

УДК 004.514.62

Костюк Д.А., Дёмин В.В.

МОДЕЛЬ МИНИ-ОКОН С ДИНАМИЧЕСКИМ ОТОБРАЖЕНИЕМ В АППАРАТНО-УСКОРЕННОМ ГРАФИЧЕСКОМ ИНТЕРФЕЙСЕ

Введение. Задача манипулирования окнами в ситуации, когда их совокупная площадь существенно превышает разрешение устройства вывода, возникла почти одновременно с графическим интерфейсом пользователя. Один из ранних способов облегчить навигацию в таких условиях получил название иконификации и сводится к тому, что окно «сворачивается» в пиктограмму соответствующего приложения, снабженную поясняющей подписью. Изначально пиктограммы иконифицированных окон располагались непосредственно по верху фонового изображения экрана. С распространением метафоры рабочего стола, содержащего пиктограммы файлов, с целью их смыслового разграничения в ряде графических оболочек пиктограммы иконифицированных окон стали располагать на специально отведенной панели или в вертикальном меню, вызываемом определенным действием. При этом и панель, и всплывающее меню часто содержат ярлыки всех окон, а не только скрытых, поскольку являются также удобным механизмом смены фокуса окна.

По мере отхода от изначальной модели большей популярностью стал пользоваться термин «минимизация». Однако концепция иконификации получила и альтернативное развитие. С ростом вычислительных возможностей процессоров и разрешающей способности экранов появились графические оболочки, отображавшие вместо пиктограммы скрытого окна его уменьшенное изображение. Такие уменьшенные копии окон получили разные названия в различных графических оболочках. Наиболее распространены термины «мини-окно» и «миниатюра окна».

Использование мини-окон требует решения ряда технических проблем; однако возможность видеть одновременно, хотя бы с уменьшенной детализацией, изображение всего рабочего пространства дает пользователю ощутимые преимущества [1, 2]. Особенно это актуально в среде динамически изменяющихся объектов, к которым относятся все многозадачные графические среды современных операционных систем.

Само по себе применение уменьшенных изображений объектов для предварительного просмотра (previews или thumbnails) — тумбнейлинга — наблюдается в современных версиях ряда популярных программных продуктов. Оно непосредственно задействует модель природного ориентирования человека в пространстве [2] и на сегодняшний день является одним из наиболее окупаемых способов повышения интуитивности интерфейса.

Несмотря на явные преимущества для пользователя, технические трудности, связанные с обновлением масштабированных изображений в реальном масштабе времени, привели к тому, что модель мини-окон до настоящего времени распространена в незначительной степени либо реализована в сильно искаженном виде, затрудняющем ее использование по основному назначению.

1. Модель мини-окон в классических оконных средах. До появления аппаратно-ускоренных графических интерфейсов вычислительные затраты, необходимые для реализации мини-окон, отображающих динамику в реальном масштабе времени, в большинстве случаев представлялись неоправданными. Разработчики ограничивались компромиссным вариантом, обновляя мини-окна через заданный интервал либо вовсе оставляя в них изображение, наблюдавшееся в момент минимизации окна. Разумеется, подобные ограничения заметно снижали полезность мини-окон как модели интерфейса, а, следовательно, и популярность соответствующих программных продуктов.

Костюк Дмитрий Александрович, к.т.н., доцент кафедры «ЭВМ и системы» Брестского государственного технического университета.

Дёмин Владимир Владимирович, студент IV курса факультета электронно-информационных систем Брестского государственного технического университета.

Беларусь, БрГТУ, 224017, г. Брест, ул. Московская, 267.



а)



б)

Рис. 1. Модель мини-окон в графических оболочках X Window System – FVWM-Crystal (а) и Mezzo Desktop (б)

Одну из ранних относительно полноценных реализаций модели мини-окон в UNIX-подобных ОС можно наблюдать в оконном менеджере FVWM-Crystal – доработанной версии менеджера FVWM, одного из старейших оконных менеджеров для X Window System. В FVWM-Crystal пользователь может выбрать один из нескольких различных режимов работы (т.н. рецептов). Два из стандартных рецептов, Top-Down и Top-Line, реализуют концепцию мини-окон, называемых в FVWM тумбнейлами. При минимизации окна менеджер автоматически размещает его уменьшенное изображение в нижней части рабочего стола (рис. 1а).

Технология создания тумбнейла, задействованная в данном оконном менеджере и в ряде аналогов, включает три стандартные процедуры. Окно перемещается за пределы отображаемой на дисплее области и с помощью запуска внешней утилиты создается его скриншот. Далее, полученное изображение масштабируется с помощью процессора изображений imagemagick и выводится на экран. Поскольку создание тумбнейла сопряжено с вызовом внешних при-

ложений, его частое повторение приводит к непомерному потреблению ресурсов; поэтому интервал обновления тумбнейла обычно составляет 5–10 секунд (в различных вариантах настройки), либо обновление отключено.

Позже на базе FVWM-Crystal была разработана инновационная графическая оболочка Mezzo Desktop проекта SymphonyOS, собравшая в себе ряд передовых решений эргономики графического интерфейса [3]. Рабочее пространство Mezzo Desktop реализовано с использованием концепции мини-окон (рис. 1б) с той разницей, что их изображения снабжены дополнительным эстетическим обрамлением в виде «отражения» мини-окна от поверхности рабочего стола. Обновление изображений мини-окон не производится (не в последнюю очередь из-за дополнительной вычислительной нагрузки, вносимой данным визуальным эффектом).

Аналогичную функциональность для Windows XP (рис. 2) реализуют две сторонние программы: ThumbWin и miniMize [4]. Оба приложения перехватывают события минимизации окон и располагают их уменьшенные изображения на рабочем столе. Отсутствие визуального выделяющегося обрамления часто приводит к трудностям при идентификации мини-окон среди пиктограмм. Активное использование рабочего стола для хранения данных нарушает целостность концепции мини-окон в обоих вариантах. Сходство с пиктограммами усиливает также возможность перемещать мини-окна мышью по рабочему столу и реализованный в miniMize режим их случайного размещения (ThumbWin упорядочивает расположение мини-окон автоматически в привязке к выбранному углу экрана). Обе программы позволяют задавать размер мини-окон и режим отображения (поверх остальных окон или под ними). Обновление изображения окна не производится. Поверх мини-окна выводится уменьшенная пиктограмма приложения для его более легкой идентификации.

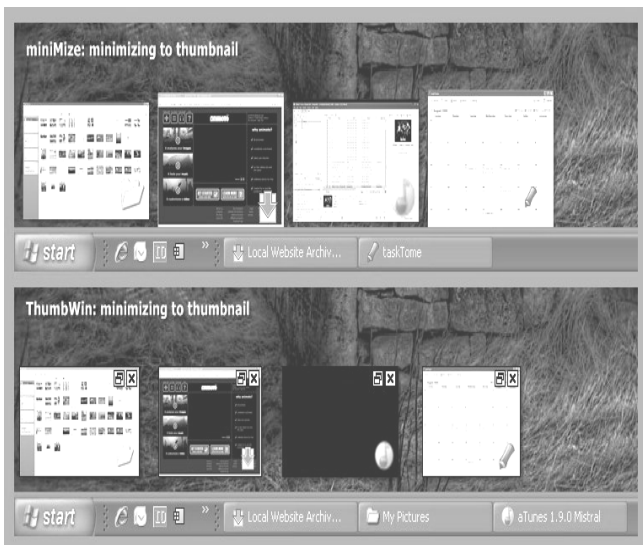


Рис. 2. Модель мини-окон в ОС Windows: программы ThumbWin и miniMize

2. Мини-окна в оболочках с аппаратно-ускоренной графикой.

С распространением аппаратного ускорения графики появилась возможность реализации мини-окон с динамически обновляемым содержимым без существенного роста загрузки центрального процессора — за счет передачи функций масштабирования и отрисовки процессору видеоадаптера [2].

Первая (и единственная на сегодняшний день) реализация динамических мини-окон под ОС Windows продемонстрирована на модифицированной конфигурации [5] графической оболочки Emerge Desktop — сторонней программы, предназначенной для замены стандартной оболочки Explorer. Боковая панель, отображающая уменьшенные изображения открытых документов, служит заменой панели задач; при наведении указателя мыши мини-окна увеличиваются в размерах, а выбор одного из них передает фокус на соот-

ветствующее окно. Отображение в реальном масштабе времени требует ОС версии Vista или 7. Панель показывает уменьшенные изображения всех окон, независимо от того, минимизированы они или нет, что в сущности нарушает рассматриваемую модель интерфейса (рис. 3а).



a)



б)

Рис. 3. Миниатюры окон, отображаемые с аппаратным ускорением: модификация оболочки Emerge Desktop (а) и уменьшенные изображения окон приложения в Windows 7 (б)

В Windows 7 реализован технологически-близкий механизм [6], отображающий уменьшенные изображения окон для выбранного на панели задач приложения. При наведении указателя мыши на кнопку соответствующего приложения в панели задач над ней отображается дополнительная панель с мини-окнами (рис. 3б). Хотя отображение мини-окон обновляется в реальном масштабе времени, они не могут использоваться для фонового отслеживания содержимого окон, т.к. не присутствуют в видимой области постоянно и требуют от пользователя дополнительных действий для показа. Поэтому речь идет о сходном механизме, а не о реализации модели.

Аналогичное представленному на рис. 3б решение можно наблюдать и в современных оконных менеджерах X Window System, взаимодействующих с 3D-акселератором. Наиболее популярный среди них на сегодняшний день оконный менеджер Compiiz реализует в качестве одной из возможностей вывод уменьшенного всплывающего изображения окна при наведении указателя мыши на соответствующую ему кнопку в панели задач. Отображение содержимого окна при этом также происходит динамически, однако всплывающие изображения показываются только для неминимизированных окон.

На базе Compiiz при участии одного из авторов статьи в 2006 году была разработана экспериментальная графическая оболочка, выполняющая масштабирование окон при их перемещении в боко-

вую область экрана. Масштаб окна при этом вычисляется на основе координат левого верхнего угла окна. Данная разработка воспроизводит интерфейсную модель дискретного масштабирования окон [2, 7] и концептуально-близка рассматриваемой модели мини-окон.

Таким образом, если использование модели мини-окон долгое время сдерживалось техническими причинами, то в настоящее время оно ограничивается скорее по исторически сложившимся обстоятельствам. Однако поскольку преимущества обновляющихся в реальном масштабе времени уменьшенных изображений неактивных окон очевидны, продолжают появляться программные решения, в том или ином виде использующие данную возможность.

3. Реализация мини-окон на базе менеджера Compoz. В настоящей работе нами предпринята попытка реализовать модель мини-окон в ее изначальном виде на базе аппаратно-ускоренного оконного менеджера (в качестве которого использован Compoz) и библиотеки OpenGL. Одно из свойств OpenGL – объекты фреймбуфера – дает оконному менеджеру эффективный доступ к окнам неактивных приложений. Для приложения объекты фреймбуфера выглядят как обычные окна, а для оконного менеджера — как текстуры, которыми можно управлять с помощью обычных команд отрисовки мультитекстуры (рис. 4).

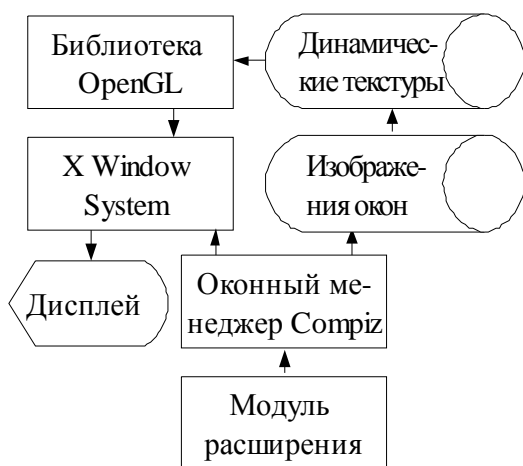


Рис. 4. Схема взаимодействия разработанного модуля с графической оболочкой и средствами поддержки аппаратно-ускоренной графики

Разработанный модуль расширения Compoz состоит из трех функциональных частей: кода инициализации, обработчиков событий и сервисных функций.

Код инициализации вызывает при загрузке модуля. При этом системе передаются указатели на конструкторы и деструкторы, вызываемые при создании и удалении контекста дисплея, каждого из соответствующих дисплею виртуальных экранов, и каждого из соответствующих виртуальному экрану окон. Доступ к контексту дисплея позволяет перехватывать события, экран дает возможность включиться в цепочку процедур обновления изображения, включающую подготовительную стадию, собственно отрисовку экрана (или экранов, если их несколько) и завершение отрисовки (рис. 5).

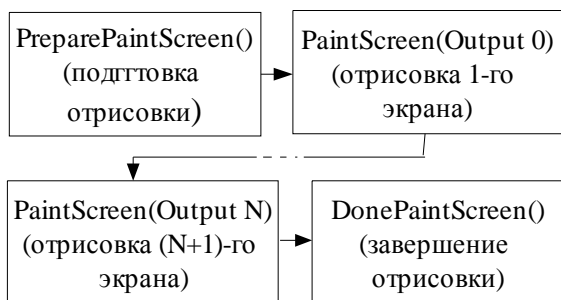


Рис. 5. Отрисовка изображения

Подготовительная стадия используется для выполнения различных подготовительных расчетов, например, для вычисления новых координат. Метод отрисовки в первую очередь должен вызвать базовую реализацию, обновляющую изображение экрана, а затем выполняет с полученным изображением любые действия, доступные средствами OpenGL; при этом он имеет доступ к устройству вывода, физическим координатам и всем открытым окнам, для каждого из которых существует соответствующая текстура. Стадия завершения отрисовки при необходимости используется для принудительной генерации события перерисовки экрана, благодаря чему выполняется отображение анимационных эффектов в реальном масштабе времени.

Формирование мини-окон при минимизации в системе X Window System сопряжено с технической проблемой, не актуальной, например, для модели дискретного масштабирования окон [7]. При наступлении события минимизации оконный менеджер уничтожает изображение окна и соответствующую динамическую текстуру, и вновь создает ее только при восстановлении окна из минимизированного состояния. Эта архитектурная особенность X-сервера позволяет экономить ресурсы на окнах, с которыми пользователь не работает в настоящий момент, и проявляется, в частности, в упомянутом выше отсутствии всплывающих миниатюр для минимизированных окон. Таким образом, поскольку показ мини-изображений таких окон невозможен, было принято решение о переназначении команды минимизации окна и закреплении ее за активированием разработанного модуля.

В ходе активации модуля запоминаются прежние координаты окна, устанавливается новое значение координат в соответствии с предназначенной для мини-окон областью внизу экрана (рис. 6), а также устанавливается значение коэффициента масштабирования, отличное от единицы. По этому коэффициенту код модуля, обрабатывающий события отрисовки, отличает минимизированные окна от стандартных.

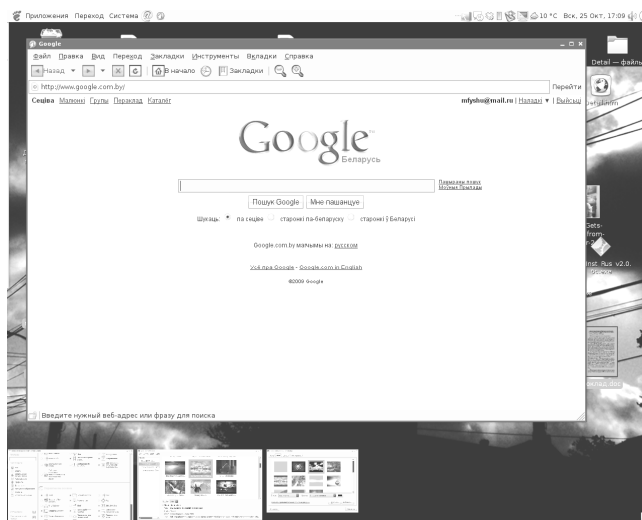


Рис. 6. Мини-окна, отображаемые разработанным модулем

При инициализации экрана модуль закрепляет собственные обработчики за необходимыми событиями. На данный момент задействованы следующие основные обработчики событий:

1. PreparePaintScreen, рассмотренный в пояснении к рис. 5;
2. PaintWindow, срабатывающий каждый раз при обновлении содержимого окна, проверяющий масштаб окна и при необходимости вносящий изменения в его матрицу трансформации;
3. DamageWindowRect, определяющий конкретный фрагмент экрана как требующий перерисовки (таким образом достигается дополнительное снижение аппаратных затрат по сравнению с полным обновлением экрана).

Также служебная функция, встроенная в цикл обработки событий, отслеживает щелчок мышью по мини-окну для возвращения его к единичному масштабу и прежним координатам на экране.

Изменение масштаба окна затрагивает только его изображение. Очевидно, что для самого окна при этом не происходит никаких изменений в размерах, иначе была бы нарушена обратная совместимость с существующими приложениями. Однако аналогичная ситуация с размерами окна сохраняется и для X-сервера. Поэтому с точки зрения системы все элементы управления окна сохраняют свои прежние координаты, и для взаимодействия с этими элементами необходимо помещать указатель мыши в ту область экрана, где они находились бы, сохраняя окно единичный масштаб. Что еще более нежелательно, манипуляция мышью за пределами мини-окна должна приводить к активации его различных элементов.

Один из способов решения данной проблемы подразумевает использование нестандартной версии X-сервера, позволяющей выполнять перенаправление ввода. Существует экспериментальная модификация кода X-сервера, предложенная в 2007 г. для реализации такого перенаправления. Для этого оконный менеджер должен предоставить X-серверу последовательность (т.н. «сетку») из треугольных элементов [8]. Четные элементы описывают треугольник на поверхности трансформированного окна, а нечетные — соответствующий ему треугольник, являющийся частью исходного окна.

Помимо некоторых проблем в стабильности, препятствующих включению данной модификации в стандартный релиз X-сервера, подход имеет еще как минимум один недостаток: существующая реализация позволяет работать только с одной сеткой. Таким образом, первый загруженный модуль расширения Compriz, использующий перенаправление ввода, монополизировать данную функцию.

Альтернативный способ избежать рассогласования в координатах, реализованный в т.ч. и в представляемой разработке, предполагает перехват событий мини-окна, связанных с указателем мыши. Это не позволяет взаимодействовать с элементами окна, но такая возможность для мини-окон и не требуется. События мыши, приходящие на область масштабированного изображения, обрабатываются разработанным модулем расширения Compriz (для подсветки мини-окна при движении указателя или перевода в немасштабированное состояние по щелчку), а остальные — блокируются. Данный механизм поддерживается непосредственно через расширение XShape, изначально предназначенное для приложений, используя-

щих канал прозрачности для отображения окон непрямоугольной формы (события мыши, попадающие на внешние прозрачные участки окна, блокируются). Механизм известен как формирование входных сигналов и задействован в Compriz при реализации ряда стандартных модулей.

При этом для мини-окон остаются по-прежнему доступными события клавиатуры и смена фокуса стандартными средствами переключения окон, что позволяет в ряде случаев взаимодействовать с мини-окном без его возвращения к стандартному масштабу.

СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ

1. Раскин, Дж. Интерфейс: новые направления в проектировании компьютерных систем. — СПб.: Символ-Плюс, 2003. — 272 с.
2. Борушко, И.Н. Применение модели периферического зрения в графическом интерфейсе пользователя / И.Н. Борушко, Е.В. Гоманова, Д.А. Костюк. — Современные информационные компьютерные технологии: сб. науч. ст. — Гродно: ГрГУ, 2006. — С. 22–27.
3. J. Spisak. Laws of Interface Design. Isptec / SymphonyOS Project. 2004. <http://web.archive.org/web/20050610031031/http://www.symphonyos.com/uilaws.html>
4. S. Kurdi. ThumbWin and miniMize: two apps that can minimize open windows as desktop thumbnails. August 25, 2008. <http://www.freewaregenius.com/2008/08/25/thumbwin-and-minimize-two-apps-that-can-minimize-open-windows-as-desktop-thumbnails/>
5. G. Trapani. Windows Vista with a Live Thumbnail Sidebar. Oct 6 2008. <http://lifelifehacker.com/5058949/windows-vista-with-a-live-thumbnail-sidebar>
6. Trying out Windows 7 Release Candidate – Part 2. May 10, 2009. <http://itsbytor.wordpress.com/2009/05/10/trying-out-windows-7-release-candidate-part-2/>
7. Гоманова, Е.В. Применение аналогии периферического зрения в аппаратно-ускоренном графическом интерфейсе пользователя / Е.В. Гоманова, Д.А. Костюк Д.А. // Вестник БрГТУ. — №5: Физика, математика, информатика. — 2007. — С. 33–35.
8. Input Redirection Update. 21/10/2008. <http://smspillaz.wordpress.com/2008/10/21/input-redirection-update/>

Материал поступил в редакцию 03.11.09

KOSTIUK D.A., DIOMIN V.V. Model of Mini-windows Dynamically Mapped in Hardware-accelerated Graphic Interface

An interface model based on usage of scaled-down images of minimized windows is considered. Carried out analysis of graphic environments, applying mini-windows or close decisions shows the absence of dynamically refreshing mini-windows in their pure form despite there are needed technical abilities in modern hardware. Developed external module for Compriz window manager is presented, which uses hardware-accelerated graphics for real-time mini-windows mapping.

УДК 621.9.08

Грисевич Л.Н., Костюк Д.А.

ПРОГРАММНАЯ СИСТЕМА АВТОМАТИЗАЦИИ ЭКСПЕРИМЕНТАЛЬНЫХ ИССЛЕДОВАНИЙ АКУСТИЧЕСКОЙ СПЕКТРОСКОПИИ

Введение. Персональные компьютеры в последнее время широко используются среди потребителей измерительной техники. Использование программного обеспечения в качестве виртуальных средств измерений в сочетании с дополнительной платой-адаптером или внешним модулем позволяет быстро превратить серийный компьютер в полноценный измерительный комплекс с широкими возможностями измерений, отображения, обработки и хранения полученной информации.

Существует ряд особенностей, делающих виртуальные измерительные приборы более предпочтительными по сравнению с традиционными (имеющими собственный экран и органы управления) устройствами [1]:

- использование компьютера для задания режимов измерения, отображения и первичной обработки результатов позволяет в ряде случаев добиться экономии средств, места и веса (поскольку компьютер и так должен присутствовать в лаборатории);
- обычно результаты измерений необходимо обрабатывать и протоколировать. В случае виртуального прибора данные сразу находятся в памяти компьютера, и для их пересылки не требуются дополнительных манипуляций;
- настройки современных приборов становятся все более сложными и разнообразными, поэтому настройка компонент измерительного комплекса для решения конкретной задачи может потребовать значительных временных затрат. Различные вариан-

Грисевич Лариса Николаевна, магистрант кафедры ЭВМиС Брестского государственного технического университета. Беларусь, БрГТУ, 224017, г. Брест, ул. Московская, 267.