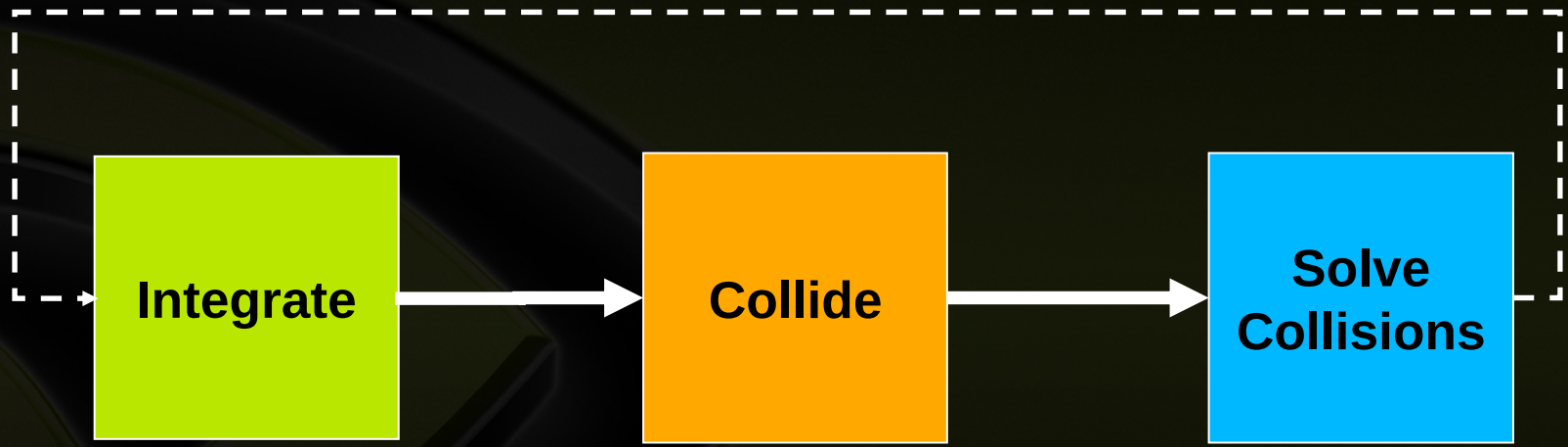
Slide courtesy of Andrew Bond, Havok

Is Game Physics A Data Parallel Task?



Slide courtesy of Andrew Bond, Havok

Physics Is A Data Parallel Task



100% data parallel

70% data parallel

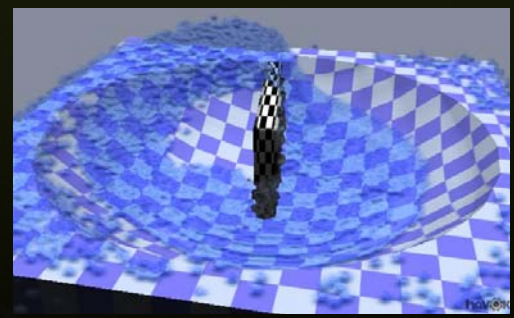
99% data parallel

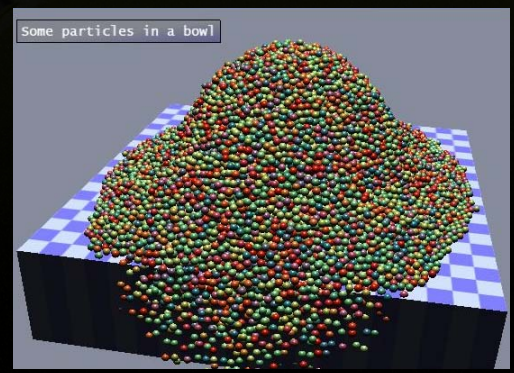
Havok FX

- Havok FX is World's first GPU-accelerated game physics SDK
  - Part of Havok 4 SDK
- Already being adopted by game developers
- Massive performance improvement over CPU implementation

Havok FX Features Overview

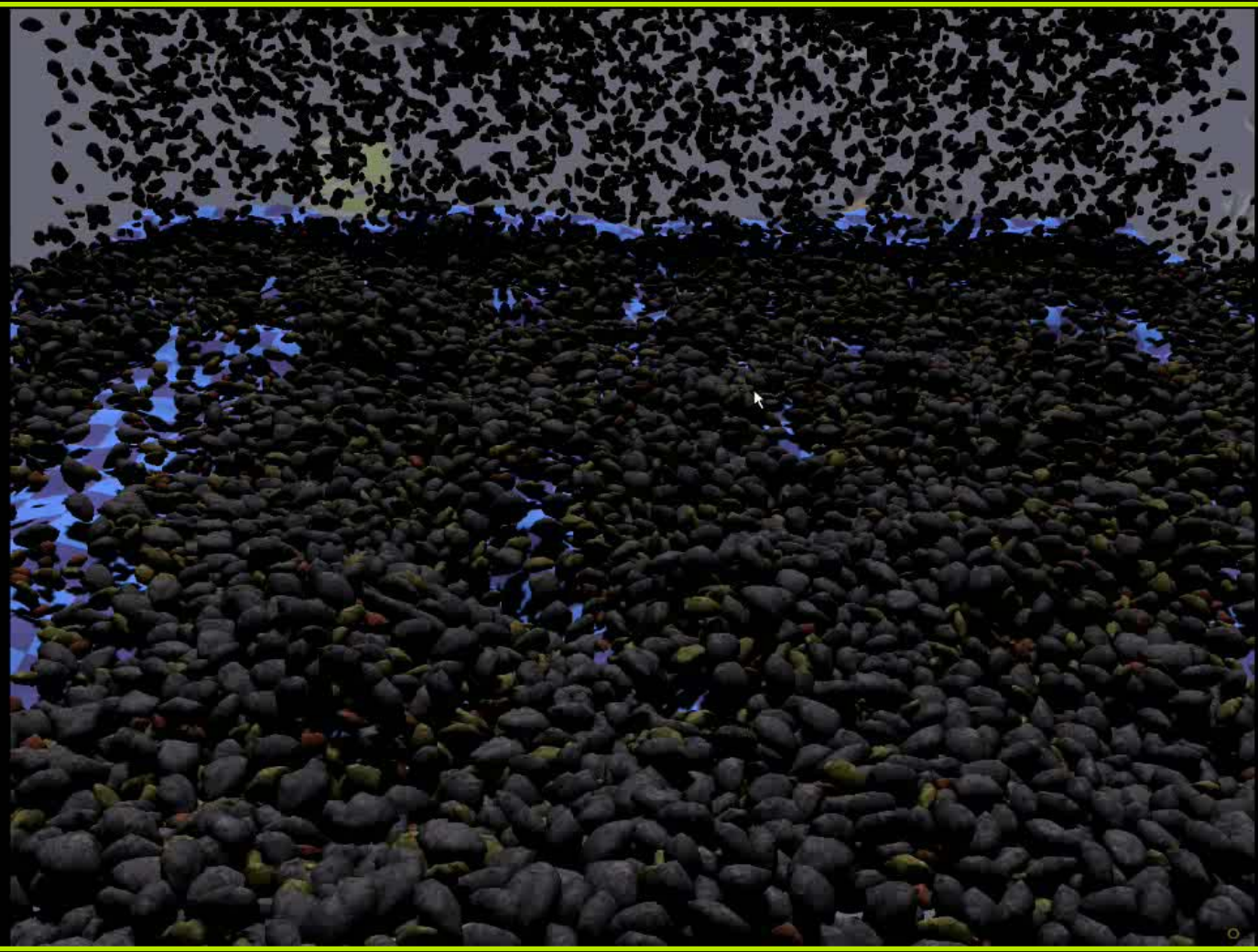
- Rigid Bodies
  - Convex collision bodies
  - Stable stacking
- Particles
  - Collisions
  - Fluid, Cloth etc.
- Lightweight Framework
- Fully integrated with Havok 4
  - Everything collides with everything else
- Integrated Toolchain
  - Max, Maya, XSI



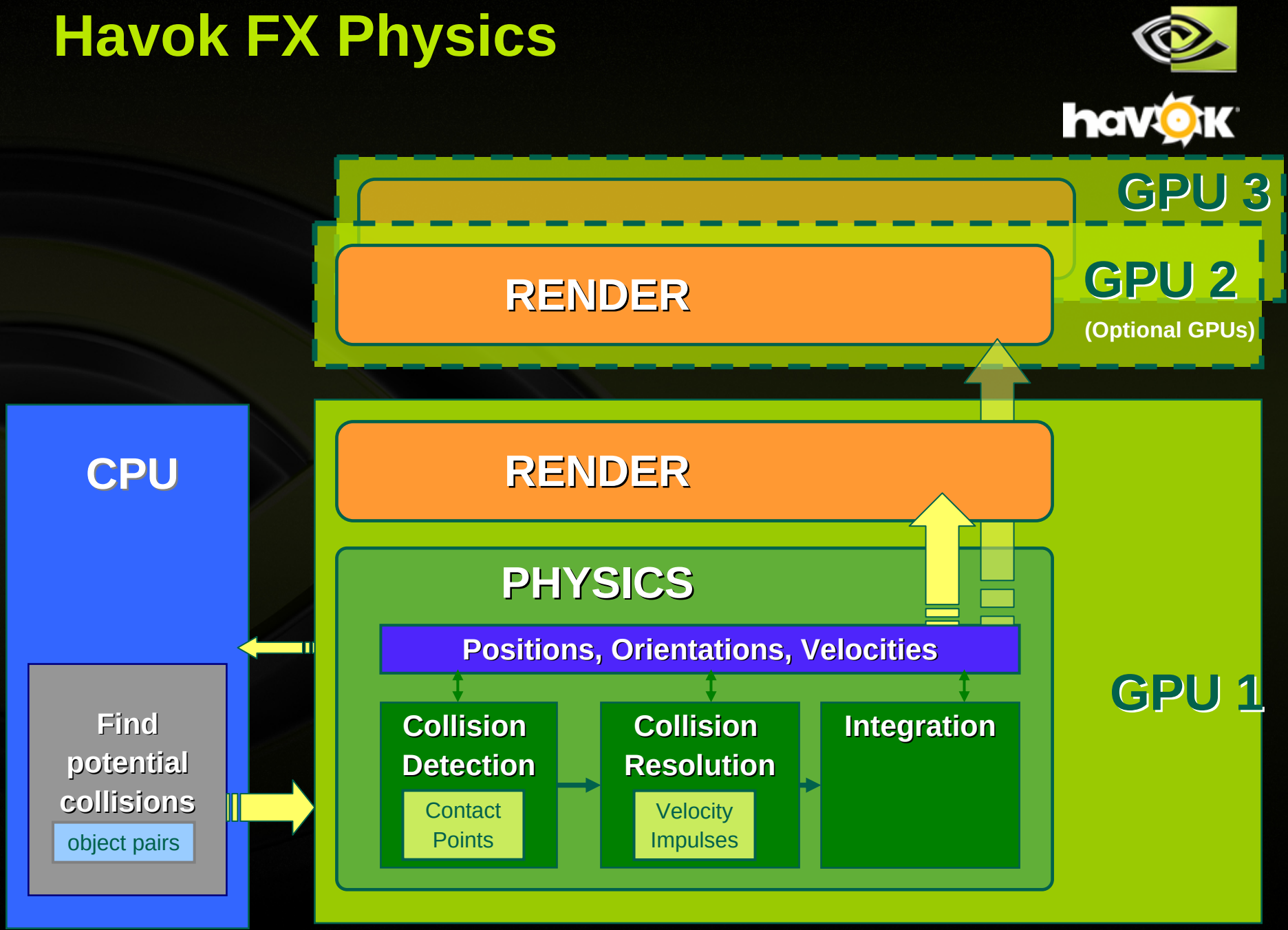


Dedicated Performance For Physics

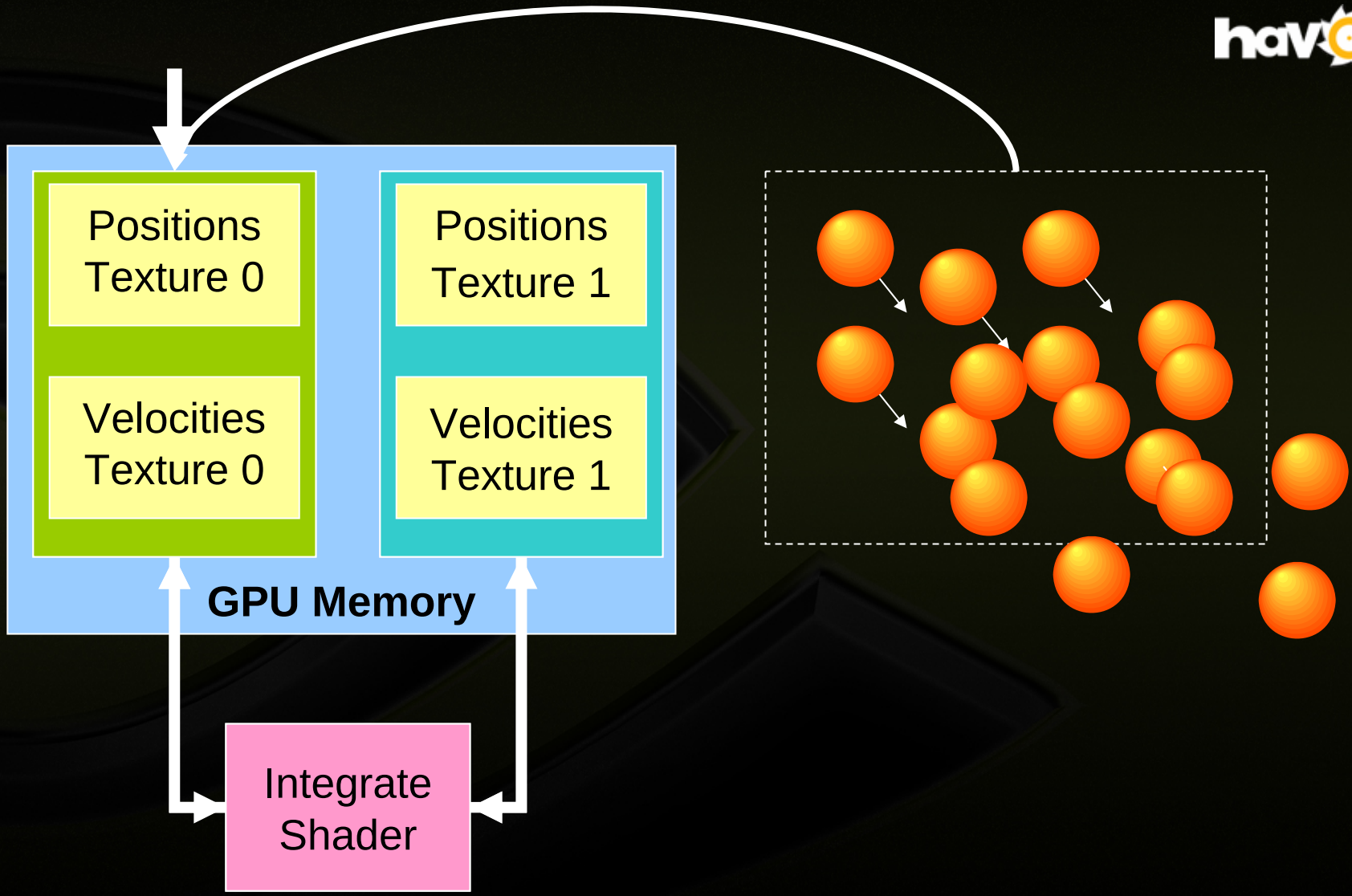
Performance Measurement



Havok FX Physics



Data Stays On The GPU



Custom Behaviors

Gameplay physics interaction

NVIDIA Technology for Physics

- Shader Model 3 GPUs

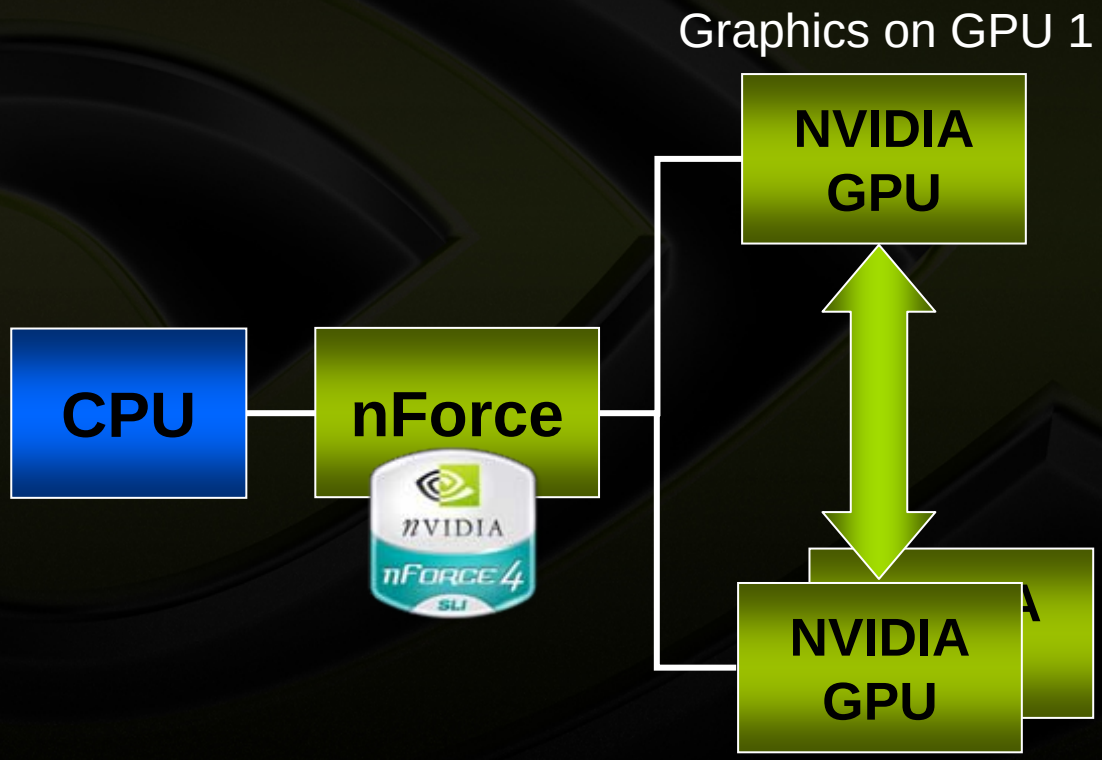
- SLI multi-GPU technology

- Cg Compiler

- New driver technology for physics

Game Physics on Multiple GPUs

Second GPU can be used for SLI graphics, second monitor or physics simulation



NVIDIA GPU Physics

- Multi-GPU configurations, mixed or same GPU type
  - One GPU does both graphics and physics
  - One GPU for graphics, one for physics
    - Enables extra GPU for rendering when FX is not active
  - Two GPUs for graphics, one for physics
    - Full speed rendering with full speed physics simulation

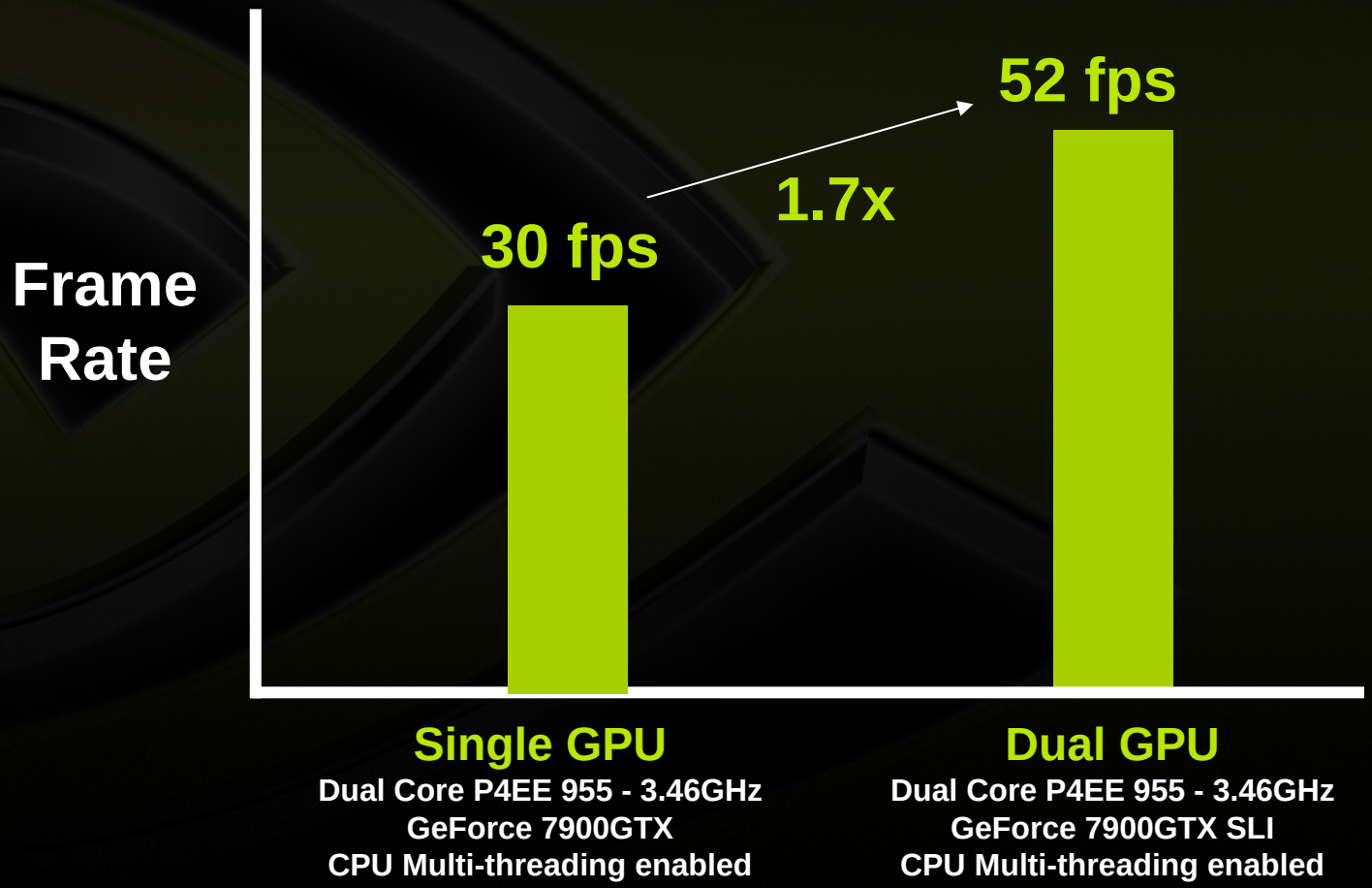
GeForce 7600 GS

GeForce 7900 GTX SLI

GeForce 7950 GX2 +
GeForce 7600

SLI Performance Scaling

Performance Measurement
15,000 Boulders with Shadows



Rendering

- Rendering is fully controlled by application
- Havok FX returns vertex buffers with position, velocity and optional user data
  - Supports OpenGL and Direct3D
- Rigid bodies rendered using instancing
  - Direct3D or OpenGL NVX_instanced_arrays extension
- Particles rendered as point sprites
  - Supports motion blur
  - Can modify particle color or size over time
  - Can use texture atlases for particle animation

The Future of GPU Physics

- Distributing physics across multiple GPUs
  - e.g. 2 GPUs for physics, 2 for rendering
- Brittle fracture
- Advanced smoke/cloud rendering
  - Volumetric shadowing
- Advanced fluids
  - Smoothed particle hydrodynamics
  - Isosurface extraction using DirectX 10 Geometry Shader

Questions?


[Paused]
Resume Demo
3 Restart Demo
1 Toggle Statistics
Settings
Toggle Help
Single Step
2 Quit