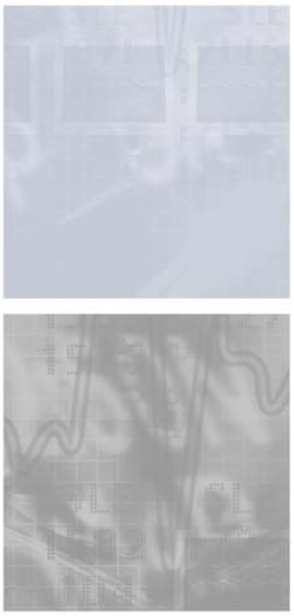


Asymmetric Physics Processing with ATI CrossFire™

Asymmetric Physics Processing with CrossFire™

Introduction

Recently, the PC industry has witnessed a steady building of interest in configuring PCs with more than one graphics processing unit. More and more motherboards are now shipping with multiple high-bandwidth PCI Express slots. This trend has largely been driven by the insatiable desire for better gaming experiences and more realistic 3D graphics.

ATI CrossFire™ is the most advanced technology available today for combining the power of multiple GPUs in a single PC. Until now, the focus of this technology has been on allowing the latest games with cutting-edge 3D graphics to run at high resolutions (to take advantage of the latest display technologies) with maximum image quality settings.

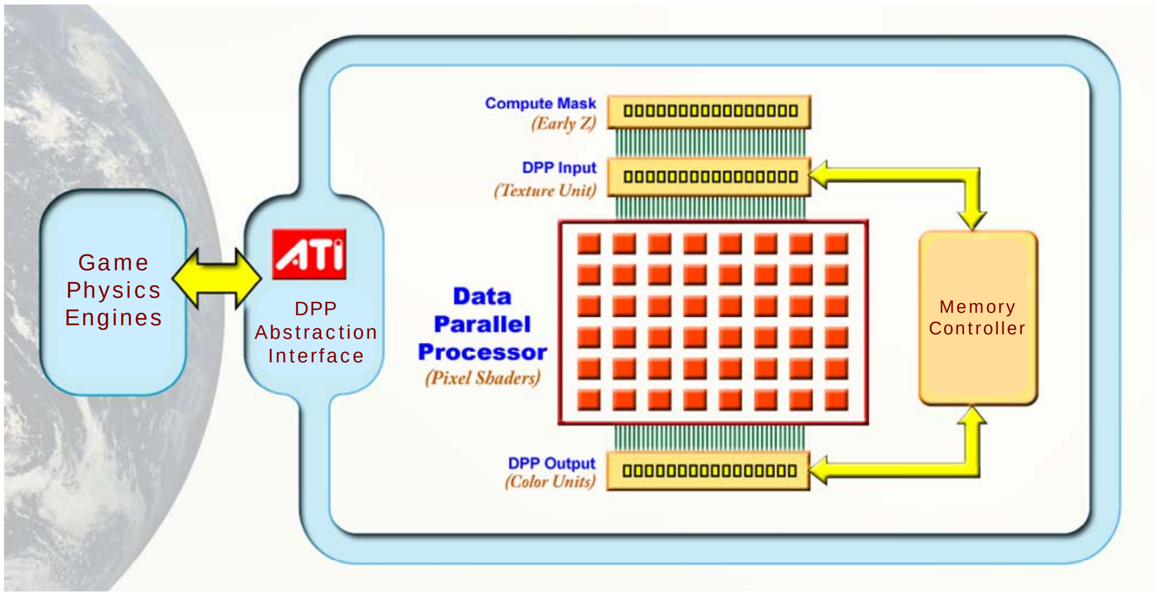
However, as GPUs have become more flexible and powerful, their potential for handling a wider range of processing tasks beyond just 3D rendering is starting to be realized. It can no longer be assumed that the GPUs in a system will necessarily be processing a single task at any given time. Asymmetric processing technology is a new feature of CrossFire™ that addresses this new environment, by allowing two or more GPUs with differing capabilities and feature sets to simultaneously handle different data parallel computing tasks, such as rendering and game physics, in a single system.

Game Computing

3D games are made up of a number of different tasks, including input processing, game state updating, artificial intelligence, physics, rendering, networking, audio, and more. While all of these tasks could run on the CPU in theory, special purpose processors can enhance a game by providing generous amounts of additional computing power for certain specific tasks. Even better, these special purpose processors can free up valuable CPU cycles to spend more time on other tasks, allowing them to be improved as well.

The first GPUs were designed to accelerate a very limited set of graphics rendering tasks. Modern GPUs are much more flexible and powerful than their predecessors. In particular, they excel at data parallel processing (DPP) tasks, where a common set of instructions is executed simultaneously across a large set of input data. Besides rendering, the detailed physics simulations that enhance the experience of recent 3D games also happen to fall into this category. Now, the technology has been developed to allow GPUs to accelerate these simulations, and so today's GPUs are able to take on an expanded role in game computing.

To expose the data parallel processing capabilities of the Radeon® X1000 family of GPUs to game physics engines and other applications that can take advantage of it, ATI has designed a DPP abstraction interface. This interface makes the GPU appear as a simplified data parallel processor, as illustrated in the diagram below.



ATI Data Parallel Processing Architecture for Physics Acceleration

A single data parallel processing task can be distributed across two or more GPUs to improve execution speed. If each GPU has an identical feature set and performance level, it is relatively straightforward to scale up performance in this way. The problem becomes significantly more difficult, however, when attempting to distribute a task across GPUs with differing characteristics.

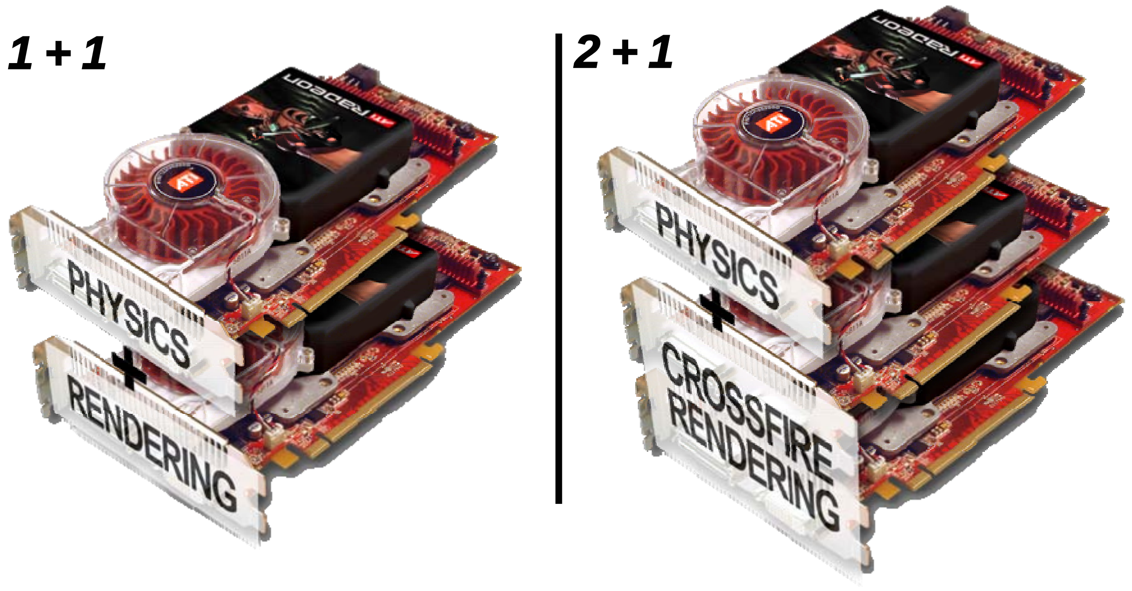
If each GPU has a different level of performance, then the process of load-balancing and keeping them both busy becomes more complex. This can introduce a significant amount of processing overhead, which in turn reduces the performance gain that can be realized from additional GPUs. If each GPU has a different feature set, then the application must be aware of this and make sure to only use features that are common to all of them. Optimizing performance in such cases can be extremely difficult.

Today's multi-GPU technologies avoid these issues by allowing only closely matching GPUs to be combined to accelerate 3D rendering. However, once the GPUs start being made responsible for game physics tasks in addition to 3D rendering, new options become available for distributing the workload between them efficiently.

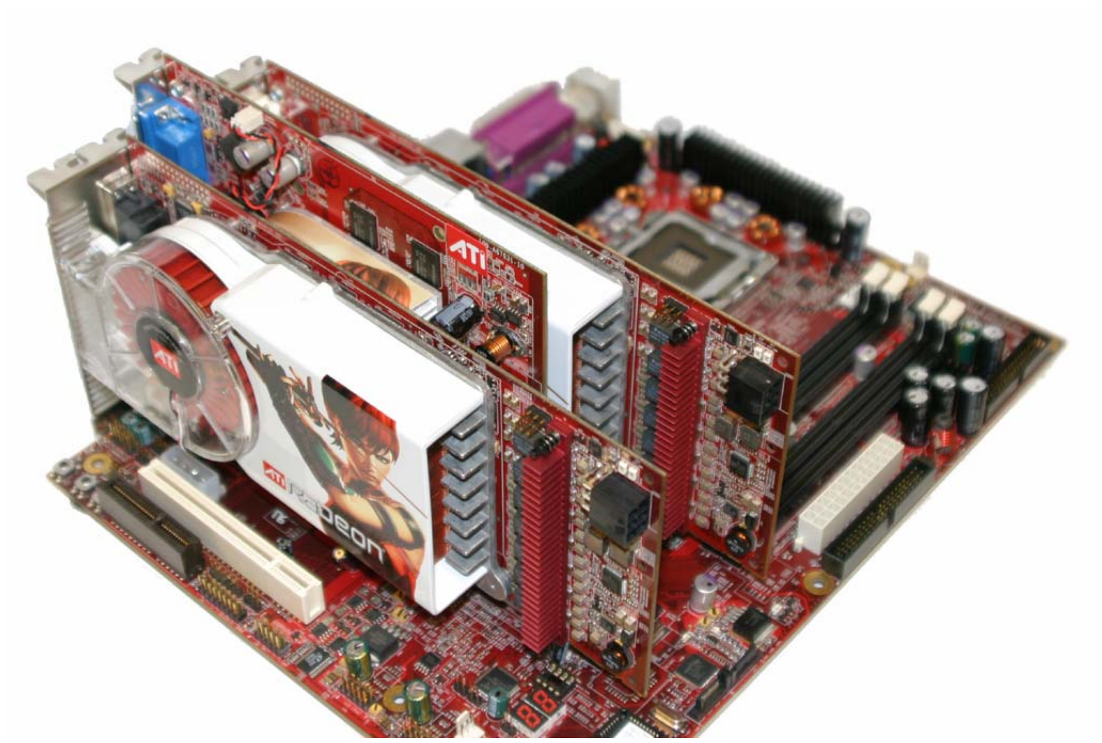
Asymmetric Processing

Asymmetric processing refers to the use of two or more GPUs in a single system, each with differing performance and feature sets, to simultaneously execute multiple tasks. These tasks may both be part of the same application, such as the rendering and physics processing tasks of a 3D game, but they are independent in the sense that no direct communication between them is required.

Asymmetric processing support is useful because it provides a simpler and more flexible upgrade path for multi-GPU systems. For example, it enables the possibility of combining a high-end graphics card with an entry-level graphics card and still realizing a compelling benefit. It also means that when upgrading from an older graphics card to a newer one, it may be possible to continue using both cards together, instead of having to sell or discard the older one.



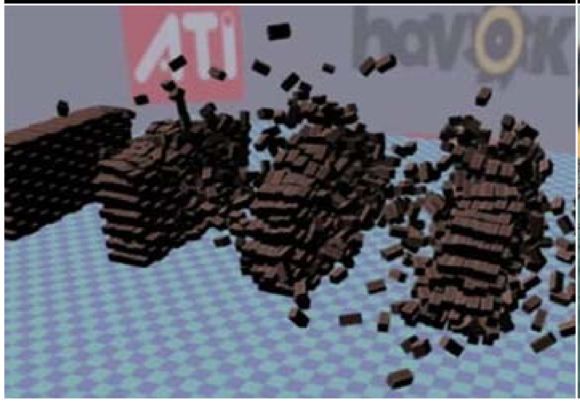
Some of the different asymmetric configurations possible with CrossFire™, using two or three graphics cards.



Example of a motherboard with an asymmetric CrossFire™ configuration, featuring dual Radeon® X1900 XTX GPUs for 3D rendering, plus a single Radeon® X1600 Pro GPU for physics processing.

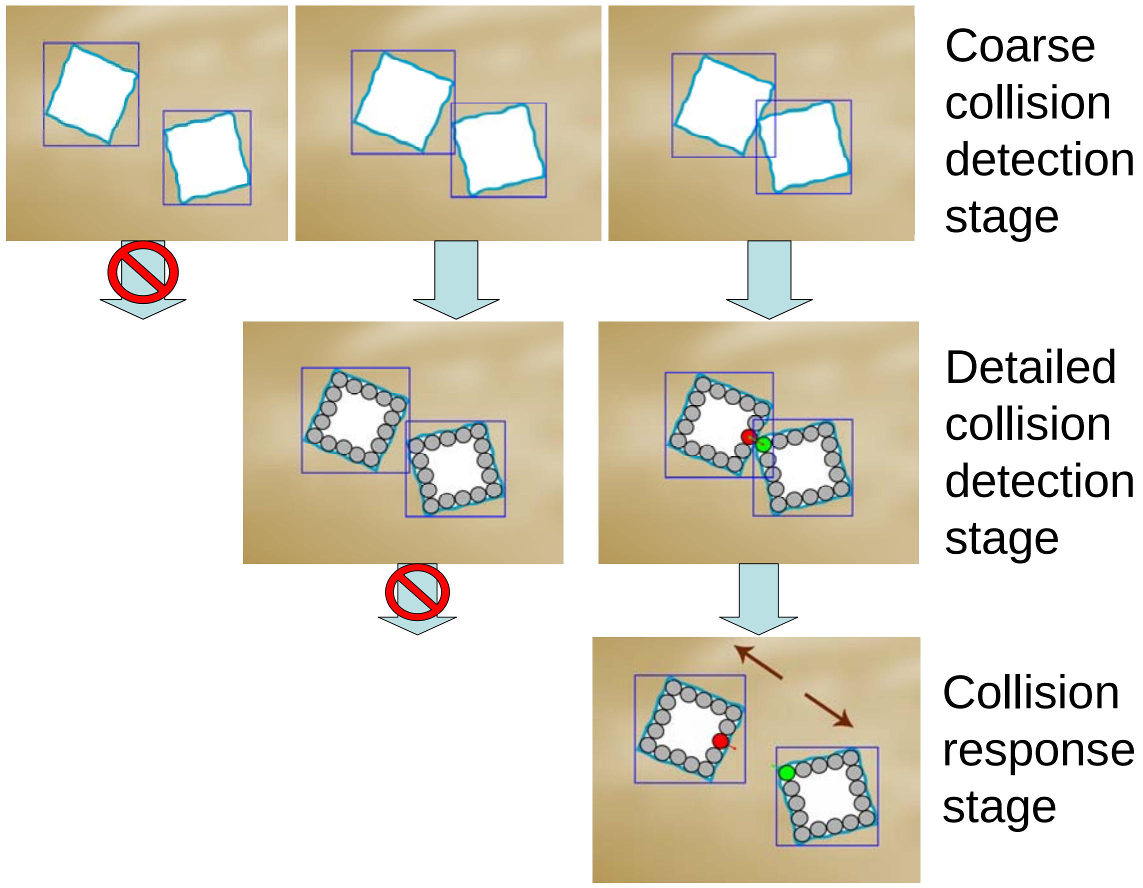
Game Physics Processing

Game physics processing is a good example of a data parallel processing task. In this case, the input data consists of either a large number of objects (such as boulders, debris, or particle systems), or a single deformable object with a large number of control points (such as smoke, fluids, hair, or cloth). For each iteration of the simulation, a set of forces is applied to each object or control point, which modifies their positions and velocities. They are also checked for collisions against other objects or surfaces.

Cloth	Smoke
Rigid Body Collisions	**Water**
	

Examples of 3D effects simulated with GPU physics processing.

The key to accelerating this type of simulation is to do it in multiple stages. The first stage determines a coarse approximation of areas where a collision will occur. These areas are then passed on to a second stage, where a more detailed simulation is done to determine the exact points of contact. Finally, these contact points are passed on to a collision response stage. This approach allows a short and fast shader codepath to be executed on non-colliding objects or control points, while progressively longer and more time-consuming shader codepaths are executed in later stages only on the specific locations where collisions occur.


Coarse collision detection stage
Detailed collision detection stage
Collision response stage

GPU Physics Processing Stages

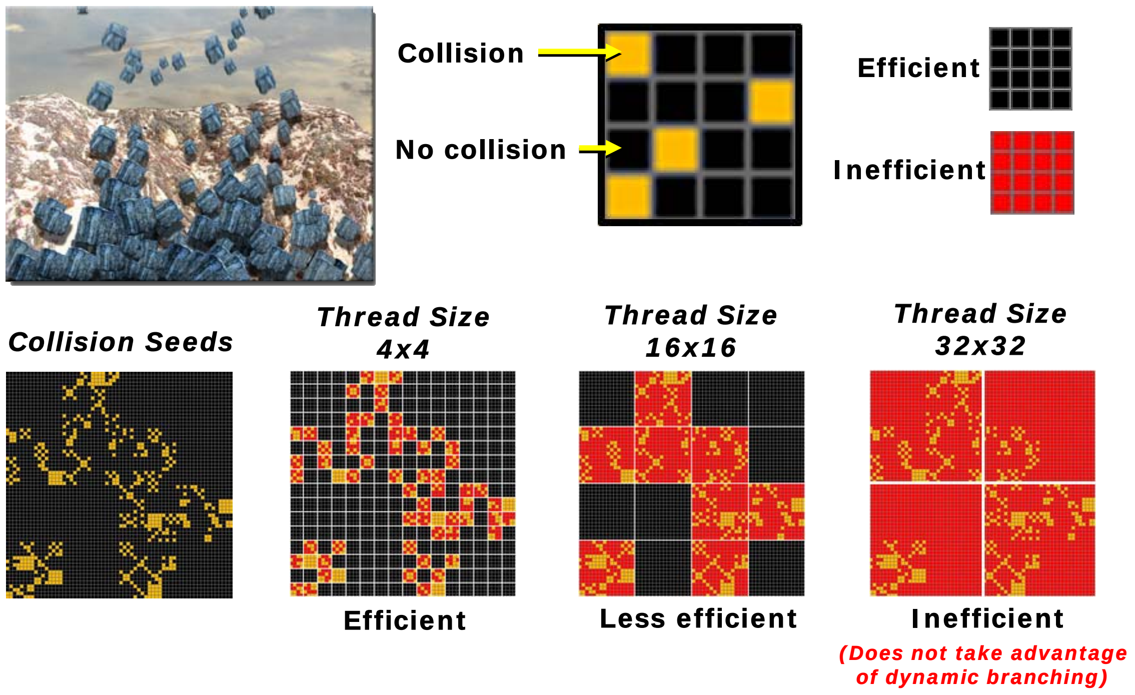
A modern GPU with well-designed shader processors can perform this kind of physics simulation much faster than a CPU can, since it can operate on many more objects or control points simultaneously. Games can take advantage of this fact by adding more physically interacting objects for greater realism. Removing the heavy burden of physics processing from the CPU also gives it more time to spend on other tasks, such as more sophisticated A.I. and richer gameplay.

ATI's latest Radeon® X1000 series GPUs possess characteristics that allow them to excel at physics processing. These include highly efficient dynamic branching with fine-grained thread sizes, and a very high level of shader processing capability. The former enables fast determination of which locations pass from earlier to later processing stages, while the latter is necessary to handle the complex collision calculations in the later stages efficiently.

One characteristic of the algorithms used for physics processing is their high arithmetic intensity. This means that a large number of operations must be performed on each piece of input data. In GPU architectures, the input data is fetched by texture units, and the arithmetic operations are handled by shader processors. The latest Radeon® X1000 series GPUs feature a high ratio of shader processors to texture units, and thus are very well suited to deal with these algorithms.

The illustration below shows the importance of thread size and dynamic branching to GPU physics processing. Dark squares indicate parts of the current frame where no objects are colliding – only a simple position and velocity update is required for objects in these areas.

Orange squares indicate places where collisions may be occurring, and therefore the more detailed collision detection algorithm must be executed on objects in these areas. Red squares indicate areas where detailed collision detection is being performed unnecessarily, due to thread size limitations of the GPU. The larger the thread size, the more red areas there are, and the lower the overall processing efficiency.



Effect of thread size and dynamic branching efficiency on GPU physics processing.

As a result of these characteristics, even the entry-level products in the Radeon® X1000 family are capable of accelerating physics processing well beyond what a high-end CPU would be capable of alone. In many cases, they can even outperform specialized physics processing units. In a gaming PC with an asymmetric multi-GPU configuration, it will be possible to achieve the ultimate gaming experience using one or two high-end GPUs for 3D rendering together with a third GPU working as a physics co-processor.

Conclusion

The ATI CrossFire™ platform is the only multi-GPU technology available today that supports asymmetric configurations for physics processing. This technology enables GPUs with different performance levels and feature sets to share game computing tasks. The result is a much wider variety of viable multi-GPU system configurations, and an expanded role for the GPU in delivering more compelling gaming experiences.

ati.com