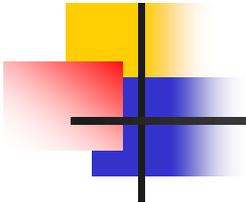


Новые тенденции использования GPGPU

Арсаев Марат

Video Group

CS MSU Graphics & Media Lab

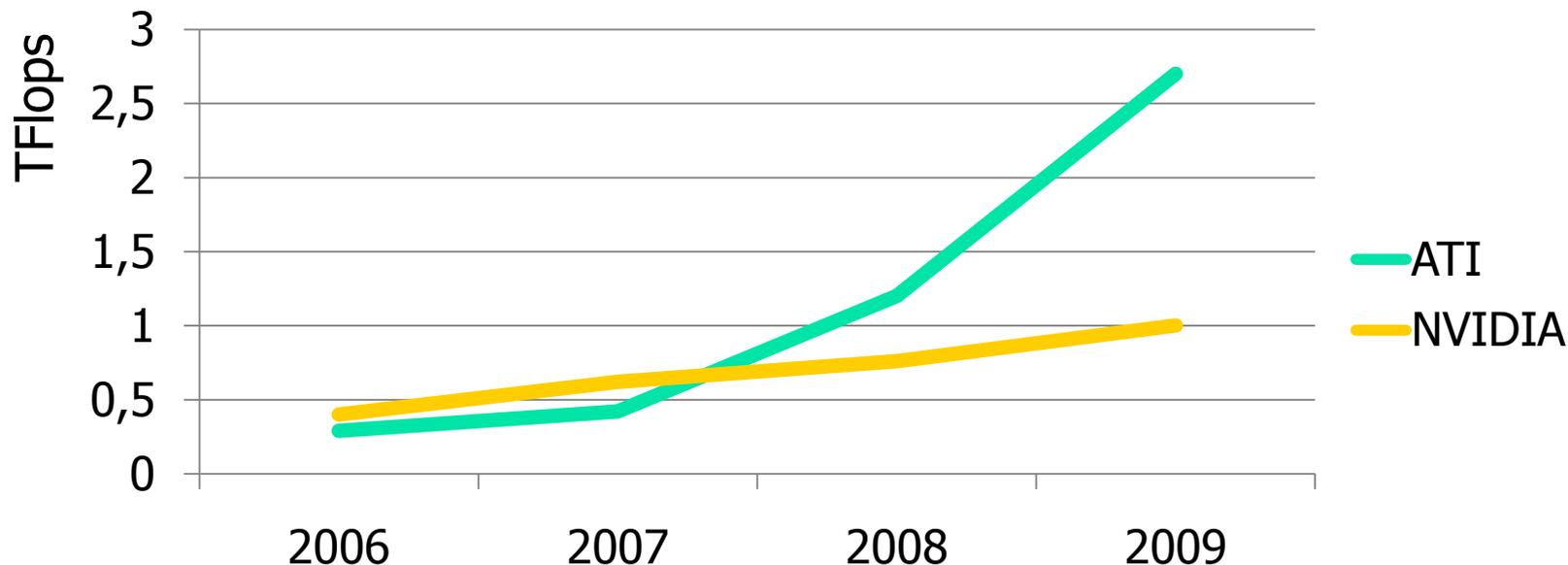


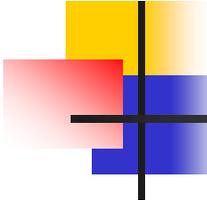
Содержание

- **Текущее положение GPGPU**
- Сравнение GPU API
- Средства разработки
- Обработка видео на GPU
- Будущее GPGPU

Текущее положение GPGPU

GPGPU – техника программирования на графическом процессоре задач общего назначения

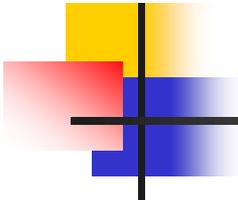




Особенности современных GPU



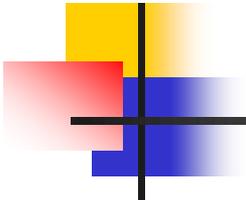
- Наличие множества архитектур
- Наличие множества API для программирования
- Сложно сразу перейти на программирование на GPU



Развитие GPGPU

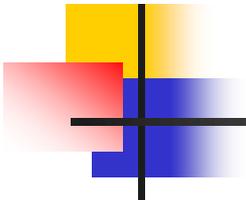
Производители продвигают GPU в массы:

- Разработка C-подобных языков
- Разработка библиотек на GPU
- Создание отладчиков\профилировщиков
- Создание специализированных сред



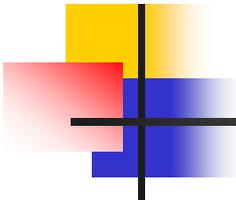
Содержание

- Текущее положение GPGPU
- **Сравнение GPU API**
 - OpenGL
 - CUDA
 - OpenCL
- Средства разработки
- Обработка видео на GPU
- Будущее GPGPU



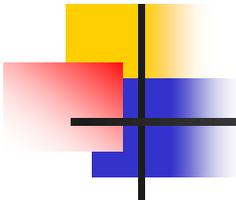
OpenGL

- Язык – OpenGL Shading Language
- Некоторая запутанность использования
- Можно использовать графические возможности GPU
- Поддерживается устаревшими моделями
- Достаточен для небольших проектов



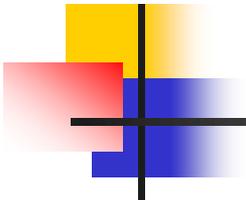
CUDA

- Язык – расширение над C
- Более гибкая работа с памятью:
 - Shared memory
 - Произвольный доступ к памяти
- Привязка к NVIDIA
- При хорошем понимании архитектуры можно получить значительный прирост производительности



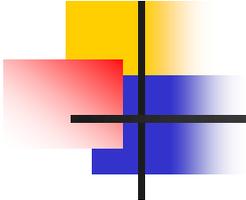
OpenCL

- Расширение языка C, похож на CUDA
- Код может выполняться на любом OpenCL-поддерживающем устройстве – главное преимущество и недостаток OpenCL
- Сложности с расширениями



OpenCL

- У NVIDIA – обертка на CUDA
- У AMD – разработана с нуля (Stream SDK)
- Приближение производительности к CUDA
- Скорая возможность полной унификации AMD и NVIDIA



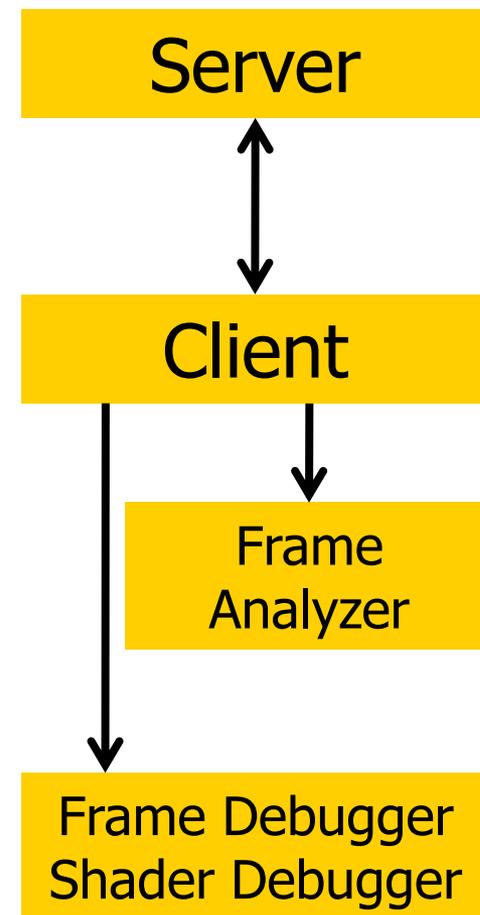
Содержание

- Текущее положение GPGPU
- Сравнение GPU API
- **Средства разработки**
 - Профилировщики и отладчики
 - IDE
 - Библиотеки
- Обработка видео на GPU
- Будущее GPGPU

AMD GPU Performance Studio



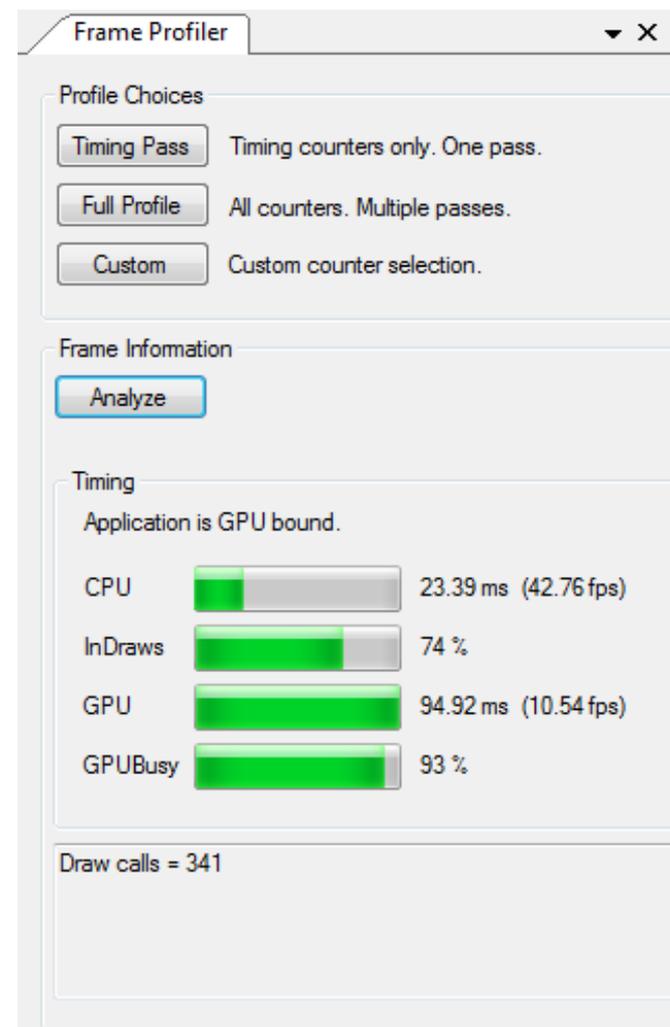
- Система клиент-сервер
- Мощное средство оценки производительности программы на DirectX 10 и DirectX 10.1
- Только для карт ATI начиная с серии HD2000



AMD GPU Performance Studio



- Быстрая или углубленная оценка производительности приложения
- Возможность отладки HLSL и ассемблера



Intel Graphics Performance Analyzer



- Отладчик и профилировщик DirectX 9 и DirectX 10 приложений для графических чипов Intel
- Важен для разработки под мобильные решения Intel
- Структура схожа с AMD GPU Performance Studio

Intel Graphics Performance Analyzer

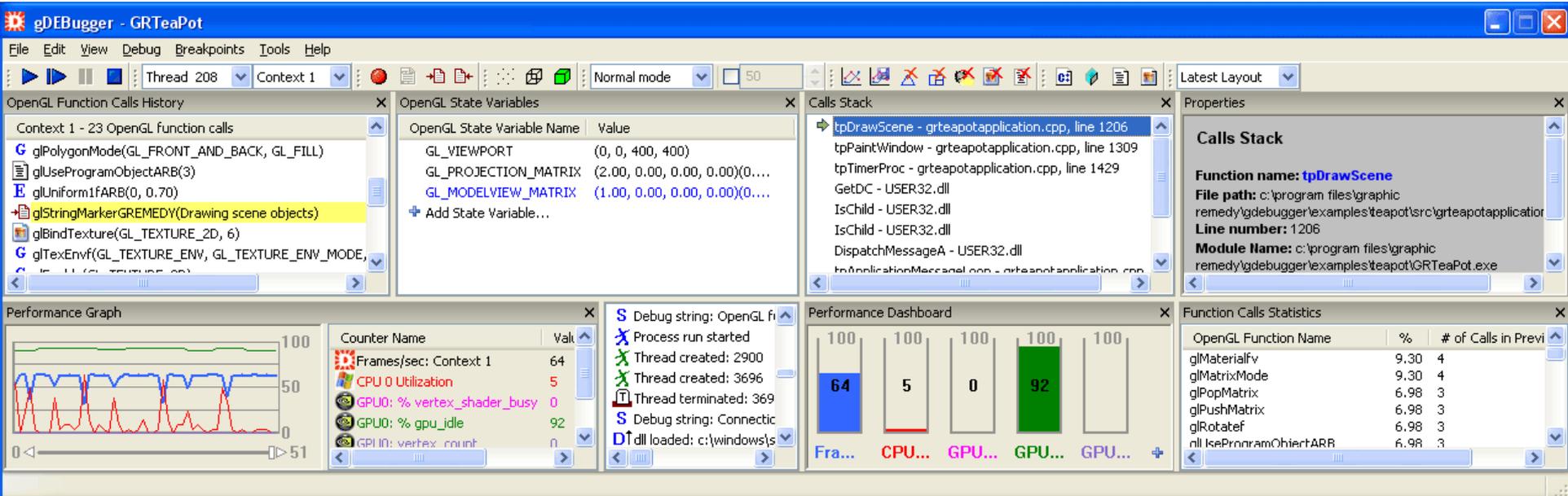


- Два варианта работы – онлайн мониторинг и покадровый анализ
- Удобный режим внесения изменений – просмотр результата



gDEBugger

- Отладчик для OpenGL (GLSL до 1.4)
- Ныне хорошо поддерживается и развивается



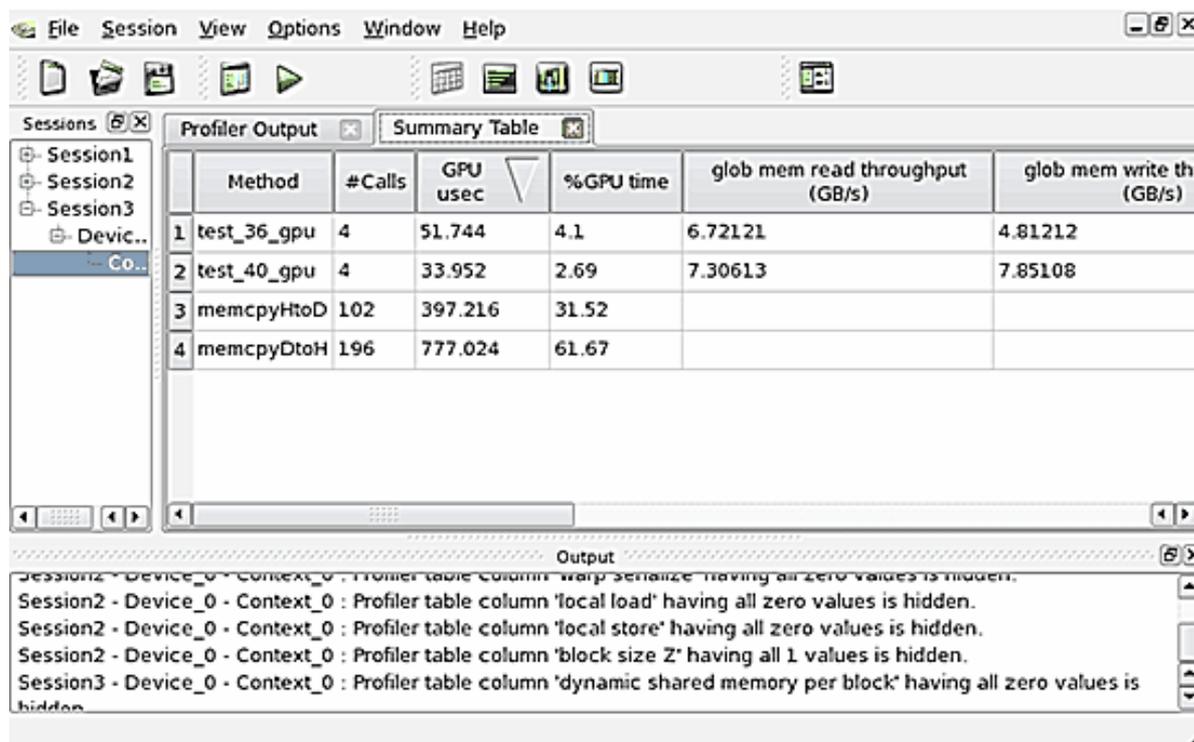
The screenshot shows the gDEBugger - GRTeaPot application interface. The main window displays the following panels:

- OpenGL Function Calls History:** Lists recent OpenGL function calls, including `glPolygonMode`, `glUseProgramObjectARB`, `glUniform1fARB`, `glStringMarkerGREMEDY`, `glBindTexture`, and `glTexEnvf`.
- OpenGL State Variables:** Shows the current state of OpenGL variables, such as `GL_VIEWPORT` (0, 0, 400, 400), `GL_PROJECTION_MATRIX`, and `GL_MODELVIEW_MATRIX`.
- Calls Stack:** Displays the current call stack, with `tpDrawScene - grteapotapplication.cpp, line 1206` selected.
- Properties:** Shows details for the selected function, including the function name (`tpDrawScene`), file path, line number (1206), and module name.
- Performance Graph:** A line graph showing performance metrics over time, including frames per second, CPU utilization, and GPU utilization.
- Performance Dashboard:** A bar chart showing performance metrics for different components, including `Fra...` (64), `CPU...` (5), `GPU...` (0), `GPU...` (92), and `GPU...` (100).
- Function Calls Statistics:** A table showing the percentage and number of calls for various OpenGL functions.

OpenGL Function Name	%	# of Calls in Previ
glMaterialfv	9.30	4
glMatrixMode	9.30	4
glPopMatrix	6.98	3
glPushMatrix	6.98	3
glRotatef	6.98	3
glUseProgramObjectARB	6.98	3

Cudaprof

- Профилировщик ядер CUDA
- Отображает основные и самые нужные показатели



The screenshot shows the Cudaprof application window with a menu bar (File, Session, View, Options, Window, Help) and a toolbar. The main window is divided into a left sidebar with a tree view of sessions and a main area displaying a 'Summary Table' of profiler output. The table has columns for Method, #Calls, GPU usec, %GPU time, glob mem read throughput (GB/s), and glob mem write thr (GB/s). Below the table, an 'Output' window shows log messages for various sessions and contexts.

	Method	#Calls	GPU usec	%GPU time	glob mem read throughput (GB/s)	glob mem write thr (GB/s)
1	test_36_gpu	4	51.744	4.1	6.72121	4.81212
2	test_40_gpu	4	33.952	2.69	7.30613	7.85108
3	memcpyHtoD	102	397.216	31.52		
4	memcpyDtoH	196	777.024	61.67		

Output window content:

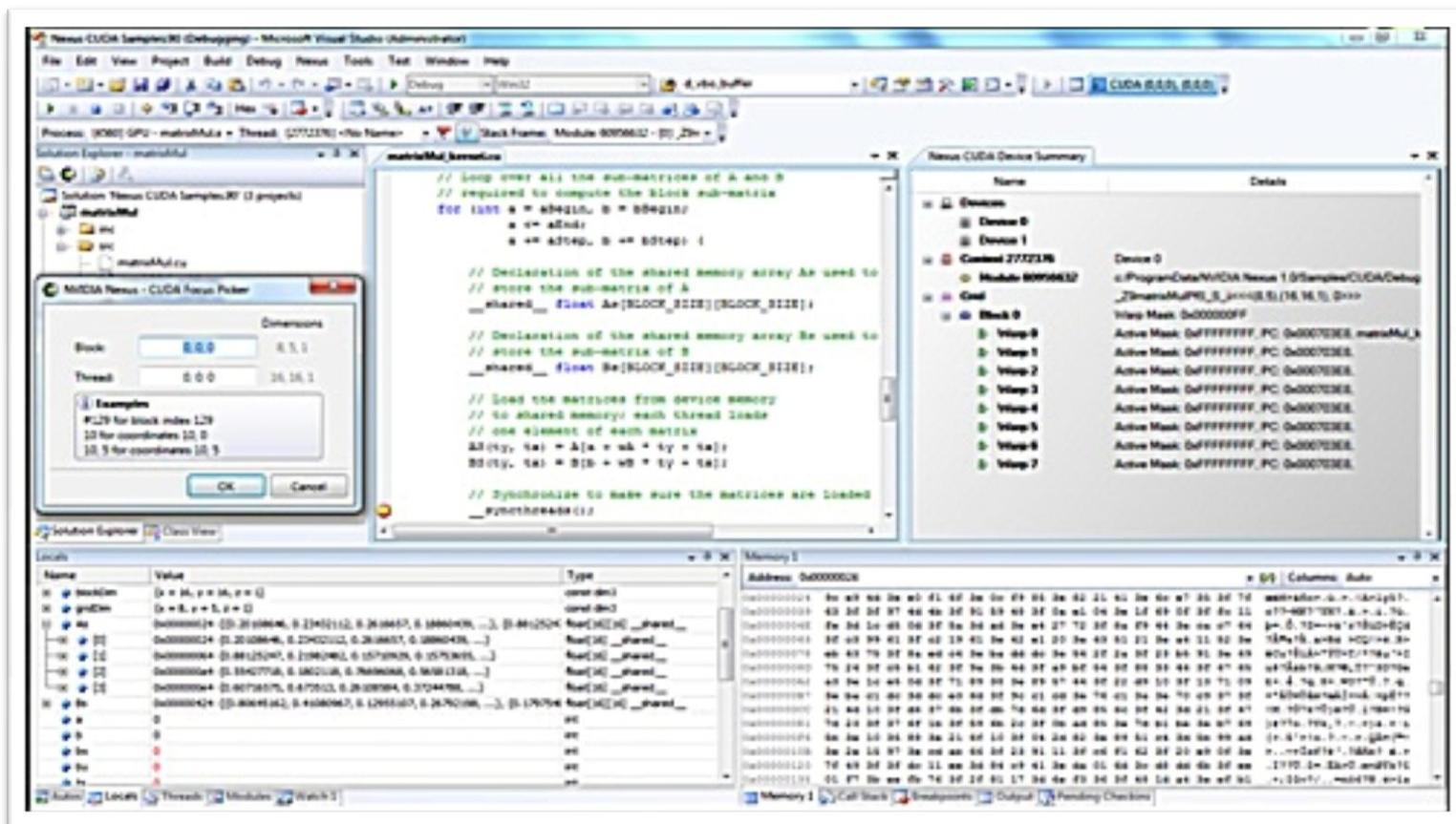
```

Session2 - Device_0 - Context_0 : Profiler table column 'local load' having all zero values is hidden.
Session2 - Device_0 - Context_0 : Profiler table column 'local store' having all zero values is hidden.
Session2 - Device_0 - Context_0 : Profiler table column 'block size Z' having all 1 values is hidden.
Session3 - Device_0 - Context_0 : Profiler table column 'dynamic shared memory per block' having all zero values is hidden.

```

NVIDIA Nexus

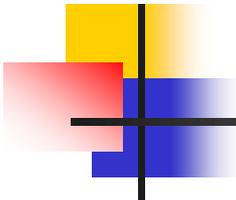
Интегрированный в Visual Studio профилировщик и отладчик



GPU библиотеки

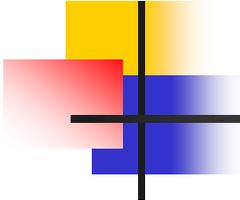
На данный существует множество вспомогательных библиотек:

- Постоянно улучшающиеся NVPP (IPP), CUBLAS, CULAtools
- ACML-GPU (для 64битных систем)
- Сторонние библиотеки



Содержание

- Текущее положение GPGPU
- Сравнение GPU API
- Средства разработки
- **Обработка видео на GPU**
- Будущее GPGPU



Обработка видео на GPU

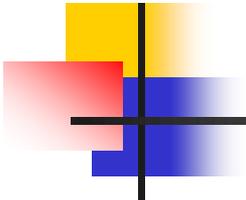
Сравним производительность похожего кода на CUDA, OpenCL, OpenGL

- Взят box filter из CUDA SDK и OpenCL SDK, измененные под обработку видео
- Написана программа на OpenGL, реализующая несепарабельный box filter

Обработка видео на GPU

Платформа: NVIDIA 9400M, Intel Atom 1.66 GHz

	CUDA	OpenCL	OpenGL
320x240	104fps	89fps	90fps
640x480	63fps	43fps	62fps



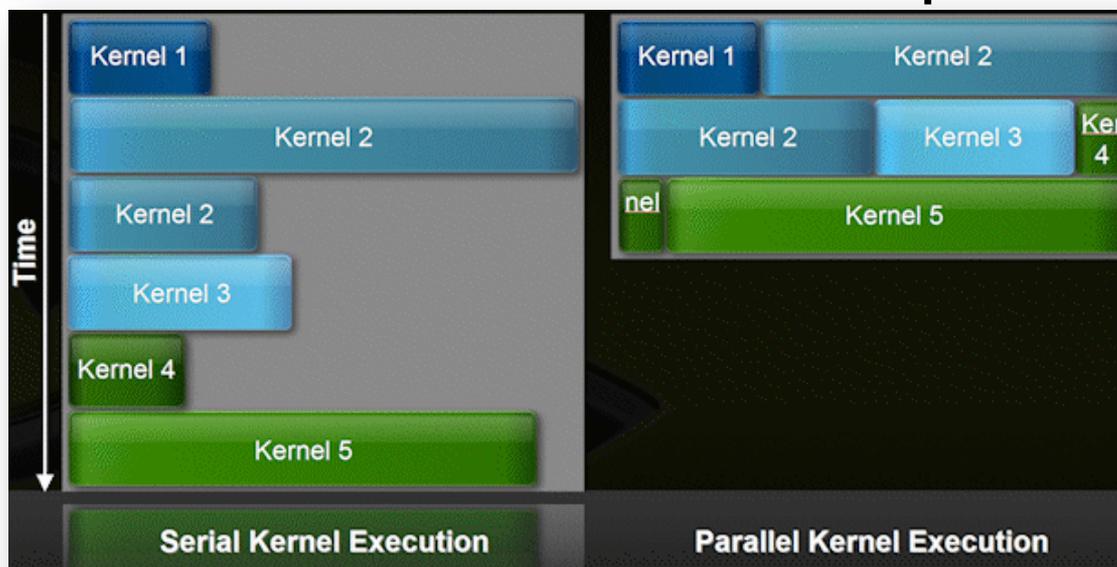
Содержание

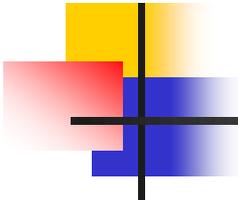
- Текущее положение GPGPU
- Сравнение GPU API
- Средства разработки
- Обработка видео на GPU
- **Будущее GPGPU:**
 - NVIDIA Fermi
 - Intel Larrabee

NVIDIA Fermi

Во многом новая архитектура:

- Программируемый кеш
- Увеличенная разделяемая память
- Параллельное вычисление ядер в одном SM:





NVIDIA Fermi

Встроенная поддержка:

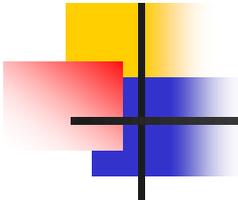
- CUDA 2.2
- C++
- OpenCL 1.0
- DirectX 11 и DirectCompute
- OpenGL 3.2
- и др.

NVIDIA Fermi



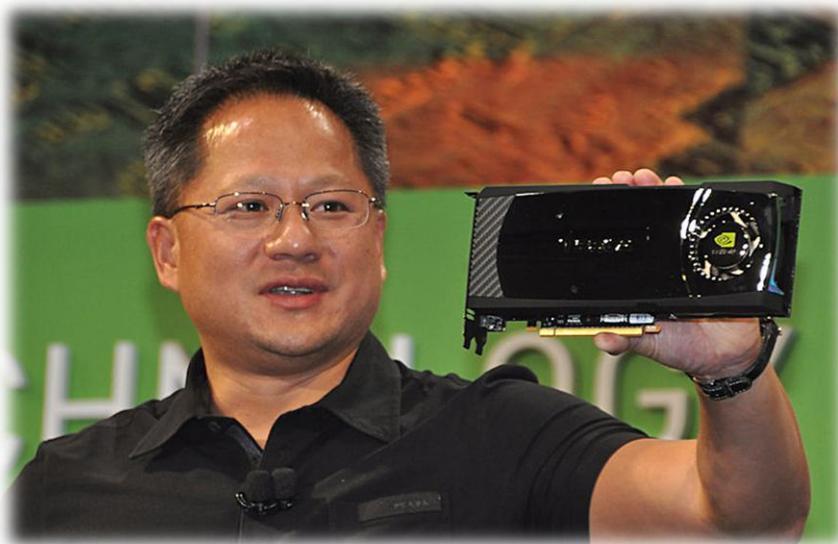
По многим характеристикам превосходит
Radeon Cypress XT (5870)

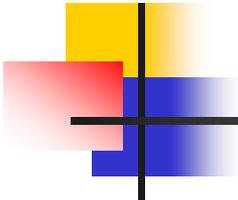
	GT200	Cypress XT	Fermi
Кол-во транзисторов	1.4 млрд	2.154 млрд	3 млрд
Потоковых процессоров	240	1600	512
Шина памяти	512 бит	256 бит	384 бит
Кеш L1	16 КБ	8 КБ	48 КБ
Кеш L2	-	512 КБ	768 КБ
Разделяемой памяти	16 КБ	64 КБ	до 48 КБ
Вычислений с плавающей точкой за такт	240 MAD	1600 MAD	512 FMA
Вычислений двойной точности за такт	30 FMA	640 FMA	256 FMA



NVIDIA Fermi

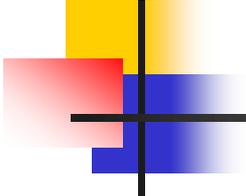
Показ на GPU Technology Conference в конце сентября вызвал множество споров





NVIDIA Fermi

- У NVIDIA есть достойный ответ Intel и ATI
- Более гибкая, быстрая и надежная архитектура
- Удобное программирование вместе с NVIDIA Nexus
- Дата выхода – март 2010



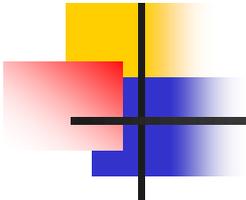
Intel Larrabee

- Larrabee – GPU на ядрах x86
- Встроенная поддержка C/C++
- Код выполняется на CPU и GPU одновременно
- Шанс Intel выйти на рынок дискретных видеокарт

Intel Larrabee

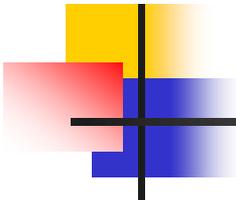
- Первый показ осенью 2009
- Все находится в разработке





Intel Larrabee

- Отсутствие окончательных программных и аппаратных спецификаций
- Отсутствие конкретных дат выхода
- Большая заинтересованность со стороны сообщества
- Разработка Larrabee Prototype Library
- Будем ждать выхода...



Выводы

- Производители стараются максимально продвигать и поддерживать GPGPU
- Все большее количество задач может быть выполнено на современных GPU
- Проблемы обработки изображений и видео очень хорошо ложатся на GPU

Список литературы

1. <http://www.fudzilla.com/content/view/15954/1/> NVIDIA Fermi Launch
2. <http://www.semiaccurate.com/2009/10/01/nvidia-fakes-fermi-boards-gtc/> NVIDIA Fermi Fake at GTC
3. <http://www.bit-tech.net/hardware/graphics/2009/09/30/ati-radeon-hd-5870-architecture-analysis/8>
HD 5870 Architecture
4. http://www.nvidia.com/object/fermi_architecture.html NVIDIA Fermi Architecture
5. <http://www.nvidia.com/nexus> NVIDIA Nexus
6. <http://developer.amd.com/gpu/acmlgpu/pages/default.aspx> AMD ACML Library
7. <http://www.intel.com/cd/corporate/pressroom/emea/rus/archive/2009/419637.htm> Larrabee press release
8. [http://en.wikipedia.org/wiki/Larrabee_\(GPU\)](http://en.wikipedia.org/wiki/Larrabee_(GPU)) Intel Larrabee GPU
9. <http://www.ixbt.com/news/all/index.shtml?12/46/03> Fermi architecture details
10. <http://developer.nvidia.com/object/get-opencl.html> NVIDIA OpenCL

Лаборатория компьютерной графики и мультимедиа



Видеогруппа это:

- Выпускники в аспирантурах Англии, Франции, Швейцарии (в России в МГУ и ИПМ им. Келдыша)
- Выпускниками защищено 5 диссертаций
- Наиболее популярные в мире сравнения видеокодеков
- Более 3 миллионов скачанных фильтров обработки видео