



ИПМ им.М.В.Келдыша РАН • Электронная библиотека

Препринты ИПМ • Препринт № 13 за 2008 г.



Ермаков А.В., Никитин К.Е.

Использование
паравиртуального монитора
Хеп в Грид-сайте ИПМ

Рекомендуемая форма библиографической ссылки: Ермаков А.В., Никитин К.Е.
Использование паравиртуального монитора Хеп в Грид-сайте ИПМ // Препринты ИПМ
им. М.В.Келдыша. 2008. № 13. 19 с. URL: <http://library.keldysh.ru/preprint.asp?id=2008-13>

**Ордена Ленина
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.Келдыша
Российской академии наук**

А.В.Ермаков, К.Е.Никитин

**ИСПОЛЬЗОВАНИЕ ПАРАВИРТУАЛЬНОГО
МОНИТОРА XEN В ГРИД-САЙТЕ ИШМ**

Москва 2008

В данной работе рассматривается применение паравиртуального монитора Xen для замены компьютерных узлов, выделенных в отладочном кластере для специализированных Грид-служб, на виртуальные машины.

Use of paravirtual monitor XEN in the grid-site of KIAM

This work illustrate how to use paravirtual monitor Xen to replace compute nodes by the virtual machines for the special grid-services.

Оглавление

1.	Введение.....	3
2.	Тестирование программного обеспечения Грид.....	5
3.	Паравиртуальный монитор виртуальных машин Xen	6
4.	Инсталляция Xen.....	8
5.	Создание гостевого домена.....	9
6.	Инсталляция ОС.....	12
6.1.	Инсталляция средствами базовой операционной системы	12
6.2.	Использование готового образа	12
6.3.	Использование диска реальной машины	13
6.4.	Создание файла конфигурации гостевого домена	14
6.5.	Загрузка домена	15
	Заключение.....	17

1. Введение

Технологии Грид стремительно развиваются. Грид предполагает коллективный режим доступа к ресурсам и к связанным с ними услугам в рамках глобально распределенных виртуальных организаций, состоящих из предприятий и отдельных специалистов, совместно использующих общие ресурсы. Технологии, поддерживающие сотрудничество с использованием общих ресурсов, привлекают все большее число специалистов из различных областей знаний, создают комфортные условия для организации такого сотрудничества.

Одним из примеров успешной реализации международного сотрудничества является проект Большого адронного коллайдера (Large Hadron Collider - **LHC**) в Европейском центре ядерных исследований (CERN). Для программно-аппаратного обеспечения этого проекта в 2002 году началась разработка проекта LHC Computing Grid (**LCG**), задачей которого стала интеграция большого числа компьютеров, размещенных в разных точках мира, с целью образования единого глобального вычислительного ресурса. Проект LCG решает не только проблему хранения и доступа к данным, но и дает необходимые вычислительные ресурсы для их анализа (а это более 40 тыс. современных компьютеров). Причем как данные, так и вычислительные ресурсы, задействованные в этом проекте, должны быть доступны тысячам ученых, участвующих в экспериментах с LHC, независимо от их географического местоположения.

С российской стороны в реализации проекта EGEE принимают участие институты Москвы, Московской области и Петербурга: НИИЯФ МГУ, ИТЭФ, ИФВЭ, ИМПБ РАН, ОИЯИ, ПИЯФ РАН, РИЦ КИ, ИПМ РАН. Участие в этом проекте Института прикладной математики подразумевает решение следующих задач:

- создание и обслуживание ресурсного центра;
- оценка и адаптация программного обеспечения;

Для решения этих задач в ИПМ существует два кластера: один для решения счетных задач и другой, дающий возможность производить оценку, отладку и адаптацию программного обеспечения. Оба этих кластера в соответствии с архитектурой Грид предоставляют конечному пользователю прозрачный доступ к вычислительному ресурсу и ресурсу хранения данных и состоят из следующих элементов (компьютеров):

- пользовательский интерфейс (UI),
- вычислительная установка (CE),
- рабочий узел (WN),
- массив данных (SE),
- элемент мониторинга (MON).

Возможность объединения большого количества компьютерных ресурсов, разнообразных по своей структуре и операционной среде, достигается за счет нового типа промежуточного программного обеспечения, специально создаваемого для Грид, и нового поколения аппаратных средств.

Программная архитектура Грид соответствует модели служб. В инфраструктуре Грид службы устанавливаются на компьютерные элементы соответствующих типов с соответствующим программным обеспечением.

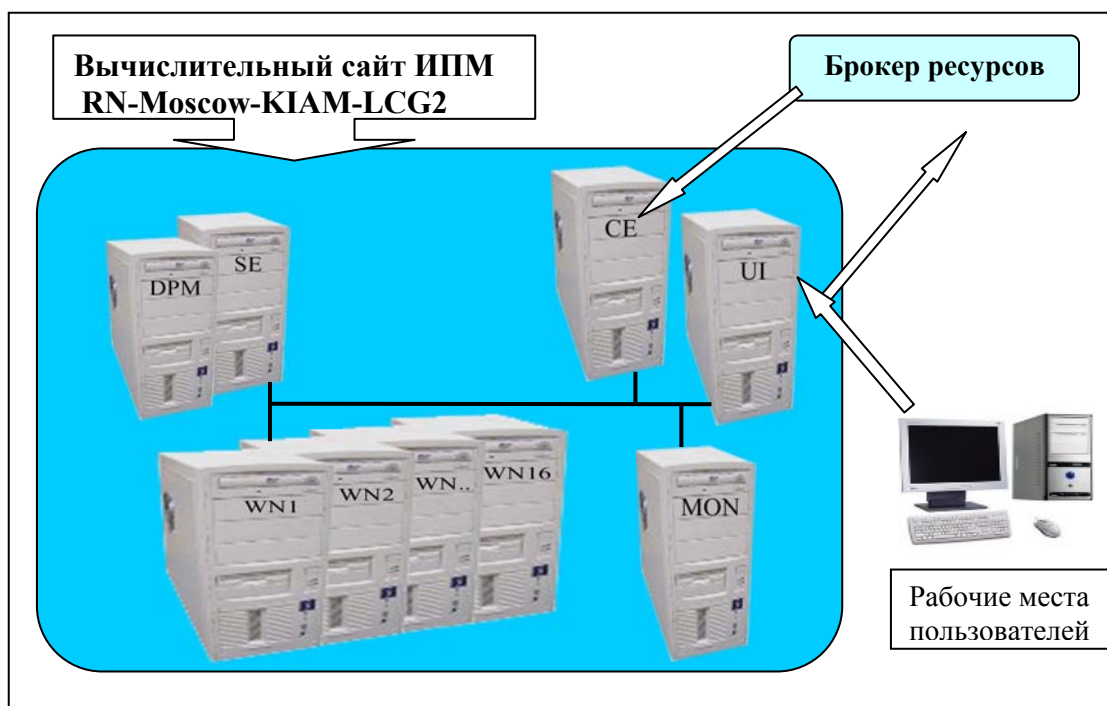


Рис.1. Инфраструктура Грид-сайта ИПМ

Пользовательский интерфейс (UI, User Interface) позволяет конечному пользователю взаимодействовать с ресурсами Грид, через него пользователь запускает свои задания и получает результаты их выполнения. Вычислительная установка (CE, Computing Element) представляет собой грид-интерфейс к локальным вычислительным ресурсам, позволяющим принимать задания, распределять их на выполнение на вычислительные (рабочие) узлы в локальном кластере и возвращать результаты.

Рабочие узлы (WN, Work Nodes) - это компьютеры, на которых непосредственно осуществляются вычисления (для этого на рабочих узлах устанавливается пользовательское программное). Рабочие узлы располагаются в локальной сети за шлюзом, управляются системой пакетной обработки и доступны в гриде только через CE. Компьютеры управления массивами данных (SE, Storage Element) предоставляют единообразный интерфейс доступа к устройствам хранения. Данная

компонента управляет и большими дисковыми массивами, и системами массовой памяти. Интерфейс SE делает прозрачным доступ к данным как пользователей, так и приложений.

Элемент мониторинга (MON, Monitoring) - на этом узле запускаются специальные приложения, позволяющие контролировать внутренние ресурсы сайта, и передавать состояние ресурсов во внешние службы мониторинга ресурсов.

Существующая версия программного обеспечения (ПО) основана на пакете gLite (Lightweight Middleware for Grid Computing), который является полным решением для грид, включая как базовые низкоуровневые программы, так и ряд служб высокого уровня. Этот пакет постепенно заменил собой пакет LCG, который работал в инфраструктуре EGEE на ранней стадии проекта.

Основной операционной системой (ОС) проекта до недавнего времени являлся Scientific Linux 3 (SL3, сделанный на основе Redhat Linux 7.3), который заменил базовую ОС во всех предыдущих версиях LCG-2. Но уже завершается переход на SL4, как на более современную и обеспечивающую более высокий уровень безопасности систему.

Пакет gLite разработан с учетом свойств интероперабельности и содержит базовые службы, облегчающие построение приложений Грид (gLite не является средством построения приложений, он поддерживает их обработку) для любых прикладных областей. В нем интегрированы компоненты из лучших на настоящий момент проектов прикладного программного обеспечения (ППО), таких, как Condor и Globus Toolkit, а также компоненты, разработанные для проекта LCG. Пакет gLite распространяется на условиях лицензии открытого кода, что облегчает его доработку и развитие в рамках международного сотрудничества.

2. Тестирование программного обеспечения Грид

При появлении новых версий программного обеспечения проекта LCG перед его установкой на компьютеры Грид-сайта ИПМ пакеты программ проходят процедуры предварительного тестирования на специально выделенном отладочном Грид-сайте. Но нагрузка на компьютерные ресурсы в отладочном Грид-сайте незначительна, что позволяет заменить физические компьютерные узлы виртуальными (рис.2).

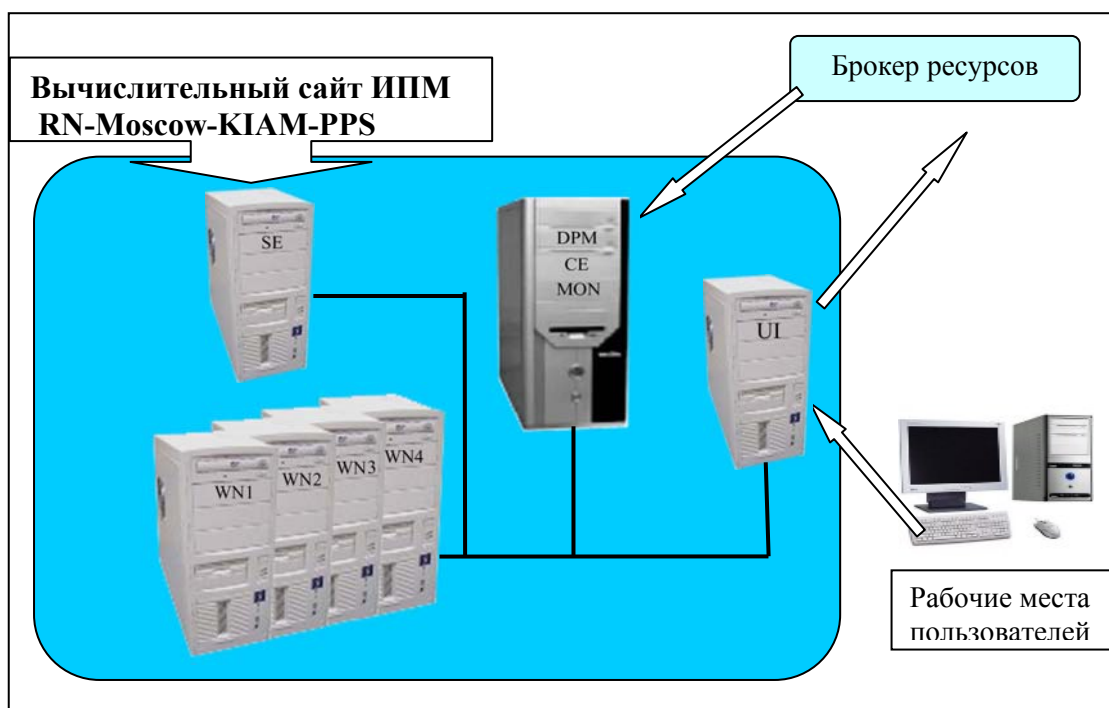


Рис.2. Использование виртуальных машин в инфраструктуре отладочного Грид-сайта

Технология виртуальных машин позволяет иметь на одной физической машине несколько функционально независимых операционных систем. Размещение Грид-служб подобным образом связано с небольшим объемом оперативной памяти (1,5 Гб) в доступной нам машине. При большем объеме оперативной памяти мы смогли бы запустить более, чем три гостевые операционные системы. Кроме того, следует отметить некоторые особенности настройки служб, не всегда позволяющие размещать данную службу иначе, как на одной машине. Например, для работы SE необходим сервис NFS, который мы не смогли запустить в гостевых ОС под управлением Scientific Linux 3 из-за некорректного определения виртуальной сетевой карты (эта проблема была впоследствии решена путем установки Scientific Linux 4). Пользовательский интерфейс UI размещен на отдельной машине в связи с тем, что использовался один для двух вычислительных кластеров. Впоследствии, в связи с переходом на Scientific Linux 4, для этого сервиса была выделена отдельная машина, которую в дальнейшем планируется оставить одну на два кластера.

3. Паравиртуальный монитор виртуальных машин Xen

В отличие от кластера для ресурсного центра, где каждый элемент установлен на отдельном компьютере, в кластере, предназначенном для исследовательских работ и проверки (отладки) программного обеспечения, используется технология виртуализации на базе «паравиртуального» (т.е. без использования аппаратных архитектурных расширений виртуализации)

монитора виртуальных машин (VMM) Xen. Такой подход позволяет снизить количество используемых для отладочных целей компьютеров,

Xen способен поддерживать одновременную работу большого числа виртуальных машин с независимыми операционными системами на одной физической машине (рабочей станции), не затрачивая значительных вычислительных ресурсов. Достигается это за счет применения технологии паравиртуализации (т.е. без использования аппаратных архитектурных расширений виртуализации).

Часто в качестве синонима VMM используют термин гипервизор (hypervisor). Гипервизор полностью эмулирует компьютер, но его основное отличие состоит в способности поддерживать выполнение более чем одной операционной системы. На VMM выполняются так называемые гостевые операционные системы (guest OS) виртуальных машин, непосредственно поддерживающие работу приложений. Собственная операционная система гипервизора называется «хостовой» (базовой).

Технология паравиртуализации позволяет гипервизору в хостовой системе управлять гостевой ОС посредством гипервызовов VMI (Virtual Machine Interface). Такой подход обеспечивает достаточно высокое быстродействие гостевых систем при малых затратах на поддержку платформы паравиртуализации. Однако, по вполне понятным причинам, далеко не все разработчики операционных систем, в частности, компания Microsoft, готовы пойти на модификацию кода своих платформ. В результате технология паравиртуализации не получила быстрого развития и широкой популярности.

На данный момент общедоступной является версия Xen 3.0, которая включена в дистрибутивы нескольких ОС, таких как Red Hat, Novell SUSE, Debian, Fedora Core, Sun Solaris. Сотрудниками Кембриджского университета и лабораторией Microsoft Research был разработан вариант операционной системы Windows XP, поддерживающий технологию паравиртуализации, который, однако, не мог быть опубликован в соответствии с лицензионной политикой Microsoft. Тем не менее, опыт, полученный в ходе работы, был зафиксирован и на данный момент доступен в документах проекта Xen.

Развитием этого проекта занимается компания XenSource, которая в последнее время ведет тесное сотрудничество с компанией Microsoft в рамках партнерских отношений, что позволит XenSource более эффективно виртуализовывать Windows-платформы, а компании Microsoft реализовать полноценную поддержку Linux в платформе Windows Virtualization с последующей интеграцией в ОС Windows Server в 2008 году (Longhorn).

Представляя собой открытую платформу виртуализации для различных архитектур (IA-32, x86-64, IA-64 и PowerPC), Xen может быть развернут на хостах с множеством различных Unix-систем. При этом поддерживаются

гостевые ОС - Linux-, Windows- и BSD-системы. В этой платформе используются как технологии паравиртуализации для запуска операционных систем, так и технологии аппаратной виртуализации (VT-x и AMD-V) для поддержки немодифицированных версий ОС в виртуальных машинах. Процессоры, поддерживающие аппаратную виртуализацию, имеют дополнительные инструкции для управления виртуальными машинами, а также два режима работы: root-mode и nonroot-mode (привилегированный и непривилегированный). Гипервизор Xen работает в режиме root-mode, напрямую общаясь с аппаратным обеспечением, и управляет гостевыми системами. Платформа Xen поддерживает до 64 процессоров в физической системе (64-way SMP).

Начиная с версии 3.0, гипервизор Xen поддерживает технологии аппаратной виртуализации AMD-V и Intel VT, что позволяет использовать операционные системы Windows в качестве гостевых, а также другие немодифицированные ОС.

В настоящее время Open Source версия платформы Xen применяется в основном в образовательных и исследовательских целях, для моделирования информационных систем с клиент-серверной архитектурой на одной ЭВМ (эмуляция компьютерной сети с помощью нескольких виртуальных машин) для защиты информации и ограничения возможностей процессов, для исследования производительности ПО или новой компьютерной архитектуры, для эмуляции различных архитектур (например, эмулятор игровой приставки) с целью оптимизации использования ресурсов мэйнфреймов и прочих мощных компьютеров (например, IBM eServer)

Сейчас свободно распространяемые версии Xen включаются в дистрибутивы многих Linux-систем, что позволяет их пользователям применять виртуальные машины для изоляции программного обеспечения в гостевых ОС с целью его тестирования и изучения проблем безопасности, без необходимости установки платформы виртуализации.

4. Инсталляция Xen

Гипервизор Xen перекладывает большинство задач по поддержке аппаратных компонентов на гостевую операционную систему, работающую в виртуальной машине. Сам Xen содержит только код, необходимый для обнаружения и запуска остальных процессоров системы, настройки обработки прерываний и нумерации PCI шины (PCI bus enumeration). Драйверы устройств работают внутри привилегированной операционной системы, а не в самом Xen. Такой подход обеспечивает совместимость с большинством устройств, поддерживаемых Linux. Сборка XenLinux по умолчанию содержит поддержку серверного сетевого и дискового оборудования, но при необходимости можно добавить поддержку других устройств, переконфигурировав Linux-ядро стандартным способом.

Для инсталляции модифицированных операционных систем, подготовленных к использованию в паравиртуальной машине Xen, никаких дополнительных требований к аппаратному обеспечению компьютера, на котором будут выполняться виртуальные машины, не предъявляется. Функционирование операционных систем в неизменном виде возможно при использовании процессоров с поддержкой виртуализации.

Дистрибутив Xen включает три главных компонента:

- собственно Xen,
- модифицированную систему Linux для работы в гостевом домене Xen,
- утилиты для управления Xen-системой.

Существует несколько вариантов инсталляции дистрибутива Xen 3.0:

- из файлов, в виде откомпилированных пакетов в самом дистрибутиве операционной системы;
- из архива бинарных файлов;
- из исходных файлов.

Архивы с бинарными и исходными файлами Xen также находятся в свободном доступе (<http://www.xensource.com/downloads/>).

Установка из пакетов дистрибутива операционной системы – это наиболее простой способ, не требующий никаких дополнительных знаний и навыков, кроме как базовых навыков владения той ОС, на которую происходит установка Xen. Для этого устанавливаются RPM-пакеты с откомпилированным для этой системы ядром, в которое включена поддержка Xen. Пакет сам добавляет все необходимые изменения в систему и модифицирует загрузчик GRUB, добавляя в стартовое меню такой вариант загрузки, в котором автоматически запускается Xen (демон xend). При загрузке этого варианта запускается базовая операционная система, и появляется возможность создания и запуска других операционных систем.

При этом, если не стоит задача поставить отличную от поддерживающих Xen операционных систем (RedHat, Novell SUSE, Debian, Fedora Core, Sun Solaris), то для создания такой системы можно воспользоваться встроенными средствами управления гостевыми доменами (например, модуль YAST в дистрибутиве SuSE Linux 10).

5. Создание гостевого домена

Под понятием «домен» будем понимать отдельно работающую на компьютере операционную систему. Домен может быть **управляющим (dom0)** или **гостевым (domU)**.

Управляющий домен **dom0** имеет больше привилегий, чем другие виртуальные машины, работающие под его управлением, и служит для конфигурирования аппаратного обеспечения. В дополнение к своим обычным функциям **dom0** управляет созданием новых доменов, поддерживает работу гостевых систем и физических устройств. В **dom0** работает «демон» **xend** - специальная служба, являющаяся центральным пунктом управления всеми ресурсами в гипервизоре Xen.

Гостевой домен - это операционная система, работающая внутри отведенного ей конфигурационным файлом пространства. Каждая гостевая операционная система (**гостевой домен domU**), установленная впоследствии, требует своего места на жестком диске для создания своей файловой системы, которая будет корневой (root filesystem) для домена **domU**.

Xen поддерживает как физические устройства (например, **/dev/hda**), так и LVM-тома (**phy:VolumeGroup/root_volume**). Но ориентация на LVM дает большую гибкость в использовании доступного дискового пространства.

В SuSe Linux вер.10 (который нам наиболее удобно использовать в качестве примера) для этого есть утилита LVM2, способствующая созданию необходимых разделов на жестком диске. С её помощью можно выделить место, как для самой системы, так и для файла подкачки (swap-раздел). Утилита имеет графический интерфейс и делает процедуру создания очень наглядной и легкой в понимании.



Рис.3. Конфигурирование ОС Linux с помощью утилиты LVM

Если же графический интерфейс не загружен по какой-то причине, то существует возможность создания файловой системы гостевых доменов с помощью командной строки.

Ниже приведен пример создания тома с именем **vm** и размером 20 GB в группе томов **xenvg**:

lvcreate -L 20g -n vm xenvg создание тома с именем **vm** и размером 20 GB.

mkfs -t reiserfs /dev/xenvg/vm создаёт файловую систему типа **reiser**.

mkdir /mount/vm создает директорию - точку монтирования гостевой ОС.

mount -t reiserfs /dev/xenvg/vm /mount/vm монтирует созданную файловую систему в точку монтирования.

При этом, исходя из надежности системы, если используются два физических диска, желательно, чтобы логические тома целиком помещались на физических дисках. Это необходимо для того, чтобы при выходе из строя физического диска пострадало как можно меньше логических томов. Отобразить информацию о разбиении можно с помощью команды **lvdisplay -m**.

Например:

> lvdisplay -m

--- Logical volume ---

LV Name /dev/xenvg1/vm11

VG Name xenvg1

LV UUID NqClrx-06Tv-vbQ8-C2Et-9TZS-pF1o-Dm1yMF

LV Write Access read/write

LV Status available

open 1

LV Size 30.00 GB

Current LE 7680

Segments 1

Allocation inherit

Read ahead sectors 0

Block device 253:0

--- Segments ---

Logical extent 0 to 7679:

Type linear

Physical volume /dev/hdd3

Physical extents 0 to 7679

В данном случае локальный том, имеющий имя `/dev/xenvg1/vm11`, принадлежащий группе `xenvg1`, объемом 30 GB расположен на физическом диске `/dev/hdd3`.

6. Инсталляция ОС

Когда файловая система создана, то на неё можно размещать гостевую ОС несколькими способами

6.1. Инсталляция средствами базовой операционной системы

Этот способ является основным для Xen и выполняется с помощью встроенных утилит системы.

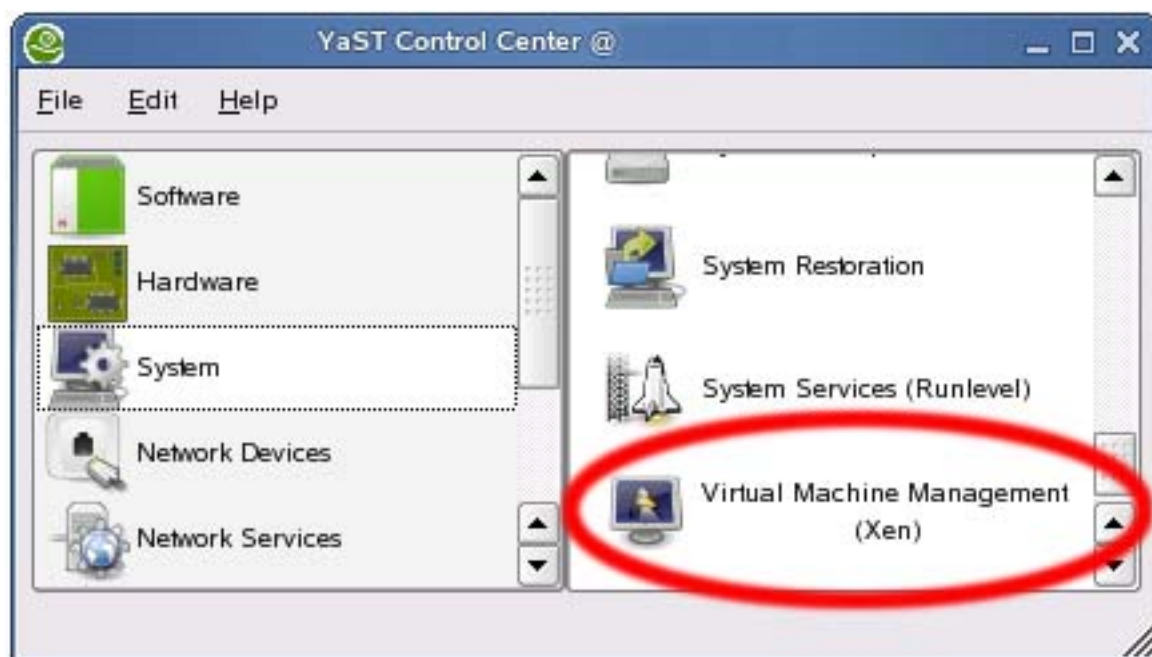


Рис.4. Инсталляция гостевой ОС с помощью встроенных утилит Xen

6.2. Использование готового образа

При создании каждого следующего домена можно взять образ существующего домена за основу и скопировать его в отдел диска, выделенный под новый гостевой домен. Другой вариант - скачать файл с

готовым образом гостевого домена из Интернета, где доступно огромное количество образов дисковых разделов, готовых для использования в качестве основных дисковых разделов в пользовательских доменах Xen.

В этом случае подготовка дискового раздела сводится к простому выкачиванию файла из сети. После того как образ получен, нужно создать конфигурационный файл домена, в котором следует указать этот образ в качестве дискового раздела. Затем запустить получившийся домен. Как правило, в комплекте с образом уже есть готовый к использованию конфигурационный файл домена. Можно использовать его, но только перед этим проверить и при необходимости изменить пути в именах файлов, в настройках ядра, и параметры домена - память, сетевые интерфейсы и т.д.

Этот способ может использоваться как для экспериментов, так и для создания полноценных гостевых доменов. Готовые образы поддаются доработке и расширению.

6.3. Использование диска реальной машины

Под понятием «реальная машина» будем понимать компьютер с единственной операционной системой. Существующую реальную машину можно запускать в виртуализированной среде Xen. (верно и обратное утверждение: пользовательский домен Xen можно очень легко превратить в реальную машину). Это можно сделать следующим образом: скопировать образ дискового раздела реальной машины в файл или дисковый раздел Xen-системы или переставить диск из реальной машины в машину с установленной Xen-системой. После этого следует создать конфигурационный файл домена, в котором указать соответствующие дисковые разделы.

При загрузке нового домена нужно использовать не старое ядро системы, а ядро, специально подготовленное для Xen. Поэтому при замене ядра необходимо скопировать модули нового ядра на диск гостевой системы.

Для этого необходимо выполнить следующие действия (команды):

cd /mnt/vm1 перейти в директорию, в которую подмонтирован диск с новой гостевой системой (в нашем случае - это /mnt/vm1/).

cp /boot/*xen* ./boot скопировать все относящиеся к Xen системе файлы из директории /boot управляющего домена **dom0** в директорию /boot гостевой операционной системы.

cp -ax /lib/modules/*xen* /var/tmp/dirinstall/lib/modules/ скопировать все относящиеся к новому ядру ОС модули из директории /lib/modules управляющего домена **dom0** в директорию /lib/modules гостевой операционной системы.

```
# echo "/dev/hda1 / reiserfs          defaults 1 1" > ./etc/fstab
```

внести изменения в файл **fstab** гостевой операционной системы.

```
# rm -f ./etc/mtab          удалить файл mtab в файловой системе гостевой операционной системы.
```

```
# cd /                      вернуться в корень файловой системы dom0.
```

```
#umount /mnt/vm1          размонтировать файловую систему гостевой операционной системы.
```

После завершения создания гостевой операционной системы можно приступить к созданию ее файла конфигурации и запуска.

6.4. Создание файла конфигурации гостевого домена

Прежде чем запускать дополнительный домен, необходимо создать его конфигурационный файл. В системе для этого Xen есть несколько базовых шаблонов:

/etc/xen/xmexample1 является простым шаблоном, служащим для описания одной единственной VM.

/etc/xen/xmexample2 предназначен для многократного использования на нескольких виртуальных машинах. В этом шаблоне необходимо установить значение переменной **vmid** в командной строке **xm** для каждой из VM.

Для запуска следует скопировать один из этих файлов под новым именем в директорию, в которой будут находиться все конфигурационные файлы. Обычно они расположены в директории **/etc/xen/vm/** и имеют название **vm1 vm2 vm3**. Если подразумевается, что загрузка гостевых доменов будет проходить в автоматическом режиме, при включении компьютера и загрузке управляющего домена, тогда эти файлы следует поместить в директорию **/etc/xen/auto/** и отредактировать его, задав требуемые значения параметров создаваемого гостевого домена.

Параметрами, используемыми для загрузки и конфигурации гостевой системы, являются:

kernel - путь до ядра, откомпилированного для использования с Xen.
Например:

```
kernel = '/boot/vmlinuz-2.6.9-xenU'
```

name – имя создаваемого виртуального домена. Например:

```
name = 'vm1'
```

memory - размер выделяемой домену памяти в мегабайтах. Например:

```
memory = 256
```

root описание реального физического устройства на котором находятся файлы. Например:

```
root = "/dev/hda3"
```

disk - первым значением задается местонахождение корневого раздела гостевого домена и параметр, являющийся ссылкой на физическое устройство (жесткий диск), Вторым значением является указатель на файл подкачки. Например:

```
disk = ['phy:xenvg/vm1,hda3,w',phy:xenvg/sw1,hdb1,w' ]
```

dhcp – параметр, определяющий динамическое выделение IP адреса от сервера DHCP. Например:

```
dhcp='dhcp'
```

vif – значение задает параметры виртуального сетевого интерфейса.

Например, можно отредактировать переменную **vif** для смены MAC адреса виртуального сетевого интерфейса

```
vif = ['mac=00:06:AA:F6:BB:B3']
```

Если вы не установили эту переменную, то **xend** автоматически сгенерирует случайный MAC из неиспользуемого диапазона.

Количество гостевых доменов ограничивается производительностью физической машины, на которую установлен Xen, в основном оперативной памятью и дисковым пространством.

6.5. Загрузка домена

Для загрузки домена возможно использование как встроенных менеджеров управления доменов, так и утилиты **xm**, которая реализует множество команд для управления доменом. Для запуска нового домена используем команду **xm** с опцией **create** (создает внутри физической машины новый гостевой домен в соответствии с параметрами, заданными в файле конфигурации).

В нашем случае запуск нового виртуального домена будет выглядеть следующим образом:

```
# xm create -c /etc/xen/vm/vm1 где /etc/xen/vm путь к конфигурационному файлу vm1.
```

Флаг **-c** указывает утилите **xm** необходимость подключиться к консоли домена после его создания, так как рекомендуется отслеживать процесс загрузки домена. В итоге на экране должно появиться приглашение на вход в систему.

Localhost login:

После входа в систему необходимо настроить сеть. Это можно сделать как стандартными способами, используемыми в Linux системах, например, с помощью утилиты **netconfig**, которая позволяет задать значение сетевых параметров (например, задать статический IP адрес, адрес шлюза, маску и т.п.).

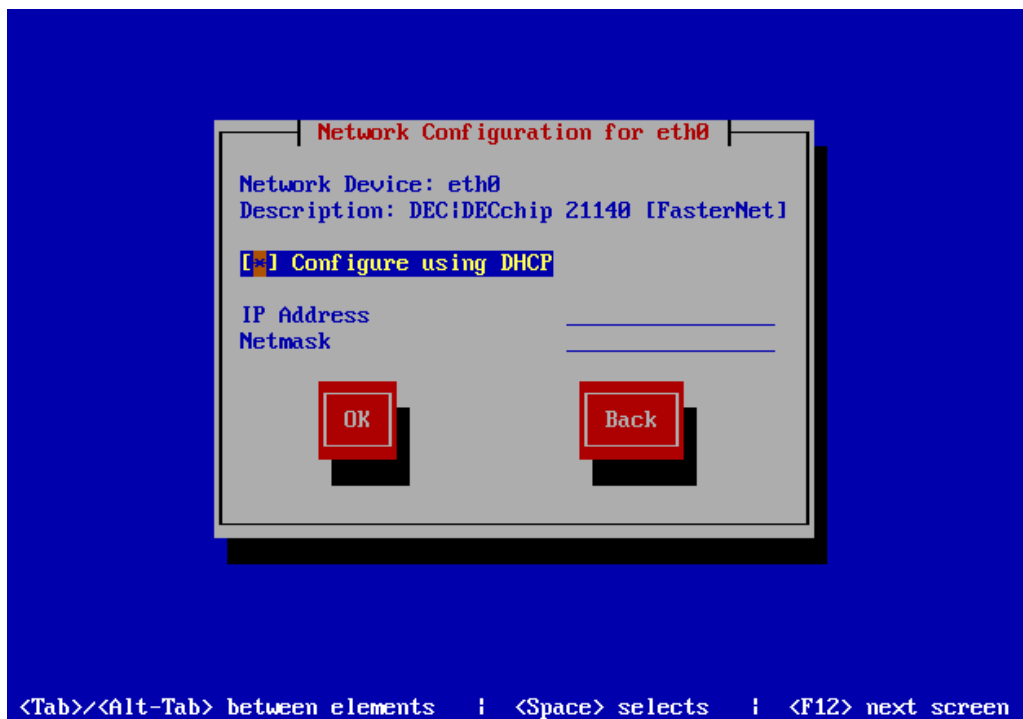


Рис.5. Настройка параметров сети с помощью утилиты **netconfig**

так и внося изменения в следующие системные файлы:

/etc/sysconfig/network задается имя хоста, адрес шлюза и имя домена.

NETWORKING=yes

HOSTNAME= pps-mon4.keldysh.ru

GATEWAY=212.193.97.65

DOMAINNAME=keldysh.ru

/etc/resolv.conf задаются параметры DNS серверов.

search keldysh.ru

nameserver 194.226.57.61

nameserver 194.226.65.3

```

/etc/sysconfig/network-          scripts/ifcfg-eth0 задаются IP
адрес, маска сети и адрес шлюза.
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=static
IPADDR=212.193.97.76
NETMASK=255.255.255.192
GATEWAY=212.193.97.65

```

После завершения процесса настройки параметров сети гостевая операционная система готова к работе.

Наблюдение за состоянием виртуальных доменов из управляющего домена **dom0** возможно с помощью команды **xm list**, показывающей информацию о запущенных гостевых доменах (имя домена, выделенный ему объём оперативной памяти, состояние домена (ready/blocked), время работы)

```
# xm list
```

<i>Name</i>	<i>ID</i>	<i>Mem(MiB)</i>	<i>VCPUs</i>	<i>State</i>	<i>Time(s)</i>
<i>Domain-0</i>	<i>0</i>	<i>980</i>	<i>2</i>	<i>r</i>	<i>1714.7</i>
<i>vm1</i>	<i>1</i>	<i>512</i>	<i>1</i>	<i>b</i>	<i>8580.2</i>
<i>vm3</i>	<i>2</i>	<i>512</i>	<i>1</i>	<i>b</i>	<i>221.9</i>

С помощью других параметров команды **xm** можно останавливать, перегружать виртуальные домены, управлять количеством памяти, отданной системой каждому из доменов, останавливать работу домена в любой момент времени и продолжать её из этой точки, отслеживать загрузку каждого домена.

Заключение

Установка, наладка и последующая эксплуатация монитора виртуальных машин Xen позволили не только лучше понять технологию паравиртуализации, но и сократить количество физических серверов, необходимых для работы кластера. Опыт эксплуатации виртуальных машин Xen позволит в дальнейшем применить эту технологию и в других инфраструктурных проектах.

Применение виртуальных серверов несколько упростило восстановление их при сбоях системы как из-за отказа оборудования, так и в

результате каких-то программных коллизий (это одна из важнейших особенностей работы в отладочном режиме). Кроме того, данная технология позволяет более гибко использовать доступное дисковое пространство.

Но кроме положительных сторон использования данной технологии есть и несколько отрицательных моментов. При сбое в работе управляющего домена следуют отказы всех виртуальных серверов, находящихся под его управлением. При интенсивном использовании серверов уменьшается срок службы физических дисков, следовательно, управляющий домен следует размещать на отдельном физическом диске. Так же к отрицательным моментам можно отнести и тот факт, что для использования в качестве гостевых ОС необходимо применять модифицированные версии ОС. Этот недостаток постепенно упраздняется выходом новых версий ОС, где поддержка Xen входит в базовый состав ОС и не требует дополнительных усилий для запуска в качестве гостевого домена.