

На правах рукописи

Лавров Валерий Александрович

**ИССЛЕДОВАНИЕ И РЕАЛИЗАЦИЯ ЭФФЕКТИВНЫХ
МЕТОДОВ ЗАПИСИ АКТИВНОСТИ
ПОЛЬЗОВАТЕЛЯ ПЕРСОНАЛЬНОГО КОМПЬЮТЕРА**

АВТОРЕФЕРАТ

диссертации на соискание учёной степени кандидата
технических наук по специальности 05.13.11 –
«Математическое и программное обеспечение
вычислительных машин, комплексов и
компьютерных систем»

Томск – 2005

Работа выполнена в Томском государственном университете на кафедре теоретических основ информатики факультета информатики

Научный руководитель: кандидат
физико-математических наук,
доцент Гладких Б.А.

Официальные оппоненты: доктор технических наук,
профессор Горчаков Л.В.,
кандидат технических наук
Кузнецов Д.Ю.

Ведущая организация – Кемеровский государственный университет.
Защита состоится 21 апреля 2005 г. в 10 часов 30 минут на заседании диссертационного совета Д 212.267.08 в Томском государственном университете.
Адрес: г. Томск, пр. Ленина 36, корп. 2, ауд. 102.

Отзывы на автореферат (в двух экземплярах, заверенные печатью) просьба направлять по адресу: 634050, г. Томск, пр. Ленина 36, Томский госуниверситет, ученому секретарю Н.Ю. Буровой.

С диссертацией можно ознакомиться в научной библиотеке Томского государственного университета.

Автореферат разослан «_15_» марта 2005 г.

Ученый секретарь
диссертационного совета,
доктор технических наук, доцент

Скворцов А.В.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Настоящая работа относится к области программного обеспечения новых информационных технологий и направлена на повышение надежности функционирования программ. Усложнение программных продуктов предъявляет более жесткие требования к их тестированию. И данная работа, в том числе, направлена на то, чтобы за счет уменьшения времени тестирования, при применении результатов данной работы, сократить сроки создания программного обеспечения. Кроме этого, использование результатов работы в области обучения помогает облегчить общение между человеком и компьютером.

Актуальность работы

В связи с большим распространением персональных компьютеров за последние годы резко возросло количество разнообразных приложений. Работа с компьютером усложняется с каждым годом, поэтому требуются специальные средства для быстрого обучения работе с приложениями, для тестирования программных продуктов и для создания рекламных роликов программ.

Существует специальный класс программ, которые могут осуществлять запись активности пользователя (изменения графического содержимого экрана, звукового сопровождения, разнообразных событий клавиатуры, мыши, других устройств ввода). Одни программы сохраняют только отдельные статические снимки экрана, другие имеют возможность создавать динамические видеоролики, которые при воспроизведении показывают все записанные действия пользователя. Весьма важным с точки зрения пользователя являются дополнительные функциональные возможности, такие как вставка маркеров во время записи, редактирования записанного видеоролика, удобные средства навигации при проигрывании и т.п.

Существующие модели основываются на том, что часть информации в ходе записи может быть утеряна, за счет чего повышается производительность и уменьшается объем результирующей записи. Но такой подход абсолютно неприменим в области тестирования сложных программных продуктов. Программист должен воспроизвести в точности все действия тестера, который обнаружил проблему. К тому же, требования эффективности, предъявляемые к таким системам, противоречивы. С одной стороны, необходимо добиться наивысшей точности воспроизведения исходной активности с точки зрения адекватности изображения и времени наступления событий, с другой – получить минимальный размер записанного видеоролика. Еще одно требование связано с минимизацией дополнительной нагрузки на процессор компьютера, на котором происходит запись активности.

Несмотря на то, что на рынке представлено несколько программ для записи активности пользователя, ни одна из них не обеспечивает качественную запись с минимальной нагрузкой на центральный процессор и минимальным размером результирующего видеоролика. Например, в области тестирования программных продуктов является важным тот факт, что записывающая программа оставляет как можно больше свободного процессорного времени для тестируемого продукта. Для видеороликов, распространяемых в интернете, желательно иметь небольшой размер, чтобы сократить время загрузки, а для демонстрационных видеороликов на компакт-дисках желательно передавать видеоролики высокого качества, несмотря на их большой размер.

Поэтому весьма актуальной является разработка новых эффективных методов записи активности пользователя персонального компьютера и реализация их в пакете прикладных программ, исполняемым под управлением современных операционных систем, а также в онлайн-режиме в интернете.

Цель работы

Целью данной работы является разработка новых эффективных методов записи активности пользователя компьютера и компактных форматов хранения генерируемых при этом данных, учитывающих специфику регистрируемых объектов.

Методика исследований

Исследование проводилось с использованием методов функционального моделирования, объектно-ориентированного и системного программирования. Правильность результатов исследования подтверждена результатами сравнительного практического тестирования систем, разработанных на основе предложенных в работе методов.

Научная новизна работы,

по мнению автора, заключается в следующем:

1. Разработан новый метод регистрации активности пользователя, основанный на синхронном детектировании событий графического ядра операционной системы.
2. Разработан новый векторно-растровый формат хранения данных об активности пользователя, отличающийся тем, что элементы данных привязаны к меткам времени, что обеспечивает высокую точность регистрации. Формат обеспечивает более высокую степень компрессии по сравнению с другими известными форматами.
3. Предложено архитектурное решение, реализующее метод регистрации активности пользователя для операционных систем семейства Windows, минимизирующее нагрузку на центральный процессор.

Практическая ценность работы

1. Предложенные в работе методы регистрации активности пользователя ЭВМ и формат хранения данных могут быть использованы для создания эффективных приложений, которым требуется функция записи активности пользователя.

2. На основе предложенных автором методов, форматов и архитектурных решений создан ряд программных продуктов для регистрации активности пользователя (BB FlashBack и др.).

Внедрение полученных результатов

Разработанный автором программный комплекс для компании Blueberry Consultants Ltd. является коммерческим продуктом и распространяется по интернету. К настоящему времени продано более 1000 копий этого комплекса.

Апробация работы

Отдельные результаты работы докладывались и обсуждались на следующих конференциях:

1. Восьмой Российско-Корейский международный симпозиум по науке и технологиям, Томск, июнь 2004 г.
2. VIII всероссийская научно-практическая конференция, Анжеро-Судженск, апрель 2004 г.
3. Всероссийская научная конференция «Наука. Технологии. Инновации», Новосибирск, декабрь 2004 г.
4. IV всероссийская научно-практическая конференция «Информационные недра Кузбасса», Кемерово, февраль 2005 г.

Структура диссертации

Диссертация состоит из введения, четырех глав, заключения, библиографического списка и трех приложений.

Публикации по работе

Основное содержание работы отражено в четырех статьях [3 – 5, 7], брошюре [2] и четырех тезисах [1, 6, 8, 9].

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

В диссертационной работе проводится анализ существующих методов и нескольких распространенных систем для записи активности пользователя. Предлагается новый метод, который особенно эффективен в области тестирования программных продуктов.

В первой главе диссертации проводится аналитический обзор областей применения систем записи активности пользователя, а также методов, использованных в этих системах.

В § 1.1 дается определение понятия записи активности пользователя. При работе пользователя за компьютером с графическим интерфейсом он всегда производит манипуляции с графическими объектами на экране компьютера, работает с мышью, клавиатурой и другими устройствами ввода данных. Все эти действия входят в понятие «активность пользователя». Под записью активности пользователя понимается процесс, в ходе которого производится запись всех действий пользователя с целью их последующего воспроизведения на любом компьютере.

Системы для записи активности пользователя могут применяться в таких областях, как создание демонстрационных видеороликов программ и систем, создание обучающих систем, тестирование пакетов программ и онлайн-службы поддержки программных продуктов.

В § 1.2 вводятся свойства идеальной системы записи активности пользователя. Идеальная система записи активности пользователя обладает следующими свойствами:

- в ходе записи не отнимается процессорное время;
- записываются все события без потери;
- результирующая запись имеет минимальный размер;
- редактор записи обладает всеми необходимыми средствами редактирования для создания привлекательных демонстраций;
- проигрыватель абсолютно точно воспроизводит все записанные действия пользователя.

Модель М1 идеальной системы записи активности пользователя изображена на рис. 1. На вход черного ящика поступают сигналы $X_1(t)$, а на выходе имеем сигнал $Y(t)$, который представляет собой готовую запись активности пользователя. Для модели идеального способа записи активности пользователя ни выходной сигнал, ни сам черный ящик никак не влияют на входные сигналы. Под входными сигналами мы можем понимать $X_1(t)$ – графические события операционной системы, $X_2(t)$ – звуковое сопровождение, а $X_3(t)$ – события мыши, $X_4(t)$ – клавиатуры и т.д.



Рис. 1. Идеальная модель систем записи активности пользователя.

Далее проводится анализ различных областей применения, в которых свойства идеальной системы невозможно полностью реализовать. Поэтому в действительности мы всегда имеем другую модель M2 (рис. 2), в которой на входные сигналы оказывается влияние самим черным ящиком.

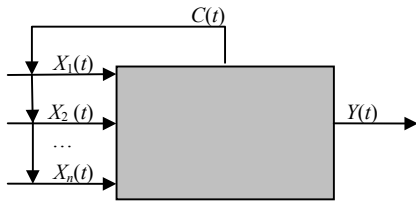


Рис. 2. Реальная модель систем записи активности пользователя.

Модели M1 и M2 характеризуются тем, что существует некоторая функция $F(X_i(t)) = Y(t)$, а также обратная ей функция $F'(Y(t)) = \{X_i(t)\}$. То есть всегда есть взаимнооднозначное соответствие между входными сигналами $X_i(t)$ и выходным сигналом $Y(t)$. Физически это означает, что при воспроизведении записи $Y(t)$ мы видим абсолютно точно то, что производилось пользователем в ходе записи.

В § 1.3 анализируются существующие модели (рис. 3) систем записи активности пользователя, делается вывод о существенных недостатках при применении систем в области тестирования программных продуктов. В этих моделях не представляется возможным из $R(t)$ получить $Y(t)$, так как не существует обратной функции $S'(t)$. А значит и невозможно точно воспроизвести сигналы $X_i(t)$. Но в этой модели существует функция $S''(t)$, которая переводит сигнал $R(t)$ в сигнал $Y'(t)$, похожий на $Y(t)$. А сигнал $Y'(t)$ в свою очередь может быть преобразован в сигналы $\{X'_i(t)\}$, которые похожи на сигналы $\{X_i(t)\}$. Физически это происходит из-за того, что поступающие сигналы $X_i(t)$ искажаются, и некоторые значения теряются. Например, все современные системы теряют часть графической информации в ходе записи, поэтому при воспроизведении мы не увидим точной картины того, что происходило в процессе записи.

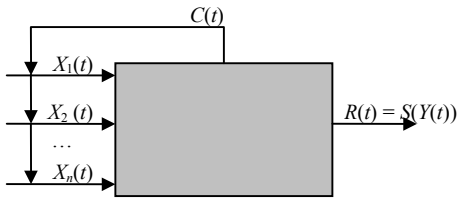


Рис. 3. Обобщенная модель существующих систем записи активности пользователя.

Существующие модели никогда не смогут создать точную запись тех действий, которые производились пользователем. Часть событий будет утеряна.

Внутренняя структура черного ящика в большинстве моделей представляется в виде схемы, изображенной на рис. 4.

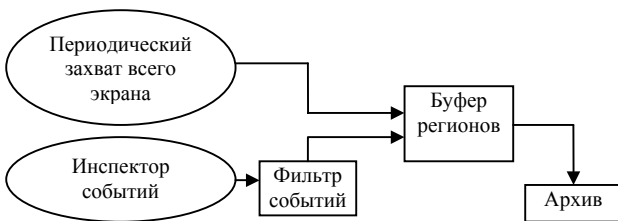


Рис. 4. Внутренняя структура черного ящика современных моделей записи активности пользователя

Используя такую структуру, невозможно создать точную запись тех действий, которые производились пользователем. Часть событий будет утеряна, так как они могут происходить между отдельными захватами экрана, или, быть незамеченными инспектором событий.

В § 1.4 проводится практический анализ современных систем записи активности пользователя Camtasia, WinCam и DemoForge. В ходе тестирования оценивались такие качества систем, как нагрузка на центральный процессор, размер результирующей записи и точность воспроизведения. По результатам тестирования видно, что ни одна из систем не позволяет решать все задачи на самом высоком уровне, так как все они отнимают много процессорного времени или теряют важную информацию в ходе записи. Для тестирования программных продуктов такие критерии являются важными, поэтому использование существующих методов в этой области не будет оправданным.

Отсюда следует, что актуальной является разработка системы записи активности пользователя по своим функциональным возможностям не уступающей протестированным системам, но обеспечивающей высокую точность воспроизведения исходной активности при минимальной нагрузке на центральный процессор и обладающей богатым набором средств редактирования записей.

Во второй главе предлагается новый метод записи активности пользователя, а также формат для хранения результирующей записи.

В § 2.1 предлагается новая модель М4 (рис. 5) записи активности пользователя на основе синхронного детектирования событий графического ядра операционной системы. В рамках этой модели не происходит потери или искажения информации, поэтому воспроизведение результирующей записи будут точно совпадать с тем, что происходило во время записи.



Рис. 5. Внутренняя структура черного ящика современных моделей записи активности пользователя

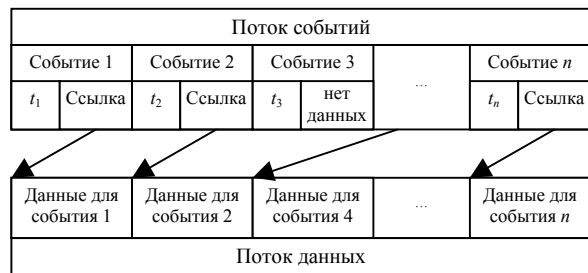


Рис. 6. Структура предлагаемого формата

Для хранения специфических данных в рамках предложенной модели § 2.2 предлагается структура формата (рис. 6).

Так как результирующая запись в нашей модели состоит из последовательности событий, а каждое событие может иметь некоторые данные, то логически формат хранения можно представить в виде двух потоков: поток описаний событий и поток двоичных данных. Каждое событие в потоке событий имеет идентификатор события, отметку времени наступления события и необязательную ссылку на данные, хранящиеся в потоке двоичных данных.

Для уменьшения общего объема данных оба потока могут быть сжаты используя различные алгоритмы потокового сжатия в режиме реального времени. Данные для событий могут быть различными. Большую часть будут занимать растровые изображения и звук. Для растровых изображений можно применять различные алгоритмы сжатия без потерь, а для звука и сжатие с потерями, например, mp3

Формат является экономичным по размеру, масштабируемым, он позволяет легко модифицировать результирующую запись.

В § 2.3 приводится физическая реализация приведенной в § 2.1 структуры комбинированного векторно-растрового формата хранения данных об активности пользователя под названием FBR. В этом формате все события делятся на три группы: описательные, графические и события управления.

Описательные события используются для передачи параметров результирующего видеоролика, таких как размер, название, автор и др. Графические события производят непосредственно изменение графического содержимого экрана компьютера. А события управления используются для позиционирования курсора мыши, состояния указателя, а также для хранения событий клавиатуры.

События также делятся на две категории. К первой категории относятся события, которые для своего исполнения не требуют внешних данных. Например, событие IDRectangle (нарисовать прямоугольник) полностью содержит всю необходимую информацию для воспроизведения этого события. Ко второй категории относятся события, требующие разнообразных внешних данных. Событие IDCursorShape (смена графической формы курсора мыши) ссылается на блок данных в двоичном потоке, содержащем битовую карту новой формы.

В третьей главе диссертации предлагается новая архитектура эффективной системы записи активности пользователя на основе модели, предложенной в § 2.1. Учитываются также требования эффективности и функциональной широты.

В § 3.1 описываются технические требования к эффективной системе и способы их реализации. Требования включают в себя следующие элементы:

1. *Поддержка всех современных ОС семейства Windows.* Из-за того, что многие пользователи как можно дольше стараются сохранить свои вложения в программные продукты, которые они используют, в мире до сих пор используется несколько операционных систем семейства Windows.

2. *Высокая точность воспроизведения при минимальном размере видеоролика.* Точность воспроизведения в свою очередь может быть декомпозирована на две составляющих: точность воспроизведения статических изображений и точность воспроизведения последовательности статических изображений, а также событий мыши и клавиатуры.

Неточность воспроизведения статических изображений может происходить как за счет закругления разрешающей способности экрана, так и за счет уменьшения глубины цвета.

Ошибки воспроизведения последовательности статических изображений происходит из-за того, что при воспроизведении различные события происходят не в те моменты времени, в какие они происходили в ходе записи. Например, при плавном перемещении курсора мыши в ходе записи мы можем наблюдать отрывистое передвижение при проигрывании уже записанного видеоролика.

3. *Минимальное воздействие на центральный процессор.* Для удовлетворения этого требования предлагается ввести механизм точного детектирования изменения состояния экрана в тот момент, когда это изменение производится операционной системой (модель M4). За счет этого отпадает необходимость периодического анализа всего экрана, что существенно снижает нагрузку на центральный процессор.

4. *Возможность записи звукового сопровождения, событий клавиатуры и мыши.* Желательно реализовать функцию записи звука как можно более универсально, чтобы один и тот же код работал под управлением различных операционных систем.

5. *Наличие разнообразных средств редактирования.* Функция редактирования является очень важной, так как позволяет существенно улучшить содержимое записанного видеоролика. Предлагаемый нами в параграфе 2.3 формат FBR дает принципиальную возможность легко изменять записанный видеоролик. Редактор был реализован и входит в состав пакета BB FlashBack.

6. *Экспорт в популярные обменные форматы (Flash, MPEG, WMV).* Формат FBR хотя и является эффективным и удобным для хранения записанных видеороликов, но большинство пользователей использует для просмотра видеофайлов форматы Flash, MPEG или WMV. Поэтому в систему необходимо добавить функции конвертации в эти форматы. Реализация этих функций описана в разделах 3.5.5 и 3.5.6.

7. *Удобство системы навигации при проигрывании.* Реализация этого требования предполагает наличие у проигрывателя функций навигации по отдельным кадрам, перемотки вперед и назад, масштабирования.

8. *Компонентность.* Особую значимость имеет реализация системы для он-лайнного использования в интернете, например, с целью поддержки программных продуктов. Реализация этого требования означает создание такого кода, который может быть использован как в виде отдельного приложения, так и в виде ActiveX-объекта.

В § 3.2 рассматриваются способы реализации требований к системе записи активности пользователя с учетом специфики структуры операционной системы Windows 98. Перехват графических событий операционной системы возможно осуществить по скрытой от пользователя фирменной технологии GDI Hooking, для чего должна быть разработана специальная программа.

В ОС Windows 2000/XP фирмой Microsoft заложена технология Mirror Driver, которая предназначена для использования системой удаленного управления рабочим столом. Для того, чтобы использовать этот механизм необходимо написать собственный дополнительный к основному видеодрайвер, которому ОС посылает копии всех графических событий. Реализация данной технологии подробно описана в § 3.3.

В операционной системе Windows NT ни тот, ни другой способы неприменимы, потому что технология Windows NT не позволяет легко встраиваться в цепочку графических вызовов. Для этой ОС возможен единственный способ, заключающийся в периодическом захвате всего содержимого видеопамати с последующим анализом. Этот способ универсален, пригоден для любой ОС, однако он не учитывает особенности команд графической подсистемы и поэтому обеспечивает меньшую компрессию, к тому же существенно нагружает центральный процессор для обработки данных. При этом существует вероятность потери событий между двумя соседними захватами. Реализация универсального метода захвата описана в § 3.4.

В § 3.5 предлагаются различные механизмы которые, с одной стороны, могут облегчить работу с системой, а с другой стороны расширить функциональные возможности, такие как:

– продолжительная запись: полезное свойство для целей тестирования программ. Запись происходит постоянно, без остановок. В случае возникновения проблем с тестируемым продуктом тестер может отослать разработчику последние *n* секунд записанной информации для исправления ошибки;

– запись звукового сопровождения: пользователь может голосом комментировать то, что происходит на экране;

– сохранение нажатий кнопок клавиатуры и мыши: удобное свойство для целей тестирования программных продуктов;

– сохранение выводимой на экран текстовой информации: для удобства навигации по записанному ролику. Записывающая программа всю текстовую информацию, выводимую на экран. Это свойство основано на возможности драйвера дисплея перехватывать функции вывода текста на экран.

В четвертой главе диссертации описываются особенности программной реализации системы записи активности пользователя, а также приводится описание метода использования системы в области поддержки программных продуктов. В конце главы проводится сравнительное тестирование разработанной нами системы записи активности пользователя и существующих.

Современные коммерческие приложения разрабатываются в основном с использованием средств разработки, основанных на языке C++. Существуют две самые распространенные среды для платформ Windows – Microsoft Visual Studio C++ и Borland C++ Builder. Каждая из сред разработки имеет достоинства и недостатки, которые описаны в § 4.1.

Приложение с графическим пользовательским интерфейсом было создано в среде Borland C++ Builder, а объекты ActiveX и свободно распространяемый проигрыватель написаны в среде Visual Studio.

Важную роль в разработанной системе играют процедуры сжатия без потерь. Существует ряд открытых библиотек сжатия, примером которых является zlib. Описание использования библиотеки zlib в реализуемой системе приводится в § 4.2.

Исходя из требования компонентности, пакет реализован в виде библиотеки классов. Данный подход обеспечивает гибкость при работе готовой системы в различных средах, а также легкое использование созданного кода для написания других приложений. Описание классов и структур приводится в § 4.3.

При разработке графического интерфейса делался упор на простоту и удобство. С одной стороны, интерфейс не должен быть перегружен ненужными элементами, а с другой стороны, необходимо достаточное количество элементов, чтобы пользоваться программой было удобно. Принципы организации пользовательского интерфейса системы описаны в § 4.4.

Вопросам взаимодействия между отдельными компонентами системы посвящен § 4.5. В нем описывается последовательность действий пользователя и соответствующих вызовов внутри системы при записи активности пользователя.

Непосредственная реализация драйвера по технологии Mirror Driver рассматривается в § 4.6. Приводятся примеры базового кода драйвера, описывается взаимодействие между отдельными частями драйвера и передача данных между драйвером и приложением пользователя Win32. Здесь же рассматривается вопрос отладки кода драйвера во время его разработки, что является непростой задачей.

Не во всех операционных системах возможно использование специального драйвера для захвата экрана. Описание работы системы в режиме совместимости с основными версиями ОС Windows приводится в § 4.7.

Особенности установки программы на компьютеры с различными операционными системами описываются в § 4.8. Под Windows 98 не требуется никаких ухищрений для установки и осуществляется простое копирование файлов.

Под Windows NT4 необходимо наличие прав администратора, чтобы создать группы для файлов и скопировать часть файлов в системную директорию.

Самая сложная установка производится под управлением Windows 2k/XP. Для установки программы необходимо иметь наличие администраторских прав и наличие в групповой политике разрешения на установку неподписанных драйверов для того, чтобы установить VB Capture видеодрайвер в систему.

Для воспроизведения FBR-файлов необходим специальный проигрыватель. Для разработки небольшого проигрывателя была использована среда MS VS C++. Вопросам создания проигрывателя и его функциональных особенностей посвящен § 4.9. Размер проигрывателя составляет около 350 Кб.

В § 4.10 предлагается способ использования системы для он-лайн службы поддержки. После установки программного продукта пользователь может столкнуться с различными проблемами, которые сам разрешить не в состоянии. Если программа является лицензионной, то пользователь может позвонить в центр технической поддержки и попросить помощи. Технический персонал просит записать пользователя видеоролик всех действий, приводящих к проблеме. Пользователь высылает ролик в фирму производителя, используя электронную почту. Технический персонал анализирует проблему и затем помогает пользователю решить её.

Работа службы поддержки может быть облегчена, если будет создан специальный веб-сервис, с помощью которого, шаг за шагом, пользователь может достоверно сообщить о проблеме и получить квалифицированную помощь. Ему даже не надо будет устанавливать программу создания видеороликов, так как она будет встроена в HTML-страницы в виде ActiveX-объекта. Необходимые параметры для правильной работы этого объекта на машине клиента могут изменяться динамически и настраиваются техническим персоналом службы поддержки.

Результаты экспериментального тестирования приводятся в § 4.11. Описывается методология тестирования и использованное для тестирования оборудование, отдельные тесты и приводятся результаты тестирования. Экспериментальное тестирование показало, что по совокупности всех критериев, система FlashBack оказалось лучшей среди всего набора тестируемых систем.

Таким образом, применение технологии Mirror Driver, разработанных методов по обработке графических данных и формата FBR позволило создать систему, которая на данный момент является самой эффективной среди рассмотренных систем в смысле размера результирующего видеоролика, минимизации нагрузки на центральный процессор и верности воспроизведения.

В заключении подводятся итоги проделанной работы.

1. Проблема записи активности пользователя персональных компьютеров возникает во многих областях их применения: при тестировании программ, создании демонстрационных и обучающих видеороликов, для поддержки программных продуктов через интернет и т. п.

2. Автором были проанализированы модели для записи активности пользователя, используемые в существующих системах, что позволило выделить сильные и слабые стороны продуктов. Проведенное экспериментальное исследование этих систем показало, что ни одна из них не обладает абсолютным преимуществом по совокупности рассмотренных критериев. В связи с этим актуальной являлась разработка новой эффективной системы, не уступающей им по своим функциональным возможностям, но обеспечивающей наивысшую точность воспроизведения при меньшем размере видеоролика и минимальной нагрузке на процессор.

3. На основе анализа была предложена новая модель регистрации активности пользователя, основанная на синхронном детектировании событий графического ядра операционной системы.

4. Для хранения специфических результирующих данных об активности пользователя в рамках предложенной модели был разработан формат, отличающийся тем, что элементы данных привязаны к меткам времени, что обеспечивает высокую точность регистрации.

5. На основе предложенной модели, а также сформулированных технических требований к функциональности и эффективности, разработана архитектура системы, учитывающая особенности операционных систем семейства Windows, включающего Windows 98, Windows 2000/XP, Windows NT.

В системе Windows 98 перехват графических событий операционной системы осуществлено по скрытой от пользователя фирменной технологии GDI Hooking, а в ОС Windows 2000/XP – по технологии Mirror Driver, которая предназначена для использования системой удаленного управления рабочим столом.

В операционной системе Windows NT ни тот, ни другой способы неприменимы. Для этой ОС был реализован способ, заключающийся в периодическом захвате всего содержимого видеопамати с последующим анализом.

Разработана компонентная модель системы, позволяющая записывающему приложению функционировать как в виде отдельного приложения, так и в виде ActiveX-объекта, встраиваемого в другие приложения, в частности, в HTML-код веб-страниц.

6. Разработанная архитектура системы записи активности пользователя была реализована в виде пакета программ BB FlashBack, включающего в себя:

- программу записи активности пользователя FlashBack Recorder,
- редактор записанных видеороликов FlashBack Editor,
- проигрыватель FlashBack Player,
- встраиваемые компоненты ActiveX FlashBack Recorder и FlashBack Player.

Общий объем исходного текста программ на языке C++ составил 12 Мбайт, из них около 50% написаны лично автором настоящей работы.

7. В ходе реализации проекта автором был предложен ряд специальных технологий, ускоряющих процесс обработки данных при записи активности в программе FlashBack Recorder, в том числе методы, позволяющие избежать накладных расходов на передачу данных из видеопамати в системную память (внутренний буфер в системной памяти с отображением всех графических операций на него), механизмы передачи данных из вспомогательного видеодрайвера.

8. Экспериментальное исследование системы BB FlashBack производилось на примерах записи активности пользователя при работе с рядом популярных пакетов прикладных программ: MS Office, Adobe Photoshop, Internet Explorer и др. По большинству критериев BB FlashBack показала преимущество перед всеми системами своего класса. FlashBack уступила только одному продукту DemoForge и только по одному критерию – размеру видеоролика. Однако при этом DemoForge не обеспечивал точное воспроизведение изображения на экране, урезая глубину цвета с 24 до 8 бит, а также грубо фиксировал временные события: смену изображений и передвижение курсора мыши.

9. Пакет BB FlashBack является коммерческой системой, состоящей из линейки специализированных продуктов и распространяемой британской компанией Blueberry Consultants Ltd. Первая коммерческая версия была выпущена в сентябре 2003 года. В настоящее время (январь 2005 года) продается версия 1.4, а всего было продано более 1000 копий. Компанией в 2003 году создан веб-сайт для ознакомления с возможностями продуктов, загрузки демонстрационных версий, сбора отзывов и замечаний, который находится по адресу <http://www.bbsoftware.co.uk>.

Основным продуктом является BB FlashBack Publisher. Он включает в себя поддержку всех разработанных свойств для системы BB FlashBack и может быть использован во многих областях.

Продукт BB TestAssistant представляет из себя специальную версию пакета, в которой отсутствуют средства редактирования, но при этом имеется специализированный набор средств для создания отчетов об ошибках возникающих при тестировании программных продуктов. Отчеты могут быть отправлены разработчикам или в специальный центр через интернет.

Пакет BB FlashBack SDK предоставляет сторонним разработчикам возможность использования технологии BB FlashBack в своих программных продуктах. В пакете имеются два объекта ActiveX: BB FlashBack Recorder и BB FlashBack Player.

Для целей он-лайн поддержки программных продуктов на основе технологии BB FlashBack был создан пакет BB SupportAssistant. Пакет позволяет пользователям программного обеспечения воспользоваться специальным веб-сервисом на веб-сервере компании Blueberry для создания запросов в компанию, разработавшую программу. Запрос может включать в себя запись проблемы пользователя в виде компактного видеоролика формата FBR, который может быть просмотрен разработчиками.

В приложении 1 диссертации приводится руководство пользователя системы BB FlashBack.

В приложении 2 представлены веб-страницы продуктов BB FlashBack, BB TestAssistant и BB FlashBack SDK.

В приложении 3 находятся отзывы различных организаций на систему BB FlashBack: Ник Пичи (Британский Совет), Росс Хопкинс (CBT Interactive), журнал «Компьютера Online», Фил Дэлглиш (клиника Spectrum Health), Мартин Грин (Blueberry Consultants Ltd.).

СПИСОК ПУБЛИКАЦИЙ ПО ТЕМЕ ДИССЕРТАЦИИ

1. Лавров В. А. Использование технологии "BB FLASHBACK" в целях образования // Наука. Технологии. Инновации (материалы науч. конф.), ч. 2. – Новосибирск, 2004. – С. 56–58.
2. Лавров В. А. Разработка методов записи активности пользователя ЭВМ и их реализация в системе BB FlashBack // Вестник Томского государственного университета: Общонаучный периодический журнал. Бюллетень оперативной научной информации. № 22. Март. – Томск: Изд-во Том ун-та, 2004. – 123с.
3. Лавров В. А. Векторно-растровый формат хранения видеоизображений // Обработка данных и управление в сложных системах. – Томск: Изд-во Том. ун-та, 2004. Вып. 6. – С. 106–117.
4. Лавров В. А. Использование Mirco драйвера для записи всех изменений экрана компьютера в файл видеоформата FBR // Вестник ТГУ. Серия «Математика. Кибернетика. Информатика». – Томск: Изд-во Том. ун-та. Декабрь 2004. № 284. – С. 190–193.
5. Лавров В. А. Написание своего Mirco видео драйвера // Вестник ТГУ. Серия «Математика. Кибернетика. Информатика». – Томск: Изд-во Том. ун-та. Декабрь 2004. № 284. – С. 194–197.
6. Лавров В. А. Обучение работе с пакетами программ с помощью систем, записывающих и воспроизводящих действия пользователей // Материалы IV всерос. научно-практ. конф. «Информационные недр Кузбасса». – Кемерово: ИНТ, 2005. – С. 153.
7. Лавров В. А. Практическое сравнение системы BB FlashBack с ее аналогами // Вестник ТГУ. Серия «Математика. Кибернетика. Информатика». – Томск: Изд-во Том. ун-та. Декабрь 2004. № 284. – С. 194–197.
8. Лавров В. А. Тестирование программных продуктов с использованием технологии FlashBack // Материалы VIII Всерос. научно-практ. конф. «Научное творчество молодежи». – Томск: Изд-во Том. ун-та, 2004. – С. 46–47.
9. Valeri Lavrov. Screen recording system for the windows desktop // 8th Korea - Russia International Symposium on Science and Technology. PROCEEDINGS. KORUS. – New York City, NY: IEEE, 2004. Vol. 1. – P. 107–109.