

Введение

Когда я сделал первую попытку превратить свой опыт работы с ядром Linux в текст книги, понял, что не знаю, как двигаться дальше. Не хотелось просто писать еще одну книгу о ядре операционной системы. Конечно, на эту тему *не так уж и много* книг, но все же я хотел сделать что-то такое, благодаря чему моя книга была бы особенной. Как достичь этой цели? Я не могу успокоиться, пока не сделаю что-нибудь особенное, лучшее в своем роде.

Наконец я решил, что смогу предложить достаточно уникальный подход к данной теме. Моя работа — изучение и разработка ядра операционной системы. Мое увлечение — изучение и разработка ядра операционной системы. Моя любовь — ядро операционной системы. Конечно, за многие годы я успел собрать много интересных анекдотов и полезных советов. С моим опытом я смог бы написать книгу о том, как нужно разрабатывать программный код ядра и как этого делать *не нужно*. Прежде всего, эта книга об устройстве и практической реализации ядра операционной системы Linux. В ней информация представлена так, чтобы получить достаточно знаний для решения реальных практических задач и чтобы эти задачи решать правильно. Я человек прагматичный, и книга имеет практический уклон. Она должна быть полезной, интересной и легко читаться.

Я надеюсь, что читатели, после прочтения этой книги, получают хорошее понимание тех правил (писанных и неписанных), которые действуют в ядре операционной системы. Я также надеюсь, что читатели сразу после прочтения этой книги смогут начать действовать и писать полезный, правильный и хороший код ядра. Конечно, эту книгу можно читать и просто ради интереса.

Это то, что касалось еще первого издания книги. Однако время идет и снова приходится возвращаться к рассмотренным вопросам. В этом издании представлено несколько больше информации по сравнению с первым: материал серьезно пересмотрен и доработан, появились новые разделы и главы. С момента выхода первого издания в ядро были внесены изменения. Однако, что более важно, сообщество разработчиков ядра Linux приняло решение¹ в ближайшем будущем не начинать разработку серии ядра 2.7. Было решено заняться стабилизацией серии ядра 2.6. Стабилизация включает в себя много моментов, тем не менее есть один важный, который касается данной книги, — книга, которая посвящена ядру серии 2.6, остается актуальной. Если изменения происходят не слишком быстро, то существует большой шанс, что “моментальный снимок” ядра останется актуальным и в будущем. В конце концов, книга сможет вырасти и стать канонической документацией по ядру. Я надеюсь, что именно такая книга и находится у вас в руках.

Как бы там ни было, книга написана, и я надеюсь, что она вам понравится.

¹ Это решение было принято на саммите разработчиков ядра Linux (Linux Kernel Development Summit), который состоялся летом 2004 года в г. Оттава, Канада.

Итак...

Разработка программного кода ядра операционной системы не требует наличия гениальной, волшебной или густой бороды Unix-хакера. Хотя ядро операционной системы и имеет некоторые свои особенности, оно незначительно отличается от любого большого программного продукта. Так же как и в случае любой сложной программы, здесь есть, что изучать, но в программировании ядра не намного больше священных или непонятных вещей, чем в создании любой другой программы.

Очень важно, чтобы вы читали программный код. Доступность открытого исходного кода операционной системы Linux — это подарок, который встречается очень редко. Однако недостаточно *только* читать исходный код. Необходимо взяться за дело серьезно и изменять этот программный код. Находите ошибки и исправляйте их! Улучшайте драйверы для своего аппаратного обеспечения! Находите слабые места и закрывайте их! У вас все получится, если вы будете сами *писать* программный код.

Версия ядра

Эта книга посвящена ядрам Linux серии 2.6 и базируется на версии ядра 2.6.10. Ядро — это “движущийся объект”, и никакая книга не в состоянии передать динамику во все моменты времени. Тем не менее базовые внутренние структуры ядра уже сформировались, и основные усилия по представлению материала были направлены на то, чтобы этот материал можно было использовать и в будущем.

Читательская аудитория

Эта книга предназначена для разработчиков программного обеспечения, которые заинтересованы в понимании ядра операционной системы Linux. Тем не менее это *не* построчные комментарии исходного кода ядра. Это также не руководство по разработке драйверов и не справочник по программному интерфейсу (API) ядра (кстати, формализованного API ядра Linux никогда не было). Целью книги является предоставление достаточной информации об устройстве и реализации ядра для того, чтобы подготовленный программист смог начать разработку программного кода. Разработка ядра может быть увлекательным и полезным занятием, и я хочу ознакомить читателя с этой сферой деятельности по возможности быстро. В книге обсуждаются как вопросы теории, так и практические приложения, она обращена к людям, которые интересуются и тем, и другим. Я всегда придерживался мнения, что для понимания практических приложений необходима теория, тем не менее я считаю, что эта книга не сильно углубляется в оба этих направления. Я надеюсь, что, независимо от мотиваций необходимости понимания ядра операционной системы Linux, эта книга сможет объяснить особенности устройства и реализации в достаточной степени.

Таким образом, данная книга освещает как использование основных подсистем ядра, так и особенности их устройства и реализации. Я думаю, что эти вопросы важны и достойны обсуждения. Хороший пример — глава 7, “Обработка нижних половин и отложенные действия”, посвященная обработчикам *нижних половин* (bottom half).

В этой главе рассказывается о принципах работы и об особенностях реализации механизмов обработки нижних половин (эта часть может быть интересна разработчикам основных механизмов ядра), а также о том, как на практике использовать экспортируемый интерфейс ядра для реализации собственных обработчиков bottom half (это может быть интересно для разработчиков драйверов устройств). На самом деле мне кажется, что обе эти стороны обсуждения будут интересны для всех групп разработчиков. Разработчик основных механизмов ядра, который, конечно, должен понимать принципы работы внутренних частей ядра, должен также понимать и то, как интерфейсы ядра будут использоваться на практике. В то же самое время разработчик драйверов устройств получит большую пользу от хорошего понимания того, что стоит за этим интерфейсом.

Все это сродни изучению программного интерфейса некоторой библиотеки наряду с изучением того, как эта библиотека реализована. На первый взгляд, разработчик прикладных программ должен понимать лишь интерфейс (API). И действительно, интерфейсы часто предлагают рассматривать в виде черного ящика. Разработчик библиотеки, наоборот, обычно интересуется лишь принципом работы и реализацией функций библиотеки. Я уверен, что обе группы разработчиков должны потратить некоторое время на изучение другой стороны предмета. Разработчик программ, который хорошо понимает операционную систему, сможет значительно лучше эту операционную систему использовать. Аналогично разработчик библиотеки должен иметь хотя бы малое представление о том, что происходит в реальной жизни, и, в частности, о тех программах, в которых будет использоваться его библиотека. Поэтому я старался коснуться как устройства, так и использования подсистем ядра не только в связи с тем, что эта книга может быть полезна для одной или другой группы разработчиков, а в надежде, что *вся материал* книги будет полезен для всех разработчиков.

Предполагается, что читатель знаком с языком программирования C и операционной системой Linux. Некоторые знания принципов построения операционных систем также желательны. Я старался объяснять все понятия, однако в случае проблем в списке литературы можно найти несколько отличных книг, которые посвящены основам построения операционных систем.

Эта книга будет полезна для студентов, изучающих основы построения операционных систем, в качестве *прикладного* пособия и вводного материала по соответствующей теории. Книга пригодна как для расширенных специальных курсов, так и для общих специальных курсов, причем в последнем случае без дополнительных материалов. Я прошу потенциальных учебных инструкторов связаться со мной; я буду очень рад оказать помощь.

Интернет-ресурс

Автор поддерживает Интернет-сайт http://tech9.net/rml/kernel_book/, содержащий информацию о данной книге, включая ошибки, расширенные и исправленные разделы, а также информацию о будущих изданиях. Всем читателям рекомендуется посетить этот сайт.

Благодарности ко второму изданию

Как и большинство авторов, я писал эту книгу, не сидя в пещере (что само по себе хорошо, потому что в пещерах могут водиться медведи), и, следовательно, многие люди оказали мне поддержку в создании рукописи своим сердцем и умом. Поскольку невозможно привести полный список этих людей, я хочу поблагодарить всех своих друзей и коллег за помощь, поддержку и конструктивную критику.

В первую очередь, я хотел бы высказать благодарность моему редактору Скотту Мейерсу (Scott Meyers) за руководство, благодаря которому второе издание книги превратилось из идеи в конечный продукт. Мне снова было очень приятно работать с Джоржем Недеффом (Georg Nedeff), производственным редактором, который во всем обеспечивал порядок. Особая благодарность литературному редактору Марго Кэтс (Margo Catts). Мы можем только желать, чтобы наше владение ядром было так же совершенно, как ее владение печатным словом.

Отдельное спасибо техническим редакторам этого издания Адаму Белею (Adam Belay), Мартину Пулу (Martin Pool) и Крису Ривере (Chris Rivera). Их знания и исправления помогли сделать эту книгу неизмеримо лучше. Если, несмотря на их неоценимые усилия, все же остались ошибки, то это вина автора. Такое же большое спасибо Заку Брауну (Zak Brown), который приложил огромные усилия к техническому редактированию первого издания.

Многие разработчики ядра отвечали на вопросы, предоставляли поддержку или просто писали программный код, интересный настолько, что по нему можно было бы написать отдельную книгу. Среди них Андреа Аркангели (Andrea Arcangely), Алан Кокс (Alan Cox), Грег Кроах-Хартман (Greg Kroah-Hartman), Даниэл Филлипс (Daniel Phillips), Дэвид Миллер (David Miller), Патрик Мочел (Patrick Mochel), Эндрю Мортон (Andrew Morton), Звене Мвейкамбо (Zwane Mwaikambo), Ник Пиггин (Nick Piggin) и Линус Торвальдс (Linus Torvalds). Особое спасибо тайному сообществу ядра (хотя никакого тайного сообщества нет).

Я хочу выразить свою любовь и признательность многим людям. Среди них Пол Амичи (Paul Amichi), Кейт Бэрбег (Keith Barbag), Дейв Эггерс (Dave Eggers), Ричард Эрикссон (Richard Erickson), Нат Фридман (Nat Friedman), Дастин Холл (Dustin Hall), Джойс Хокинс (Joyce Hawkins), Мигуэль де Иказа (Miguel de Icaza), Джимми Крел (Jimmy Krehl), Дорис Лав (Doris Love), Джонатан Лав (Jonathan Love), Патрик ЛеКлер (Patrick LeClair), Линда Лав (Linda Love), Рэнди О'Дауд (Randy O'Dowd), Сальваторэ РибAUDO (Salvatore RibaudO) и его чудесная мама, Крис Ривера (Chris Rivera), Джой Шав (Joey Shaw), Джэрэми Вандорен (Jeremy VanDoren) и его семья, Стив Вейсберг (Steve Weisberg) и Хелен Винснант (Helen Whinsnant).

И в заключение, спасибо за все моим родителям.

Желаю большого хакерского счастья!

*Роберт Лав,
г. Кембридж, штат Массачусетс.*

Об авторе

Роберт Лав (Robert Love) использует операционную систему Linux с первых дней ее существования. Он является страстным активистом сообществ разработчиков ядра и GNOME. Сейчас Роберт работает главным инженером по разработке ядра группы разработчиков Ximian Desktop компании Novell. До этого он работал инженером по разработке ядра компании Monta Vista Software.

Проекты по разработке ядра, которыми занимался автор, включают планировщик выполнения процессов, преемливое (вытесняемое) ядро (preemptive kernel), уровень событий ядра, улучшение поддержки виртуальной памяти (VM), улучшение поддержки многопроцессорного оборудования. Роберт является автором утилит `schedutils` и менеджера томов GNOME. Роберт Лав читает лекции и пишет статьи по основам построения ядра операционной системы и получает приглашения редактировать статьи в издании *Linux Journal*.

Автор получил степень бакалавра по математике и вычислительной технике в университете штата Флорида. Хотя Роберт и родился в южной Флориде, своим домом он считает Кембридж, штат Массачусетс. Роберт увлекается футболом, фотографией и любит готовить.

От издательства

Вы, читатель этой книги, и есть главный ее критик. Мы ценим ваше мнение и хотим знать, что было сделано нами правильно, что можно было сделать лучше и что еще вы хотели бы увидеть изданным нами. Нам интересно услышать и любые другие замечания, которые вам хотелось бы высказать в наш адрес.

Мы ждем ваших комментариев и надеемся на них. Вы можете прислать нам бумажное или электронное письмо либо просто посетить наш Web-сервер и оставить свои замечания там. Одним словом, любым удобным для вас способом дайте нам знать, нравится ли вам эта книга, а также выскажите свое мнение о том, как сделать наши книги более интересными для вас.

Посылая письмо или сообщение, не забудьте указать название книги и ее авторов, а также ваш обратный адрес. Мы внимательно ознакомимся с вашим мнением и обязательно учтем его при отборе и подготовке к изданию последующих книг. Наши координаты:

E-mail: info@williamspublishing.com

WWW: <http://www.williamspublishing.com>

Информация для писем из:

России: 115419, Москва, а/я 783

Украины: 03150, Киев, а/я 152

Для читателей

Более подробную информацию об этой и других книгах издательства Sams Publishing можно получить на Интернет-сайте www.nowellpress.com. Для поиска информации о книгах введите в поисковое поле код ISBN (без соединительных черточек) или название книги.