

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ
УНИВЕРСИТЕТ имени академика С.П. КОРОЛЕВА»
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

А.М. СУХОВ, Е.С. САГАТОВ

**БАЗОВЫЕ ПРИНЦИПЫ ИНТЕРНЕТ-ТЕЛЕВИДЕНИЯ
И ОРГАНИЗАЦИЯ ИНТЕРНЕТ ТЕЛЕВЕЩАНИЯ
НА БАЗЕ СВОБОДНОГО ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ В ГРИД-СРЕДЕ СГАУ**

*Методического обеспечения по курсу
«Перспективные информационные технологии»
и лабораторный практикум по разделу
«Базовые принципы интернет телевидения»*

САМАРА

2010

АННОТАЦИЯ

Движение к инновационной экономике требует внедрения новых услуг и продуктов, особенно в ИТ сфере. При этом первый, кто предлагает услугу на рынке, оказывается в огромном выигрыше. Настоящий курс предназначен для того, чтобы научиться определять перспективные с точки зрения возможного бизнеса технологии и выводить услуги на их основе на рынок.

Новый рывок в развитии информационных технологий позволил мультимедиа технологиям стать массово доступными. Никого уже не удивляет ни видеоконференцсвязь, ни интернет видео приложения. Вместе с тем, качественное интернет телевидение еще достаточно дорого и требует значительных вычислительных и телекоммуникационных ресурсов. В Самарском государственном аэрокосмическом университете технологии интернет вещания были внедрены в 2007 году, в 2010 году была проведена серьезная реконструкция системы вещания с внедрением грид технологий.

Данный курс предназначен для сотрудников тех операторов связи, которые запускают или в ближайшее время собираются запустить интернет сервисы, связанные с представлением видео контента. Курс содержит как основные теоретические сведения, так и рекомендации по внедрению конкретных технологических решений.

ОГЛАВЛЕНИЕ

| | |
|--|----|
| Аннотация | 2 |
| Основные понятия | 3 |
| Технология вещания | 5 |
| Сжатие видео | 6 |
| Передача видео в сетях протокола TCP/IP | 9 |
| Параметры видео потоков | 12 |
| Потоковые протоколы | 14 |
| Типы файлов для хранения видео | 16 |
| Сетевая инфраструктура | 17 |
| Спектр услуг | 19 |
| Источники доходов | 21 |
| Особенности контента | 22 |
| Лицензирование деятельности | 23 |
| Готовые решения интернет телевидения | 24 |
| VideoLAN свободно распространяемый продукт для организации интернет вещания | 25 |
| Устанавливаем VideoLAN | 28 |
| Прокси сервер | 44 |

Основные понятия

Новый рывок в развитии информационных технологий позволил мультимедиа технологиям стать массово доступными. Никого уже не удивляет ни видеоконференцсвязь, ни интернет видео приложения. Вместе с тем, качественное интернет телевидение еще достаточно дорого и требует значительных вычислительных и телекоммуникационных ресурсов. В Самарском государственном аэрокосмическом университете технологии интернет вещания были внедрены в 2007 году, в 2010 году была проведена серьезная реконструкция системы вещания с внедрением грид технологий.

В истории российского интернет 2007 останется годом появления многочисленных проектов интернет телевидения, однако, лишь некоторым из них суждена яркая и долгая жизнь, а также коммерческий успех. В настоящей лабораторной работе хотелось бы проанализировать полученный опыт, как техническую и экономическую его сторону, так и творческий аспект с тем, чтобы выделить наиболее значительные тенденции таких проектов. При этом авторы опирались на свой собственный опыт по руководству канала научно-образовательного интернет телевидения и исследований в области характеристик аудиовизуального трафика в глобальной сети, а также на многочисленные контакты с ведущими российскими и зарубежными специалистами в области интернет телевидения.

Начать хотелось бы с определения самого понятия интернет телевидения. Под интернет телевидением понимаются такие системы, в которых сигнал, несущий аудиовизуальную информацию, распространяется по сетям протокола TCP/IP. В свою очередь, среди подобных систем можно также выделить два основных подвида: IPTV и, собственно, Интернет ТВ. IPTV предполагает распространение телевизионного сигнала внутри сети одного оператора связи, в связи с чем облегчаются настройки сети и достигается более высокая скорость передачи телевизионного изображения и,

следовательно, его качество, а также упрощаются задачи биллинга. Отличительной чертой IPTV является просмотр телевизионных программ на экране обычного телевизионного приемника, для чего на месте приема должно быть установлено специальное устройство – TV Box, декодирующее приемный сетевой сигнал. В настоящее время все построенные и стоящиеся цифровые кабельные сети используют данный стандарт.

Интернет ТВ предполагает распространение сигнала без ограничений в глобальной сети, при этом достаточно трудно внести какие либо общие настройки и значительно улучшить передачу телевизионного изображения. Кроме того, видео трафик достаточно сильно загружает существующие магистральные сети и, с ростом качества передаваемого видео, такие нагрузки могут стать критическими. Как правило, прием видео сюжетов Интернет ТВ осуществляется на компьютерные мониторы.

Технология вещания

подавляющее большинство существующих сегодня эфирных каналов и кабельных сетей телевидения вещают в аналоговом формате, исключение – спутниковое телевидение, давно перешедшее на цифровые стандарты. Цифровая передача аудиовизуальной информации предполагает проводить все телевизионные операции при помощи последовательности нулей и единиц.

Ниже приведена общая для всех систем интернет телевидения принципиальная схема:

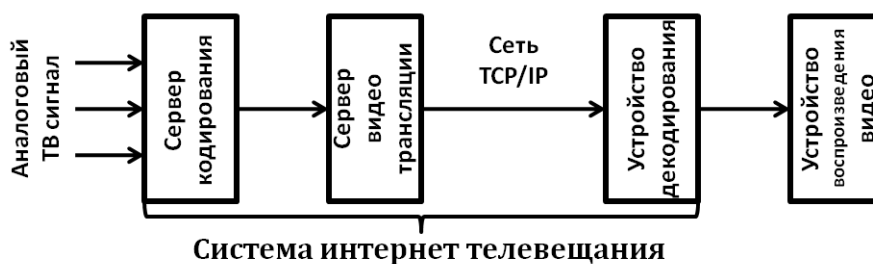


Рис. 1. Принципиальная схема интернет телевидения

Идущий из студии или предварительно записанный телевизионный сигнал сначала поступает на соответствующий интерфейс сервера кодирования видео и аудио. Специальная программа кодировщик (кодек) преобразует аналоговую информацию в цифровую, при этом сжимая ее в соответствии со стандартами MPEG2 или MPEG4. Перечислим также основные типы протоколов сжатия для аудиовизуального ряда, это H.261, H.263 от ITU-T (International Telecommunication Union) и MPEG-1, MPEG-2, MPEG-4 (Moving Pictures Experts Group) от ISO/IEC (International Organization for Standardization / International Engineering Consortium). Имеются также объединенные стандарты H.262/MPEG-2 и H.264/MPEG-4.

Сжатие видео

Реальный мир представляется трехмерным динамическим изображением, которое должно быть передано средствами телевидения. Поскольку динамическое изображение может быть представлено в виде последовательности отдельных картинок, то на данный момент в телевидении передается информация о яркости и цветности двухмерного изображения.

Главная задача телевидения – нахождение способов преобразования изображения объекта для передачи по каналам электросвязи. Электрический канал связи в аналоговом телевидении может передавать в каждый момент времени только одно значение сигнала.

Для решения проблемы преобразования изображения используются два фундаментальных принципа: пространственная и временная дискретизация. Пространственная дискретизация заключается в разбивке всего передаваемого изображения на конечное число дискретных элементов. Элементом изображения (пиксель) называется минимальная деталь изображения, внутри которой яркость и цвет считаются постоянными.

Одновременная передача информации обо всех элементах изображения требует значительной пропускной способности каналов связи. Проблему

переполнения каналов связи решает принцип последовательной во времени передачи информации об отдельных элементах изображения. Кадр это массив всех пикселей, видимых камерой в данный момент времени. Плоский кадр имеет два измерения, которые отвечают за вертикальное и горизонтальное разрешение.

Сжатие видео означает уменьшение количества данных, используемых для представления видеоизображений. Сжатие может быть без потерь, так чтобы результат декомпрессии полностью соответствовал оригиналу, однако такой тип сжатия не позволяет достичь значительных коэффициентов сжатия. Поэтому все существующие на сегодня кодеки реализуют сжатие с потерями.

Наиболее часто применяемая технология, позволяющая повысить степень сжатия, это компенсация движения. Ее использование означает, что последующие кадры в потоке используют похожесть областей в предыдущих кадрах для увеличения степени сжатия. То есть только небольшая часть кадров, называемая ключевыми (I кадр), кодируется полностью и может быть восстановлен без привлечения дополнительных данных. Для восстановления оставшихся кадров кодируется разница между соседними кадрами. Особенности кодирования данной разницы и есть основная суть стандартов на компрессию видео.

На сегодня практически все алгоритмы сжатия видео (например, стандарты, принятые ITU-T или ISO) используют дискретное косинусное преобразование (DCT) или его модификации для устранения пространственной избыточности. Использование большинства методов сжатия (таких как дискретное косинусное преобразование и дискретное вейвлет преобразование) влечет также использование процесса квантования. Квантование может быть как скалярным, так и векторным, тем не менее, большинство практических схем сжатия используют скалярное квантование, вследствие его простоты.

H.261 — стандарт сжатия видео, принятый в 1990 году международной организацией ITU. Первоначально он был разработан для передачи по каналам ISDN, на которых ширина потоков данных кратна 64 *Kbps*. Алгоритм стандарта был предназначен для работы с потоками от 40 *Kbps* и до 2 *Mbps* при очень высоких характеристиках качества связи: потери пакетов не превышают 0,005%, вариация задержки вообще отсутствует. Стандарт поддерживал CIF и QCIF размеры кадров с разрешениями 352x288 и 176x144 соответственно. H.261 предполагает восстановление кадра из предыдущего (Р кадр) методом движущихся векторов и квантования разницы для блоков 8 на 8 пикселей.

H.263 представляет собой развитие стандарта H.261, предыдущей разработки ITU-T — стандарта видеосжатия, и алгоритмов MPEG-1 и MPEG-2. Стандарт H.263 был первоначально разработан ITU-T для передачи видео с постоянной, фиксированной скоростью в разнообразных транспортных телекоммуникационных системах. Стандарт первоначально базировался на H.324 (PSTN и другие системы видеоконференций и голосовой связи), но впоследствии нашёл применение в H.323 (видеоконференции, основанные на RTP/IP), H.320 (видеоконференции, основанные на ISDN). В нем впервые было применено потоковое аудио и видео и специальный тип передачи управляющей информации SIP.

MPEG-2 — название группы стандартов цифрового кодирования видео и аудио сигналов, одобренных ISO — Международной Организацией по стандартизации/IEC Moving Picture Experts Group (MPEG). MPEG-2 используется для общего сжатия движущихся изображений и звука и определяет формат видео-потока, который может быть представлен как три типа кадра — независимо сжатые кадры (I-кадры), кадры сжатые с использованием предсказания движения в одном направлении (P-кадры) и кадры, сжатые с использованием предсказания движения в двух направлениях (B-кадры). Соответствующие группы кадров от одного I-кадра до другого образуют GOP — Group Of Pictures — группу кадров. Обычно

используются потоки в 25 или 29,97 кадров в секунду. MPEG-2 поддерживает видео и в прогрессивной, и в чересстрочной развертке.

H.264, MPEG-4 Part 10, (включая AVC, *Advanced Video Coding*) — стандарт сжатия видео, предназначенный для достижения высокой степени сжатия видеопотока при сохранении высокого качества. Он был создан ITU-T Video Coding Experts Group (VCEG) совместно с ISO/IEC Moving Picture Experts Group (MPEG) в рамках совместной программы Joint Video Team (JVT). ITU-T H.264 стандарт и ISO/IEC MPEG-4 Part 10 стандарт (формально, ISO/IEC 14496-10) технически полностью идентичны. MPEG-4 включает в себя многие функции MPEG-1, MPEG-2 и других подобных стандартов, добавляя такие функции, как поддержка языка виртуальной разметки VRML для показа 3D объектов, объектно-ориентированные файлы, поддержка управления правами и разные типы интерактивного медиа. Стандарт H.264/AVC/MPEG-4 Part 10 содержит ряд новых возможностей, основные из них: многокадровое предсказание кадров, позволяется использование до 32 ссылок на другие кадры, тогда как раннее число ссылок ограничено одним или, в случае В-кадров, двумя кадрами. Компенсация движения с переменным размером блока (от 16x16 до 4x4 пикселя) позволяет крайне точно выделять области движения. Сжатие макроблоков без потерь, гибкие функции чересстрочного сжатия, адаптивный выбор кодеком между размерами блока 4x4 и 8x8, логарифмическое управление длиной шага для упрощения распределения битрейта кодером и упрощенного вычисления обратной длины квантования, новые функции устойчивости к ошибкам и т.д.

Передача видео в сетях протокола TCP/IP

Существует два основных типа доставки аудиовизуальной информации:

- Потокное видео
- Прогрессивный формат доставки

Потокное видео используется в первую очередь для доставки видео в рамках прямых трансляций или эфирно-кабельного телевидения. Оно

предполагает постоянный поток видео данных от сервера видео трансляции. Если скорость такого потока (иначе битрейт) постоянно превышает доступную ширину канала в используемой сети, то полученное изображение не будет отличаться от того, что передает сервер трансляции. Если же пропускной способности сети не достаточно, то картинка будет временами исчезать или вообще не отражаться.

В условиях ограниченности скорости подключения используется прогрессивный способ доставки видео. Он предполагает выделение внутри исходного видео файла небольших по объему независимых фрагментов, которые могут воспроизводиться на приемной стороне по мере их передачи, не дожидаясь пока будет скачан весь файл. При небольшой скорости подключения к сети показ будет происходить отрывками, со значительными перерывами между ними. Если же канал широкий, то показ будет идти не только непрерывно, но и файл будет скачен в опережающем режиме, до завершения трансляции, так что сеть будет незагруженной значительное количество времени. Этот способ интернет видео трансляций используется преимущественно для организации видеоархивов, работающих в режиме видео по требованию.

Следует отметить, что закодированная информация не приспособлена для передачи по пакетным сетям, ярким представителем которых является сеть протокола TCP/IP. Поэтому полученные в результате кодирования данные должны быть разбиты на отдельные пакеты и к этим пакетам необходимо добавить заголовки, определяющие их поведение в сети. В сети Интернет для передачи мультимедийной информации используются потоки (stream) в основе которых лежат RTP (Real Time Protocol) пакеты. Это модернизированные UDP (User Datagram Protocol) пакеты с уникальным идентификатором потока и номером пакета в нем.

Например, раньше исходные транспортные потоки, создаваемые системами MPEG-1 и MPEG-2 (Video and Audio Packetized Elementary Streams, PES),

позволяет перейти от платы по трафику к абонентской или повременной оплате, что является неотъемлемой частью Интернет телевидения.

В настоящее время разработаны различные реализации мультикастинговой рассылки. В IPTV роль разветвителей потоков играют сетевые устройства – маршрутизаторы и коммутаторы, в программное обеспечение которых включена поддержка мультикастинговой рассылки. Для организации интернет телевидения требуется инсталляция дополнительных прокси серверов, которые будут принимать видео потоки и перенаправлять их по запросам пользователей. Такая система рассылки не зависима от провайдеров и называется наложенным мультикастингом. В идеальном случае запрос от пользователя должен перенаправляться к ближайшему прокси серверу, как это сделано в Content Distribution Network (сетях по доставке трафика). Любое программное обеспечение, позволяющее осуществлять интернет трансляции должно включать в себя реализацию прокси сервера, равно как сервер сжатия, сервер потокового видео, плеер на приемной стороне и средства включения полученной картинке в веб среду.

Параметры видео потоков

Следующий вопрос, возникающий при организации Интернет-телевидения, состоит в выборе параметров потокового телевидения. Таких параметров несколько, к ним относятся:

- скорость аудиовизуального потока в *Kbps*,
- размер изображения в пикселях по горизонтали и вертикали,
- тип протокола сжатия,
- частота смены кадров в секунду.

Разрешение видеоизображения, характеризует возможность различать близко расположенные линии в получаемом изображении, измеряется в количестве точек (pixels) по вертикали и диагонали. Например, протокол H.263 предусматривает передачу видео изображений в следующих форматах:

SQCIF (128x96), QCIF (176x144), CIF (352x288), 4CIF (704x576), and 16CIF (1408x1152), CIF это Common Intermediate Format. Основные типы протоколов сжатия для аудиовизуального ряда были перечислены выше.

Естественно, что различные типы вещания предполагают разные скорости аудиовизуальных потоков и для IPTV (кабельного ТВ) они гораздо выше. Скорость потока зависит во многом от типа применяемого оборудования, для передачи стандартной телевизионной картинки (720x576) она варьируется от 3 до 6 Mbps (Мегабит в секунду), хотя может достигать и 1,5 Mbps. Для телевидения высокой четности (1920x1080) необходима доступная полоса в районе 10 Mbps, хотя лидирующие производители достигли 4 Mbps.

При вещании на глобальную сеть параметры видео потоков гораздо скромнее. Так, применяемая нашим партнером, русскоязычной компанией МаксТВ, Флорида, США схема вещания предполагает скорость пока не превышающую 700 Kbps (Килобит в секунду). Эта схема предполагает подключение пользователей из стран с развитой сетевой инфраструктурой (США, Канада, страны ЕС, Австралия). Для вещания в странах СНГ необходимо использовать более скромные параметры потоков (идеально CIF, 225 Kbps), не превышающие 300 Kbps. Следует отметить, что применение новых кодеков позволяет добиться приемлемого качества вещания даже на этих скоростях.

Качество получаемого изображения оценивается в соответствии с рекомендациями ITU-R BT 500 по пятибалльной шкале. Часто эту шкалу называют MOS (Mean Opinion Score). Оценка - удовлетворительно означает прерывистость в движении объектов, а также их нечеткие очертания. Обычно для передачи высококачественного изображения или, по-другому, картинки телевизионного качества с $Q_{MOS} \geq 4$, требуется поток шириной в несколько Мбит/с. Однако далеко не все интернет операторы, эксплуатирующие широкополосные сети, способны предоставить подобную сетевую емкость. Поэтому последние стандарты, такие как H.323 или MPEG-4, позволяют

передавать видео ряд на скорости от 20 Кбит/с, что и используется для трансляции на специализированные мобильные терминалы - смартфоны.

Потоковые протоколы

Потоковое мультимедиа непрерывно получается пользователем от провайдера потокового вещания. Это понятие применимо как к информации, распространяемой через телекоммуникации, так и к информации, которая изначально распространялась посредством потокового вещания (например, радио, телевидение).

Основные протоколы потокового медиа: RealTime Streaming Protocol (rtsp://), Progressive Networks Streaming Protocols (pna://, pnm://), Microsoft Windows Media Streaming Protocol (mms://), а также UDP, HTTP, HTTPS, RTP. Протоколы RTSP, RTP и RTCP специально разрабатывались для передачи мультимедийной информации по сети. Последние два построены на основе UDP.

Датаграмные протоколы, такие как User Datagram Protocol (UDP), отправляют поток медиаинформации как поток отдельных маленьких пакетов. Он прост и эффективен; в то же время, в спецификации протокола нет гарантии доставки данных получателю. Это очень сильно затрудняет поиск и исправление получаемых данных принимающим информацию приложением. При потере данных поток может быть отключен.

Надежные протоколы, такие как TCP, гарантируют корректность получаемых данных клиентов потокового вещания. Однако при большом количестве ошибок при соединении/подтверждении получаемой информации передаваемая информация может стать неактуальной. Это также может вызвать значительные задержки при передаче информации на время, затраченное на пересылку поврежденной информации. Одним из решений данной проблемы является буферизация информации на стороне клиента.

Протокол RTP (Real-time Transport Protocol) работает на транспортном уровне и используется при передаче трафика реального времени. Протокол

был разработан Audio-Video Transport Working Group в IETF и впервые опубликован в 1996 году как RFC 1889, и заменён в RFC 3550 в 2003 году. Протокол RTP переносит в своём заголовке данные, необходимые для восстановления голоса или видеоизображения в приёмном узле, а также данные о типе кодирования информации (JPEG, MPEG и т. п.). В заголовке данного протокола, в частности, передаются временная метка и номер пакета. Эти параметры позволяют при минимальных задержках определить порядок и момент декодирования каждого пакета, а также интерполировать потерянные пакеты. В качестве нижележащего протокола транспортного уровня, как правило, используется протокол UDP.

Установление и разрыв соединения не входит в список возможностей RTP, такие действия выполняются сигнальным протоколом (например, RTSP или SIP протоколом).

Потоковый протокол реального времени (Real Time Streaming Protocol, RTSP), разработанный IETF в 1998 году и описанный в RFC 2326, является прикладным протоколом, предназначенным для использования в системах, работающих с мультимедиа данными. Он позволяет клиенту удалённо управлять потоком данных с сервера, предоставляя возможность выполнения команд, таких как «Старт», «Стоп», а также доступа по времени к файлам, расположенным на сервере.

RTSP не выполняет сжатие, а также не определяет метод инкапсуляции мультимедийных данных и транспортные протоколы. Передача потоковых данных сама по себе не является частью протокола RTSP. Большинство серверов RTSP используют для этого стандартный транспортный протокол реального времени (RTP), осуществляющий передачу аудио- и видеоданных.

Microsoft Media Server (MMS) — протокол мультимедиа вещания корпорации Microsoft, используемый в Microsoft Media Service (ранее назывался NetShow Services). MMS может доставляться через протоколы UDP или TCP. По

умолчанию используется порт UDP/TCP 1755. Изменённая версия HTTP через TCP называется MMSH.

Типы файлов для хранения видео

Закодированная информация с телевизионными передачами может быть сохранена и в дальнейшем быть использована для организации трансляции. Для хранения цифровых аудиовизуальных данных разработаны специальные типы файлов с тем, чтобы максимально сократить размер файла и его битрейт. В настоящее время существует целый ряд конкретных реализаций для серверов хранения видео из общей схемы интернет телевидения. Здесь бы хотелось перечислить наиболее распространенные форматы хранения видео:

- Формат FLV, разработанный фирмой Macromedia, поглощенной впоследствии Adobe
- Формат WMV, созданный Microsoft
- Форматы RV, RMVB, созданные одним из пионеров интернет видео корпорацией Real

Кроме перечисленных форматов есть еще и другие, мало используемые для хранения видеоданных на интернет серверах, такие как avi, matroska, mov, vob и т.д. Эти форматы основаны на контейнерном принципе, при создании файлов можно использовать практически любой кодек, что зачастую приводит к трудностям при просмотре полученного видео.

FLV (Flash Video) - формат файлов, используемый для передачи видео через Интернет. Используется такими сервисами, как YouTube, Google Video, RuTube, Tube.BY, Муви, Obivu и др.

Файлы в формате FLV можно просматривать в большинстве операционных систем, потому что он использует Adobe Flash и плагины к большинству браузеров, и поддерживается многими программами для воспроизведения видео, работающими с помощью DirectShow.

FLV-файл — это битовый поток, который является вариантом видеостандарта H.263. Flash Player 8 и более новые редакции поддерживают потоковое видео On2 TrueMotion VP6. On2 VP6 обеспечивает более качественное изображение, особенно при использовании низкого битрейта. Начиная с Flash Player 9 Update 3 поддерживается новый формат мультимедиафайла ISO Base MPEG-4 Part 12, с новым видеокодеком — H.264. Этот стандарт видеосжатия при том же низком битрейте выдаёт значительно более детализированное и «ясное» изображение, особенно в динамических сценах. Недостаток это повышение требований к вычислительным ресурсам и платные патенты.

Windows Media Video — название системы видео кодирования, разработанной компанией Microsoft для хранения и трансляции видеоинформации в проприетарных форматах Microsoft.

RealMedia — проприетарный стандарт на потоковое вещание и на формат медиафайлов, принадлежащий фирме «RealNetworks Products and Services». С версии 6 начиная, из всех существующих стандартов вещания долго оставался единственным, позволявшим осуществлять произвольную «перемотку» по оси времени лежащих на http сервере файлов, в том числе и при работе через прокси. Это обусловило активное выкладывание пользователями RealMedia файлов на дешёвых и бесплатных хостингах. Из-за приемлемого качества изображения и разборчивости речи при сверхнизких битрейтах видеопотока данный формат встраивается в наиболее распространенные смартфоны.

Сетевая инфраструктура

В этой части работы хотелось бы рассмотреть вопрос о размещении основных серверов, оказывающих услуги интернет телевидения. Для этого сначала выделим основные компоненты подобных проектов. К ним можно отнести:

- Система «видео по требованию», которая предполагает установку, как минимум, двух серверов. Один из серверов предназначен для хранения видео, второй сервер отвечает за управление: доступ пользователей, биллинг, организацию откликов от зрителей и т.д.
- Система прямых интернет трансляций состоит из сервера потокового видео с места трансляции и системы наложенного мультикастинга, представляющего из себя иерархическую систему прокси ретрансляторов, разветвляющих видео поток. Сами ретрансляторы должны устанавливаться на узле с широким обратным каналом.
- Система IPTV вещания предназначена трансляции своей программы в сетях кабельного телевидения. Для этого организуется видео поток непосредственно из аппаратного комплекса вещания, он доставляется при помощи глобальной сети оператору кабельного вещания, который просто ретранслирует его.
- Вещание на сотовые компании и для мобильного ТВ (DVB-H) предполагает организацию видео потоков специального формата, которые совместимы с мобильными стандартами, но принципиальных отличий от предыдущего пункта нет.

Сразу же встает вопрос, а где размещать сервера услуг? Ясно, что часть из них требуется установить непосредственно в программно-аппаратном комплексе вещания. Для этого необходимо подведение широкополосного, желательного оптоволоконного канала связи. Минимальные требования к каналу это его пропускная способность в 2 Mbps (Megabit per second). Но это только минимальная цифра для проведения прямых трансляций. Система видео по требованию и прокси ретрансляторы требуют стартовой десяти мегабитной полосы.

В связи с этим, часть серверов системы интернет телевидения может размещаться в специализированных дата центрах, где предоставляются все необходимые телекоммуникационные услуги: высокая скорость

подключения, место в стойке для размещения оборудования, бесперебойное питание.

Спектр услуг

При помощи технологий интернет телевидения возможно оказание ряда новых, инновационных услуг

- Услуги по предоставлению прямых включений с места события в сетку телевидения. Наша компания участвовала в проекте "Открытый диалог" на телеканале СКАТ (Самара). В рамках данной программы происходит общение лица, приглашенного в студию, с жителями губернии, которые собираются в специально оборудованные помещения, разбросанные по территории области.
- Организация прямых интернет трансляций в глобальную сеть для освещения важнейших событий: съездов, конференций, семинаров, презентаций и т.д. При наличии сети интернет такая трансляция организуется в течение 15 минут. Особых требований к качеству канала нет, достаточно 300 Кбит/с, так как в качестве сети распространения сигнала уже готовая сеть.
- Организация оперативной системы сбора и обработки мультимедийной информации на основе услуг мобильной связи, включая услуги MMS. Запуск информационных сервисов (оперативные новости с места событий, народные новости, информация на дорогах, конкурс клипов-поздравлений и т.д.). Подобная система позволяет организовать принципиально новую службу новостей, которая сможет освещать события с невиданной ранее оперативностью.
- Организация и ведение видеоархива телевизионных программ, как собственных, так и сторонних организаций. Это вид услуг предназначен для расширения аудитории, в том числе и за счет зрителей из других регионов и даже стран.

- Предоставление готовых телевизионных программ со шлюза Интернет провайдерам и сотовым операторам. Программа выдается в виде потокового вещания, поддерживающего мультикастинговую рассылку, параметры потока (размер изображения, скорость потока) могут быть согласованы с оператором. Возможна организация услуги видео по требованию.
- Подготовка потока IPTV для распространения в сетях цифрового кабельного телевидения или в составе пакета международных программ. В рамках узла телекомпании возможно организовать готовый поток данных требуемого формата и доставить его посредством глобальной сети для дальнейшего распространения на сервер потокового видео.
- Помощь в организации студии собственного интернет вещания и сети распространения сигнала информационным холдингам, включая радиостанции, домовым сетям. Возможно сотрудничество в подготовке журналистов.
- Проектирование, строительство и помощь при оформлении разрешительной документации сети Интернет - телевещания.
- Построение систем корпоративного телевещания и видеосвязи, в том числе для проведения селекторных совещаний и дистанционного обучения.

В течение ближайшего года в Самаре, как и многих других крупных городах России, ожидается бум широкополосных интернет подключений и появление множества безлимитных тарифов, причем этот процесс уже начался. Это означает, что потребуется дополнительное информационное наполнение региональных сетей, желательно мультимедийное, генерирующее значительные объемы трафика, что открывает новые направления в медиа бизнесе.

На мой взгляд, каждый из крупных медиа холдингов, должен в ближайшее время создать интернет подразделение, производящее мультимедийный

контент на базе существующих журналистских коллективов. Приблизительная стоимость создания такого подразделения составит от полутора до двух миллионов рублей, включая подготовку кадров и затраты на персонал в течение первого года проекта. Основная проблема – кадровая, готовых специалистов найти очень трудно, а если учесть, что они должны закрывать разные области: администрирование серверов, веб-программирование и редактирование видео, трудно разрешаемая.

Источники доходов

Основным источником дохода телевизионных компаний в России служат доходы от рекламы, в то время как в большинстве стран с развитой рыночной экономикой и устоявшимся правовым полем основной источник финансирования это средства зрителей.

Доходы от рекламы зависят от числа зрителей и ее эффективности. К сожалению, существующие проекты интернет телевидения пока не могут обеспечить ни того, ни другого. Отсутствие же значительных доходов сказывается на качестве предлагаемых программ. Выход из этого замкнутого круга видится в предложении ряда новых, нестандартных услуг.

Прежде всего, это касается рынка прямых трансляций, который совсем не развит из-за недостатка технических возможностей по предоставлению услуг. Прямые трансляции могут быть двух типов:

- Первая из них предполагает трансляцию в прямой телевизионный эфир различных значимых событий: съездов политических партий, различных массовых развлекательных мероприятий, карнавалов, народных гуляний и т.д. То есть тех, у организаторов которых есть деньги.
- Вторая возможность это организация прямых интернет трансляций событий, что гораздо дешевле. В качестве дополнения возможны прямые включения в эфир новостных программ.

Необходимо также поработать на рынке продажи контента. Здесь также несколько направлений для деятельности, основные из них работа с юридическими и частными лицами (оптовый и розничный рынки). Прежде всего, необходимо попробовать договориться о сотрудничестве с крупными компаниями интернет вещания об организации трансляционного потока для включения его в программные пакеты, например, специализированный пакет для русскоязычной диаспоры. Особо стоят сотовые компании, разворачивающие сети 3G и мобильное телевидение стандарта DVB-H.

Рядовые интернет пользователи, хотя и избалованы огромным количеством бесплатного наполнения, иногда готовы заплатить за оперативную и полную информацию, особенно если она носит эксклюзивный характер. Тем более что современные технологии предлагают целый ряд простых платежных систем, начиная с сервиса на базе SMS. При определенном формате запроса со счета снимаются деньги, и ему высылается пароль. Возможны также варианты оплаты по web money, кредитным картам и т.д. Эта услуга может быть популярна для пользователей за границей, при условии предоставления им видео повышенного качества.

Однако при запуске проекта интернет телевидения следует всегда помнить, что первые доходы компания получит не сразу, а до текущей самоокупаемости расходов пройдет не менее года со дня запуска проекта. Поэтому на первоначальном этапе расходы должны быть максимально минимизированы, в том числе и на техническое оснащение компании. Идеально начало проекта интернет телевидения в рамках крупных медиа холдингов, включающих газеты, радиостанции, другие средства массовой информации. В них уже есть журналистские коллективы, службы рекламы, часть инфраструктуры.

Особенности контента

Как оказалось, программная политика в области интернет телевидения должна сильно отличаться от обычного эфирного телевидения. Если IPTV

ориентировано на просмотр с программ на экране обычного телевизора, то интернет телевидение смотрят, как правило, на компьютерных мониторах, что изменяет поведение зрителя.

Уютное расположение на диване способствует длительному просмотру телевизионных программ, в то время как сидение перед компьютерным монитором требует динамического развития событий. Общепринято, что сюжеты интернет телевидения не должны превышать пяти минут. Отсюда вывод, новостные выпуски любой тематики есть наиболее приемлемый формат передачи, также подходят отдельные номера развлекательных программ, любые динамические сюжеты.

Важнейшее условие успешного проекта интернет телевидения – это оперативность подачи информации, используя мобильные технологии возможно добиться практически мгновенного получения мультимедийной информации с места события. При этом следует помнить о том, что интернет телевидение не может пока позволить себе дорогих сюжетов, связанных с работой больших съемочных групп, желательно, чтобы всю техническую работу на месте выполнял журналист при помощи мобильного телефона и простой видеокамеры.

Изменения в политику интернет телевидения придут с новым поколением ТВ боксов, позволяющих навигацию в интернет и, следовательно, просмотр на обычных телевизорах.

Лицензирование деятельности

Предоставление услуг интернет телевидения требует получения ряда лицензий. После консультаций с юристами удалось установить минимальный набор необходимой документации. В него входят:

1. Регистрация собственного средства массовой информации
2. Лицензия на предоставление телематических услуг связи
3. Лицензия на предоставление услуг связи для целей кабельного ТВ
4. Лицензия на телевидение

Если получение первых трех лицензий не вызывает особого труда, то последняя лицензия даже без эфирной компоненты достаточно сложна в получении.

Готовые решения интернет телевидения

Существует целый ряд готовых решений для организации интернет телевидения. Эти решения могут быть проприетарными с продажей лицензий на программный продукт, так и распространяемыми по общедоступной лицензии GNU. Среди коммерческих программных продуктов лидируют решения от фирм Microsoft, RealNetworks, Adobe, стоит упомянуть и российскую разработку Видикор. Однако стоит выделить свободно распространяемый продукт VideoLAN, ставший во многом стандартом для трансляций.

В принципе, для реализации схем вещания, как это показано на Рисунках 2 и 3 необходимо наличие четырех программных продуктов:

1. Сервера кодирования для оцифровки видео
2. Сервера потокового видео для разбиения видео на пакеты и их пересылке по сети
3. Прокси сервера для организации наложенного мультикастинга
4. Плеера на приемной стороне

Все эти четыре компонента реализованы у пионера потокового видео компании RealNetworks: RealProducer осуществляет оцифровку, HelixServer организует видео поток, HelixProху перенаправляет запросы, а RealPlayer воспроизводит видео.

Остальные компании пошли путем объединения функциональности, так Microsoft объединил серверы и представил Expression Encoder, также три сервера объединены в одном продукте и у компании Видикор.

Но дальше всех пошли разработчики VideoLAN, которые объединили все четыре продукта в один, который используется, в том числе и в качестве

программы для просмотра (плеера). В следующих разделах будет подробно рассмотрен этот программный продукт.

VideoLAN свободно распространяемый продукт для организации интернет вещания

VideoLAN - многофункциональный комплекс, портированный практически под все операционные системы, поддерживающий множество протоколов, форматов и контейнеров, который можно использовать и как локальный аудио/видеоплеер, и как сервер трансляции (см. рис. 3).

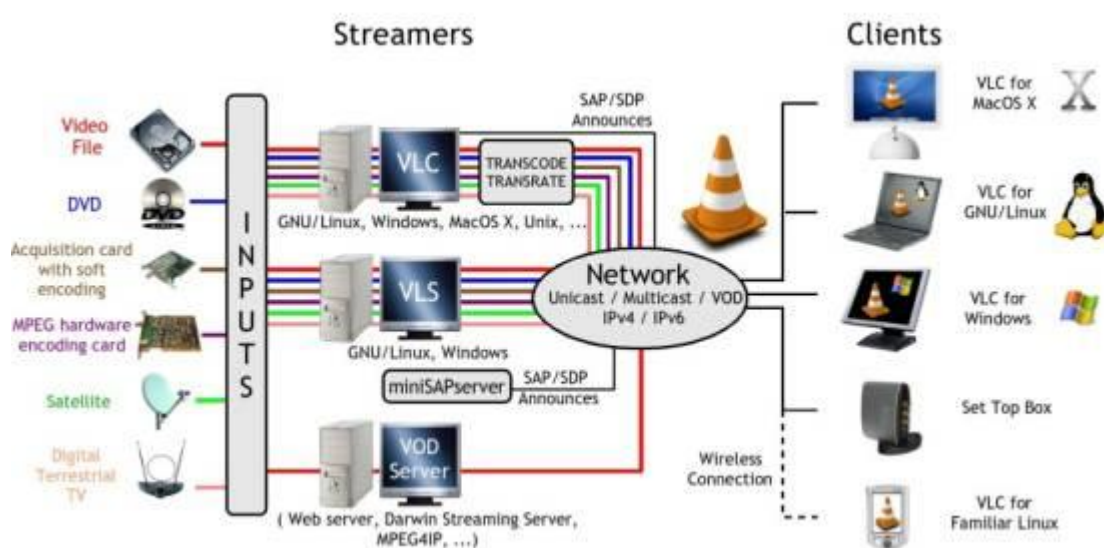


Рисунок 3. Потокное вещание в локальных/глобальных сетях.

Обзор основных возможностей

VideoLAN - это некоммерческий проект, бесплатную версию которого (вместе с исходными текстами и готовыми бинарными сборками) всегда можно скачать с официального сервера <http://www.videolan.org/>.

Клиентская и серверные части исправно работают под Linux, Windows, Mac OS X, BeOS, xBSD, Solaris, Familiar Linux, Yopy/Linupy и QNX, однако их функциональность различна и в зависимости от выбранной платформы варьируется в очень широких пределах (подробнее см. рис. 2).

| Interfaces and control | | | | | | |
|------------------------|-----------|-------|-----------|--------|-----------|---------|
| | Windows | X | Linux | OS | Mac | Android |
| Default | WxWidgets | Cocoa | WxWidgets | Native | WxWidgets | GPE |
| Qt 4 ¹¹ | ✓ | ✗ | ✓ | - | ? | ✗ |
| Skins | ✓ | ✗ | ✓ | ✗ | ✓ | ✗ |
| Web | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Telnet | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Command line | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Infrared | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ |

| Miscellaneous | | | | | | |
|--------------------------------|---------|---|-------|----|-----------------|---------|
| | Windows | X | Linux | OS | Mac | Android |
| SAP/SDP announces | ✓ | ✓ | ✓ | ✗ | ✓ | ✗ |
| Bonjour protocol | ✗ | ✓ | ✓ | ? | ? | ? |
| Nozilla/Firefox plugin | ✓ | ✓ | ✓ | ✗ | ✓ | ✗ |
| ActiveX plugin | ✓ | - | - | - | - | - |
| SVCD Menu | ✓ | ✗ | ✓ | ✗ | ✓ | ✗ |
| Localization | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| CD-Text ¹¹ | ✓ | ✗ | ✓ | ✗ | ✓ | ✓ |
| CDDB CD info | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ |
| IGMPv3 ¹² | ✓ | ✗ | ✓ | ✗ | ✓ | ✓ |
| IPv6 ¹² | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ |
| MLDv2 ¹³ | ✓ | ✗ | ✓ | ✗ | ✓ | ✓ |
| CPU acceleration ¹⁴ | ✓ | ✓ | ✓ | ✓ | ✓ ¹⁵ | ✗ |

Рисунок 4. Возможности программы VideoLAN на каждой из поддерживаемых ею платформ.

Поддерживаются следующие входные форматы данных: MPEG-1, MPEG-2, MPEG-4/DivX (считываемые с локального жесткого диска или CD/DVD); "настоящие" DVD и VCD; спутниковые карты, работающие по стандарту (DVB-S); потоковое видео, "упакованное" в MPEG-1, MPEG-2 и MPEG-4 (то есть, VideoLAN может работать не только как сетевой транслятор, но и как **ретранслятор** чужого контента с возможностью сохранения последнего на жесткий диск).

В настоящий момент реализованы два основных протокола трансляции: **Unicast** ("узконаправленное" вещание с доставкой контента только одному целевому узлу) и **Multicast** (групповая трансляция с доставкой одного и того же контента множеству узлов). Также (формально) имеется возможность широковещательной рассылки контента всем узлам локальной сети (для этого достаточно указать в качестве целевого IP-адреса 255.255.255.255), но с высокой степенью вероятности она будет задавлена брандмауэрами и маршрутизаторами, так что без их радикальной перестройки сеанс вещания не состоится даже в рамках локальной сети.

Еще имеется ограниченная поддержка видео-по-требованию (Video-on-Demand или, сокращенно, VoD) с возможностью выбора контента по HTTP

или TELNET интерфейсам, однако эта возможность обычно используется исключительно администраторами для удаленного управления сервером трансляции.

Кстати, об интерфейсах. VideoLAN поддерживает широкий ассортимент, способный удовлетворить даже самых изысканных гурманов: GUI, NCUSERS, командная строка, HTTP и даже плагины для некоторых популярных браузеров.

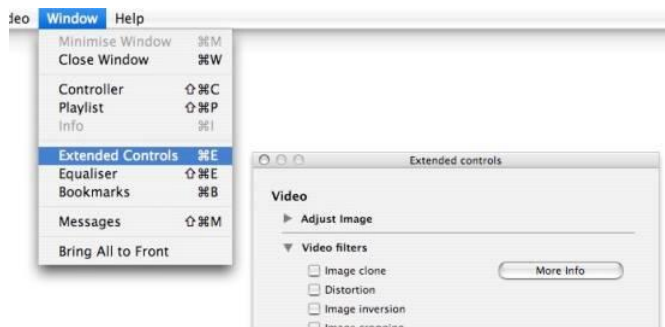


Рисунок 5. Графический интерфейс программы VideoLAN.



Рисунок 6. Текстовый NCURSES интерфейс.

Контейнеры, в которые помещается транслируемый поток, зависят от типа трансляции, допустимые комбинации которых перечислены в таблице на рис.

5. Естественно, все это хозяйство работает как с IPv4, так и с IPv6.

| | UDP | RTP | MPEG | MMS2 | File |
|-----------|-----|-----|------------|------|------------|
| ES | ✗ | ✓ | ✓ | ✗ | ✓ |
| TS | ✓ | ✓ | ✓ | ✗ | ✓ |
| QoS | ✗ | ✗ | ✓ | ✗ | ✓ |
| ASF | ✗ | ✗ | ✓ | ✓ | ✓ |
| MMS | ✗ | ✗ | ✗ | ✗ | ✓ |
| QuickTime | ✗ | ✗ | ✗ | ✗ | ✓ |
| Raw | ✓ | ✓ | ✓ | ✗ | ✓ |
| MP3EG | ✗ | ✗ | 🟢 SVT only | ✗ | 🟢 SVT only |

Рисунок 7. Допустимые комбинации протоколов трансляции с контейнерами, в которые упаковывается транслируемый медиа-поток.

Устанавливаем VideoLAN

Установка VLC производится на самую новую на 08.11.2010г. версию операционной системы Debian GNU/Linux 5.0.6 со всеми обновлениями. Установка производилась в базовом серверном варианте без графической среды и каких-либо дополнительных компонентов. После установки никаких изменений не производилось. Производится первый запуск системы. Вход выполнен с административными правами.

Определяем, установлен ли Midnight Commander. По умолчанию в Debian GNU/Linux он не установлен.

```
debian:~# mc
-bash: mc: команда не найдена
```

Для установки Midnight Commander и любых других пакетов, которые нам понадобятся в дальнейшем, воспользуемся утилитой aptitude.

```
debian:~# aptitude_
```

Aptitude — оболочка для Advanced Packaging Tool, части системы управления пакетами в операционной системе Debian и её производных. Имеет псевдографический интерфейс и интерфейс командной строки.

```
Действия Откат Пакет Решатель Поиск Параметры Окна Помощь
C-T: Меню ?: Справка q: Выход u: Обновление g: Загр/Устан/Удал пакетов
aptitude 0.4.11.11
--- Установленные пакеты (234)
--- Неустановленные пакеты (22763)
--- Виртуальные пакеты (2473)
--- Задачи (732)

Эти пакеты уже установлены на вашем компьютере.
В этой группе содержится 234 пакета.
```

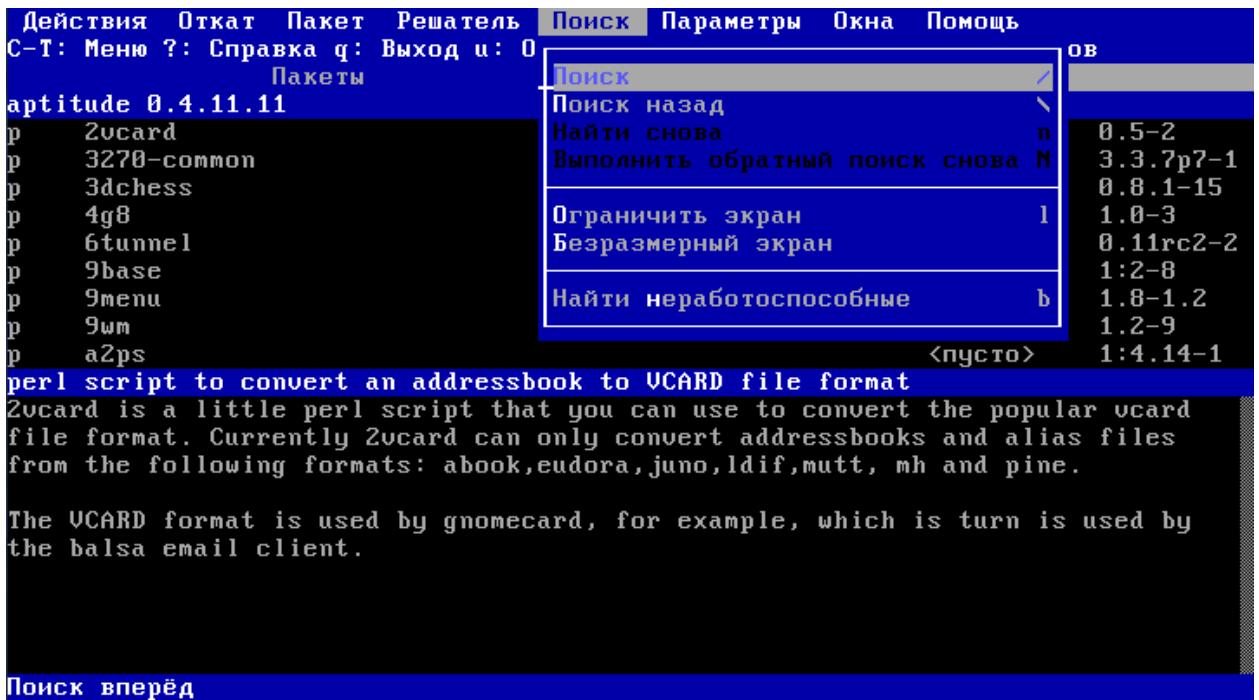
Для удобства поиска необходимых пакетов, выйдем в меню программы, используя сочетание клавиш Ctrl+T, и откроем плоский список пакетов.

```
Действия Откат Пакет Решатель Поиск Параметры Окна Помощь
C-T: Меню ?: Справка q: Выход u: Обновление g: Загр/Устан/Удал пакетов
aptitude 0.4.11.11
--- Установленные пакеты (234)
--- Неустановленные пакеты (22763)
--- Виртуальные пакеты (2473)
--- Задачи (732)
Следующее f6
Предыдущее f7
Закреть q
Новое окно Пакета
Проверка рекомендаций
Новый плоский список пакетов
Новый Debtags браузер
Новый браузер категорий
Пакеты

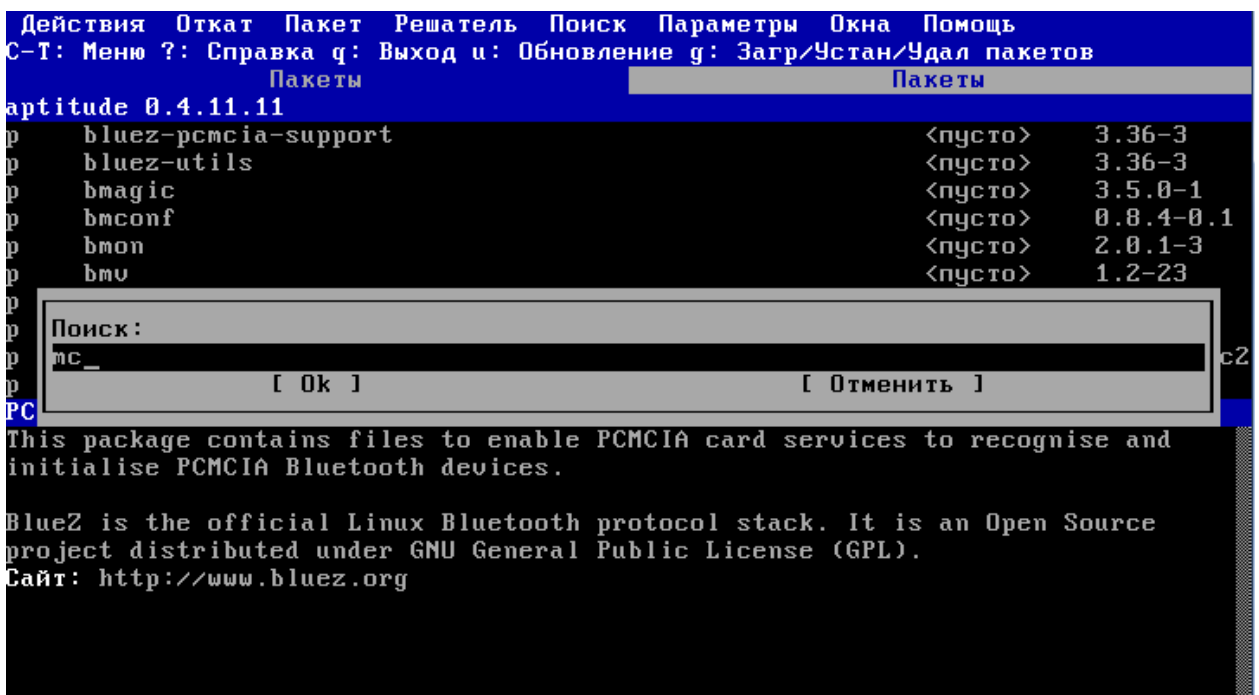
Эти пакеты уже установлены на вашем компьютере.
В этой группе содержится 234 пакета.

Показать все пакеты в системе одним списком без категорий
```

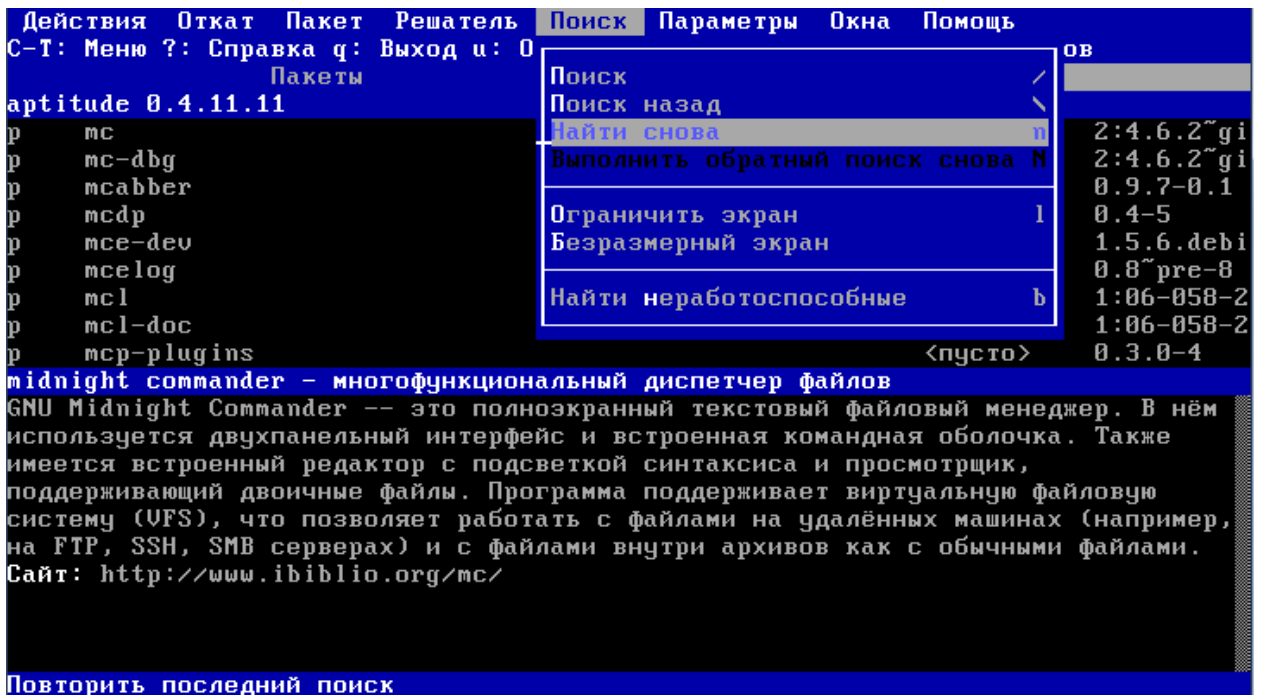
Проведем поиск по плоскому списку нажав клавишу “/”.



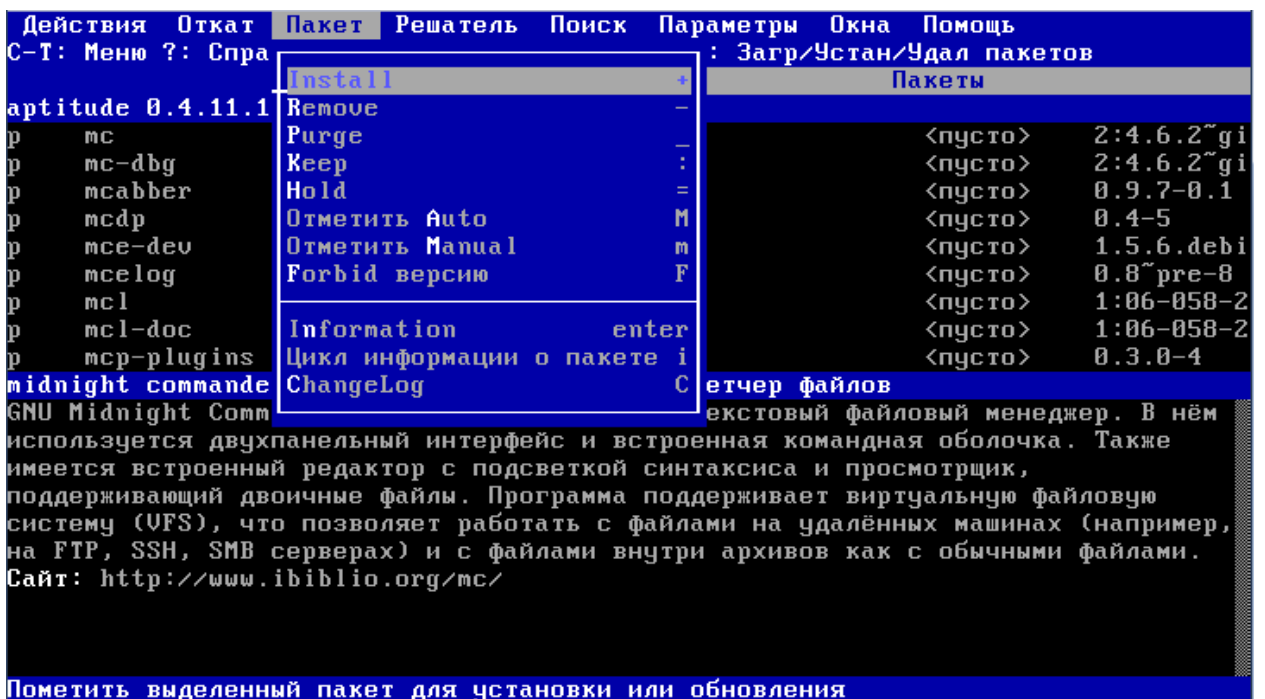
Необходимо ввести несколько букв из искомого названия пакета.



Для поиска следующего совпадения жмем клавишу “n” пока не найдем необходимый пакет. Все пакеты в плоском списке упорядочены по алфавиту.



Устанавливаем курсор на пакете, который хотим установить и жмем клавишу “+” либо Install в меню.



Пакет mc выделен зеленым цветом для установки.

```

Действия Откат Пакет Решатель Поиск Параметры Окна Помощь
C-T: Меню ?: Справка q: Выход u: Обновление g: Загр/Устан/Удал пакетов
Пакеты
aptitude 0.4.11.11 Будет использовано 10,9МБ д DL размер: 3598кВ
pi mc +6402кВ <пусто> 2:4.6.2~gi
p mc-dbg <пусто> 2:4.6.2~gi
p mcabber <пусто> 0.9.7-0.1
p mcdp <пусто> 0.4-5
p mce-dev <пусто> 1.5.6.debi
p mcelog <пусто> 0.8~pre-8
p mcl <пусто> 1:06-058-2
p mcl-doc <пусто> 1:06-058-2
p mcp-plugins <пусто> 0.3.0-4
p mcpp <пусто> 2.7.1
midnight commander - многофункциональный диспетчер файлов
GNU Midnight Commander -- это полноэкранный текстовый файловый менеджер. В нём
используется двухпанельный интерфейс и встроенная командная оболочка. Также
имеется встроенный редактор с подсветкой синтаксиса и просмотрщик,
поддерживающий двоичные файлы. Программа поддерживает виртуальную файловую
систему (VFS), что позволяет работать с файлами на удалённых машинах (например,
на FTP, SSH, SMB серверах) и с файлами внутри архивов как с обычными файлами.
Сайт: http://www.ibiblio.org/mc/

```

Мы можем отметить другие пакеты, которые хотим установить, удалить, либо обновить. Затем необходимо нажать клавишу “g” или выбрать пункт меню Установить/удалить пакеты.

```

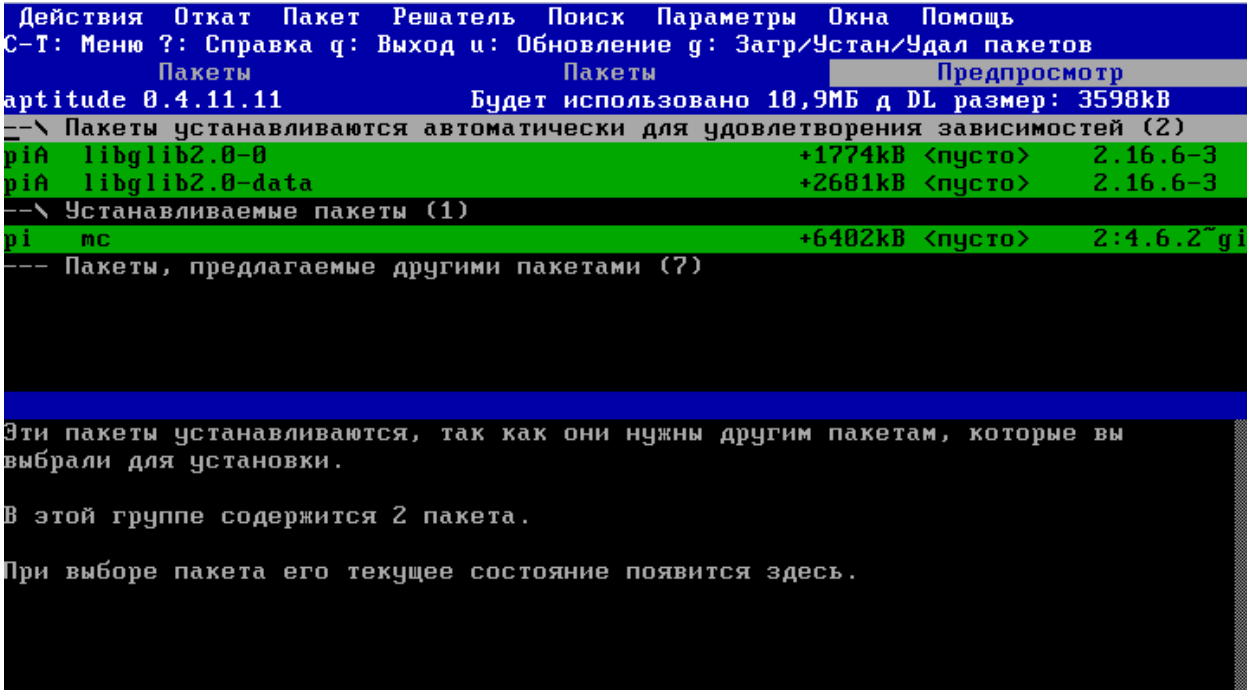
Действия Откат Пакет Решатель Поиск Параметры Окна Помощь
Установить/удалить пакеты g
Обновить список пакетов u
Пометить обновляемые U
Забудь о новых пакетах f
Отменить все незаконченные действия
Очистить кеш пакетов
Очистить устаревшие файлы
Играть в сапёра
Стать суперпользователем
Выйти Q
вление g: Загр/Устан/Удал пакетов
Пакеты
использовано 10,9МБ д DL размер: 3598кВ
+6402кВ <пусто> 2:4.6.2~gi
<пусто> 2:4.6.2~gi
<пусто> 0.9.7-0.1
<пусто> 0.4-5
<пусто> 1.5.6.debi
<пусто> 0.8~pre-8
<пусто> 1:06-058-2
<пусто> 1:06-058-2
<пусто> 0.3.0-4
<пусто> 0.3.0-4
ный диспетчер файлов
ранный текстовый файловый менеджер. В нём
и встроенная командная оболочка. Также
имеется встроенный редактор с подсветкой синтаксиса и просмотрщик,
поддерживающий двоичные файлы. Программа поддерживает виртуальную файловую
систему (VFS), что позволяет работать с файлами на удалённых машинах (например,
на FTP, SSH, SMB серверах) и с файлами внутри архивов как с обычными файлами.
Сайт: http://www.ibiblio.org/mc/
Выполнить все незаконченные установки и удаления

```

Aptitude попытается самостоятельно удовлетворить все зависимости для выбранных пакетов. К примеру, для работы программы mc требуется библиотека libglib2.0. Aptitude автоматически добавила для установки пакеты, содержащие данную библиотеку. Аналогично для удовлетворения

зависимостей aptitude может удалить конфликтующие, старые или неостребованные пакеты, обновить либо переустановить пакеты. Бывают случаи, когда утилита находит несколько вариантов удовлетворения зависимостей. Тогда выбирается вариант по умолчанию, который можно изменить на данном экране. Последней строкой отображается свернутый список дополнительных пакетов, которые предлагаются для установки другими пакетами. К примеру, для mc это различные архиваторы. Без них mc работоспособен, но не работает с архивами.

Пакет bzip2 необходим для распаковки архива исходных кодов VLC.



```
Действия Откат Пакет Решатель Поиск Параметры Окна Помощь
С-Т: Меню ?: Справка q: Выход u: Обновление g: Загр/Устан/Удал пакетов
Пакеты Пакеты Предпросмотр
aptitude 0.4.11.11 Будет использовано 10,9МБ д DL размер: 3598кВ
--\ Пакеты устанавливаются автоматически для удовлетворения зависимостей (2)
piA libglib2.0-0 +1774кВ <пусто> 2.16.6-3
piA libglib2.0-data +2681кВ <пусто> 2.16.6-3
--\ Устанавливаемые пакеты (1)
pi mc +6402кВ <пусто> 2:4.6.2~gi
--- Пакеты, предлагаемые другими пакетами (?)

Эти пакеты устанавливаются, так как они нужны другим пакетам, которые вы
выбрали для установки.

В этой группе содержится 2 пакета.

При выборе пакета его текущее состояние появится здесь.
```

После того, как все зависимости удовлетворены снова нажимает кнопку “g”, либо выбираем пункт меню Установить/удалить пакеты.

```

Действия  Откат  Пакет  Решатель  Поиск  Параметры  Окна  Помощь
Установить/удалить пакеты  g  вление g: Загр/Устан/Удал пакетов
Обновить список пакетов  и  акеты  Предпросмотр
Пометить обновляемые  U  использовано 10,9МБ д DL размер: 3598кВ
Забудь о новых пакетах  f  ски для удовлетворения зависимостей (2)
Отменить все незаконченные действия
Очистить кеш пакетов
Очистить устаревшие файлы
Играть в сапёра
Стать суперпользователем
Выйти  Q  +1774кВ <пусто> 2.16.6-3
+2681кВ <пусто> 2.16.6-3
+6402кВ <пусто> 2:4.6.2~git
тами (?)
<пусто> 3.10.22-6
<пусто> 1.0.5-1+le
<пусто> 1.0.4-1
й работы системы, но они могут
какими-то другим программам, которые вы
устанавливаете в данный момент.
В этой группе содержится 7 пакетов.
При выборе пакета его текущее состояние появится здесь.
Выполнить все незаконченные установки и удаления

```

```

Выбор ранее не выбранного пакета libglib2.0-0.
(Чтение базы данных... на данный момент установлено 21394 файлов и каталогов.)
Распаковывается пакет libglib2.0-0 (из файла ../libglib2.0-0_2.16.6-3_i386.deb)
...
Выбор ранее не выбранного пакета libglib2.0-data.
Распаковывается пакет libglib2.0-data (из файла ../libglib2.0-data_2.16.6-3_all
.deb)...
Выбор ранее не выбранного пакета mc.
Распаковывается пакет mc (из файла ../mc_4.6.2~git20080311-4_i386.deb)...
Обрабатываются триггеры для man-db ...
Настраивается пакет libglib2.0-0 (2.16.6-3) ...
Настраивается пакет libglib2.0-data (2.16.6-3) ...
Настраивается пакет mc (2:4.6.2~git20080311-4) ...
Нажмите Ввод для продолжения.

```

Установка выбранных пакетов завершена. Нажимаем Enter для возврата в окно плоского списка. Для закрытия всех окон и выхода из aptitude используем клавишу “q” или меню программы.

По вышеописанному алгоритму работы с aptitude устанавливаются и другие пакеты, необходимые для VLC. В дальнейшем, процедура установки пакетов подробно описываться не будет.

Чтобы установить VLC на Debian GNU/Linux, необходимо скачать с официального сайта программы исходные коды и скомпилировать их под свои задачи и оборудование.

Ссылку на скачивание можно получить на странице <http://www.videolan.org/vlc/download-sources.html>. Скачивание исходных кодов можно произвести с помощью программы wget.

Wget — свободная не-интерактивная консольная программа для загрузки файлов по сети. Поддерживает протоколы HTTP, FTP и HTTPS, а также поддерживает работу через HTTP прокси-сервер. Программа включена почти во все Linux-дистрибутивы.

```
debian:~# wget -O vlc-1.1.4.tar.bz2 http://sourceforge.net/projects/vlc/files/1.1.4/vlc-1.1.4.tar.bz2/download_
```

После скачивания архива с исходными кодами VLC откроем Midnight Commander, набрав в консоли mc. Заходим внутрь архива клавишами курсора и Enter и распаковываем директорию с исходными кодами в домашний каталог root клавишей F5.



Клавишей F10 закрываем Midnight Commander.

Переходим в директорию с исходными кодами.

```
debian:~# cd vlc-1.1.4_
```

Для сборки минимальной конфигурации VLC необходимо установить следующие пакеты так, как это было описано ранее: make, automake, binutils, cpp, gcc, g++, libdvbpsi5-dev. Для удовлетворения зависимостей aptitude автоматически установит дополнительные пакеты.

Установка VLC выполняется в 3 этапа:

configure – конфигурирование компонентов VLC под конкретную задачу, ОС, аппаратную часть.

make – компилирование всех необходимых частей программы с учетом конфигурации.

make install – установка компонентов программы в ОС.

Минимальная конфигурация для работы VLC в качестве прокси сервера выглядит следующим образом:

```
./configure '--build=i686-linux-gnu' '--prefix=/usr' '--enable-fast-install' '--enable-sout' '--disable-nls' '--disable-dbus' '--disable-mtp' '--disable-lua' '--disable-avformat' '--disable-avcodec' '--disable-postproc' '--disable-hd1000v' '--disable-hd1000v' '--disable-swscale' '--disable-gme' '--disable-xcb' '--disable-remoteosd' '--disable-glx' '--disable-libproxy' '--disable-update-check' '--disable-fb' '--disable-ggi' '--disable-sdl' '--disable-mad' '--disable-jack' '--disable-pulse' '--disable-lirc' '--disable-a52' '--disable-aa' '--enable-dvbpsi' '--disable-mozilla' '--disable-dvb' '--disable-ogg' '--disable-vorbis' '--disable-shout' '--disable-qt4' '--disable-flac' '--disable-skins2' '--disable-freetype' '--disable-mkv' '--disable-speech' '--disable-caca' '--disable-live555' '--disable-libmpeg2' '--disable-fribidi' '--disable-mod' '--disable-theora' '--disable-dvdnav' '--disable-gnutls' '--disable-ncurses' '--disable-smb' '--disable-gnomevfs' '--disable-bonjour' '--disable-mpc' '--disable-vcd' '--disable-vcdx' '--disable-notify' '--disable-twolame' '--disable-faad' '--disable-zvbi' '--disable-telx' '--disable-atmo' '--disable-taglib' '--disable-libass' '--disable-dca' '--disable-realrtsp' '--disable-dv' '--disable-x264' '--disable-alsa' '--disable-v4l' '--disable-v4l2' '--
```

```
disable-pvr' '--disable-svgalib' '--disable-dirac' '--disable-mux_ogg' '--disable-
schroedinger' '--disable-fluidsynth' '--disable-libxml2' '--disable-svg' '--disable-
portaudio' '--disable-upnp' '--disable-goom' '--disable-libgcrypt' '--disable-udev' '--
disable-dc1394' '--disable-projectm' '--without-kde-solid' '--disable-
osso_screensaver' '--disable-sqlite' 'build_alias=i686-linux-gnu' 'CFLAGS=-g -O2'
'LDFLAGS=-Wl,--as-needed' 'CPPFLAGS=' 'CXXFLAGS=-g -O2' >> /dev/null
```

где

./configure – скрипт настройки VLC в каталоге с исходными кодами программы.

'--build=i686-linux-gnu' '--enable-release' '--prefix=/usr' '--enable-fast-install' 'build_alias=i686-linux-gnu' 'CFLAGS=-g -O2' 'LDFLAGS=-Wl,--as-needed' 'CPPFLAGS=' 'CXXFLAGS=-g -O2' – инструкции настройки компилятора на максимальную производительность компилируемых программ.

--disable – отключить компонент. Компонент не будет скомпилирован, библиотеки для его компиляции и работы ставить не нужно, но и функциональность данного компонента в программе будет отключена.

--enable – включить компонент. Компонент будет скомпилирован, необходимо установить библиотеки для его компиляции (пакеты, оканчивающиеся на -dev), функциональность данного компонента будет присутствовать в программе.

>> /dev/null – позволяет отбросить статистические сведения о конфигурации компонентов и видеть только ошибки и предупреждения. Без данной команды предупреждения могут остаться незамеченными.

Для упрощения ввода данной команды можно опубликовать скрипт на доступном сервере, затем скачать его в директорию VLC командой wget и выполнить командой sh scriptname. Обратите внимание, что файл должен находиться в директории /root/vlc-1.1.4/ !

В процессе проверки конфигурации может появиться сообщение:

configure: WARNING: libcrypt support required for RAOP plugin

Не стоит обращать на него внимание, так как это всего лишь напоминание. Модуль RAOP нам не требуется.

Выполнение команды **configure** может занять несколько минут. По окончании запустим компиляцию программы с помощью команды **make**. Выполнение данной команды также может занять много времени. Не останавливайте процесс компиляции.

По окончании выполните команду **make install** для установки программы в операционную систему. Установка может занять много времени.

Обращаю ваше внимание, что:

- Команду **configure** можно выполнять сколь угодно раз, до достижения необходимого результата.
- После выполнения команды **make** вы можете выполнить **make install** для установки или **make clean** для удаления результатов компиляции. Не выполняйте **configure** до **make clean**!
- После выполнения **make install** вы можете выполнить **make uninstall**, либо удалить папку, из которой производилась установка, так как она больше не требуется для работы программы. Не выполняйте **configure**, **make** или **make install** до **make uninstall**.
- VLC не работает под пользователем **root**. Это специально сделано разработчиками, чтобы обезопасить систему.

Пришло время запустить VLC. Для этого необходимо перейти в режим обычного пользователя. Это можно сделать двумя способами:

1. Выполнить **exit** и заново авторизоваться под другим пользователем.
2. Выполнить **su user** и сразу перейти под учетную запись **user** без ввода пароля. Для возврата под учетную запись **root** необходимо набрать **exit**.

Запуск VLC в режиме прокси сервера производится следующей командой:

```
cvlc --loop http://89.186.245.201:2000 --sout '#duplicate{dst=std
{access=http,mux=ts,dst=0.0.0.0:8080},dst=std{access=msh,mux=asfh,dst=0.0.0
.0:8081}}'
```

где

--loop //Зацикливание воспроизведения. В случае обрыва связи VLC будет снова и снова пытаться восстановить подключение.

http://89.186.245.201:2000 //С какого адреса и порта и по какому протоколу брать видео

8080 //Порт для трансляции HTTP

8081 //Порт для трансляции MMS

Закрывать VLC можно сочетанием клавиш Ctrl+C.

Если в конце команды запуска поставить амперсant, то прокси будет запущен в фоне, и можно будет вводить другие команды консоли. Пример:

```
cvlc --loop http://89.186.245.201:2000 --sout '#std{access=http,
mux=ts,dst=0.0.0.0:8080}'&
```

Другие консольные команды, которые могут понадобиться для управления VLC.

ps ax | grep vlc //Узнаем запущен ли VLC и какой номер его процесса

kill -9 XXXX //Закрываем процесс VLC. XXXX – номер процесса из прошлой команды.

fg cvlc //Вывести VLC на передний план. Ввод команд в консоль становится недоступен.

Закрывать VLC можно сочетанием клавиш Ctrl+C.

Для использования VLC в качестве сервера кодирования необходимо выполнить команду:

```
cvlc --loop http://89.186.245.201:2000 --sout '#transcode{vcode=DIV3,
vb=512,fps=10,scale=0.5,acodec=mp3,ab=64,channels=1,samplerate=44100}:http
{mux=asf,dst=0.0.0.0:8080}'
```

где

`transcode` – указывает на то, что видеопоток будет перекодироваться;

`vcode=DIV3` – видеокodeк (h264, VP80, dirac, theo, mp2v, WMV2, none и др.);

`vb=512` – битрейт видеопотока в кбит/с;

`fps=10` – кадров в секунду;

`scale=0.5` – размер изображения от оригинала (0.5 – в два раза меньше оригинала, 0.25 – в четыре раза меньше оригинала, 2 – в два раза больше оригинала);

`acodec=mp3` – аудиокодек (mp4a, vorb, flac, mpga, wma2, s16l и др.);

`ab=64` – битрейт аудио потока в кбит/с;

`channels=1` – каналов аудио в выходном потоке;

`samplerate=44100` – частота дискретизации аудио потока;

`http` – протокол трансляции;

`mux=asf` – инкапсуляция (ts, webm, ffmpeg{mux=flv}, ogg, wav и др.);

`dst=0.0.0.0:8080` – адрес для подключения. В протоколах HTTP и MMS адрес 0.0.0.0 означает все сетевые интерфейсы сервера, аналогично он может быть не задан. В протоколе RTP в качестве адреса должен быть либо конкретный адрес клиента, либо широковещательный адрес.

Транслировать закодированное видео можно сразу по нескольким протоколам. К примеру:

```
cvlc --loop http://89.186.245.201:2000 --sout '#transcode{vcode=DIV3,
vb=512,fps=10,scale=0.5,acodec=mp3,ab=64,channels=1,samplerate=44100}:
```



```
:duplicate{dst=http{mux=asf,dst=:8080/},dst=std{access=msh,mux=asfh,dst=0.0.0.0:8080}}
```

Если вы попытаетесь запустить кодирование на нашем прокси сервере, то будут выведены сообщения:

ts demux error: MPEG-4 descriptor not found

stream_out_transcode stream out error: cannot find audio decoder

stream_out_transcode stream out error: cannot create audio chain

Сообщения дают нам понять, что VLC не может найти кодеки для декодирования и кодирования потока. Это означает, что нам нужно перекомпилироваться VLC с дополнительными кодеками. Для этого необходимо выполнить следующие действия:

1. Закрывать VLC.
2. Выйти из обычного пользователя командой `exit` и войти пользователем `root`.
3. Войти в каталог с исходниками VLC командой `cd /root/vlc-1.1.4`
4. Выполнить `make uninstall`
5. Выполнить `make clean`
6. Затем нужно внести изменения в команду `configure` таким образом:

Например, чтобы добавить кодеки необходимо активировать одну из опций в зависимости от необходимого кодека: `'--enable-avformat', '--enable-avcodec', '--enable-ogg', '--enable-vorbis', '--enable-flac', '--enable-mkv', '--enable-libmpeg2', '--enable-theora', '--enable-twolame', '--enable-x264', '--enable-dirac', '--enable-mux_ogg'`.

Чтобы добавить эффекты улучшения изображения, логотип канала, бегущую строку или изменить размер изображения необходимо активировать: `'--enable-postproc', '--enable-swscale'`.

Чтобы использовать в качестве устройства ввода порт FireWire (IEEE1394): '--enable-dv', '--enable-dc1394'.

Для добавления дополнительных протоколов трансляции используются модули: '--enable-live555', '--enable-realrtsp'.

Обратите внимание: после включения каких-либо модулей необходимо установить соответствующие библиотеки в системе. Иначе они будут требоваться при выполнении команды `configure`.

К примеру для активации большинства кодеков необходимо включить опции '--enable-avformat', '--enable-avcodec'. Но после их включения команда `configure` выдает следующее сообщение и прерывается:

```
configure: error: Could not find libavcodec or libavutil. Use --disable-avcodec to ignore this error.
```

Это означает, что мы должны установить при помощи `aptitude` пакет `libavcodec-dev` или `libavutil-dev`. В дальнейшем конфигуратор потребует `libavformat-dev`, поэтому данный пакет лучше сразу установить. А также для успешного прохождения конфигурации необходима утилита `pkg-config`, которую также необходимо установить.

Обращаю Ваше внимание, что в момент написания статьи в стабильную версию Debian GNU/Linux входит устаревшая версия `libavcodec`, `libavutil` и `libavformat`. Получить обновленную версию данных библиотек можно следующим образом:

1. В файл конфигурации менеджера пакетов `/etc/apt/sources.list` добавляем репозитории из новой версии Ubuntu Linux, а уже записанные репозитории Debian GNU/Linux отключаем, поставив в начале каждой строки “#”.

```
File: sources.list   Line 1 Col 0   856 bytes   100%
#
# deb cdrom:[Debian GNU/Linux 5.0.3 _Lenny_ - Official i386 DVD Binary-1 2009090
5-08:48]/ lenny contrib main
# deb cdrom:[Debian GNU/Linux 5.0.3 _Lenny_ - Official i386 DVD Binary-1 20090905
-08:48]/ lenny contrib main
# deb http://mirror.yandex.ru/debian/ lenny main non-free contrib
# deb-src http://mirror.yandex.ru/debian/ lenny main non-free contrib
# deb http://security.debian.org/ lenny/updates main contrib non-free
# deb-src http://security.debian.org/ lenny/updates main contrib non-free
# deb http://volatile.debian.org/debian-volatile lenny/volatile main contrib non-
free
# deb-src http://volatile.debian.org/debian-volatile lenny/volatile main contrib
non-free
deb http://ru.archive.ubuntu.com/ubuntu/ maverick main restricted universe multi
verse
deb http://ru.archive.ubuntu.com/ubuntu/ maverick-updates main restricted univer
se multiverse
1Help 2UnWrap 3Quit 4Hex 5Line 6RxSrch 7Search 8Raw 9Unform 10Quit
```

2. Затем открываем aptitude и нажимаем клавишу “u” либо в меню Обновить список пакетов.

```
Actions Undo Package Resolver Search Options Views Help
u: Update g: Download/Install/Remove Pkgs
Install/remove packages g
Update package list u (2)
Mark Upgradable U
Forget new packages f (18557)
Cancel pending actions eated Packages (24)
Clean package cache
Clean obsolete files
Play Minesweeper
Become root
Quit Q ckages is available.
This group contains 242 packages.
Check for new versions of packages
```

3. Теперь обновляем libavcodec-dev, libavutil-dev и libavformat-dev, как это описывалось ранее.

4. По окончании можно вернуть прежнее состояние /etc/apt/sources.list и снова обновить список пакетов aptitude.

При помощи сочетания клавиш Alt+F1..6 можно перелистывать консоли. Очень удобно подбирать параметры конфигурации в одной консоли, добавлять компоненты в aptitude в другой, а проверять конфигурацию в третьей.

Для запуска трансляции из файла (скачать его на компьютер можно при помощи wget) воспользуемся командой:

```
cvlc --loop /root/file.avi --sout '#std{access=http,mux=ts,dst=0.0.0.0:8080}'
```

где /root/file.avi – путь к видеофайлу в файловой системе сервера.

VLC может на лету перекодировать файл с видео. Пример команды:

```
cvlc --loop /root/file.avi --sout '#transcode{vcode=DIV3, vb=512,fps=10,scale=0.5,acodec=mp3,ab=64,channels=1,samplerate=44100}:http {mux=asf,dst=0.0.0.0:8080}'
```

Для трансляции видео потока напрямую с камеры через FireWire интерфейс можно использовать следующую команду:

```
cvlc --loop dv/rawdv:///dev/raw1394 --sout '#transcode{vcode=DIV3, vb=512,fps=10,scale=0.5,acodec=mp3,ab=64,channels=1,samplerate=44100}:http {mux=asf,dst=0.0.0.0:8080}'
```

где

dv/rawdv:// - протокол получения данных.

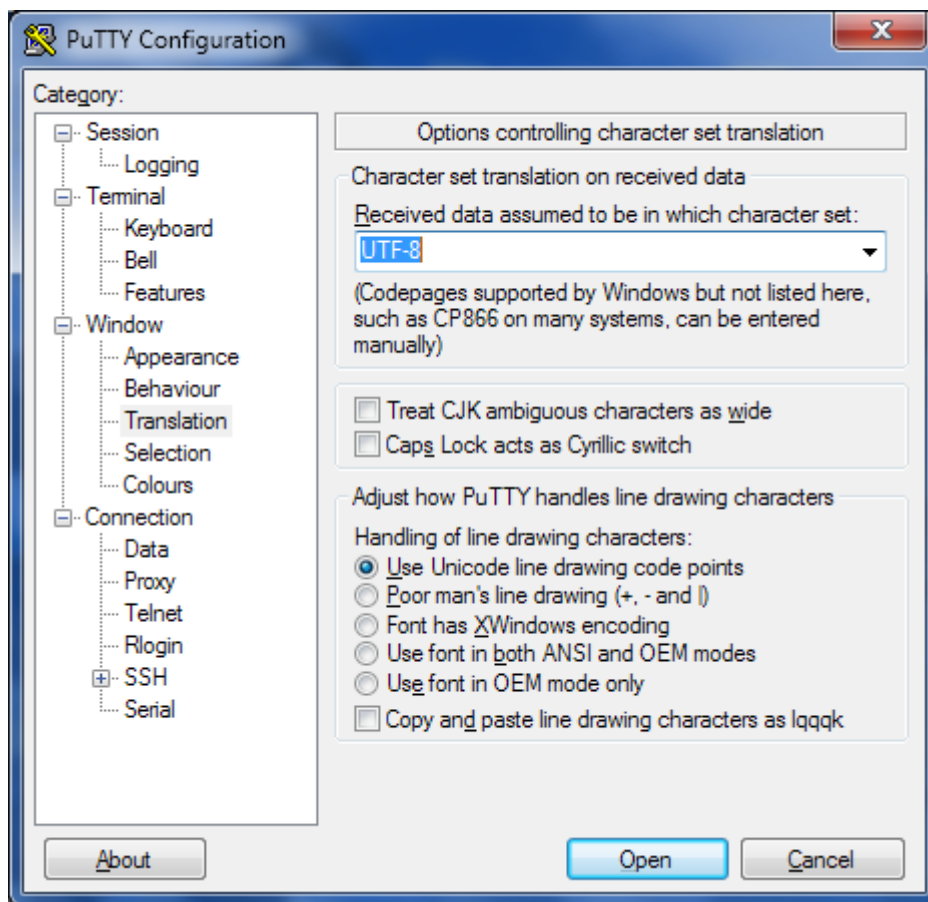
/dev/raw1394 – устройство, с которого получают данные. Устройство создается системой автоматически при подключении камеры к FireWire порту.

Для использования описанных протоколов и кодеков не забудьте активировать соответствующие функции VLC перед компиляцией программы, а также установить необходимые библиотеки ОС.

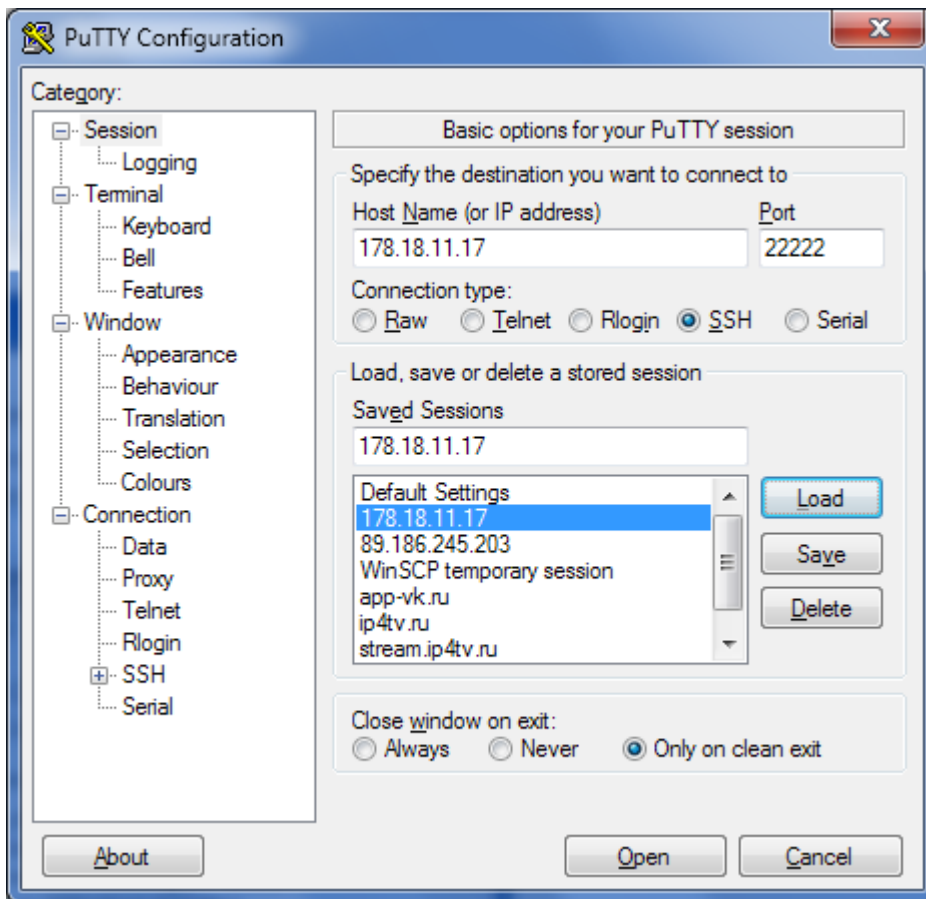
Прокси сервер

Запускаем putty.exe

Исправляем кодировку Window->Translation->Ставим UTF-8



Переходим на вкладку Session и вводим IP сервера и порт 22222



Жмем Open.

Login as: user //Если попытаетесь набрать в русской раскладке, то не залогинитесь.

root@178.18.11.17's password: //вводите пароль, он не отображается

```
cvlc --loop http://109.169.194.139:2000 --sout '#duplicate{dst=std{access=http,mux=ts,dst=0.0.0.0:8080},dst=std{access=mms,h,mux=asfh,dst=0.0.0.0:8081}}'
```

Строка запуска VLC прокси, где

http://109.169.194.139:2000 //С какого адреса и порта и по какому протоколу брать видео

8080 //Порт для трансляции HTTP

8081 //Порт для трансляции 8081

Необходимо открыть подключения к портам 8080 и 8081 из внешней сети!!!

Также можно устанавливать порты 1234 и 1235.

Закрывать VLC можно сочетанием клавиш Ctrl+C.

Если в конце команды запуска поставить амперсant, то прокси будет запущен в фоне и можно будет вводить другие команды консоли. Пример:

```
cvlc --loop http://109.169.194.139:2000 --sout '#duplicate{dst=std{access=http,mux=ts,dst=0.0.0.0:8080},dst=std{access=mmsh,mux=asfh,dst=0.0.0.0:8081}}'&
```

Этот способ предпочтительней, так как при непредвиденном завершении сеанса консоли (обрыв связи), трансляция не оборвется.

```
ps ax | grep vlc //Узнаем запущен ли VLC и какой номер его процесса
```

```
kill -9 XXXX //Закрываем процесс VLC. XXXX – номер процесса из прошлой команды.
```

```
fg cvlc //Вывести VLC на передний план. Ввод команд в консоль становится недоступен. Закрывать VLC можно сочетанием клавиш Ctrl+C.
```

Для закрытия сессии используется команда exit.

Проверьте настройки своего фаервола. Мне кажется он обрывает постоянные подключения через несколько минут. Соответственно через несколько минут будет обрываться и видео.

Литература

[1] Cisco Systems, “Cisco Visual Networking Index: Forecast and Methodology, 2008–2013”, 9 июня 2009 (http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360.pdf).

[2] X. Hei, C. Liang, and et. al., A Measurement Study of a Large-Scale P2P IPTV System, IEEE Trans. on Multimedia, vol. 9, 8, 2007.

[3] Y. Won and M. Choi, End-User IPTV Traffic Measurement of Residential Broadband Access Networks, in IEEE NOMS, 2008.

- [4] S. Bradner, J. McQuaid, RFC2544 - Benchmarking Methodology for Network Interconnect Devices, March 1999.
- [5] International Telecommunication Union, "Methodology for the subjective assessment of the quality of television pictures (ITU-R BT.500-11)" (2002).
- [6] Д. Ватолин, А. Паршин, О. Петров, А. Титаренко, "Субъективное Сравнение Современных Видеокодеков" (http://www.compression.ru/video/codec_comparison/pdf/msu_subjective_codecs_comparison_ru.pdf).
- [7] Haghani, E. De, S. Ansari, N., VoIP Traffic Scheduling in WiMAX Networks, in: IEEE Global Telecommunications Conference, 2008.
- [8] M. Claypool and J. Tanner, "The Effects of Jitter on the Perceptual Quality of Video," in ACM Multimedia, 1999.
- [9] P. Calyam, M. Sridharan and et. al., "Performance Measurement and Analysis of H.323 Traffic," in PAM Workshop, 2004.
- [10] T. Silverston, O. Fourmaux, A. Botta, A. Dainotti, A. Pescapé, G. Ventre, K. Salamatian, Traffic analysis of peer-to-peer IPTV communities, Computer Networks, Vol. 53, pp. 470-484, 2009.
- [11] S. Tao, J. Apostolopoulos, R. Guerin, Real-Time Monitoring of Video Quality in IP Networks, IEEE Trans. on Networking, Vol. 16, No. 5, pp. 1052-1065, 2008.
- [12] A. Sukhov, P. Calyam, W. Daly, A. Ilin, Towards an analytical model for characterizing behavior of high-speed VVoIP applications, Computational Methods in Science and Technology, 11(2), 161-167 (2005).
- [13] VideoLAN team, "VideoLAN, Free streaming and multimedia solutions for all OS!" (<http://www.videolan.org/>).
- [14] Wireshark Foundation, "Wireshark. Go deep.". (<http://www.wireshark.org/>)
- [15] Avery Lee, "Welcome to virtualdub.org! - virtualdub.org" (<http://www.virtualdub.org/>).
- [16] "Main Page - Avisynth". (http://avisynth.org/mediawiki/Main_Page)

[17] “Traces of video stream in wireless networks (WiMAX, 3G, WiFi)”
(<http://www.ip4tv.ru/stati/aaa.html>).