

УДК 004.75+004.272.26+004.925.3

КОНФИГУРИРОВАНИЕ ВЫЧИСЛИТЕЛЬНОГО КЛАСТЕРА ДЛЯ РАСПРЕДЕЛЕННОГО РЕНДЕРИНГА ВЫСОКОРЕАЛИСТИЧНЫХ ЗД СЦЕН

Р.Р. Газизов¹, Д.И. Костюк², В.В. Кугуракова³

¹⁻³ Казанский (Приволжский) федеральный университет

¹starkindustries14579@gmail.com, ²xdxnkx@gmail.com, ³vlada.kugurakova@gmail.com

Аннотация

Описан способ реализации распределенных вычислений с использованием GPU и современных инструментов администрирования и управления расчетами для ферм рендеринга. Проведен сравнительный анализ традиционного метода вычислений на CPU и современных подходов. Рассмотрены различные конфигурации вычислительного кластера. Определен наиболее производительный способ выполнения визуализации.

Ключевые слова: визуализация, распределенный рендеринг, вычислительный кластер, конфигурация.

ВВЕДЕНИЕ

Трёхмерная графика играет в современном мире всё большее значение. Благодаря ей можно создавать фотореалистичные или стилизованные изображения, получать представления о реальных или вымышленных объектах, а также изображения того, чего обычно человек не в силах увидеть невооружённым глазом. Такими примерами могут служить модель галактики или множество форм микроорганизмов. Благодаря этому компьютерная графика активно используется во многих отраслях жизни, таких, как архитектура, промышленность, киноиндустрия, игровая индустрия, рекламная продукция, наука, медицина.

В процессе работы с трёхмерной графикой приходится так или иначе совершать следующие шаги:

- моделирование – создание трёхмерной математической модели сцены и объектов в ней;

- текстурирование – назначение поверхностям моделей растровых или процедурных текстур (подразумевает также настройку свойств материалов – прозрачность, отражения, шероховатость и пр.);
- освещение – установка и настройка источников света;
- анимация – придание движения объектам;
- динамическая симуляция – автоматический расчёт взаимодействия частиц, твёрдых/мягких тел и пр. с моделируемыми силами гравитации, ветра, выталкивания и др., а также друг с другом;
- рендеринг (визуализация) – построение проекции в соответствии с выбранной физической моделью;
- композитинг (компоновка) – доработка изображения.

Рендеринг является важным этапом в работе с трёхмерной графикой, так как от его результата зависят реалистичность и эффектность итоговой работы. Сам по себе этот процесс очень ресурсоёмкий, вследствие чего он занимает продолжительное время вычислений. К примеру, визуализация видеоролика длиной в несколько минут в высоком разрешении может выполняться несколько дней или недель, в зависимости от сложности сцен. Поэтому зачастую не хватает вычислительной мощности одного компьютера для выполнения поставленной задачи визуализации. Возникает необходимость решения этой проблемы.

Один из способов решения трудоемких вычислительных задач – это использование вычислительного кластера или группы компьютеров, предназначенных для распределенных вычислений в параллельном режиме.

Заранее предпросчитанный рендеринг может быть использован в подготовке удалённых планов для крупномасштабных культурно-исторических реконструкций, например, [1].

РАСПРЕДЕЛЕННЫЙ РЕНДЕРИНГ

Техника распределенного рендеринга позволяет распределить задачи визуализации между несколькими компьютерами в сети, то есть производить вычисления мощностями нескольких компьютеров (см. рис. 1). При распределенной визуализации анимации (секвенции) между компьютерами распределяются диапазоны кадров. Каждая машина проводит рендеринг своей части, кадр за кадром, пока не решит поставленную задачу. В случае с визуализацией одного кадра, но очень большого разрешения, можно применить следующий способ:

основная идея состоит в том, что кадр, который подвергается рендерингу, делится на небольшие участки (бакиты) (см. рис. 2); каждой машине в сети раздается некоторое их количество; результаты расчетов собираются в итоговое изображение.

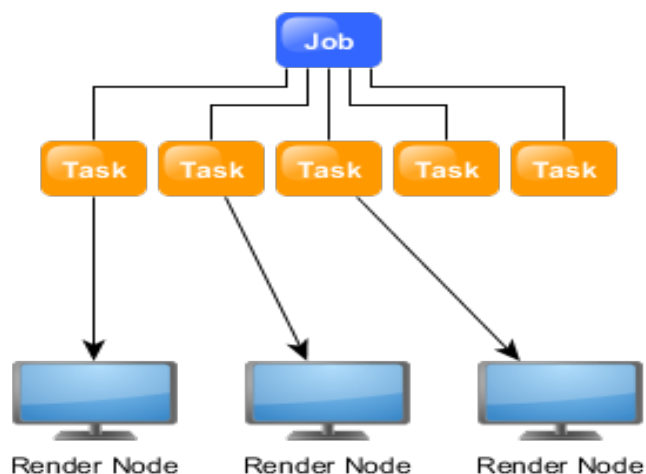


Рис. 1. Распределение работы между нодами

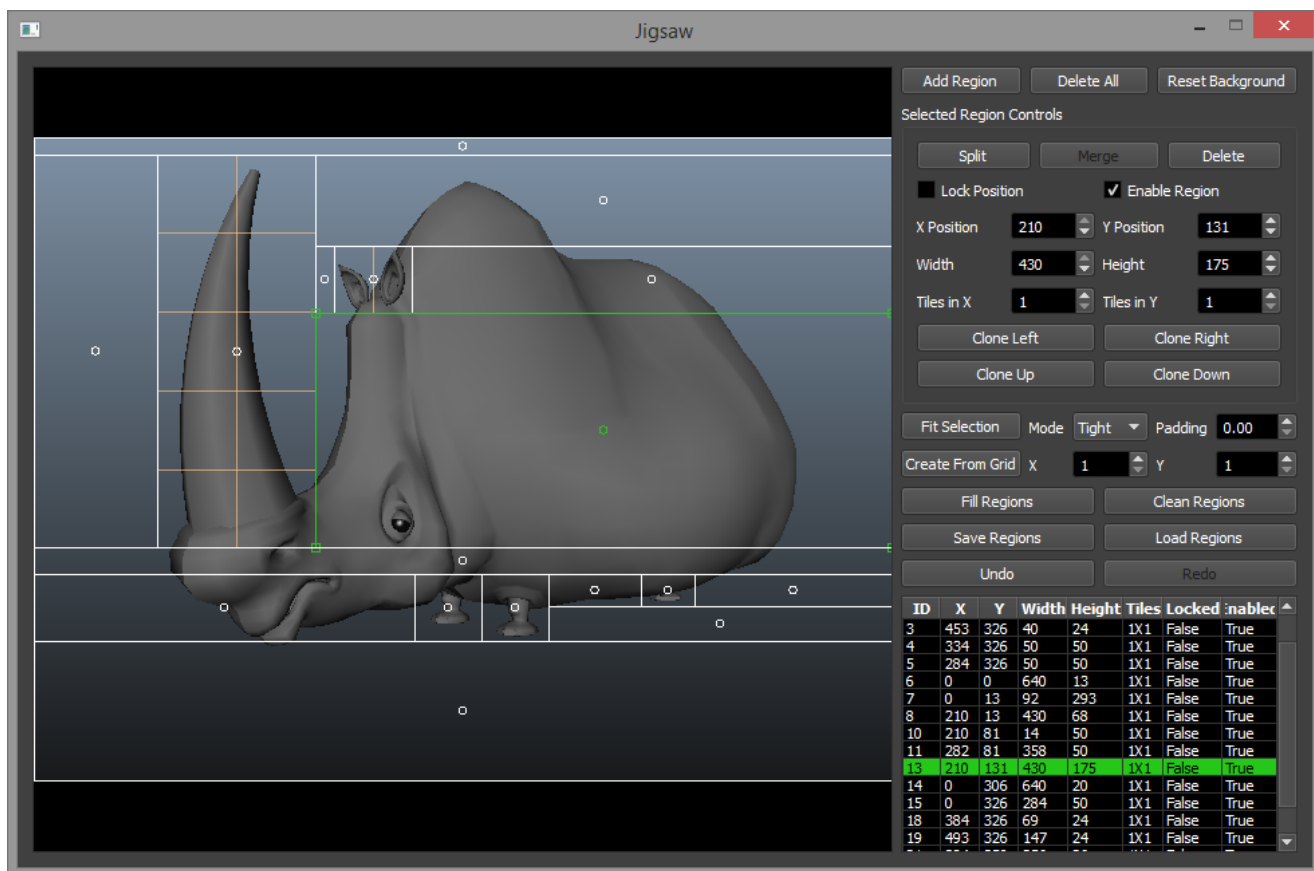


Рис. 2. Разбиение кадра на бакиты

ВЫЧИСЛИТЕЛЬНЫЙ КЛАСТЕР

Что же представляет из себя вычислительный кластер? Каждый вычислительный узел – это компьютер или нода, которые имеют собственную оперативную память, операционную систему и т. д. Чем больше компьютеров находится в вычислительной сети, тем меньше времени требуется на визуализацию сцены [2].

Узлы объединены высокопроизводительной коммуникационной сетью, обладающей минимальными задержками и широкой полосой пропускания информации, то есть компьютеры «видят» друг друга через сетевое окружение. На все машины установлены идентичное программное обеспечение и плагины. Если компьютеры не в прямой зоне досягаемости, то необходимо используются утилиты, обеспечивающие удаленный доступ, для работы с удаленными машинами.

Все необходимые данные для распределенного рендеринга должны храниться на сервере или общем сетевом диске, к которому будут обращаться вычислительные узлы. Этот диск хранит файлы сцен, текстуры, настройки. Он же принимает выходные данные – итоговый результат. Он виден всем машинам под одним и тем же именем.

В настоящее время описанные кластеры находят широкое применение по всему миру.

СРЕДСТВА ВИЗУАЛИЗАЦИИ

Визуализацию можно выполнять как на CPU, так и на GPU. Мы выбрали два наиболее популярных на данный момент решения из специализированного программного обеспечения для визуализации: Redshift (GPU); Arnold (CPU). Каждый из этих визуализаторов (рендеров) поддерживает распределенные вычисления. Они производят визуализацию с использованием алгоритма трассировки луча, что позволяет получить фотореалистичные изображения.

Визуализатор Arnold использует при рендеринге в качестве вычислительной платформы центральный процессор CPU. Преимущества Arnold – это высокая оптимизация, гибкость и система плагинов, к которым относятся процедурные функции, шейдеры, свои драйверы и прочее. Однако визуализация замедляется из-за данных, которые находятся в кэш-памяти, это увеличивает используемые оперативную и дисковую память.

Программа Redshift использует графический процессор GPU. Это первый в мире рендер, полностью ускоренный на GPU. Для его работы необходима видеокарта от Nvidia. Он поддерживает следующие алгоритмы глобального освещения: Brute-force GI; Photon Mapping; Irradiance Cache; Irradiance Point Cloud. При визуализации данные подгружаются из оперативной памяти или жесткого диска в видеопамять графического процессора. Достаточно 1 гигабайта видеопамети, чтобы провести рендеринг сложных сцен [3, 4].

Достоинством GPU-процессоров является огромная вычислительная мощность. При большом объеме информации, которая обрабатывается, GPU имеет явное преимущество. Поэтому область применения GPU бесконечна.

```
Render dml_itis@ainur (id=3):
Engine = "2.2.2
IPv4 Address: 10.150.144.52
Status: Online Busy
Priority = 99
Capacity = 100 of 1100 ( 1000 used )
Max Tasks = 10 ( 1 running )
Idle Time = Thu 01 Jun 14:11.08
Busy Time = Thu 01 Jun 14:11.08
Launched at: Tue 30 May 13:00.45
Registered at: Tue 30 May 13:00.38
Host:
OS="windows 64"
Capacity = 1100, Max Tasks = 10, Power = 1000
Idle CPU = 10%
Busy CPU = 50%
Idle Mem = 50%
Busy Mem = 90%
Idle Swap = 30%
Busy Swap = 80%
Idle HDD = 1 Gb free
Busy HDD = 1 Gb free
Idle HDD I/O = 50%
Busy HDD I/O = 95%
Network Interfaces:
{446B58DA-48E7-4803-918D-118EC0267E1C}: d4:3d:7e:dc:be:0e
IPv6 Address: fe80:::
IPv6 Address: fe80:::6c8b:69f:deb3:b4e6
IPv6 Address: ff00:::
IPv4 Address: 0.0.0.0
IPv4 Address: 10.150.144.0
IPv4 Address: 10.150.144.52
IPv4 Address: 10.150.147.255
IPv4 Address: 224.0.0.0
IPv4 Address: 255.255.255.255
```

Рис. 3. Основные ресурсы

ИНСТРУМЕНТЫ АДМИНИСТРИРОВАНИЯ И УПРАВЛЕНИЯ РАСЧЕТАМИ

В нашей работе использовалось готовое программное решение Afanasy. С его помощью мы легко создали систему распределенного рендеринга и получили доступ к таким инструментам администрирования, как:

- контроль работы для распределенных вычислений;

- отслеживание ресурсов, используемых нодами (использование центрального процессора, оперативной памяти, жесткого диска, сетевого трафика, скорости работы дисковых операций ввода-вывода (см. рис. 3); это помогает определить, что замедляет процесс визуализации);
- удаленное отслеживание выполняемых задач через Веб и стационарные интерфейсы (см. рис. 4);
- возможности учета собственных нужд программным путем.

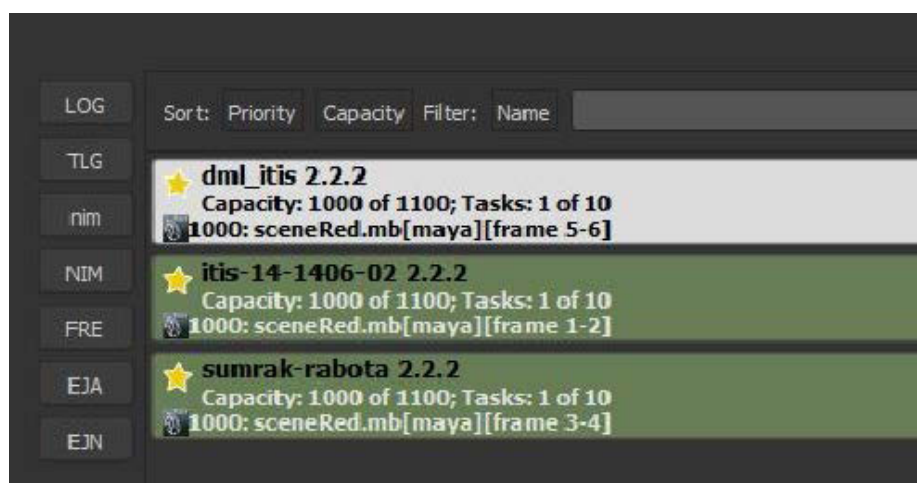


Рис. 4. Перечень задач рендеринга

НАСТРОЙКА РАБОЧЕЙ СИСТЕМЫ

Вычислительный кластер, на котором была проведена визуализация, состоит из трех нодов. На каждом из них CPU – Intel Core i7-4770 3.40GHz 8ГБ Ram и GPU – NVIDIA GeForce GTX 660 1.5 GB. На каждом были установлены программы Arnold, Redshift и программный комплекс Afanasy. После установки программ на все машины были настроены сеть и общий сетевой диск, определены сервер и клиенты. Подробный процесс настройки описан ниже.

*В корневом каталоге установленной программы запустить файл **start.cmd**. На панели задач закрепится меню программы Afanasy. Выбрать в меню **set to server**, указать **IP-адрес** сервера. Указать путь к программе Autodesk Maya, выбрав в меню **software->setup soft->maya**. Указать файл **maya.exe**.*

*Теперь можно запустить сервер. Перейти в каталог **\start\AFANASY** и запустить файл **_afserver.cmd**. Далее через меню программы Afanasy запустить **start watch**. Появится окно для мониторинга процесса рендеринга.*

Запустить в меню **submit job**. Появится окно для настройки работы, которая будет выполняться.

Запустить в меню **local render**. Добавить ноду сервер для визуализации.

Теперь сервер настроен полностью и готов выполнять работу.

Аналогично запустить на клиентах **start.cmd**. Указать **IP-адрес** сервера и путь к программе Autodesk Maya.

Запустить визуализацию **local render**. Убедиться, что в командной строке указано **render registered**. Также в окне мониторинга во вкладке **renders** появились клиенты.

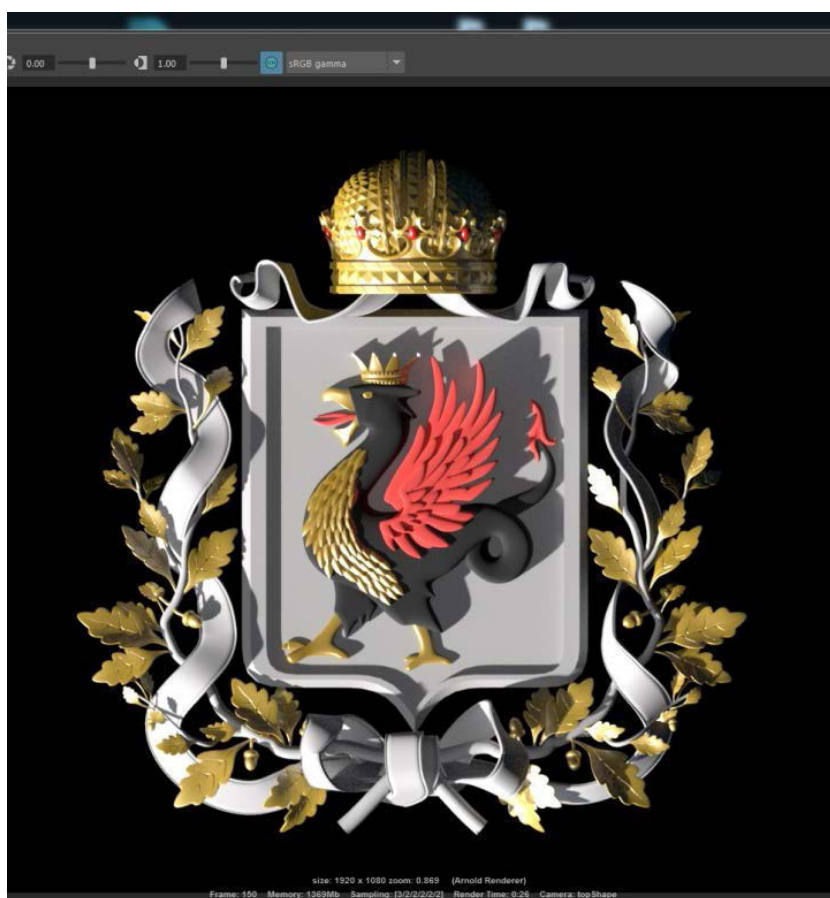


Рис. 5. Сцена с гербом после процедуры рендеринга

АНАЛИЗ ВЫЧИСЛЕНИЙ И РЕЗУЛЬТАТЫ

Оценка производительности выполнялась в несколько этапов. На каждой итерации оценивалось время, затраченное на растеризацию конечной работы. Техническая спецификация приведена в соответствующих таблицах 1 и 2.

Таблица 1. Сравнение затраченного времени на визуализацию сцены с гербом с использованием GPU и CPU

Количество полигонов	635732	635732
Разрешение	1920x1080	1920x1080
Вычислительная платформа	Intel Core i7-4770 3.40GHz 8ГБ Ram	GPU – NVIDIA GeForce GTX 660 1.5 GB 8ГБ Ram.
Визуализатор	Arnold	Redshift
Время выполнения	26 сек	9.05 сек

На первом этапе оценивался рендеринг сцены с достаточно большим количеством полигонов. Конечное изображение с гербом продемонстрировано на рис. 5. Результат рендеринга при использовании разных вычислительных решений представлен в таблице 1. Видно, что рендеринг на видеокарте опережает своего конкурента на CPU практически в 3 раза.

На втором этапе производился рендеринг сцены с содержанием большого количества флюидов на примере с торнадо. Конечное изображение продемонстрировано на рис. 6. Результат рендеринга при использовании разных вычислительных решений представлен в таблице 2.

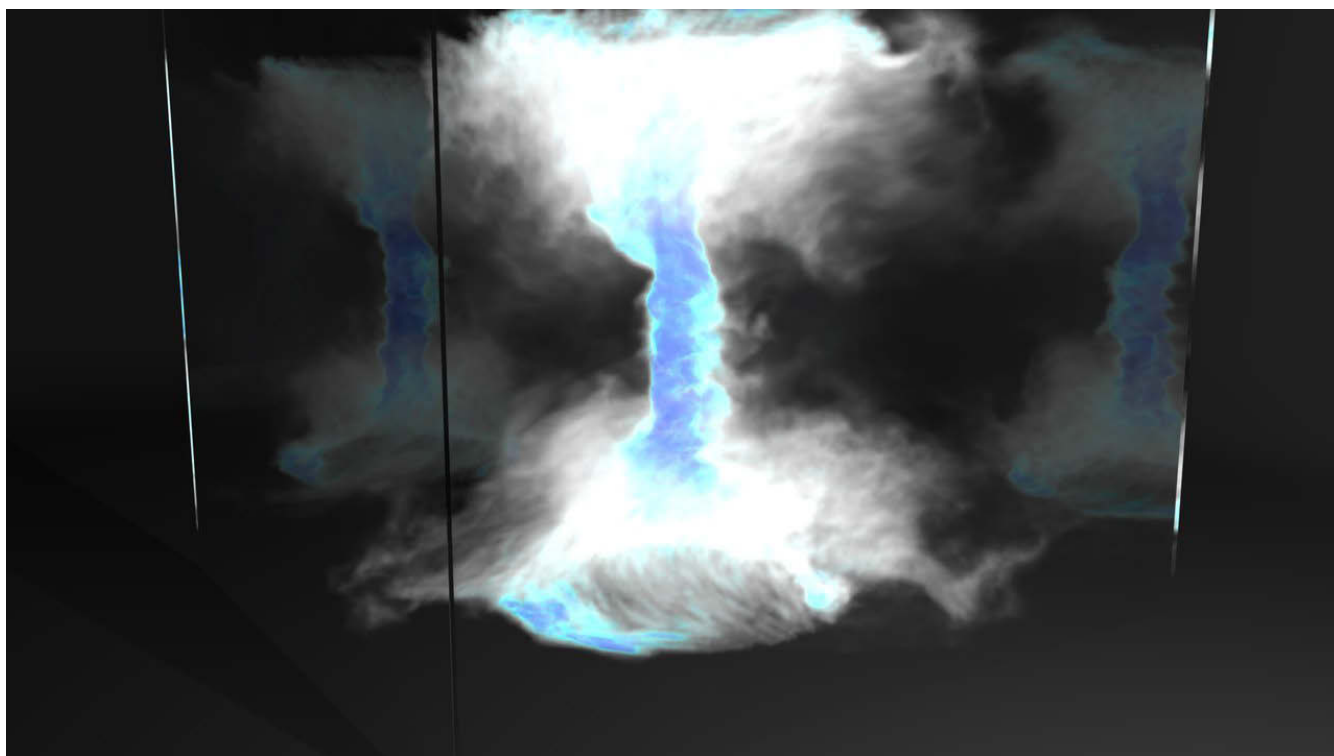


Рис. 6. Рендеринг сцены с торнадо

Таблица 2. Сравнение затраченного времени на визуализацию сцены с торнадо с использованием GPU и CPU

Количество полигонов	(Флюидные симуляции)	(Флюидные симуляции)
Разрешение	1920x1080	1920x1080
Вычислительная платформа	Intel Core i7-4770 3.40GHz 8ГБ Ram	GPU – NVIDIA GeForce GTX 660 1.5 GB 8ГБ Ram.
Визуализатор	Arnold	Redshift
Время выполнения	52 мин 56 сек	1 мин 56 сек

В очередной раз установлено, что вычисления на GPU занимают лидирующую позицию, в данном случае отрыв составляет 27 раз.

На 3-ем этапе использовали систему распределенного рендеринга для визуализации ролика длиной в 50 кадров с тем же торнадо. Результаты представлены в таблице 3. Пример работы системы мониторинга визуализации продемонстрирован на рисунках 7 и 8.

Таблица 3. Сравнение затраченного времени при распределенной визуализации сцены с торнадо с использованием GPU и CPU

Количество нодов	3	3
Разрешение	1920x1080	1920x1080
Количество фреймов	50	50
Вычислительная платформа	Intel Core i7-4770 3.40GHz 8ГБ Ram	GPU – NVIDIA GeForce GTX 660 1.5 GB 8ГБ Ram.
Визуализатор	Arnold	Redshift
Среднее время рендеринга 1 фрейма	53мин	2 мин 08 сек
Время выполнения	14 часов 42 мин 16 сек	36 мин 30 сек
Суммарное затраченное время	44 часа 12 мин 41 сек	1 час 46 мин 30 сек

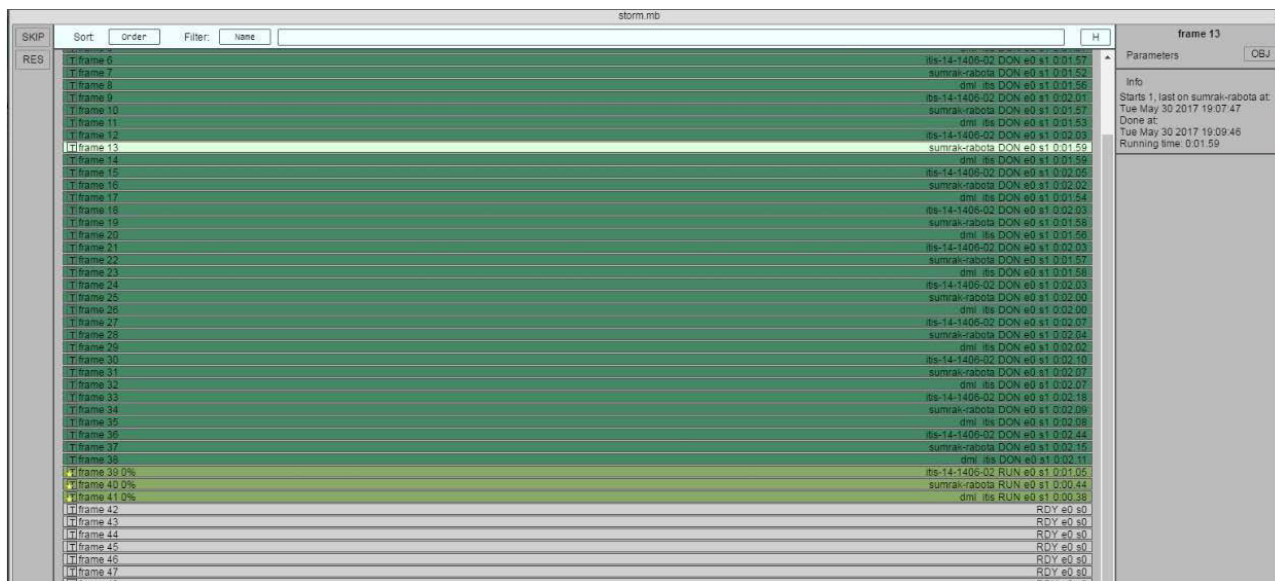


Рис. 7. Отслеживание рендеринга каждого кадра

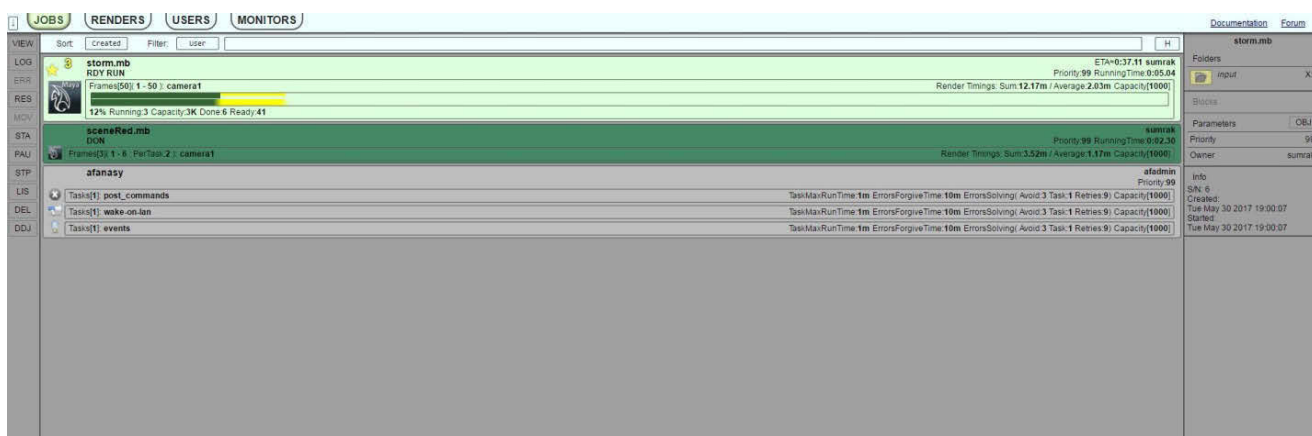


Рис. 8. Мониторинг работы

Исходя из полученных результатов, можно сделать вывод, что визуализация с использованием графического процессора в качестве вычислительной платформы позволяет сократить время на рендеринг примерно в 25 раз. Однако при визуализации ненагруженных сцен разница в скорости незначительна при двух конфигурациях. Использование технологии распределенного рендеринга позволяет сократить время, затрачиваемое на рендеринг, пропорционально количеству задействованных нод.

ЗАКЛЮЧЕНИЕ

Выполнение правильной настройки вычислительного кластера позволило значительно сократить время выполнения визуализации. При выполнении следующей визуализации не потребуется вносить какие-то кардинальные изменения в систему, нужно лишь перенести необходимые данные в общую сетевую папку.

Созданный сервис позволяет получить доступ к ресурсам рендер-фермы, эффективно и удобно взаимодействовать с ней. Он может эффективно использоваться для рендеринга анимационных роликов, высокореалистичных изображений и других проектов.

Вычислительная мощность GPU по сравнению с CPU намного выше. Наличие большего количества вычислительных ядер по сравнению с CPU и наличие собственной памяти с высокоскоростной шиной являются достоинствами графического ускорителя. Благодаря большому количеству ядер GPU выдает высокую производительность на распараллеливаемых задачах.

СПИСОК ЛИТЕРАТУРЫ

1. *Razuvalova E., Nizamutdinov A.* Virtual Reconstruction of Cultural and Historical Monuments of the Middle Volga// *Procedia Computer Science*. 2015. V. 75. P. 129–136.

2. *Shih M., Rizzi S., Insley J., Uram T., Vishwanath V., Hereld M., Papka M.E., Ma K.* Parallel Distributed, GPU-Accelerated// *Advanced Lighting Calculations for Large-Scale Volume Visualization*. 2017. P. 47.

3. ИТКН.РУ Информационные технологии и коммуникации: Вычислительные кластеры, кластер для расчета трехмерных моделей и сцен. <http://www.itkn.ru/solutions/clusters>

4. *Zheng R., Jia J., Jin H., Lv X., Yang S.* Memory-Based Data Management for Large-Scale Distributed Rendering. 2017. P. 123.

CONFIGURATION OF A COMPUTER CLUSTER FOR DISTRIBUTED RENDERING OF HIGH-REALISTIC 3D SCENES

R.R. Gazizov¹, D.I. Kostyuk², V.V. Kugurakova³

¹⁻³ Higher School ITIS. Kazan Federal University

¹starkindustries14579@gmail.com, ²xdxnkx@gmail.com, ³vlada.kugurakova@gmail.com

Abstract

The article discusses a method for implementing distributed computing using the GPU and modern administration tools and calculation management for the rendering farms. A comparative analysis is carried out between the traditional method of computing on the CPU and the modern approach to the GPU. Various configurations of the computing cluster are considered. The most productive way of rendering is determined.

Keywords: *visualization, distributed rendering, computational cluster, configuration.*

REFERENCES

1. Razuvalova E., Nizamutdinov A. Virtual Reconstruction of Cultural and Historical Monuments of the Middle Volga// Procedia Computer Science. 2015. V. 75. P. 129–136.
2. Shih M., Rizzi S., Insley J., Uram T., Vishwanath V., Hereld M., Papka M.E., Ma K. Parallel Distributed, GPU-Accelerated// Advanced Lighting Calculations for Large-Scale Volume Visualization. 2017. P. 47.
3. ITKN.RU Информационные технологии и коммуникации: Вычислительные кластеры, кластер для расчета трехмерных моделей и сцен. <http://www.itkn.ru/solutions/clusters>
4. Zheng R., Jia J., Jin H., Lv X., Yang S. Memory-Based Data Management for Large-Scale Distributed Rendering. 2017. P. 123.

СВЕДЕНИЯ ОБ АВТОРАХ



ГАЗИЗОВ Рим Радикович – лаборант-исследователь Высшей школы информационных технологий и информационных систем КФУ. Сфера научных интересов – распределенный рендеринг высокореалистичных 3D сцен и вопросы их моделирования.

Rim Radikovich GAZIZOV – assistant-researcher, KFU, Higher School of Information Technologies and Information Systems. Research interests include distributed rendering of highly realistic 3D scenes and questions of their modeling.

email: starkindustries14579@gmail.com



КОСТЮК Даниил Иванович – младший научный сотрудник, ассистент кафедры интеллектуальной робототехники. Сфера научных интересов – процедурная генерация сцен, моделирование, рендеринг.

KOSTYUK Daniil Ivanovich – junior researcher, assistant of the Department of Intelligent Robotics. Research interests include procedural scenes generation, modeling, rendering.

email: xdxnxkx@gmail.com



КУГУРАКОВА Влада Владимировна – старший преподаватель Высшей школы информационных технологий и информационных систем, руководитель лаборатории «Виртуальные и симуляционные технологии в биомедицине». Сфера научных интересов – реалистичность визуализации и симуляций, иммерсивность VR.

Vlada Vladimirovna KUGURAKOVA, Senior Lecturer of Higher School of Information Technology and Information Systems, Head of Laboratory “Virtual and simulation technologies in biomedicine”. Research interests include realism of visualization and simulation, immersion VR.

email: vlada.kugurakova@gmail.com

Материал поступил в редакцию 30 мая 2017 года