



КубГАУ

Виртуализация

доцент кафедры ИС, Петров Антон Александрович



Виртуализация «на пальцах»

- Виртуализация компьютера означает, что можно заставить компьютер казаться сразу несколькими компьютерами одновременно или совершенно другим компьютером.
- Виртуализацией также называется ситуация, когда несколько компьютеров представляются как один отдельный компьютер.
- Другое использование виртуализации заключается в симуляции процессора. Это, так называемая, P-code (или pseudo-code) машина. P-code – это машинный язык, который выполняется на виртуальной машине, а не на реальном оборудовании.



Основной аспект виртуализации

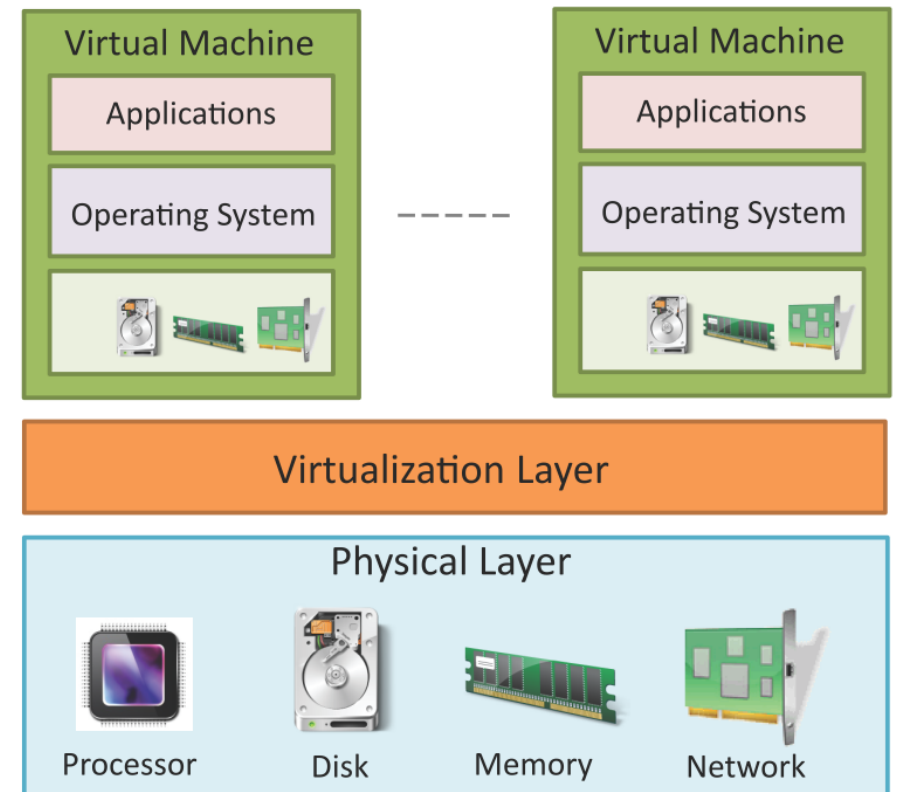
- виртуальные команды переводятся (транслируются) на физические команды основного оборудования.
 - Обычно это происходит динамически.

Виртуализация



КубГАУ

- Виртуализация относится к разделению ресурсов физической системы (такой как процессов, память, сеть и диск) на несколько виртуальных ресурсов.
- Ключевые облачные технологии позволяют объединять разные типы ресурсов в пулы.
- В облаке ресурсы объединяются в пулы для обслуживания нескольких пользователей (multi-tenancy).





КубГАУ

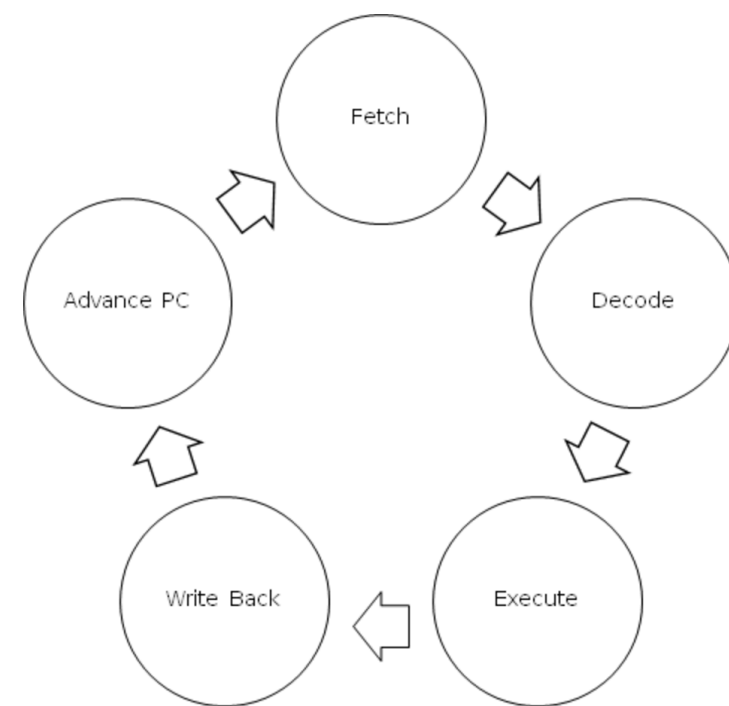
Терминология

- Симуляция
- Эмуляция
 - Интерпретация
 - Двоичная трансляция
- Виртуализация
 - Хозяйская система (Host)
 - Гостевая система (Guest)



Интерпретация на примере ЦП

- Fetch — чтение машинного кода из памяти.
- Decode — декодирование текущей функции, заключённой в инструкции, а также её аргументов — операндов.
- Execute — исполнение функции над аргументами.
- Writeback — запись результатов в память.
- Advance PC — продвижения регистра-указателя инструкций (PC, program counter).





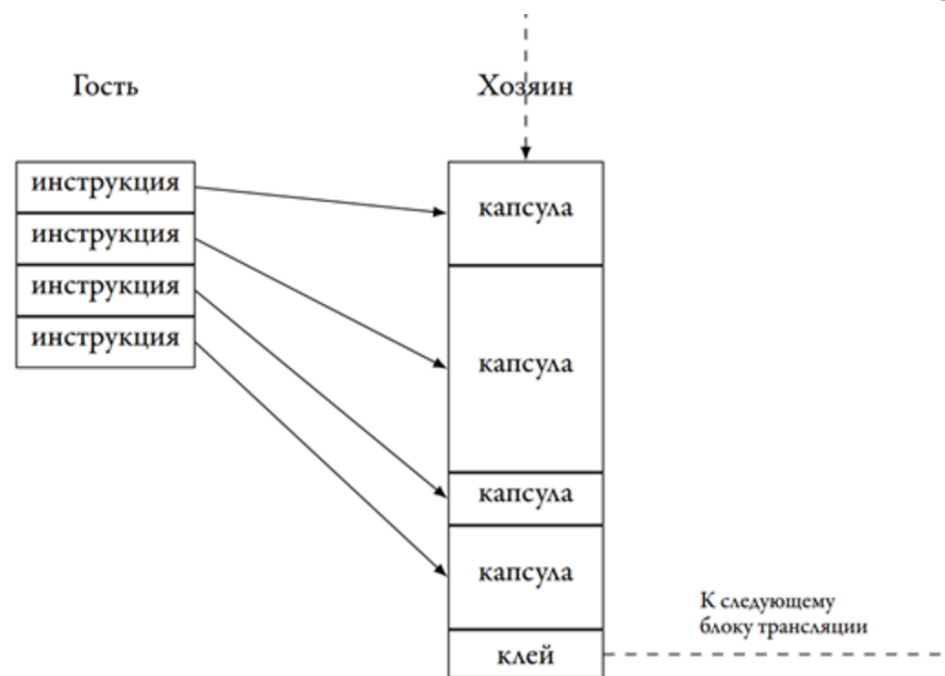
Интерпретация

- Практически всегда первый тип модели, создаваемый для новой архитектуры процессора
- Плюсы
 - Простота
- Минусы
 - Низкая скорость работы



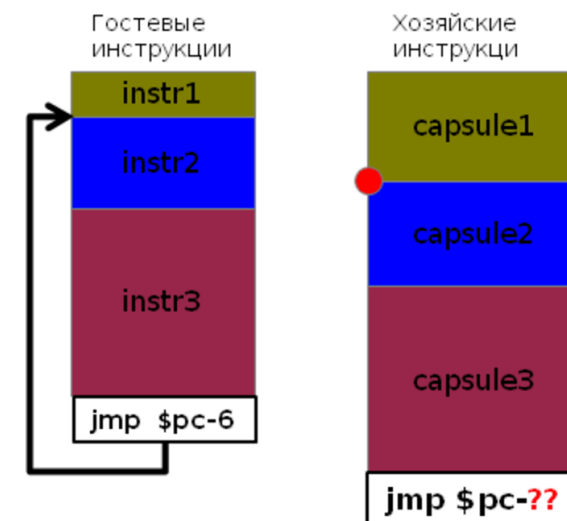
Динамическая трансляция

- Повторяемость кода
- Капсулы хозяйского кода
- Блок трансляции



Проблема динамической трансляции

- Проблема обнаружения кода
 - В оперативной памяти данные (переменные, массивы, константы, строки и т.д.) и код программ хранятся вместе. Двоичная трансляция блоков данных (рассматриваемых как код!) бесполезна: управление на них никогда не будет передано.
 - В архитектурах, допускающих переменную длину инструкций, очень важен адрес, с которого начинается их декодирование. Сдвиг даже на один байт приводит к полной смене смысла последовательности.
 - Результат декодирования зависит от режима процессора. Например, для архитектуры ARM есть фактически два набора инструкций — полный 32 битный и урезанный 16-битный Thumb, переход между которыми происходит с помощью команды **BX**
- Самомодифицирующийся код
 - Устаревание транслированных блоков
- Ограниченность оптимизаций





Прямое исполнение

- Архитектуры гостя и хозяина совпадают (или почти совпадают)
- Проблемы:
 - Доступы к памяти. Адресное пространство гостя занимает лишь часть памяти симулятора. Данные и код симулируемой системы не обязательно будут находиться по тем же адресам, по которым они располагались в реальности.
 - Возврат управления. Как можно «заставить» симулируемое приложение отдать управление обратно симулятору?
 - Привилегированные инструкции. Симулятор работает в непривилегированном режиме пользовательского приложения, а гостевой код может содержать инструкции системных режимов. Попытка их исполнения приведёт к аварийному завершению симулятора.
- Существуют программы *двоичной instrumentation*, позволяющие «незаметно» для гостевого приложения подменять машинный код выбранных инструкций.



Типы виртуализации

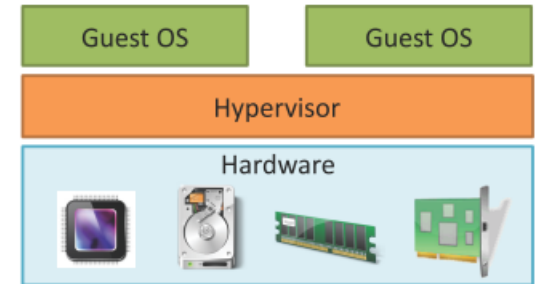
- Полная виртуализация (Динамическая трансляция)
 - При полной виртуализации гостевая операционная системы полностью отделяется от управления инфраструктурой хоста. Гостевая ОС не требует никаких изменений, и не подозревает, что она запущена в виртуальном окружении. Полная виртуализация предоставляет возможность прямого исполнения запросов пользователя, и бинарной трансляции запросов ОС.
- Паравиртуализация
 - В паравиртуализации гостевая ОС модифицируется для коммуникации с гипервизором с целью улучшения производительности и эффективности работы. В ядре гостевой ОС модифицируются не виртуализуемые инструкции на гипервызовы, которые напрямую отправляются в гипервизор.
- Аппаратная виртуализация
 - Аппаратные средства поддержки виртуализации являются опциональным свойством производимого оборудования. Например, средства виртуализации компании Intel (VT-x) и AMD (AMD-V), встраиваемые в соответствующие оборудование. При аппаратной виртуализации определённый набор вызовов отлавливается на уровне оборудования и отправляется напрямую в гипервизор. Следовательно, отпадает необходимость в бинарной трансляции, или пара-виртуализации.

Гипервизор

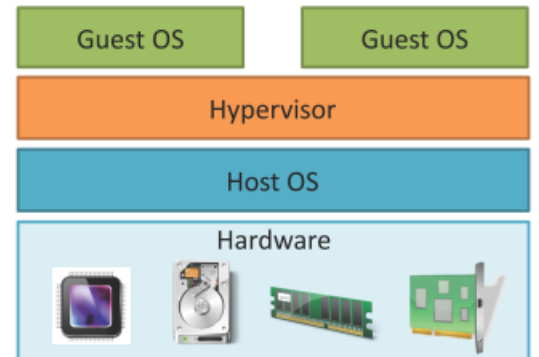


КубГАУ

- Виртуализация представляется гипервизором или монитором виртуальных машин (VMM).
- Гипервизор представляет собой виртуальную платформу управления гостевыми операционными системами (ОС).
- Первый тип гипервизоров
 - Нативные гипервизоры, которые запускаются напрямую на оборудовании хоста, контролируют это оборудования, и осуществляют мониторинг гостевые операционные системы.
- Второй тип гипервизоров
 - Гипервизоры, которые запускаются поверх операционной системы хоста и осуществляют мониторинг гостевой операционной системы.



Type-1 Hypervisor



Type-2 Hypervisor



Гипервизор Xen

- Гостевой гипервизор разработанный в 2003 в Кембриджском университете.
- Использует пять подходов для запуска гостевой VM:
 - Full virtualization
 - paravirtualization
 - Full virtualization with paravirtualization drivers
 - Paravirtualization with full virtualization drivers
 - Paravirtualization in a container
- Первый и Второй тип гипервизора





Гипервизор KVM

- KVM (Kernel-based Virtual Machine)
 - Linux host OS
 - Компонент ядра KVM включен в Linux, начиная с 2.6.20, 2007.
 - Требует аппаратной виртуализации ЦП (Intel VT-x, AMD-V)
 - Использует интерфейс /dev/kvm для:
 - Установки гостевой VM
 - Запуска симулятора I/O
 - Запуска симулятора видео дисплея на хосте
 - Второй тип гипервизора
 - KVM может создавать несколько гостевых VM
 - запускать не модифицированные Linux or Windows образы.





КубГАУ

Гипервизор VirtualBox

- Oracle VirtualBox, разработан компанией Innoteck в 2007, а после поглощён компанией Sun microsystems.
 - Software и hardware техника виртуализации
 - Paravirtualization и full virtualization
 - ОС хоста: Linux, OSX, Windows
 - Гостевые ОС: Linux, OSX, Windows
- Второй тип гипервизора



VirtualBox



Преимущества виртуализации

- Сокращение затрат на приобретение и поддержку оборудования
- Сокращение серверного парка
- Сокращение штата IT-сотрудников
- Простота в обслуживании
- Клонирование и резервирование

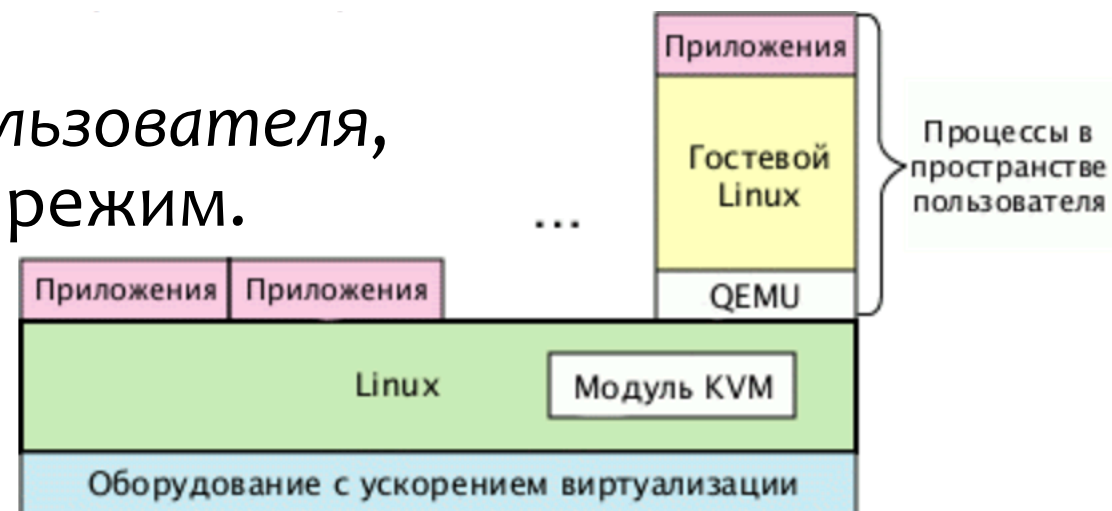
Проекты виртуализации для Linux

Проект	Тип	Лицензия
Bochs	Эмуляция	LGPL
QEMU	Эмуляция	LGPL/GPL
VMware	Полная виртуализация	Проприетарное
z/VM	Полная виртуализация	Проприетарное
Xen	Паравиртуализация	GPL
UML	Паравиртуализация	GPL
Linux-VServer	Виртуализация уровня операционной системы	GPL
OpenVZ	Виртуализация уровня операционной системы	GPL



Linux KVM (Kernel Virtual Machine)

- KVM -- решение для полной виртуализации
- Превращает ядро Linux в гипервизор, используя модуль ядра
- Помимо режима ядра и режима *пользователя*, KVM представляет новый *гостевой режим*.





Подготовка к установке KVM

- `$ egrep -c '(vmx|svm)' /proc/cpuinfo`

```
roger@roger-laptop: ~  
File Edit View Terminal Help  
roger@roger-laptop:~$ egrep -c '(vmx|svm)' /proc/cpuinfo  
4  
roger@roger-laptop:~$
```

```
roger@roger-laptop: ~  
File Edit View Terminal Help  
roger@roger-laptop:~$ /usr/bin/kvm-ok  
INFO: Your CPU supports KVM extensions  
INFO: /dev/kvm exists  
KVM acceleration can be used  
roger@roger-laptop:~$
```



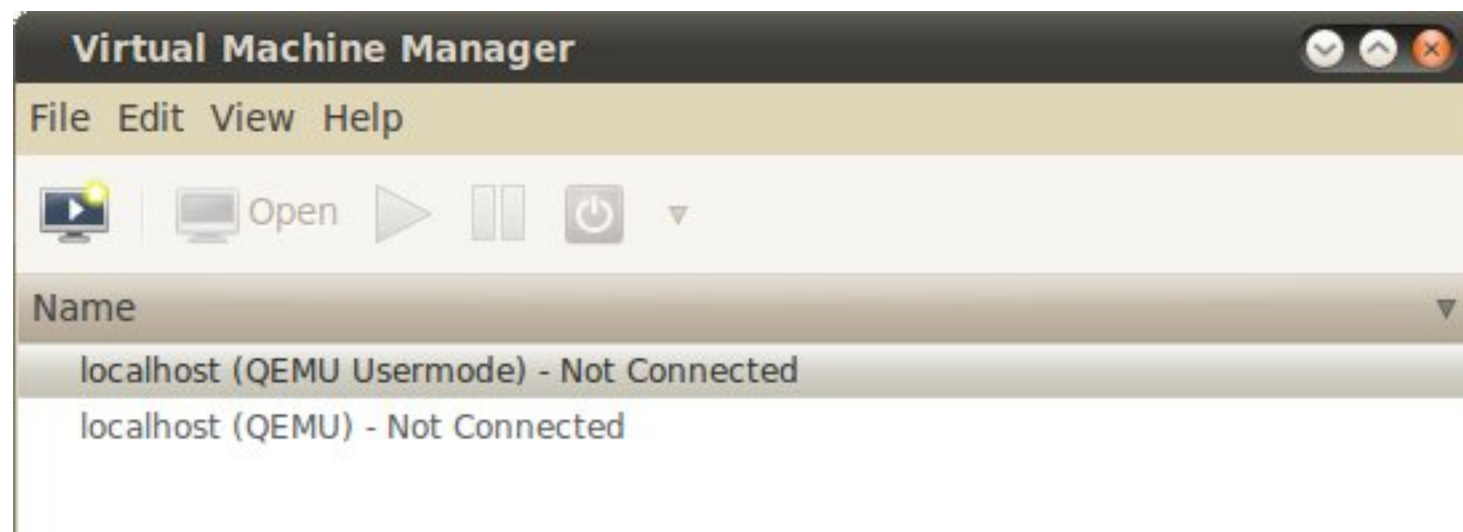
Скачивание и установка KVM

- `$ apt-get install qemu-kvm libvirt-bin`
- `$ apt-get install bridge-utils virt-manager python-virtinst`
- (опционально) `$ apt-get install ubuntu-vm-builder`



Использование VMM

- При первом запуске программы вы увидите две категории, обе не подключенные. Это ссылки на стандартные модули KVM, пока не работающие. Для их использования щелкните правой кнопкой мыши и выберите "connect".



Использование VMM

Add Connection

Hypervisor: QEMU/KVM

Connection: Local

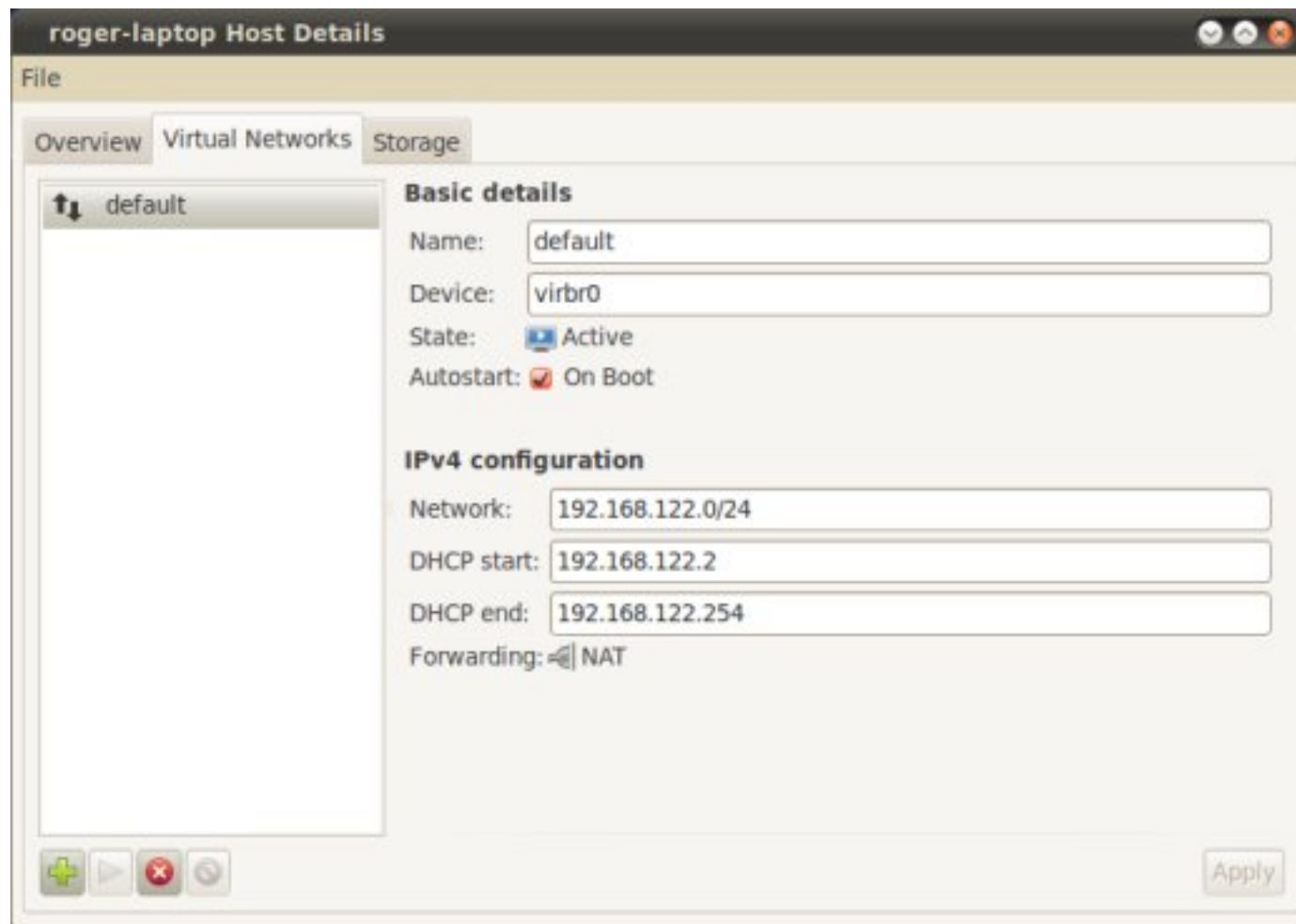
Name

Hostname:

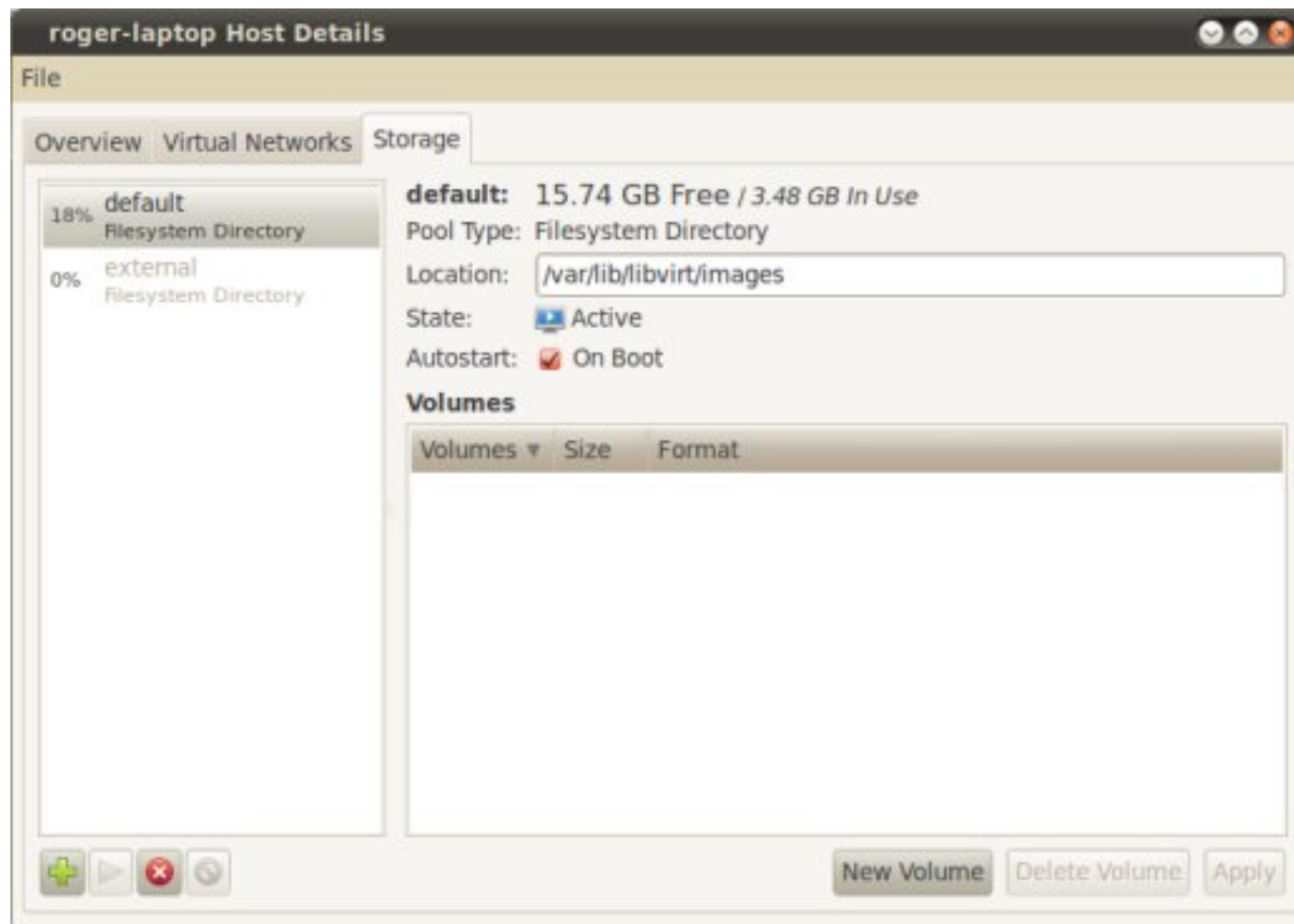
Autoconnect: ☒

Cancel Connect

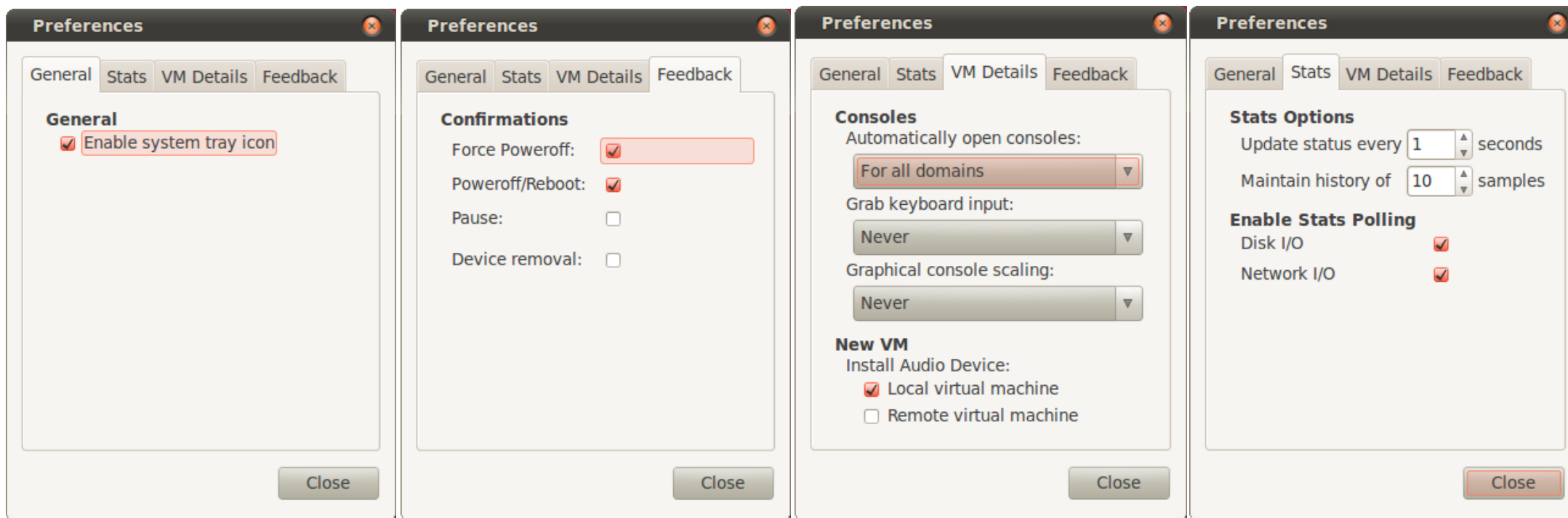
Использование VMM



Использование VMM



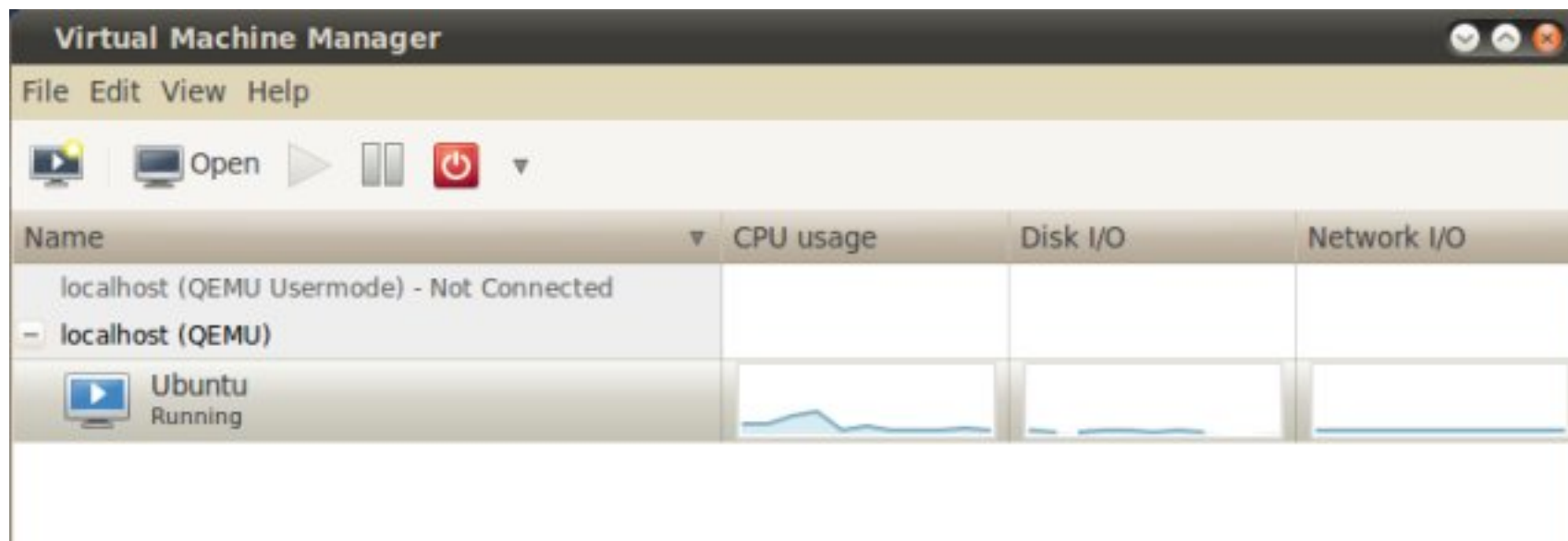
Использование VMM



The image displays four sequential screenshots of the VMM Preferences dialog box, each showing a different tab. Each window has a title bar with 'Preferences' and a close button. The tabs are 'General', 'Stats', 'VM Details', and 'Feedback'.

- General Tab:** Shows the 'General' section with a checked checkbox for 'Enable system tray icon'.
- Confirmations Tab:** Shows the 'Confirmations' section with checkboxes for 'Force Poweroff:', 'Poweroff/Reboot:', 'Pause:', and 'Device removal:'. The first two are checked.
- Consoles Tab:** Shows the 'Consoles' section with settings for 'Automatically open consoles:' (set to 'For all domains'), 'Grab keyboard input:' (set to 'Never'), and 'Graphical console scaling:' (set to 'Never'). It also includes a 'New VM' section with 'Install Audio Device:' checked for 'Local virtual machine'.
- Stats Options Tab:** Shows the 'Stats Options' section with 'Update status every' set to 1 seconds and 'Maintain history of' set to 10 samples. It also includes the 'Enable Stats Polling' section with checkboxes for 'Disk I/O' and 'Network I/O', both of which are checked.

Использование VMM





КубГАУ

Создание виртуальной машины

New VM

Create a new virtual machine
Step 1 of 5

Enter your virtual machine details

Name:

Connection: localhost (QEMU Usermode)

Choose how you would like to install the operating system

☒ Local install media (ISO image or CDROM)

☐ Network Install (HTTP, FTP, or NFS)

☐ Network Boot (PXE)

Cancel Back Forward

Создание виртуальной машины

New VM

Create a new virtual machine
Step 2 of 5

Locate your install media

☐ Use CDROM or DVD

☒ Use ISO image:

OS type: Generic

Version: Generic

Cancel Back Forward



КубГАУ

Создание виртуальной машины

New VM

Create a new virtual machine
Step 2 of 5

Locate your install media

☐ Use CDROM or DVD

☒ Use ISO image:

Browse...

Choose an operating system type and version

OS type: Linux

Version: Generic 2.6.x kernel

Cancel Back Forward

Виртуализация

Создание виртуальной машины

New VM


Create a new virtual machine
Step 4 of 5

☒ Enable storage for this virtual machine

☒ Create a disk image on the computer's hard drive

8.0 GB

14.3 Gb available in the default location

☒ Allocate entire disk now 

☐ Select managed or other existing storage

Browse...

Cancel Back Forward

Создание виртуальной машины

New VM

Create a new virtual machine
Step 5 of 5

Ready to begin installation of **Ubuntu**

OS: Generic 2.6.25 or later kernel with virtio

Install: Local CDROM/ISO

Memory: 1024 MB

CPUs: 2

Storage: 4.0 Gb /home/roger/ubuntu.img

☐ Advanced options

Virtual network 'default' : NAT

☒ Set a fixed MAC address

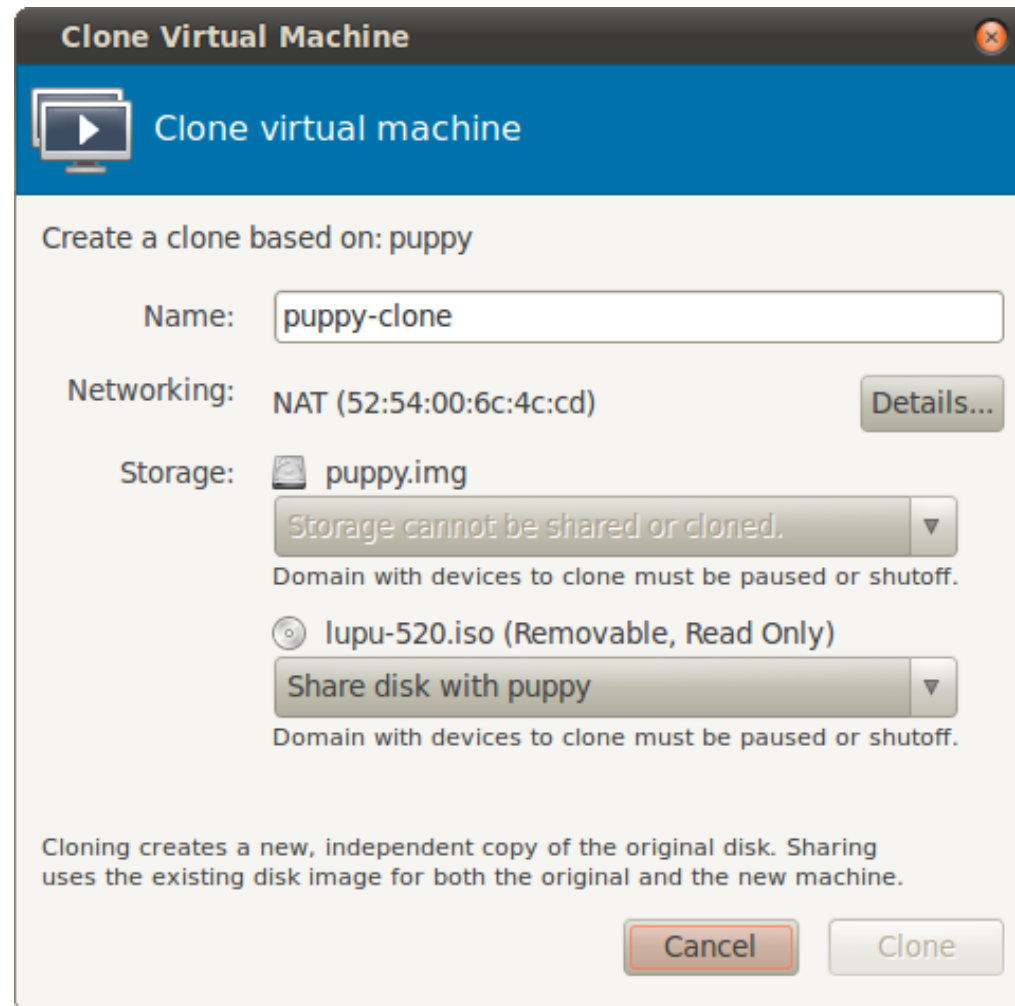
52:54:00:0c:4f:be

Virt Type: kvm

Architecture: i686

Cancel Back Finish

Клонирование виртуальной машины





КубГАУ

Миграция виртуальной машины

Migrate the virtual machine

Migrate 'Ubuntu'

Name: Ubuntu
Original host: roger-laptop
New host:

Migrate offline: ☐

☐ Advanced options

Tunnel migration through libvirt's daemon: ☐

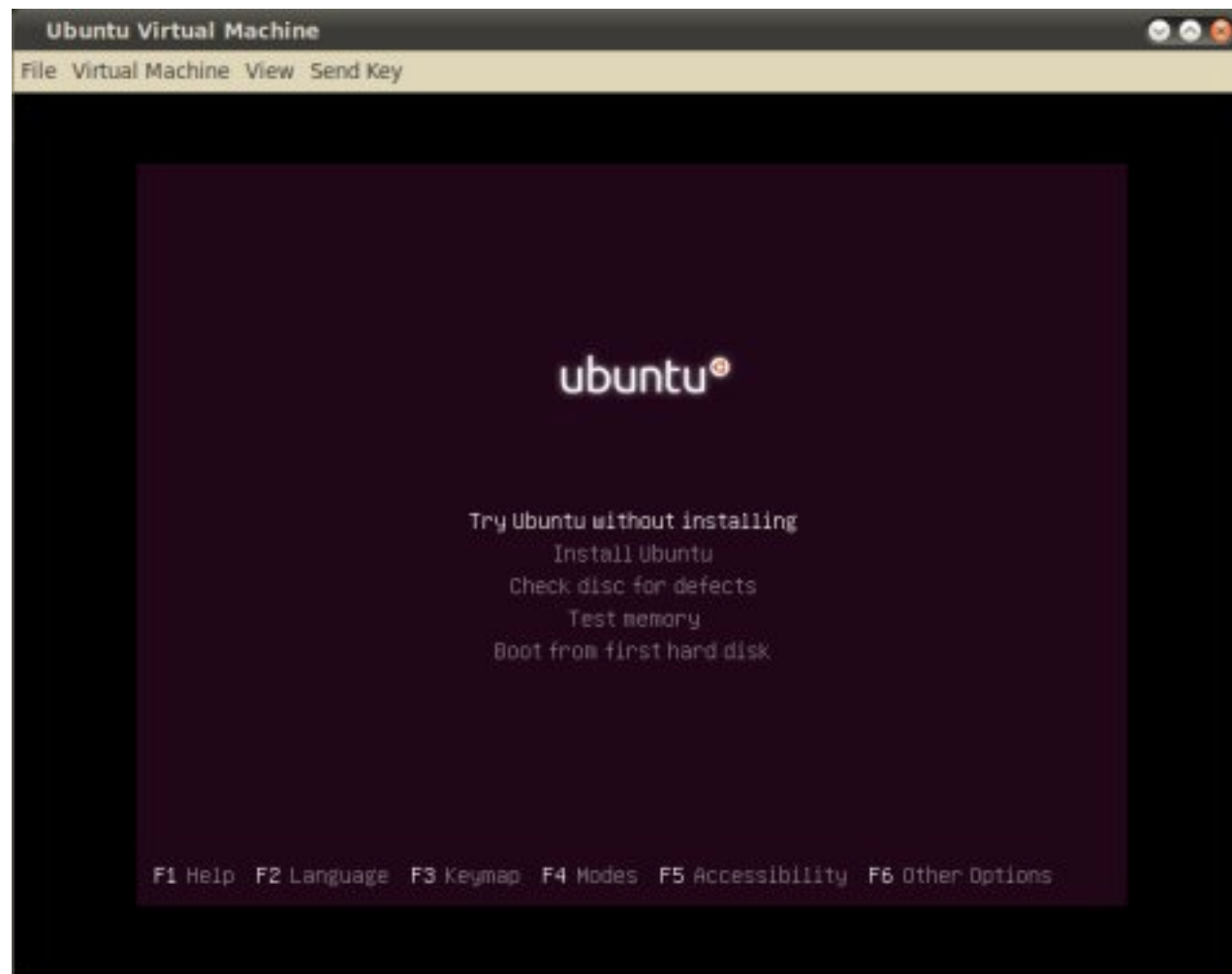
Connectivity

Address: ☐

Port: ☐ 49152

Bandwidth: ☐ 0 Mbps

Запуск виртуальной машины



Контейнерная виртуализация (Лекция)

- OS level virtualization method for running isolated Linux systems on a single host/kernel
 - Released in 2008.
 - CPU, Memory, I/O, network limitations and prioritization for each container without the need of a VM hypervisor.
- Focus:
 - Offer distribution and vendor neutral environments for hosting Linux systems.
 - Offer environments similar to VMs hosted by VMMs, but without the overhead of running separate kernels and/or simulating hardware interfaces.

