Вычислительные кластеры: проблемы и тенденции

А.Е. Шевель

Аннотация

Ниже предлагается краткий обзор сообщений по вопросам построения вычислительных кластеров на совещании Spring HEPiX 2007, которое состоялось 23-27 марта 2007 в институте ядерных исследований **DESY** (Гамбург, ФРГ). Материалы совещания представляют собой несомненный интерес для тех, кто вовлечён в организацию использования вычислительных ресурсов, несмотря на то, что часть сообщений предполагают некоторое знакомство с реалиями организации обработки данных в экспериментальных исследованиях по физике высоких энергий. В оригинальной форме материалы совещания представлены на сайте http://hepix2007.desy.de/, там можно найти слайды презентаций и посмотреть видео самих презентаций. Состав тем, представленных в данном обзоре, находится на совести автора обзора.

Серия совещаний HEPiX

Совещания **HEPiX** начались в 1991 году. В то время цель этой серии совещаний была сформулирована следующим образом.

«HEPiX есть международная группа сотрудничающих исследовательских институтов, которые экстенсивно расширяют использование **ОС UNIX** в экспериментах по физике высоких энергий. Усилия группы сфокусированы на:

- обмене опытом;
- информировать поставщиков вычислительной техники и организации, готовящие стандарты;
- исследовании возможных решений тех проблем, которые имеют место в использовании **UNIX** в качестве основной **ОС** на главных вычислительных установках в институтах членах группы.»

Первоначальные темы, предложенные на первом совещании, состоявшемся в **FNAL** 23-25 сентября 1991 года, были следующими:

- пакетная обработка заданий;
- работа с магнитными лентами;
- управление программным обеспечением и распространение программ;
- переносимость программного окружения в физических вычислениях;
- поддержка оборудования и программного обеспечения распространённых персональных компьютеров.

На первом совещании присутствовало около 40 участников из **ANL**, **BNL**, **FNAL**, **HEPnet**, **INFN**, **KEK**, **LAMPF**, **LBL**, **NIKHEF**, **SCRI**, **SLAC**, **SSCL**, **SURA/CEBAF**, **TRIUMF**. Более подробную информацию об этой серии совещаний можно найти на сайте http://wwwhepix.web.cern.ch/. В настоящее время такого рода совещания проводятся дважды в год – весной и осенью. Число участников колеблется от 40 до 150 человек. Все эти люди являются специалистами, которые отвечают за планирование развития кластеров (иначе **ЦОД**ов) и/или за каждодневное круглосуточное функционирования вычислительной инфраструктуры в своих институтах. До сих пор организация совещания была весьма традиционной: небольшой зал, кафедра, экран, электронный проектор, презентации с компьютера, ведущего заседание.

Весеннее совещание HEPiX 2007

Совещание проходило по программе, состоящей из нескольких тематических разделов:

- Состояние/изменение основных вычислительных мощностей в различных институтах. Здесь представлялись общие сведения об изменениях в компьютерной инфраструктуре, установках нового оборудования и/или программного обеспечения, пр. Иными словами, что происходит в конкретных Центрах Обработки Данных (**ЦОД**) или вычислительных кластерах. Всего 13 сообщений по 15-20 минут.
- Решения и архитектуры. Всего 6 сообщений по 30 минут каждое.
- Вторичная память и файловые системы. Всего 13 сообщений по 20-30 минут и одна дискуссия на эту тему 40 минут.

- Системный менеджмент. Всего 8 сообщений по 25-30 минут.
- Научный Линукс (Scientific Linux). Была представлена информация о Scientific Linux 5 и о Scientific Linux в CERN. Всего 2 сообщения по 20-30 минут и дискуссия на эту тему 40 минут.
- Различные измерения на ЦОДах и родственные проблемы. Всего 6 сообщений по 20-25 минут.
- Разное. На данном совещании здесь были представлены соображения по увеличению надёжности вычислительных установок с использованием тестирования и методики предсказания отказов, а также новости о сетевой безопасности.

Автор статьи обратил внимание на следующие темы.

- Текущее состояние вычислительных мощностей.
- Доступ к распределённым данным.
- Кластерные файловые системы.
- Вопросы виртуализации.
- Системный менеджмент.
- Надёжность хранения данных на дисках.
- Измерения производительности.
- Состояние дистрибутива Scientific Linux.

Состояние/изменение основных вычислительных мощностей

Здесь имели место сообщения из следующих институтов: LAL/IN2P3 (Франция), LAPP (Франция), PSI (Швейцария), CASPUR (Италия), BNL (США), GSI (ФРГ), RAL (Англия), GridKa (ФРГ), PDSF (США), ScotGrid (Англия), TRIUMF (Канада), INFN-T1 (Италия), CC-IN2P3 (Франция), CERN (Швейцария), DESY (ФРГ), SLAC (США). В таблице 1 приведены важные параметры нескольких **ЦОД**ов (вычислительных кластеров, или просто кластеров).

	Пакетная система выполнения заданий	Число машин в кластере	Ёмкость памяти на дисках (TB)	Ёмкость массовой памяти на лентах (PB)	Ёмкость внешних каналов связи
LAL/IN2P3	BQS	~900	~450	~1.6	23 Gbit
TRIUMF	LSF	~200	~300		15 Gbit
BNL	Condor/LSF	~2400	~900	~4.0	
DESY	LSF	~1200	~430	~2.0	transition to 10 Gbit
INFN-T1	LSF	~1600	~950	~0.75	30 Gbit
RAL			~950		20 Gbit
GridKa			~1550		10 Gbit
SLAC		~1700	~750		

Таблица показывает масштаб некоторых (не всех) кластеров уровня **T1** (пустые ячейки означают, что автор не располагает соответствующей свежей информацией). В таблице упоминается не более трети кластеров уровня **T1**, которые планируются к использованию для обработки данных с ускорителя в **CERN** (Женева, Швейцария). Естественно, что все кластеры имеют необходимо программное обеспечение, программный инструментарий **Globus** + **VDT** + масса прочего, для реализации архитектуры **Grid**. В настоящее время в этой архитектуре имеются более сотни виртуальных организаций (**VO**) и ежедневно выполняются много тысяч заданий. Очень важным является организация доступа к данным в условиях упомянутой архитектуры **Grid**.

Доступ к распределённым данным в Grid

В вычислительных системах типа **Grid** (во всяком случае, для физики высоких энергий) данные хранятся в так называемых Storage Elements (**SE**) – элементах памяти. Это сервер или мощный кластер, который содержит необходимую вторичную память, например, много петабайтов (PB). Таким образом, **SE** – это

родовое название функциональной единицы сети **Grid**. В качестве памяти используется дисковые массивы **RAID5** или **RAID6**. Кроме дисковых массивов на больших вычислительных кластерах используется роботизированная ленточная память с несколькими десятками магнитофонов.
Общим местом для дисковых массивов является использование кластерных файловых систем: **GPFS**, **Lustre** и т.д. Массовая передача информации (имеется в виду объёмы передач от сотен **GB** до многих десятков **TB** и более) с одного **SE** на другой производится с использованием систем управления памятью (Storage Resource Management – **SRM**). Как на данном совещании, так и в литературе часто употребляется один термин – **SRM** в двух смыслах: **SRM** как стандарт на интерфейс к системе хранения/передачи данных и **SRM** как конкретная программная система. В настоящее время используется стандарт **SRM** 2.2 (см. http://sdm.lbl.gov/srm-wg/doc/SRM.v2.2.pdf). Имеется несколько реализаций систем типа **SRM:**

- **CASTOR2** – иерархический сервер памяти базирующийся на использовании ленточной роботизированной памяти; совместная разработка **CERN** и **RAL**.
- **dCache** - иерархический сервер памяти, может использовать различные роботизированные ленточные системы памяти; разработка **DESY** и **FNAL**.
- **DPM** - иерархический сервер памяти, использует только дисковую память; разработка **CERN**.
- **StoRM** - иерархический сервер памяти, использует только дисковую память; разработка **INFN** и **ICTP**. Имеет интерфейс для работы с различными файловыми системами: **GPFS**, **Lustre**, **XFS** и др.
- **BeStMan** - иерархический сервер памяти, использует только дисковую память; разработка **LBNL**.

Все перечисленные системы имеют интерфейс **SRM** 2.2. Поскольку интерфейс сравнительно молод, то различные системы хранения/передачи данных с разной скоростью приближаются к описанному стандарту. Интерфейс **SRM** к различным реализациям систем памяти в **Grid**, естественно, предполагает использование Grid Security Infrastructure (**GSI**) и других компонентов распределённой вычислительной среды.. Любая операция **SRM** использует многие десятки или сотни элементов, таких как сегменты линий связи, серверы, всевозможные программные сервисы и т.д., распределённые географически по различным городам, странам и континентам. Чтобы видеть реальное состояние таких распределённых систем, необходимо периодическое тестирование, которое заключается в выполнении набора относительно простых операций. В презентации «SRM update: experiment requirements, status and deployment» представлена информация о таком наборе операций и показаны результаты. Тестирование различных систем показывает, что со временем они становятся всё более надёжными (как видно на рисунке 1).

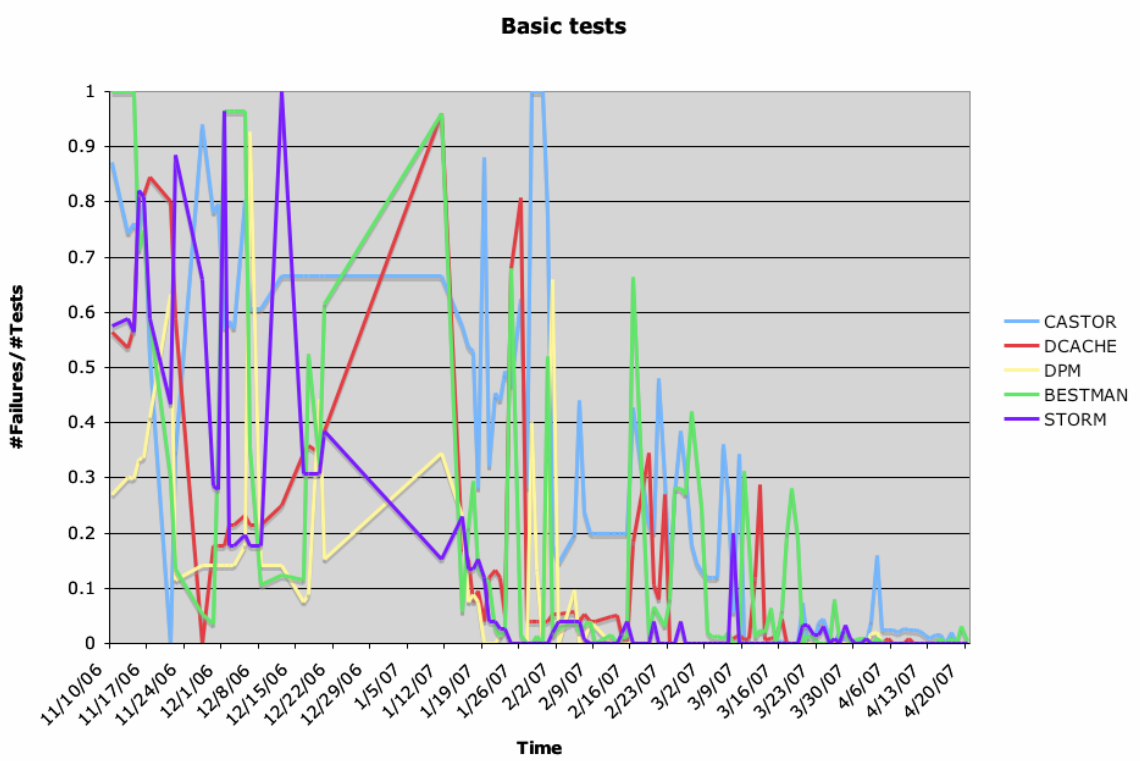

Рисунок 1. Число ошибок в тестировании различных систем (рисунок из презентации Steve Traylen).

На совещании были представлены также сообщения о состоянии ряда проектов по **dCache**, **DPM**.

Очевидно, что при таких больших объёмах данных в распределённых вычислительных установках особую важность приобретают кластерные файловые системы.

Кластерные файловые системы

Традиционные файловые системы, такие как *ext3, xfs, reiserfs* и другие обычно ассоциированы с отдельным физическим устройством (или его частью), например, дисковым накопителем. Такие файловые системы были разработаны и использовались в рамках одного внешнего устройства на одном узле. В вычислительных кластерах, состоящих из сотен компьютеров, необходимо использовать иные файловые системы, которые удовлетворяют следующему перечню требований:

- высокую пропускную способность по чтению/записи (главным образом за счёт параллельного выполнения операций ввода/вывода);
- масштабируемость для большого числа клиентов, т.е. узлов кластера;
- поддержка файлов большого размера (несколько ТВ);
- целостность данных;
- единое пространство имён файлов и директорий;
- единое логическое пространство памяти;
- способность к продолжению работы после выхода из строя отдельных элементов, например, дисководов.

Обычно кластерная файловая система содержит большой объём данных распределённых по нескольким (или многим) серверам памяти, которыми пользуются множество клиентов для одновременного и независимого выполнения операций записи и/или чтения данных.

Среди рассмотренных систем были представлены в различных аспектах системы **PANASAS**, **GPFS**, **Lustre**. Все эти системы решают (каждая по-своему) вышеперечисленные

задачи. Каждая из упомянутых систем представляет собой большой программный комплекс и имеет свои достоинства и недостатки. Насколько автору известно, никто не производил прямых сравнений этих систем, поскольку серьёзное экспериментальное сравнение может занять несколько специалистов самой высокой квалификации на год или более. Авторы презентаций приводят различные данные о своих системах. Высказывались планы сокращения использования **PANASAS** и перехода на **GPFS** или другие системы. Обращает на себя внимание факт, что достигнутая средняя скорость записи/чтения данных в кластерных файловых системах составляет 1 GB/sec и более.

В презентации Damian Hazen «**GPFS/HPSS Integration»** представлены интересные попытки интеграции файловой системы **GPFS** и роботизированной ленточной памяти **HPSS**. В частности, разработчики планируют совместить эту интеграцию с процессом резервного копирования. Иными словами, если какие-то файлы уже записаны в ленточный робот, например, в связи с низкой частотой использования, то они не должны восстанавливаться при рутинном восстановлении, например, после аварии с дисковым накопителем.

Дойдя до этого места, читатель может обратить внимание, что и кластерные файловые системы и гридовский интерфейс к системам управления хранением данных представляет собой некоторые варианты виртуализации систем памяти: пользователь может сосредоточиться на содержании памяти, не вникая в технические детали того, как память организована и где она географически размещается.

Виртуализация

На совещании были представлены сообщения: Thomas Finnern “Highly Available Central Services. A Virtulization Approach” и отчёт о совещании по виртуализации Owen Synge “Virtualization Users Workshop Report”. Речь идёт о виртуальных компьютерных узлах, когда несколько виртуальных узлов могут сосуществовать на одном реальном узле. В общем случае на каждом виртуальном узле может быть отдельная ОС, не совпадающая по типу с ОС на соседнем виртуальном узле. Сейчас, такого рода виртуализация может быть организована с использованием различных программных систем, например, **VMware**, **Xen**, других. Авторы различают термины **виртуализация** и **паравиртуализация**. **Виртуализация** есть общий метод предоставления пользователю абстракции памяти, периферийного устройства или вычислительной машины, невзирая на внутреннюю организацию имеющегося оборудования. **Паравиртуализация** - метод виртуализации компьютера, который предоставляет пользователю абстракцию виртуальной машины с программными интерфейсами, которые похожи, но не идентичны нижележащим аппаратным интерфейсам, например, **Xen**. В известном смысле последний термин – это просто уточняющее название для прежних идей виртуализации (вспомним, например, пару **Control Program** – **Conversational Monitor System** из OS VM на вычислительных установках IBM370 и более поздних).

Среди преимуществ использования виртуализации на вычислительных кластерах авторы указывают на следующие:

- большая степень изолированности каждого отдельного задания на физическом узле, поскольку каждое задание выполняется на отдельном виртуальном узле, что намного снижает вероятность несанкционированного использования данных или прокси одного задания другим заданием на том же физическом узле;
- большее количество заданий можно пропустить на кластере той же производительности, что определяется следующими соображениями:
    - портабильность - при использовании виртуализации легче администрировать заданиями, например, ввести любое задание в состояние «*пауза*» создать копию образа задания и запустить его на другой машине;
    - возможность использования более, чем одну ОС - пользователь имеет возможность выбирать операционные системы, в которых он хотел бы выполнить задание; в то же время администратор может использовать ту ОС, которая более подходит для администрирования;
- большая изоляция пользователя и пользовательских программ от реальных периферийных устройств.

Оба автора тестировали несколько вариантов виртуализации. Оба уверены, что это наличие виртуальных узлов в вычислительных кластерах есть ближайшее будущее архитектур типа **Grid**.

Отмечено, что несколько сильных организаций активно экспериментируют в области виртуализации вычислительных узлов больших кластеров. Перечислены программные продукты, которые используются в различных экспериментах по виртуализации на больших кластерах:

- **Xen**
- **VMware**
- **Solaris** containers
- User Mode Linux (**UML**)
- **Vserver**.

Чаще всего упоминались **Solaris** и **Xen**. Рассматривались пять моделей виртуализации. Отмечается, что виртуализация влечёт пренебрежимо малое снижение производительности. Подчёркивается, что одним из основных достоинств является поддержка различных операционных систем на одной и той же машине. Предполагается, что в недалёком будущем **Xen** и **Solaris containers** будут взаимозаменяемы. Нужно сказать, что виртуализация является «горячей» темой, которая обсуждается на многих митингах. С другой стороны, следует помнить, что виртуализация находится в стадии более или менее интенсивного экспериментирования.

Итак, мы видим, что количество **ОС**, сервисов и сложность сервисов растёт на больших кластерах как снежный ком. Как следить за всем этим многообразием и как устанавливать всё это на сотнях кластерных узлах?

Системный менеджмент

Этой проблематике были посвящены несколько презентаций. В презентации Ian Neilson “Overview of WLCG Grid Services Monitoring Working Group**”** представлен краткий обзор деятельности рабочей группы по мониторингу сервисов в **Grid**. Основной целью деятельности данной группы разработчиков является повышение надёжности работы **Grid** путём сбора, обработки и представления корректной информации о состоянии различных сервисов в **Grid** администраторам и пользователям. Важность работы определяется тем, что распределённая (на разных континентах) сеть крупнейших вычислительных кластеров является довольно большой:

- более дюжины кластеров с более, чем тысячью машин в каждом;
- с более, чем тысяча ТВ дисковой памяти в среднем в каждом;
- с огромными роботизированными ленточными хранилищами в каждом.

В этой сети ежедневно выполняется несколько десятков тысяч заданий от сотен пользователей. Передаются десятки ТВ данных. Нетрудно видеть, что требуется постоянно следить за состоянием такой сети, чтобы быть в состоянии предпринять контрмеры при возникновении неплановых ситуаций: снижение пропускной способности по передаче данных между кластерами, отключение питания, выход из строя маршрутизатора и другие возможные неприятности. Группа представила несколько прототипов и находится в поиске наиболее эффективных решений по мониторингу. Естественным решением группы является использование обкатанных решений на конкретных кластерах, например, такие подсистемы как **GridView**, **GridICE**, **GOCDB**, **Gstat**, **MonaLisa**, **Nagios@CEE**, которые уже используются на конкретных кластерах. Сложность задачи определяется несколькими условиями:

- методами сбора данных о работе отдельных компонентов **Grid** (например, локально или удалённо);
- форматами данных о работе отдельных компонентов **Grid** и протоколами передачи этих данных;
- организацией хранения собранных данных;
- формами отображения данных.

Все перечисленные пункты содержат массу деталей, с которыми можно ознакомиться на сайте https://twiki.cern.ch/twiki/bin/view/LCG/. Часто недооценивается значение последнего пункта в приведённом перечислении. Формы представления и состав самих данных различны для различных групп пользователей. Например, руководители крупного эксперимента могут интересоваться лишь общей загрузкой крупных вычислительных кластеров в мире, игнорируя множество деталей, относящихся к конкретным пакетам заданий. Отдельный экспериментатор может следить только за своим пакетом заданий (или отдельным заданием), не тратя время на изучение информации о глобальных параметрах **Grid**.

Другие презентации касались использования ряда мониторирующих и кластерных систем: **Nagios**, **Quattor**, **cfengine**, etc. На большинстве кластеров наличие таких систем как **Nagios** и/или **Ganglia** является общим местом.

Собранные мониторирующие данные могут занимать немалый объём, особенно если планируется хранить сведения о состоянии вычислительной системы за длительное время (например, несколько лет). Здесь возникает интересный вопрос о надёжности доступа к данным на дисках. В большинстве случаев молчаливо предполагается, что данные на дисках не изменяются со временем. Интересно выяснить, что происходит на самом деле.

Надёжность хранения данных на дисках

При больших объёмах памяти, исчисляемой в сотнях терабайтов и более, в силу вступают законы больших чисел. Так, если что-то нежелательное, например, сбой при операции чтения, случается раз в год на нескольких дисках, то в системе с тысячами дисков такое может происходить несколько раз в день. Какие изменения происходит в системе после непланового отключения питания? На маленьких установках, после аварии происходит простая проверка целостности локальной файловой системы. Если попытаться запустить такую проверку в большой кластерной файловой системе, то она может занять несколько дней. Именно по этим причинам среди сообщений, посвящённых файловым системам, горячий интерес вызвала презентация Klement Peter “Silent corruption” («Тихое искажение»). Такое необычное название выбрано из соображений, что, по мнению автора презентации, ряд искажений данных в больших файловых системах остаются неизвестными до того времени, пока они не приводят к каким-то неприятностям.

В презентации рассматривается серия экспериментов с записью файлов на диск, последующим чтением только что записанных файлов с диска, последующим сравнением прочитанных файлов с теми, что были записаны. В данной работе были использованы специальные тестовые файлы. Запись, считывание и сравнение производились фоновым процессом, который запускался на большом числе машин (в данном эксперименте на 3500 машинах). Результат был впечатляющим: прочитанная с диска информация ***не совпадала*** с ранее записанным оригиналом примерно в тысяче случаях при общем объёме передачи из оперативной памяти на диск 41 PB. Ошибки (несовпадения) были встречены на 170 машинах. Средний темп несовпадений = 2-5 в день. Не удалось найти строгую корреляцию с каким-то типом программного обеспечения. В некоторых случаях можно было говорить о сбоях на конкретных машинах. Однако в большинстве случаев не просматривается корреляции с типом оборудования.

Источники ошибок интуитивно очевидны. Так, в технических данных на многие дисководы можно найти, что вероятность ошибок на дисках 10**-14, т.е. одна ошибка на 10**14 бит при выполнении операции ввода/вывода. Таким образом, возникает примерно одна ошибка (один бит неверный) в среднем при считывании примерно 11.4 TB. Если учесть, что информация с поверхности диска считывается примерно со скоростью 900 Mbit/s (как максимум), то в среднем можно ожидать одну ошибку за 30.9 часов непрерывного чтения. В реальных дисковых системах при использовании многих сотен дисководов, это время может сократиться до минут. Кроме того, ошибка в считывании управляющей информации, описывающей расположение данных на диске, может привести и приводит к считыванию полностью искажённой информации, объём которой кратен размеру стрипа в **RAID** (например, 64 KB или 128 KB). Напомним, в дополнение, о вероятном существовании погрешностей в микропрограммах контроллеров периферийных устройств (сетевых, дисковых, проч.). В презентации приводится типизация ошибок и некоторые методы предотвращения искажения информации. В частности, отмечается эффективность в смысле устойчивости по отношению к ошибкам ввода/вывода файловой системы **ZFS (**см., например, http://www.sun.com/software/solaris/zfs_lc_preso.pdf). В ряде презентаций сообщались сравнительные характеристики файловых систем **ZFS** (контрольные суммы, повышенная вероятность обнаружения искажения данных) и **XFS**. К моменту совещания **ZFS** была доступна лишь на **Solaris** 10. Есть информация, что может появиться версия под **Linux**. Несколько кластеров уже используют **ZFS,** и ряд организаций планируют это в ближайшее время, несмотря на то, что данная файловая системы ещё находится в весьма юном возрасте. Очевидно, что проблема, затронутая в сообщении, требует повышенного внимания и изучения.

После надёжности нелишне обратить внимание на серию презентаций по измерению быстродействия вычислительных машин.

Измерения быстродействия машин

Этому аспекту было уделено заметное внимание, поскольку необходимо иметь подходящие процедуры оценок производительности конкретных вычислительных машин. Существующие методы оценки (например, **SPEC**) естественно не идеальны и часто не отражают изменение производительности на конкретном классе задач, в данном случае для задач физики высоких энергий. В данной предметной области до последнего времени набор тестов **SPECint2000** считался одним из приемлемых способов сравнения производительности разных машин почти повсеместно – от разработчиков до фондирующих организаций. Однако, в ряде презентаций предлагается больше внимания обратить на **SPECint2006**, который значительно больше соответствует современной архитектуре микропроцессоров (многоядерность, большой кэш второго уровня на каждом ядре и т.п.).

Среди параметров, характеризующих производительность, используется ещё один важный показатель – производительность машины на ватт потребляемой мощности. Это неудивительно, поскольку экономия хотя бы 1 ватт на машину при заказе 500 машин может вылиться в экономию около 2 КВт или больше, учитывая потребление электроэнергии самими машинами, системой охлаждения, системами бесперебойного питания, прочим сопутствующим оборудованием.

Было уделено внимание разнице 2-4-ядерных машин и многоядерных машин (8-16 ядер). Отмечалось, что многоядерные машины имеют в настоящее время недостаточное число сетевых портов, оперативной памяти. Обсуждаются как минимум 2 GB оперативной памяти на ядро.

Состояние дистрибутива Scientific Linux

Отчёт о состоянии этой дистрибуции вызвал некоторую дискуссию, поскольку эта версия **Linux** является фактическим стандартом для физики высоких энергий. Дистрибуция доступна на сайте http://www.scintificlinux.org, а также имеется русскоязычная версия на сайте http://www.linux-ink.ru. Фактически эта версия, в значительной степени, производная от RedHat. Коллектив, занимающийся поддержкой **Scientific Linux**, компилирует исходные тексты, которые входят в **RedHat** и которые являются свободно распространяемыми текстами, что-то убирает и добавляет что-то своё. В настоящее время был представлен и обсуждался **Scientific Linux** 5.x. Заметим попутно, что примерно половина представленных на совещании кластеров использует **Scientific Linux** 3.x, а другая половина использует смесь 3.x и 4.x. Практически все организации, представленные на совещании, планируют переход в ближайшее время на **Scientific Linux** 4.x.

Этот дистрибутив, естественно отличается от других Linux дистрибутивов. Обсуждать его достоинства или недостатки в отрыве от конкретной предметной области (в данном случае – это экспериментальные исследования по физике высоких энергий) представляется малопродуктивным. Основными требованиями в данной области, в особенности на вычислительных кластерах, является следующее:

- высокая надёжность работы базовых программных средств (1/2 года или около этого без перезагрузки и без проблем с десятками-сотнями заданий каждый день является нормой);
- доступность и надёжное функционирование популярных кластерных систем;
- доступность и надёжная работа файловой системы AFS, некоторых других компонентов (полный список пакетов, которые добавлены и которые удалены, приведён на сайте дистрибутива);
- приемлемый уровень поддержки, который, в основном, реализуется в виде дискуссионных списков.

Заключительные замечания о тенденциях

Внутренние сети вокруг кластеров имеют коммуникации на основе 10 Gbit Ethernet. Внешние каналы как правило имеют ёмкость 10 Gbit или более. Ёмкость внешних каналов связи наращивается быстро. По ряду оценок к 2013 году ожидается достижение ёмкости около 1 Tbit для каналов между кластерами на разных континентах.

На большинстве кластеров переходят к кластерным файловым системам и прекращают использование **NFS**. В качестве кластерной файловой системы используются в основном **GPFS** и **Lustre.** Несколько кластеров используют **PANASAS,** обсуждается переход от этой файловой системы к другим.

На всех кластерах планируется увеличение объёмов памяти и вычислительной мощности примерно на 30% в ближайший год. Общее для всех вычислительных кластеров – быстро обновляется оборудование. В

среднем в год устанавливается дополнительно примерно 1/3 новых машин. Большая часть вновь приобретаемых машин базируется на Intel Xeon 5160 (Woodcrest 3 GHz) или близкий по параметрам Opteron. По мнению автора данной статьи – это очень правильное решение: поддерживать возраст важных компонентов кластеров не более 3 лет. Таким образом, весь кластер поддерживается на современном уровне.

Почти все кластеры работают под **Scientific Linux**. Многие используют версию 3.x, но все планируют переход к **Scientific Linux** 4.x.

Пакетные системы различны: используются **PBS** и его клоны, **LSF**, **Condor**, **SGE**.

В отношении дисковой памяти очень популярно решение в виде «дискового ящика» (например, Sun Fire X4500 – 48 x 500 GB на шасси 4U).

Закончилась гонка по частоте процессоров, началась гонка по количеству ядер процессоров. По общему мнению участников совещания простая мультипоточность (multithread) процессора не является достаточной для параллельных вычислений с MPI или OpenMP. В частности, в связи с ростом количества ядер в процессоре растёт популярность использования программного **RAID**.

В ряде презентаций сообщается о проблемах с установкой большого количества машин: проблемы с охлаждением раков, потребляемой мощностью, подготовкой помещений. В связи с этим предложено интересное решение [см. рисунок 1 и. 2]. Как показано на рисунках, вычислительный кластер размещён в стандартном контейнере. После установки контейнера к нему подключается электрическую сеть, компьютерную сеть, вода (для охлаждения) и кластер готов к работе. Если требуется большой кластер, то можно поставить несколько (или много) контейнеров.

Рисунок 2. Кластерный чёрный ящик - вид снаружи.

Рисунок 3. Кластерный чёрный ящик – вид внутри.