

УДК 004.75

А. А. Беляева, Е. В. Биряльцев, М. Р. Галимов, Д. Е. Демидов,  
А. М. Елизаров, О. Н. Жибрик

## **Кластерная архитектура программно-технических средств организации высокопроизводительных систем для нефтегазовой промышленности<sup>1</sup>**

**АННОТАЦИЯ.** Проанализирован процесс проектирования и создания опытного комплексного программно-аппаратного решения организации высокопроизводительных вычислений, обеспечения хранения больших данных и трехмерной визуализации в реальном времени для обеспечения производственных процессов в нефтегазовой отрасли. Приведены промежуточные технические решения и результаты, полученные в процессе работ по проектированию соответствующего специализированного комплекса, рассмотрены проблемы и направления дальнейшего развития названного технологического направления.

*Ключевые слова и фразы:* численное моделирование, распределенные вычисления, распределенное хранение, нефтегазовый комплекс, платформа высокопроизводительных вычислений

### **Введение**

Современные успехи в развитии высокопроизводительных аппаратных средств и программных технологий позволяют считать задачу создания технической базы массового суперкомпьютеринга для промышленного применения [1] выполненной, что способствует внедрению в традиционное промышленное производство таких ресурсоемких технологий, как численное моделирование. Одной из массовых отраслей промышленности, в которой методы численного моделирования актуальны уже в настоящее время, является нефтегазовая промышленность.

Усложнение геологического строения месторождений, вводимых в промышленную эксплуатацию на фоне снижения цен на углеводороды, требует повышения эффективности геологоразведочных работ и

---

<sup>1</sup> Работа выполнена при финансовой поддержке РФФИ и Правительства Республики Татарстан в рамках научных проектов 15-07-08380, 15-47-02343

© А. А. Беляева, Е. В. Биряльцев, М. Р. Галимов, Д. Е. Демидов, А. М. Елизаров, О. Н. Жибрик, 2017

© ООО «Градиент технологий», ЗАО «Градиент», НИИСИ РАН, КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ, 2017

© [Программные системы: теория и приложения](#), 2017

снижения рисков бурения и эксплуатации. Одной из наиболее перспективных технологий, которая решает названные задачи, является полноволновая инверсия сейсмических данных. В отличие от технологии общей глубинной точки, применявшейся на протяжении 50 лет, технология полноволновой инверсии работоспособна в геологических условиях любой сложности и позволяет непосредственно получить данные о механических свойствах геологической среды, что необходимо для обоснованного проектирования схемы разработки и конструкции скважин.

Полноволновая инверсия основана на подборе такого распределения механических параметров исследуемого объема геологической среды, которые, при решении прямой задачи распространения сейсмических волн, дают модельный сигнал, минимально расходящийся с полевыми наблюдениями. Подбор характеристик среды производится, как правило, ньютоновскими или квазиньютоновскими оптимизационными алгоритмами, минимизирующими функционал расхождения полевых и модельных сигналов. Для работы оптимизационных алгоритмов необходимо вычислять гессиан или, как минимум, градиент функционала расхождения по параметрам среды. Количество таких параметров среды, определяющее размерность задачи, достигает нескольких десятков тысяч. В силу значительной «овражности» оптимизируемого функционала, оптимизационная процедура, как правило, сходится к решению лишь за несколько сотен шагов. Таким образом, для проведения полноволновой инверсии необходимо итерационно решать несколько десятков тысяч относительно малоразмерных прямых задач распространения сейсмических волн.

Полноволновая инверсия является развивающейся технологией, и в ее практическом внедрении существенную роль играют малые и средние инновационные геофизические компании с ограниченными финансовыми и техническими ресурсами. Таким компаниям малодоступны «тяжелые» высокопроизводительные системы. Ниже представлены техническая и программная архитектуры, доступные названным компаниям и позволяющие промышленно решать не только задачи полноволновой инверсии, но и такие сходные по системным требованиям и востребованные в текущих условиях задачи, как оптимизация схемы разработки и управление рисками при бурении, требующие решения множества малоразмерных прямых задач.

## 1. Формулировка проблемы

Рассмотрим, какие основные требования предъявляются сегодня к комплексной высокопроизводительной системе. Эти требования сформировались на основе нашего опыта практической эксплуатации высокопроизводительных систем в геофизической компании ( см. [2–4]).

Основные функциональные требования:

- обеспечение эффективных высокопроизводительных вычислений для решения задач численного моделирования и статистической обработки сигналов;
- надежное хранение и управление данными большого объема (десятки и сотни терабайт), включающими наборы полевых записей геофизического оборудования, результаты численного моделирования в виде статических и динамических кубов данных сложной природы, вплоть до тензоров, и реляционной базой данных, содержащей учетную и справочную информацию;
- организация коллективной работы с возможностью удаленного доступа заказчиков, контрагентов, удаленно работающих специалистов компании по сетям общего пользования, обеспечивающая визуализацию объемных двухмерных и трехмерных сцен, в том числе динамических.

Дополнительные требования:

- возможность гибкого масштабирования и реконфигурации системы, в том числе при изменении характера нагрузки;
- возможность интеграции в общий бизнес-процесс унаследованного и покупного программного обеспечения, в том числе работающих в системе Windows ранних версий;
- обеспечение высокой надежности системы, отсутствие единой точки отказа;
- минимально возможная сложность и стоимость технических и программных решений и их сопровождения.

На современном техническом уровне развития программно-технических средств каждое из названных основных функциональных требований обеспечивается использованием апробированных технических решений.

Высокопроизводительные вычисления для решения задач численного моделирования и статистической обработки эффективно

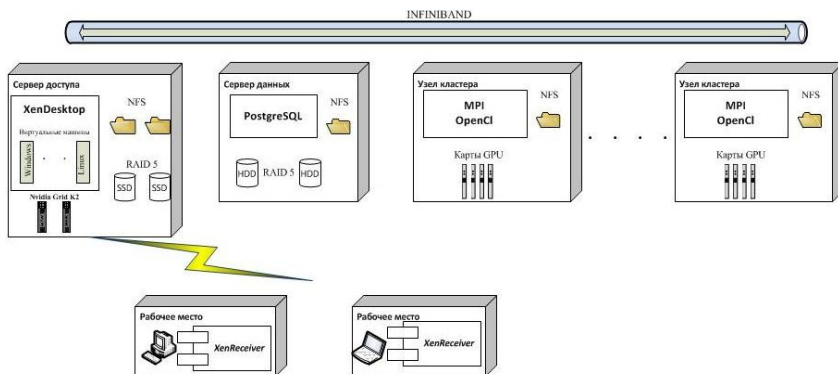
решаются на локальных и кластерных системах с использованием графических процессоров (GPU).

Надежное хранение больших объемов данных обеспечивается архитектурами сетевого хранения данных SAN (Storage Area Network), а при необходимости быстрого доступа к данным архитектурой DAS (Direct Attached Storage) с организацией дисковых массивов уровня RAID 5, 6 и выше.

Доступ по сетям общего пользования к данным большого объема, когда их передача клиенту из мест хранения невозможна, обеспечивается виртуализацией рабочих столов. При необходимости выполнения трехмерной или «тяжелой» двухмерной визуализации больших данных, виртуальный рабочий стол может быть установлен с использованием технологий графических карт на стороне сервера, например по технологии nVidia GRID.

Технологии виртуализации позволяют также решить задачу интеграции унаследованного и покупного программного обеспечения за счет организации на виртуальной машине необходимого системного окружения.

Программно-технический комплекс, основанный на названных технических решениях, был реализован с участием авторов, и эксплуатируется в течение двух лет, что было, в частности, отмечено ведущими производителями графических ускорителей [6-7]. Общая архитектура технического и программного обеспечения показана на рис 1.



**Рисунок 1. Архитектура кластера с выделенными серверами доступа и данных**

Опыт эксплуатации этого комплекса показал принципиальную работоспособность выбранных технических решений для промышленной эксплуатации.

Вместе с тем были выявлены следующие проблемы.

*В примененной архитектуре присутствуют единые точки отказа, каковыми являются сервер баз данных и сервер доступа.*

- примененная для надежного хранения организация дисков в RAID-массивы в целом не страхует от выхода узла из строя;
- аналогичная ситуация с сервером доступа: дублирование разделяемых графических карт nVidia K2 в целом не гарантирует от выхода из строя сервера доступа.

*Использование оборудования не является оптимальным.*

- в силу поэтапной организации работ, когда численное моделирование составляет только один из этапов, узлы с GPU и сервер доступа оказываются циклически перегруженными или недогруженными; при этом загрузки вычислительных узлов и сервера приложений находятся в противофазе, при обработке данных перегружены узлы кластера, а при интерпретации перегружен сервер доступа;
- процессорные мощности сервера данных и узлов кластера недогружены: отсутствуют сложные аналитические запросы к серверу данных, а в узлах кластера основные вычисления вынесены на графические устройства, поэтому универсальные процессоры фактически загружены обменными операциями.

- при применении RAID-массивов объемные временные данные, в частности промежуточные результаты численного моделирования, хранятся с избыточным резервированием.

*Масштабирование сервера доступа и хранилища данных затруднено: количество дисков и графических карт на один сервер ограничено, для масштабирования требуется достаточно дорогостоящее наращивание на уровне специализированных узлов.*

*Техническая и программная архитектуры сложны и требуют сопровождения коллективом специалистов с специальной квалификацией.*

## **2. Предлагаемое решение**

Как видим, основной причиной возникающих проблем является наличие в архитектуре высокопроизводительного комплекса выделенных узлов. Джим Грей [5] сформулировал подход к технической и программной архитектуре массового суперкомпьютинга, который основан на массиве однородных, легко заменяемых и наращиваемых «вычислительных кирпичей» (своего рода RAID-суперкомпьютер, в котором, благодаря однородности используемых аппаратных узлов, легко выполняются требования надежности, масштабируемости и балансировки решаемых задач).

С точки зрения аппаратной архитектуры типовой узел должен содержать вычислительные мощности, устройства хранения данных и графическую подсистему. При решении задач численного моделирования привлекательным представляется совмещение в графических устройствах узла как средств визуализации трехмерных данных, так и средств высокопроизводительных вычислений. Узлы должны быть объединены высокоскоростной сетью, необходимой как для эффективного использования вычислительных мощностей, так и для высокоскоростного доступа к распределенным данным.

Для обеспечения всего комплекса названных требований программная архитектура должна базироваться на технологии виртуализации. Основой системного программного обеспечения является установленный на всех узлах гипервизор с набором виртуальных машин, в которые по мере необходимости загружаются

операционные системы (ОС) узлов виртуального кластера (они могут использовать графические устройства узла), либо пользовательские ОС, (в них функционирует клиентское программное обеспечение, в том числе покупное и унаследованное). Также на узлах могут быть организовны серверные виртуальные машины, обеспечивающие работу файл-сервера, СУБД и сервера приложений. В такой архитектуре емкости хранения отдельных узлов объединяются сетевой файловой системой в единое пространство хранения.

Структурно описанная схема программно-технического обеспечения представлена на рис 2.

Концептуально данное решение устраняет выявленные проблемы, однако практическая его реализация наталкивается на несколько технических препятствий. В настоящий момент времени практически отсутствуют предложения готовых универсальных программно-аппаратных продуктов, которые позволяли бы быстро разворачивать высокопроизводительные системы и легко адаптировать их к имеющимся требованиям. В основном предлагаемые решения ориентированы лишь на одну из составляющих таких систем (хранение, визуализация или вычисления) с неочевидными ограничениями и особенностями функционирования и проблемами совместимости.

Ниже описаны результаты выбора и анализа совместимого комплекса технических и программных средств, реализующих RAID-суперкомпьютер.

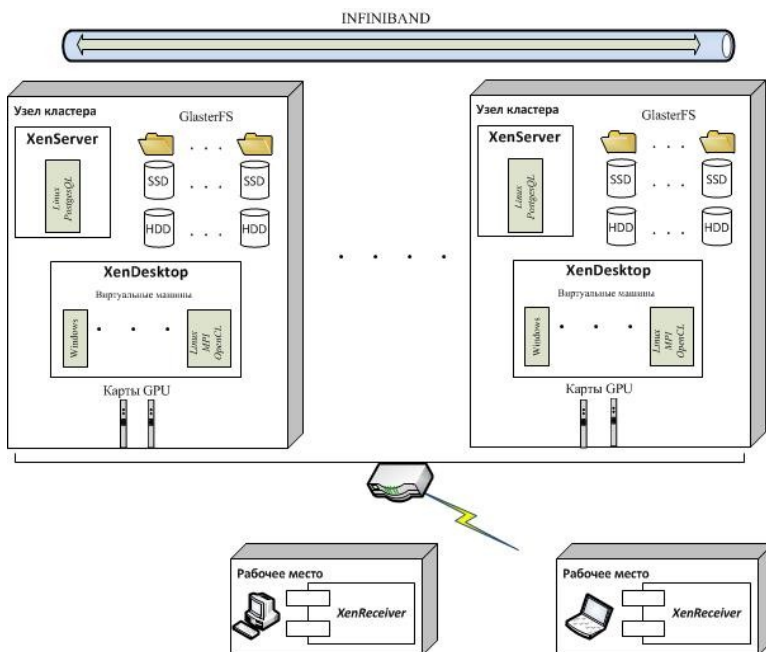


Рисунок 2. Архитектура кластера с однородными узлами

### 3. Аппаратная архитектура

При определении архитектуры технических средств производился выбор и обоснование следующих компонент.

*Тип используемых графических карт.* Как было сказано выше, для достижения полной универсальности узлов использование графических карт должно быть возможным в двух режимах - визуализации трехмерной графики и вычислений. На определенном этапе развития технологий GPU игровые графические карты поддерживали вычисления с двойной точностью и объемом памяти, сопоставимой с картами, ориентированными на вычисления. Последней картой такого типа была AMD HD 7970. Затем стала проявляться специализация, графические карты, ориентированные на визуализацию, стали оснащаться урезанной подсистемой расчетов с двойной точностью. Карты, ориентированные на вычисления и обеспечивающие вычисления с двойной точностью, были



выпущены компаниями AMD и nVidia с увеличенным объемом графической памяти (в настоящее время до 32 Гбайт с тенденцией к дальнейшему увеличению). В рамках технологии nVidia GRID были выпущены карты, поддерживающие коллективное использование и в режиме визуализации (K1 и K2) и в режиме вычислений CUDA (M10 и M60).

Особенностью рассматриваемых задач численного моделирования является относительно небольшая размерность решаемых задач, объем данных которых позволяет поместить их в память игровых карт. Также хорошо известно, что максимальная скорость вычислений на графических устройствах достигается на задачах большой размерности. Возникает вопрос, насколько графические карты, ориентированные на визуализацию, отличаются по скорости расчетов на задачах малой размерности от специализированных графических ускорителей.

**Таблица 1. Производительность графических карт на задачах малой размерности**

Тест	HD 7970 (Tahiti)	R9 NANO (Fuji)	W9100 (Hawaii)	Xeon 1630v4
Явный метод 280x280x140 элементов, 10 <sup>3</sup> шагов	1261 сек	418 сек	666 сек	1154 сек
Неявный метод, 128x128x128 элементов, 1 шаг	1.034 сек	0.717 сек	1.151 сек	3.046 сек
Метод Верле, 20 000 элементов, 10 <sup>4</sup> шагов	42 сек	67 сек	60 сек	128 сек
Обращение плотной матрицы 2000x2000	27 сек	18 сек	23 сек	148 сек

Было произведено сравнение производительностей профессиональной GPU-карты AMD W9100, универсальной карты AMD HD 7970 и современной карты AMD R9 Nano, ориентированной на визуализацию. В таблице 1 приведено сопоставление времени выполнения некоторых реальных задач малой размерности (под малой размерностью мы понимаем задачу, которая может выполняться на одной карте). Как видим, на задачах малой размерности преимущество W9100 перед HD 7970 совсем невелико, а R9 NANO в некоторых задачах даже превосходит профессиональную карту предыдущей архитектуры.

Таким образом, для рассматриваемого класса задач возможно использование в качестве вычислительных элементов карт, ориентированных на визуализацию.

*Платформа.* В рамках выбранной концепции принято решение отказаться от достижения максимально эффективной плотности производительности на узел, что достигалось ранее установкой на двухпроцессорный сервер большого количества (4 и более) GPU-карт. Однако такое решение увеличивает стоимость отдельного узла, зависимость функционирования всего кластера от его состояния, снижает возможности быстрого расширения кластера за счет закупки новых узлов из-за их высокой стоимости, и, наконец, приводит к неравномерному распределению мощности между CPU-процессорами и GPU-картами из-за большего количества последних (так как использование GPU, например, в подсистеме хранения все еще не нашло широкого применения, что означает снижение доступной мощности кластера для данной подсистемы). В качестве ограничивающего фактора выступает также количество линий PCI. Шина PCI создает в настоящее время "бутылочное горлышко" в общей производительности системы с GPU. Пропускная способность шины значительно ниже как скорости графической карты, так и скорости доступа к ОЗУ.

На рис 3 приведен график скорости доступа к памяти одно и двухпроцессорных систем при копировании массива объемом 100 мегабайт, что типично для решаемой задачи. Видно, что скорость доступа к памяти на 1 поток во всех случаях выше, чем пропускная способность шины PCI.

Для достижения максимальной производительности желательно выделять на карту 16 линий PCI. С учетом наличия в современных и перспективных процессорах 40 линий PCI, количество GPU-карт на одном узле было ограничено двумя единицами. Это решение, несмотря

на увеличение числа необходимых узлов, обеспечивает ряд других требований, в частности, доступность равной мощности для различных подсистем, удешевление единичного узла, масштабируемость решения.

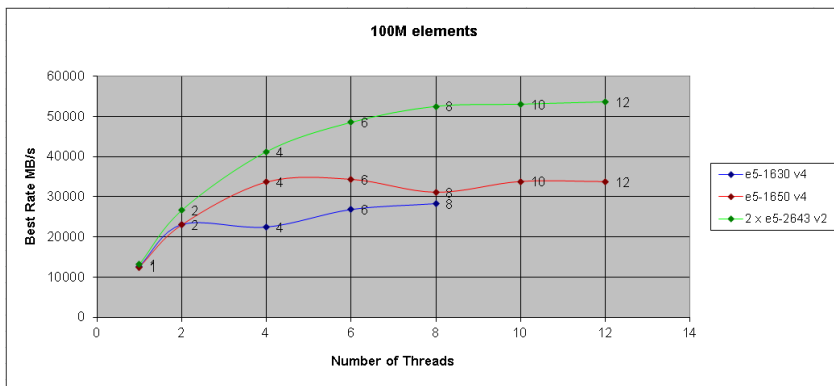


Рисунок 3. Скорость доступа к оперативной памяти

*Организация хранения данных.* Использование выделенных узлов для хранения данных диктует (для повышения эффективности использования аппаратной базы) включение в состав комплекса максимального числа дисков, объединенных, для надежности, в дисковые RAID-массивы. Как отмечено в разделе 2, такое решение приводит, с одной стороны, к определенной избыточности, а с другой стороны, не устраняет единую точку отказа. Кроме того, восстановление RAID-массива большого объема происходит достаточно длительное время, в течение которого общая производительность подсистемы хранения снижается. Централизованная система хранения также является «бутылочным горлышком» при одновременном сохранении результатов моделирования со многих расчетных узлов кластера.

В рамках концепции однородной вычислительной среды предполагается децентрализованное хранение данных на всех узлах кластера. С аппаратной точки зрения это позволит обеспечить большую производительность и масштабируемость. Кроме того, программное обеспечение децентрализованного хранения, как будет показано ниже, позволяет обеспечить регулируемую пообъектную надежность хранения

данных, что делает нецелесообразным объединение локальных дисков в RAID-массив.

Подсистема хранения должна обеспечивать как большую скорость последовательного доступа, так и малую латентность при произвольном доступе. Высокая скорость последовательного доступа в рамках распределенной архитектуры обеспечивается параллельной записью/чтением на множестве устройств хранения, в том числе с достаточно медленным доступом к единичному устройству и наличием высокоскоростной коммуникационной подсистемы. Низкая латентность, требуемая при работе с базой данных, может быть обеспечена наличием в системе твердотельных накопителей SSD либо организацией RAM-дисков, копируемых на жесткий диск и обратно в оперативную память при начале и конце работы. Последнее решение целесообразно при наличии очень интенсивной работы с дисками. Для решения рассматриваемых задач подобная интенсивность работы пока не требуется, что позволяет ограничиться наличием в системе регулируемого числа SSD.

#### **4. Система виртуализации**

Такая система должна обеспечивать:

- динамическую реконфигурацию количества и типа виртуальных машин, в том числе поддерживать клиентские ОС Windows различных версий;
- использование графических устройств узлов кластера в режимах вычислений, т.е. допускать работу с технологиями OpenCL и применять эти же карты в режиме удаленной аппаратной трехмерной виртуализации с использованием технологий OpenGL, DirectX;
- поддержку высокопроизводительных сетей Infiniband.

Основываясь на в целом положительном опыте эксплуатации системы виртуализации компании Citrix (XenServer/XenDesktop), в режиме выделенного сервера доступа, для новой архитектуры мы также выбрали продукты Citrix.

Проведенные вычислительные эксперименты показали, что данная система в целом удовлетворяет большинству требований, изложенных выше.

Ранее в системе с выделенным сервером доступа мы использовали карты K2 и технологию nVidia GRID для обеспечения виртуализации приложений с «тяжелой» графикой. В рамках реализации проекта с

однородными узлами мы отказались от использования «тяжелых» разделяемых решений. Для обеспечения визуализации и расчетов с использованием графических карт была использована технология проброса карты в ОС виртуальной машины. Проведенные эксперименты показали возможность данной операции как в режиме визуализации, так и в режиме вычислений. Вычислительные эксперименты показали, что снижение производительности составляет 5-7% на типовых задачах, что представляется вполне допустимым.

Одной из проблем использования продуктов компании Citrix является недостаточная поддержка сети Infiniband. В [3] нами отмечена проблема с подключением сервера виртуализации к вычислительным узлам и серверу БД через Infiniband FDR, однако ее удалось решить. В однородном кластере мы использовали Infiniband QDR, и пока не удается связать все узлы сетью Infiniband, однако надежда решить эту проблему остается.

Установка гипервизора на все узлы позволяет чисто программно конфигурировать виртуальные кластеры, содержащие необходимое количество расчетных узлов с запуском на них требуемых операционных систем и другого системного окружения, что важно при использовании покупного и унаследованного программного обеспечения.

Также данная архитектура позволяет разворачивать на том же аппаратном обеспечении клиентские приложения, в том числе, использующие «тяжелую» графику. Клиентское приложение запускается на загружаемой виртуальной машине с требуемой для него операционной системой.

Реконфигурация виртуальных кластеров и рабочих станций производится чисто программно, что способствует оптимальному использованию оборудования кластера. При необходимости можно использовать графические карты либо в виртуальных счетных узлах, либо - в клиентских приложениях.

Особенностью описанной архитектуры является то, что практически отсутствует возможная единая точка отказа, т. е. выход из строя отдельного аппаратного средства или сразу нескольких серверов не является критическим событием (при корректно настроенном программном обеспечении хранения данных верхнего уровня), потому что необходимые функции могут быть оперативно переданы другим серверам за счет динамического перераспределения виртуальных машин в условиях сократившихся аппаратных ресурсов. Такой вариант решения также позволяет быстро подключать новые мощности при их недостатке

или динамически изменять распределение аппаратных узлов между подсистемами для выравнивания нагрузки между ними.

## 5. Система хранения данных

Принципиально новым компонентом представленной разработки программного комплекса, не апробированном в предыдущих решениях, стала система управления распределенными данными, в том числе файловая система и СУБД.

### *Файловая система*

Ранее мы отдавали предпочтение сетевой файловой системе NFS из-за доступности развертывания и эксплуатации, а надежность хранения данных обеспечивалась за счет аппаратных RAID-массивов. При этом на узлах кластера локальная дисковая подсистема отсутствовала. Однако переход к концепции универсального кластера потребовал отказа от централизованной модели хранения в пользу распределенной модели. Также было выявлено, что RAID-системы большой емкости обладают слишком большим временем восстановления, узлы хранения являются единой точкой отказа, имеются проблемы с масштабированием. В связи с этим в рамках универсального кластера мы рассмотрели возможность использования распределенных файловых систем.

Для организации распределенного хранилища данных были проанализированы наиболее популярные варианты – распределенные файловые системы Lustre, Ceph и Gluster. Выбор производился на основе соответствия следующим критериям:

- производительность и масштабируемость;
- отказоустойчивость;
- использование типового оборудования;
- децентрализованность;
- простота развертывания и эксплуатации;
- наличие механизмов дедупликации и сжатия данных;
- прозрачность клиентского доступа.

Каждое из названных решений обеспечивает высокие производительность и масштабируемость. Системы Ceph и GlusterFS по умолчанию поддерживают механизмы репликации, в отличие от системы Lustre, в которой избыточность хранения рекомендуется организовывать на аппаратном уровне. Следовательно, файловая система Lustre

выступает, скорее, как надежный механизм организации скоростного доступа к данным, например, в вычислительных кластерах, нежели как механизм создания программных хранилищ данных.

Система Ceph представляет собой RADOS-based (Reliable Autonomic Distributed Object Store) платформу для организации распределенного хранилища (Ceph Storage Cluster) и предоставляет три интерфейса для работы с ним: CephFS, Ceph Block Device и Ceph Object Gateway, которые в свою очередь используют клиентские узлы.

Система Ceph Object Gateway не представляет интереса в контексте проектируемого кластера в силу своего функционала.

Мы исследовали возможность работы с CephFS на модельных задачах и не выявили проблем в эксплуатации. Однако по заявлениям разработчиков текущая версия CephFS является недостаточно стабильной для использования в промышленных масштабах, ее применение несет неконтролируемые риски.

Ceph Block Device организует доступ к хранилищу в виде виртуальных блочных устройств и пока представляется наиболее подходящим решением. Блочные устройства Ceph поддерживают тонкое резервирование, динамическое изменение размера и чередование (striping) между физическими узлами хранилища. Кроме того, эта система наследует все возможности RADOS, включая снимки данных, репликацию и согласованность данных (consistency).

Однако, Ceph Block Device не занимается организацией многопользовательского доступа к файлам и, таким образом, предполагает развертывание распределенной файловой системы поверх созданных блочных устройств.

GlusterFS является полноценной распределенной файловой системой, обеспечивающей реально многопользовательский доступ к файловому хранилищу. Доступ к этой файловой системе из Linux-систем организуется с помощью Gluster Native Client, из Windows-систем – с помощью сервиса Samba. GlusterFS является FUSE-файловой системой, что, с одной стороны, значительно упрощает ее использование на клиентских узлах, вплоть до монтирования через NFS v3; с другой стороны, это свидетельствует о наличии всех недостатков, присущих FUSE, главным из которых является снижение производительности.

На данный момент времени ни Ceph, ни GlusterFS не имеют механизмов дедупликации данных. Тем не менее, дедупликация на уровне файлов перечислена в списке идей проектов, предложенных Gluster в качестве пригодных для реализации в рамках производственных практик, например, GSOC, в то время как в системе Ceph, согласно

комментариям разработчиков, не планируется в обозримом будущем реализация данного функционала.

В настоящий момент времени окончательный выбор между рассмотренными системами не сделан – обе системы изучаются на тестовом полигоне на устойчивость и производительность. Окончательный выбор можно будет сделать лишь после более длительной эксплуатации названных систем. Кроме того, большое значение будет иметь эффективность использования файловой системы в качестве базовой для используемой базы данных.

### ***СУБД***

Очевидным решением видится переход на использование кластерных СУБД, т. е. систем, распределенных по массивам узлов. Такие системы позволяют хранить значительные объемы данных, увеличивать надежность и доступность информации за счет использования механизмов репликации, а также повышать скорость обработки с помощью технологии шардинга. Однако необходимость работы с унаследованными системами, реализованными на традиционных клиент-серверных архитектурах, требует рассматривать возможность перехода с классических СУБД на кластерную версию той же СУБД. Для систем, использующих СУБД, которые не имеют кластерных версий, организация работы возможна только путем запуска виртуальной машины с требуемой СУБД, использующей в качестве хранилища сетевую файловую систему.

Мы рассмотрели оба варианта миграции на кластерную архитектуру на примере СУБД PostgreSQL.

PostgreSQL имеет кластерную версию Postgres-XL. Как правило, текущая версия Postgres-XL основана на последней версии PostgreSQL с отставанием в несколько месяцев. С точки зрения пользователя, Postgres-XL выглядит как один инстанс БД, т. е. все запросы на клиентской стороне идут через стандартное подключение. Архитектурно Postgres-XL состоит из трех типов компонентов: глобальный монитор транзакций (GTM), координатор (coordinator) и узел данных (datanode).

На тестовой площадке в виртуальной распределенной среде был развернут тестовый кластер Postgres-XL, который был интегрирован с унаследованной системой управления информацией. В результате экспериментов с этой СУБД сделаны следующие выводы:

- развертывание системы не представляет значительных сложностей;



- интеграция системы с унаследованным программным обеспечением может потребовать существенной переработки последнего.

Были выявлены следующие проблемы интеграции с унаследованным ПО: отсутствие поддержки необходимого типа данных Large Objects и конфликты с ORM-технологией Hibernate. Отметим, что применение указанных типа данных и технологии доступа к ним достаточно широко распространено, а отсутствие их поддержки «из коробки» является критичным.

Были также исследованы возможности функционирования классической СУБД PostgreSQL поверх файловой системы GlusterFS. На виртуальную машину с Linux была установлена СУБД PostgreSQL, файлы базы данных которой размещались в файловой системе GlusterFS, доступ к которой осуществлялся через Gluster Native Client.

Это решение показало принципиальную работоспособность и приемлемую производительность. Детальное исследование производительности будет проведено при окончательном запуске конфигурации кластера, с осуществлением обмена между узлами по Infiniband.

Были также исследованы возможности организации «холодного» резерва на случай отказа узла с виртуализованной СУБД. На двух узлах были развернуты основной и резервный инстансы PostgreSQL, настроенные на одни и те же файлы базы данных, расположенные на GlusterFS. На основном инстансе эмулировалась работа клиента с базой данных, после чего узел отключался и поднимался резервный инстанс во втором узле. После переключения клиента СУБД на резервный инстанс работа последнего была продолжена без сбоев. Таким образом, проблема СУБД, как единой точки отказа может быть решена за счет «холодного» резервирования, со временем переключения порядка нескольких минут после выявления проблемы, что вполне допустимо для решаемого класса задач.

## **Заключение**

Проведены работы по проектированию опытного высокопроизводительного комплекса на основе принципа использования универсальных вычислительных модулей (аппаратных серверов) с переносом всех основных программных функциональных компонент в виртуализированную среду.

Общая аппаратная и программная (на системном уровне) архитектура реализованного комплекса выглядит следующим образом:

- аппаратная база – однотипные сервера с несколькими GPU-картами для организации вычислений и визуализации и дискретными HDD и SSD дисками для хранения данных;
- на каждом из серверов установлено программное обеспечение организации среды виртуализации Citrix Xen Server с созданием общего пула серверов;
- каждый сервер виртуализации обеспечивает функционирование нескольких виртуальных машин, специализация которых (организация вычислений, хранения или визуализация) определяется в зависимости от оперативных задач и может изменяться динамически.

Проведенные эксперименты показали возможность комплексирования существующего аппаратного и программного обеспечения для реализации вычислительного кластера на базе однородных модулей с возможностью динамического распределения ресурсов под виртуальные вычислительные кластеры и виртуальные рабочие станции с возможностью удаленного доступа к ним через сети общего пользования.

Установлено также, что основные проблемы в промышленном использовании подобных архитектур заключаются в слабой поддержке средой виртуализации высокопроизводительных сетей Infiniband, а также в отсутствии кластерной файловой системы, обеспечивающей выполнение всех требований к ней, в частности, поддержки дедубликации.

Дальнейшие направления исследований данной архитектуры мы видим в изучении возможностей новых технических и программных решений для оптимизации функциональных возможностей, производительности и процессов сопровождения однородного кластера.

## Список литературы

[1] Бетелин В. Б., Велихов Е. П., Кушниренко А. Г. Массовые суперкомпьютерные технологии – основа конкурентоспособности национальной экономики в XXI веке // Информационные технологии и вычислительные системы. 2007. № 2. С. 3–10

[2] Биряльцев Е. В., Богданов П. Б., Галимов М. Р., Демидов Д. Е., Елизаров А. М. Программно-техническая платформа высокопроизводительных вычислений для нефтегазовой промышленности // Программные системы: теория и приложения: электрон. научн. журн. 2016. Т. 7, № 1 (28), С. 15–27.

[3] Биряльцев Е. В., Богданов П. Б., Галимов М. Р., Демидов Д. Е., Елизаров А. М. Опыт разработки и эксплуатации суперкомпьютерного комплекса для решения обратных задач сейсморазведки. Вычислительные технологии в естественных науках// Методы суперкомпьютерного моделирования. Сборник трудов. 17–19 октября 2015 года, Россия, Таруса. Под редакцией Р. Р. Назирова, Л. Н. Щура, 2015, Ч. 3. С. 18–33.

[4] Биряльцев Е. В., Галимов М. Р. Системотехнические проблемы применения методов математического моделирования в промышленности на примере новых сейсмических технологий// Семинар OS Day/TMPA-2015 «ОС реального времени», г. Иннополис (9–10 июня 2015 года). URL:<http://osday.ru/biryaltsev.html#speaker>


[5] Hey T., Tansley S., Tolle K. The Fourth Paradigm: Data-Intensive Scientific Discovery, - 2009.

[6] Innovative Graphics Compute Helps Oil and Gas Industry. AMD Business blog, 01.04.2016, URL:<https://community.amd.com/community/amd-business/blog/2016/04/01/innovative-graphics-compute-helps-oil-and-gas-industry>

[7] Облачная графика улучшает геологоразведке доступ к сложнодоступным регионам. Новости NVidia, URL:<http://www.nvidia.ru/object/gradient-case-studies-ru.html>

*Об авторах:*

	<p><b>Анастасия Алексеевна Белева</b>          Специалист в области развертывания и поддержки системного программного обеспечения, ведущий инженер-программист ЗАО «Градиент»,  <i>e-mail:</i> <a href="mailto:anastasia.blv@gmail.com">anastasia.blv@gmail.com</a></p>
	<p><b>Евгений Васильевич Биряльцев</b>          Специалист в области специализированных информационных систем, к.т.н., автор более 50 публикаций, в том числе 3 свидетельств о регистрации программ, 2 изобретений. Генеральный директор компании ООО «Градиент технолоджи» (резидент Сколково).  <i>e-mail:</i> <a href="mailto:Igenbir@yandex.ru">Igenbir@yandex.ru</a></p>
	<p><b>Марат Разифович Галимов</b>          Специалист в области разработки программного обеспечения для нефтегазовой отрасли, к.т.н., заместитель директора ООО «Градиент технолоджи»  <i>e-mail:</i> <a href="mailto:glmvmt@gmail.com">glmvmt@gmail.com</a></p>
	<p><b>Денис Евгеньевич Демидов</b>          Специалист в области высокопроизводительных вычислений с использованием технологий GPGPU, к.ф.-м.н, с.н.с НИИСИ РАН(КазФ МСЦ РАН).  <i>e-mail:</i> <a href="mailto:dennis.demidov@gmail.com">dennis.demidov@gmail.com</a></p>
	<p><b>Александр Михайлович Елизаров</b>          Доктор ф.-м. н., профессор Казанского (Приволжского) федерального университета, Заслуженный деятель науки Республики Татарстан, директор Казанского филиала НИИСИ РАН (КазФ МСЦ РАН), член Американского математического общества (AMS), Немецкого общества математиков и механиков (GAMM) и Международного общества по индустриальной и прикладной математике (SIAM). Автор более 200 публикаций, в том числе 12</p>

	монографий. <i>e-mail: <a href="mailto:amelizarov@gmail.com">amelizarov@gmail.com</a></i>
	<b>Ольга Николаевна Жибрик</b> Специалист в области разработки программного обеспечения для нефтегазовой отрасли, ведущий инженер-программист ООО «Градиент технолоджи» <i>e-mail: <a href="mailto:olgazhibrik@gmail.com">olgazhibrik@gmail.com</a></i>

*Образец ссылки на публикацию:*

А. А. Беляева, Е. В. Биряльцев, М. Р. Галимов, Д. Е. Демидов, А. М. Елизаров, О. Н. Жибрик. Кластерная архитектура программно-технических средств организации высокопроизводительных систем для нефтегазовой промышленности // Программные системы: теория и приложения: электрон. научн. журн. 201?. Т. ?, № ??(??), с. ??–??.

URL: <http://psta.psiras.ru/read/???>

A. A. Belyaeva, E. V. Biryaltsev, M. R. Galimov, D. E. Demidov, A. M. Elizarov, O. N. Zhibrik. Architecture of HPC clusters for Oil&Gas Industry

**ABSTRACT.** We consider design and architecture of a complex integrated system for high performance computing, data storage, and 3D visualization targeted at oil and gas industry.

We describe technical details and results obtained while creating such an integrated software and hardware solution, and talk about challenges and directions for further progress in the technological field.

*Key Words and Phrases:* numerical simulation, distributed computing, distributed storage, oil and gas industry, HPC platform.