

МОДЕЛЬ СИСТЕМЫ ФОРМИРОВАНИЯ ИЗОБРАЖЕНИЙ АБОНЕНТСКОГО ТЕРМИНАЛА СОВРЕМЕННЫХ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННЫХ СЕТЕЙ

MODEL OF THE SYSTEM OF FORMATION OF IMAGES OF THE USER TERMINAL OF MODERN INFORMATION-TELECOMMUNICATION NETWORKS

V. Bessoltsev

Summary. This scientific article is devoted to the description of the imaging model of subscriber terminals displayed in modern web browsers.

Keywords: web browser, subscriber terminal, information and telecommunications network.

Бессольцев Виталий Евгеньевич

Адъюнкт, Военно-космическая академия имени

А. Ф. Можайского (г. Санкт-Петербург)

v.bessoltsev@gmail.com

Аннотация. Данная научная статья посвящена описанию модели формирования изображений абонентских терминалов, выводимых в современных веб-обозревателях.

Ключевые слова: веб-обозреватель, абонентский терминал, информационно-телекоммуникационная сеть.

Стремительное развитие современных информационно-коммуникационных сетей неизбежно влечет за собой создание и модернизацию существующих стандартов и протоколов передачи, и представления данных. Так, несколько лет назад было анонсировано и практически всеми веб-обозревателями оперативно осуществлена поддержка стандарта HTML5. В настоящее время разработчики активно используют возможности этого стандарта для улучшения уровня поддержки мультимедиа-технологий.

Самым важным новым инструментом для расширенных приложений HTML5 [1] является поверхность для рисования — Canvas, на которой при помощи кода на JavaScript происходит отрисовка графического изображения. При использовании Canvas с сервера загружается не картинка, а набор точек (или алгоритм прорисовки), по которым веб-обозреватель прорисовывает картинку.

Таким образом, Canvas, является средством программирования, которое позволяет рисовать как простейшие графические примитивы — линии, фигуры, текст, так и создавать различные эффекты мультимедиа, такие как игры, картографические инструменты и динамические графики, музыкально-световые представления и эмуляторы физических процессов.

Отрисовка изображений в Canvas возможна при наличии в веб-обозревателе поддержки WebGL — кроссплатформенного API для работы с графикой в веб-

обозревателе. То есть, WebGL — это программный интерфейс для доступа к графическому оборудованию в рамках веб-обозревателя, причём без установки каких-либо дополнительных расширений. Основанный на OpenGL, он позволяет программисту определить объекты и операции для создания высококачественных графических изображений, в частности цветных изображений трёхмерных объектов.

Рассмотрим модель формирования изображений в веб-обозревателях абонентских терминалов современных информационно-телекоммуникационных сетей (рисунок 1).

При передаче запроса на сервер в веб-обозреватель поступают данные [2] совместно с кодом на отрисовку изображения $I_{[L,L]}$ в Canvas (рисунок 2).

В веб-обозревателе происходят преобразования полученных данных для передачи на следующий уровень обработки, при этом учитываются параметры Φ_1 , которые зависят от конкретной реализации веб-обозревателя (движок веб-обозревателя).

В любой компьютерной системе взаимодействия прикладных программ с аппаратным обеспечением реализуется при помощи операционной системы.

Веб-обозреватель с целью получения вычислительных ресурсов для обработки полученного изображения $I_{[L,L]}$, которое представлено исходным кодом шейдера,

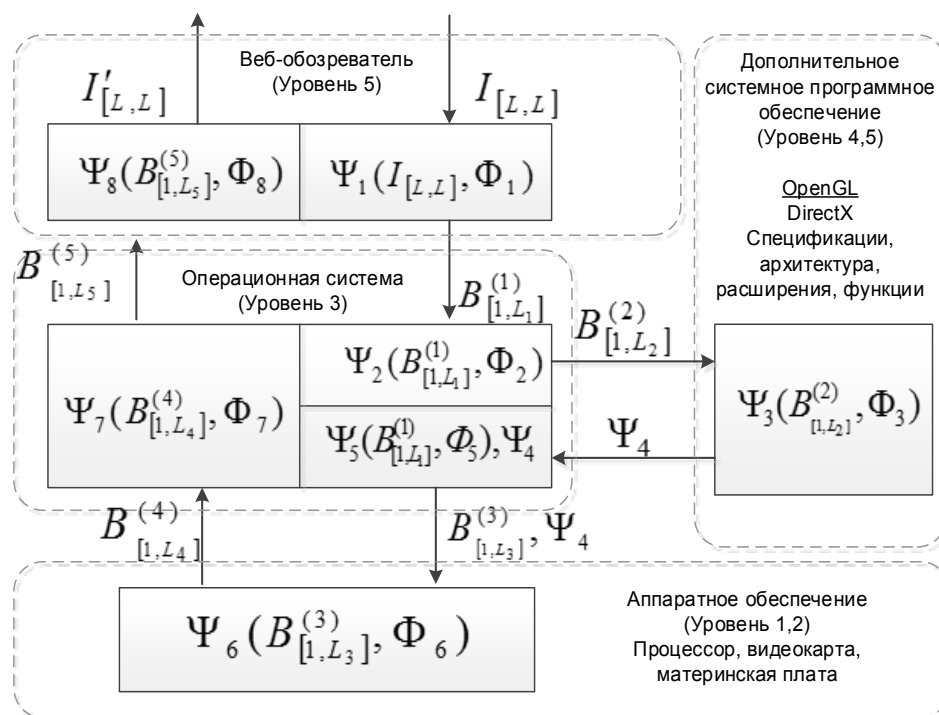


Рис. 1. Модель программно-аппаратной системы обработки графических изображений абонентского терминала ИКС

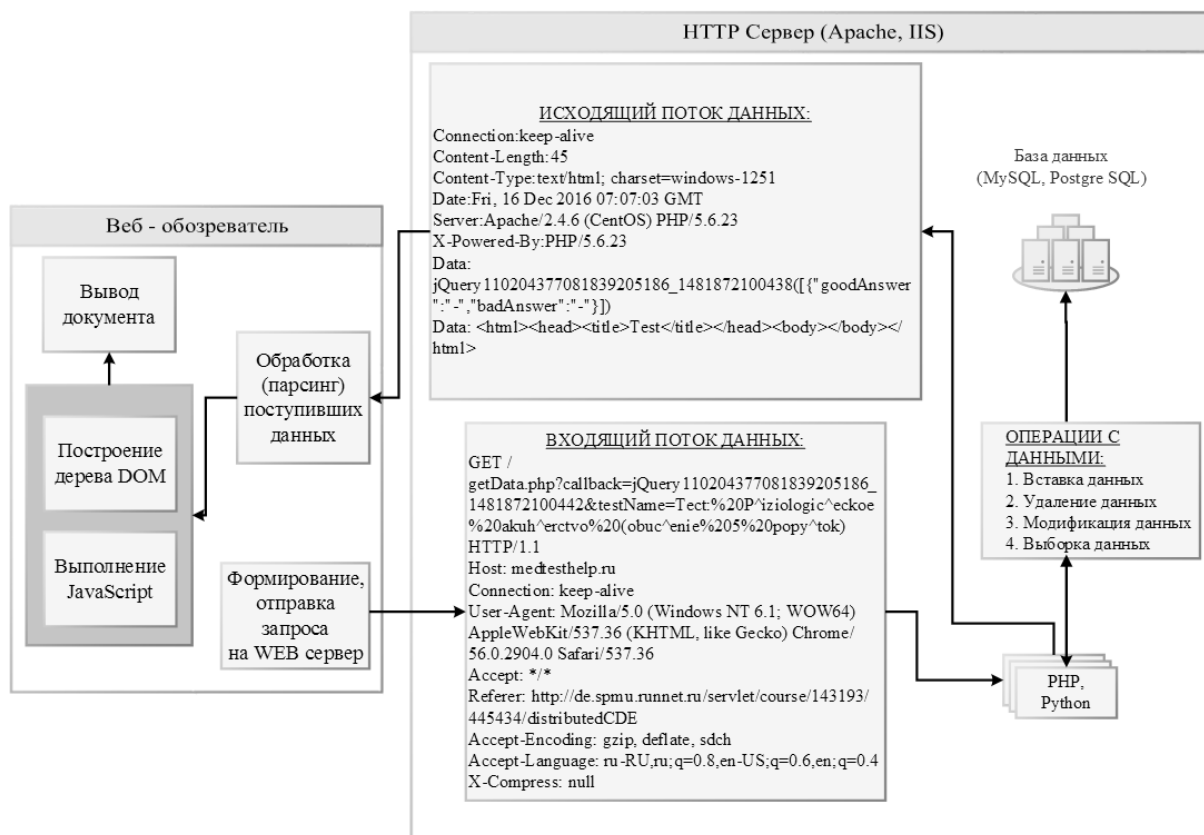


Рис. 2. Схема взаимодействия веб-обозревателя абонентского терминала с веб-сервером

создает запрос к операционной системе. Запрос содержит обработанные веб-обозревателем данные и последовательность команд для процессора $B_{[1,L_1]}^{(1)}$. Операционная система преобразовывает поступившие данные и команды и формирует процедуры взаимодействия с дополнительным системным программным обеспечением $\Psi_2(B_{[1,L_1]}^{(1)}, \Phi_2)$, при этом учитываются функционал и параметры операционной системы Φ_2 . Сформированная последовательность, включающая набор необходимых действий над изображением, и преобразованные данные изображения $B_{[1,L_2]}^{(2)}$ поступают на следующий уровень обработки.

Дополнительное системное программное обеспечение предназначенное для работы с изображениями представлено спецификациями OpenGL и DirectX [3], определяющими независимый от языка программирования программный интерфейс для написания приложений, использующих двумерную и трёхмерную компьютерную графику. Производители оборудования на основе этих спецификаций создают реализации — библиотеки функций, соответствующих набору функций спецификации.

На уровне 4,5 производится выработка инструкций для аппаратного оборудования $\Psi_3(B_{[1,L_2]}^{(2)}, \Phi_3)$, а именно определяется алгоритм вычислительных операций для получения из исходного кода шейдера последовательности данных для отрисовки графического изображения. При этом на итоговый алгоритм Ψ_4 влияют особенности реализации спецификаций конкретными производителями Φ_3 .

Операционная система на основании Ψ_4 формирует последовательность команд Φ_5 для аппаратного оборудования и определяет порядок взаимодействия до-

полнительного системного программного обеспечения с видеокарты $\Psi_5(B_{[1,L_1]}^{(1)}, \Phi_5), \Psi_4$ [4].

Множество возможных реализаций аппаратной части абонентского терминала и отсутствие строгой стандартизации технологического процесса их производства и использования комплектующих деталей порождает множество параметров Φ_6 которые влияют на параметры формируемого графического изображения. При этом производитель может применять фирменные алгоритмы обработки Ψ_6 поступающих данных $B_{[1,L_3]}^{(3)}$, но при этом должны учитываться правила обработки Ψ_4 сформированные на вышележащих уровнях.

После вычислительных операций $\Psi_6(B_{[1,L_3]}^{(3)}, \Phi_6)$ обработанные данные $B_{[1,L_4]}^{(4)}$ передаются операционной системе, которая выполнив соответствующие преобразования $\Psi_7(B_{[1,L_4]}^{(4)}, \Phi_7)$ передает их приложению выполнившему запрос, то есть, веб-обозревателю. Веб-обозреватель, в свою очередь, выводит в Canvas итоговое графическое изображение.

Таким образом, современный абонентский терминал информационно-коммуникационных сетей представляет собой сложную многоуровневую систему, которая в результате обработки поступающего изображения производит поэтапные процедуры преобразования данных. При этом, на каждом этапе происходит внесение некоторых изменений, обусловленных программно-аппаратной реализацией конкретного абонентского терминала. Множество возможных реализаций в совокупности с многоэтапностью преобразований [5] позволяет сделать вывод об уникальности системы формирования изображений абонентского терминала современных информационно-телекоммуникационных сетей.

ЛИТЕРАТУРА

1. Основы HTML5 — URL: <https://www.ibm.com/developerworks/ru/library/wahtml5fundamentals/index.html> (дата обращения: 14.01.2019).
2. Идентификация абонентов анонимных компьютерных сетей посредством выявления уникальных параметров веб-обозревателя / К. В. Сазонов и др. // Труды Военно-космической академии имени А. Ф. Можайского. — СПб.: ВКА им. А. Ф. Можайского, 2018. — Вып. 665. — С. 99–111.
3. Кулаков В. Г., Программирование на аппаратном уровне: специальный справочник, 2003. — 847с.
4. Алгоритмы антиальсинга в реальном времени — URL: <https://habr.com/ru/post/343876/> (дата обращения: 12.01.2019).
5. Вероятностные алгоритмы. — URL: <http://datareview.info/article/veroyatnostnyie-algoritmyiveroyatno-eto-rabotaet-chast-2> (дата обращения: 24.12.2018).

© Бессольцев Виталий Евгеньевич (v.bessoltsev@gmail.com).

Журнал «Современная наука: актуальные проблемы теории и практики»