

Руслан Богатырёв

## Язык как основа архитектуры. Проект "Кронос" и путь к технологиям XDS

Богатырев Р. Язык как основа архитектуры // ComputerWeek-Moscow. 1998. № 20.

В предыдущей статье из серии "Язык как основа архитектуры" ("Искусство программирования", сентябрь 2003) речь шла о проекте Lilith, связанном с созданием новой специализированной платформы — языка Modula-2, ОС Medos и персонального компьютера Lilith. Теперь давайте перенесемся на 15 лет назад, в Новосибирский академгородок. Здесь, в недрах Новосибирского государственного университета (НГУ) и ВЦ СО АН СССР, спустя два года после разработки первой партии Lilith зародился еще один проект, одновременно и похожий и непохожий на своего предшественника.

### Kronos Research Group

Сегодня нередко можно услышать, будто бы в области компьютерных технологий мы навсегда отстали от Запада (и от Востока), причем и в аппаратуре, и в программном обеспечении. Что наш удел ныне — лишь настройка готовых систем под потребности отечественного рынка, в лучшем случае — сборка из импортных блоков больших и малых информационных приложений. Что российские программисты занимаются одним "коленочным" программированием, задача которого — заткнуть бреши и дать надежду на получение работы в приличной западной компании. Но так ли это?

Как правило, дыма без огня не бывает. Подобные утверждения вполне обоснованны, в то же время такая категоричность наносит колоссальный вред развитию той индустрии, которая, по всей видимости, станет доминирующей в грядущем столетии. Причем вред не только в масштабах нашей страны, но, как это ни странно, и всего мира. Жаль, что мы часто опускаем руки, слабо представляя себе наш интеллектуальный потенциал и крайне редко используя его по назначению.

Я далек от мысли обращать читателя в иную веру. Эта серьезная, я бы даже сказал, стратегически важная тема требует не быстрых кавалерийских наскоков, а обстоятельной дискуссии с привлечением самых разных специалистов. Но в любом случае, прежде чем делать далеко идущие выводы, стоит разобраться в уже известных фактах.

В данной статье мы затронем лишь один такой пример — речь пойдет о малоизвестной российской команде из Новосибирского академгородка, со временем превратившейся в три самостоятельные компании: XDS ([www.xds.ru](http://www.xds.ru)), xTech Ltd. ([www.xtech.ru](http://www.xtech.ru)) и ProPro Group ([www.propro.ru](http://www.propro.ru)). Вот уже несколько лет они успешно сотрудничают с западными фирмами, причем как в области прикладного, так и системного программирования. Созданный за 15 лет фундамент оказался столь солидным, что позволил не просто разработать (и не продать!) уникальные технологии, воплотить их в виде целого ряда законченных коммерческих продуктов, но и выполнить немало отнюдь не рядовых проектов, прежде всего в области космической и телекоммуникационной промышленности.

При этом несмотря на неизбежную миграцию кадров (в разное время разные американские фирмы смогли уговорить ряд талантливых новосибирцев покинуть свою альма-матер; сразу несколько человек были приглашены компанией Microsoft), сравнительно небольшой коллектив, не утратив жизнеспособности, ведет интенсивные научные исследования (!) и продолжает расширять сферу своей деятельности.

На примере новосибирской команды Kronos Research Group (KRG) мы с вами проследим этапы развития передовых технологий и узнаем, какой трудный путь подчас предстоит преодолеть, прежде чем они окажутся востребованными компьютерным рынком. KRG примечательна еще и тем, что в середине 80-х годов сумела разработать отечественную 32-разрядную рабочую станцию, собственную UNIX-подобную операционную систему, а также реализовать программную и аппаратную части транспьютерной архитектуры, ставшей одной из ключевых звеньев проекта MAPC — советского ответа на японский вызов, брошенный миру под лозунгом создания компьютеров пятого поколения (Fifth Generation).



представлены: Никлаус Вирт — в центре, И. В. Поттосин — в нижнем ряду справа, а также члены команды KRG.)

Об этих работах в нашей стране было написано крайне мало — можно найти лишь несколько небольших статей и ряд препринтов самих участников проекта MAPC. Здесь нелишне подчеркнуть, что американцы всерьез интересовались результатами тех исследований, но в открытой печати, насколько мне известно, дело ограничилось двумя статьями одного из инициаторов проекта MAPC, Вадима Евгеньевича Котова, написанными им в 1991 г. по просьбе известного журнала Communications of the ACM. (На фото

### Процессор "Кронос-1"

В январе 1998 г. мне довелось встретиться с Алексеем Недорей — одним из той новосибирской четверки, которая стояла у истоков проекта "Кронос". Он поведал о многих не известных мне ранее деталях того проекта. Думаю, настало время познакомить наших читателей с довольно интересной и поучительной историей создания "Кроноса".

Взяв за основу работы швейцарской группы под руководством Никлауса Вирта ("Искусство программирования", сентябрь 2003), новосибирцы пошли дальше — они сделали 32-разрядный процессор, ориентированный на Modula-2, и использовали его впоследствии для реализации базовых компонентов транспьютерной архитектуры. Промежуточным результатом оказалось создание платы "Кронос" для "Электроники-60". Так что в отличие от швейцарцев и распространение, и серийное производство наладить было куда проще. Но обо всем по порядку.



История "Кроноса" началась летом 1983 г., когда в Новосибирске появились материалы по Modula-2 и Lilith. В это время Алексей Недоря закончил третий курс механико-математического факультета НГУ и начал делать диплом — компилятор с языка Edison (придуманного Пер Бринч Хансеном, автором языка Concurrent Pascal). В то же время Дмитрий Кузнецов приступил к созданию компилятора Modula-2 (под руководством И. В. Поттосина). Схожие задачи и общая творческая атмосфера способствовали образованию неформального коллектива: вместе собрались несколько человек, которым было интересно программирование вообще и компиляторы в частности. Новосибирцы даже организовали свой семинар в общежитии, он назывался Intruders Club. Основным заводилой всей клубной деятельности стал Кузнецов.

Где-то весной 1984 г. начала формироваться более узкая группа, куда наряду с Недорей и Кузнецовым вошли Евгений Тарасов и Владимир Васекин. Оба с физфака НГУ, с третьего курса. Евгений уже приступил к своей работе — процессору клеточной логики. Именно в этот момент, как вспоминает Алексей, "мы нахально и решили сделать себе нечто похожее на Lilith: каждому хотелось заполучить домой собственную машину".

Работа была организована так: Кузнецов писал компилятор на Burroughs 6700 и интерпретатор "Кроноса". Недоря реализовывал маленькую ОС под управлением интерпретатора, а Тарасов с

Васекиным делали процессор. Решение было принято самое простое — создать его в виде платы для "Электроники-60" (bit-slice). В июле 1984 г. Недоря написал на языке Паскаль микроассемблер для микропрограммирования "Кроноса" и к тому времени уже защитил диплом (компилятор Edison работал и порождал код для "Эльбруса").

В начале декабря 1984 г. процессор "Кронос-1" был готов. Название "Кронос" появилось совершенно случайно: оно понравилось новосибирцам своим звучанием. "Наше руководство на ВЦ СО АН СССР (прежде всего И. В. Поттосин и А. Г. Марчук) на протяжении всех работ над "Кроносом" оказывало нам большую поддержку", — вспоминает Недоря.

Для организации кросс-разработки по последовательному каналу "Кронос" был подключен к Burroughs, и начался мучительный процесс отладки. На "Кроносе" работала пультовая микропрограмма. На Burroughs (который в шутку называли "Бахус") была написана маленькая связанная утилита, которая позволяла загрузить программу в "Кронос" и запустить ее. Программы для "Кроноса" писались с самого начала на Modula-2.

Поскольку работал "Кронос" кое-как, было принято решение сделать теперь все более грамотно: изменить систему команд, процессор, компилятор и все остальное. Так что "Кронос-1" был повешен на стенку, и началась разработка "Кроноса-2".

### "Кронос" в МАИ

Здесь я ненадолго прерву рассказ Алексея, намеренно нарушу хронологию и расскажу про свои впечатления. Они связаны с практическим использованием как раз того самого "Кроноса-2". В 1984—1989 гг. в Московском авиационном институте небольшая группа из четырех человек, в которой мне довелось работать, занималась созданием ПО нескольких стендов технологического оборудования, предназначенного для научных экспериментов. Помимо меня, в группу входили И. Егоров (ныне ведущий программист в Maddox Games, технический лидер проекта "ИЛ-2 Штурмовик"), А. Люльчев и А. Русов.

Создание систем реального времени (real-time systems) всегда отличалось особой сложностью, прежде всего из-за необходимости соединять несоединимое — эффективность и надежность. Подавляющее большинство работ выполнялось на ассемблере, при этом требовалась не просто реализация специфических драйверов устройств, а увязывание всех программных элементов в единую целостную систему, работающую при минимальном вмешательстве человека.

Времени на раскочку явно не хватало, ведь стендов было много, а нас мало. Ждал непочтатый край работы, так что ни ассемблер, ни Си нас не устраивал. Главная проблема этих языков — надежность. И даже когда рука уже набита, и многое делается на автомате, часто разработка на ассемблере и Си чрезмерно затягивалась или даже заходила в тупик. Дело осложнялось тем, что оборудование (в том числе и оговоренные интерфейсы) постоянно изменялось. Так что разработчики как аппаратного, так и программного обеспечения вынуждены были постоянно заниматься притиркой созданных своими руками компонентов.

В 1985 г., в то самое время, когда команда Kronos Group уже перешла к созданию "Кроноса-2", нам попала на глаза книга Вирта на английском языке, посвященная Modula-2. Это было то, что мы искали, — разбиение программных модулей на интерфейсные и исполнительные части, а также система более строгого программирования с сохранением удобного доступа к низкоуровневым средствам. Появилась возможность наладить нормальное коллективное взаимодействие и вычленив на уровне языка системнозависимые части.

Продолжая работы по обкатанной технологии, мы стали перестраивать свое инструментальное хозяйство на ходу. За пару месяцев удалось разыскать работающий компилятор (это были приемники виртовской системы для PDP-11: M22 и M23 для RT-11). Система понравилась, единственное, что разочаровало, — не очень удачная реализация генератора объектного кода. Даже ради красивой идеи мы не могли позволить себе транжирить системные ресурсы.

Выход был найден очень простой — вместо переделки швейцарского компилятора мы реализовали несложную методику ассемблерных вставок. С помощью трассировки и накопленного опыта выявлялись критические процедуры (по скорости и объему кода), после чего они переписывались вручную на ассемблере. Оставалось только обмануть защиту компоновщика

и написать небольшую программку, которая прямо с листинга в кодах заносила "оптимизированные" процедуры в соответствующее место и корректировала все контрольные суммы, обеспечивающие целостность объектного кода. Все работало как часы.

Язык Modula-2, таким образом, стал своеобразным каркасом, в который в случае необходимости органично вписывались ассемблерные части. Первым делом мы реализовали свой собственный диспетчер процессов, а затем вдохновленные успехом написали свои библиотеки и разработали технологию, ориентированную на встроенные системы (с присущим им параллелизмом). Она максимально использовала достоинства Modula-2. Спустя некоторое время все это вылилось в целую технологию ЛОТОС, построенную на принципах методологии Mascot-3 (разработка Министерства обороны Великобритании), идеях межзадачного взаимодействия языка Ada и на формальном аппарате сетей Петри.

Все бы ничего, даже не особенно страшны были и ограничения в 64 Кбайта для целевой машины (это была "Электроника-60" и другие отечественные разновидности DEC LSI-11). Однако производительность нашего труда хотелось поднять, но как? Вот тут-то мы и узнали о существовании платы "Кронос". Подумать только: обычная плата в конструктиве "Электроники-60", превращающая наш комплекс в 32-разрядную станцию, да еще полностью настроенную на Modula-2. Об этом мы даже и не мечтали. И хотя пришлось тогда изрядно подсуетиться, чтобы раздобыть ее (а затем вдобавок и контроллер RK-дисков), наши усилия были вознаграждены.

Весь комплекс в целом превосходил по удобству и быстродействию инструментария платформу PC XT/AT (о чем тут говорить, если та же XT уступала на наших тестах по обработке и переключению контекстов процессов даже отечественным вариантам LSI-11). Помнится, наш диспетчер для DOS позволял запускать в 640 Кбайт памяти более 1000 асинхронных процессов, часть которых могла выгружаться на диск. Кросс-разработка была делом привычным, так что различие инструментальной ("Кронос", PC) и целевой (LSI-11) платформ до поры до времени нас не смущало. Но настал момент, когда мы всерьез призадумались над использованием "Кроноса-2" уже в качестве и целевой платформы. Смущал лишь дефицит таких плат (достать их было непросто), в то же время при наличии немалого парка "Электроники-60" перспективы подобного решения для наших задач выглядели весьма заманчиво.

Если даже опустить нашу чисто исследовательскую работу и обратиться к практической отдаче, то "Кронос-2" очень сильно помог, в частности, при создании в МАИ вполне конкретной встроенной системы для управления установкой плазменного напыления, точнее говоря, осуществляющей активный контроль за процессами плазменной и магнетронной обработки поверхностей различных материалов. Этот комплекс уникального оборудования, созданный руками инженеров МАИ, позволял вести работы по изучению свойств новых перспективных покрытий для аэрокосмической индустрии. Помимо привычных для систем реального времени функций он обеспечивал и поддержку БД режимов технологических процессов. Через некоторое время комплекс разросся, и мы создали уже специализированную сеть целевых машин.

Я привел лишь один из примеров применения "Кронос-2", а ведь отдача от той разработки была солидной, несмотря на малое количество плат (а потом и отдельных станций "Кронос-2.6"). Хотя, что считать малым — за все время было выпущено не менее 200 плат "Кронос-2". Да и, кроме нас, были другие команды, использовавшие многие положительные стороны "Кроноса".

Как ни жаль, развитие линии PC (в отличие от "Кроноса") пошло такими темпами, что многие прелести отечественного чуда на их фоне постепенно исчезли. То была беда нашей электронной промышленности, проморгавшей или попросту не захотевшей развивать перспективную новинку.

## **Кронос-2**

Настало время вернуться к прерванному рассказу о создании "Кроноса-2". Разработка его началась с системы команд, которую новосибирцам долгое время никак не удавалось друг с другом согласовать. В какой-то момент был найден компромисс, который тут же закрепили в виде отдельной спецификации. К лету 1985 г. процессор "Кронос-2" был готов.

Имея раскрученный "Кронос-1", не было необходимости повторять мучительный путь с Burroughs. Теперь раскрутку новой системы стали вести прямо через "Кронос". На Паскале был завершен

Кронос-ассемблер, причем довольно необычный — он был структурным и поддерживал понятия процедуры и раздельной компиляции.

На этом ассемблере были написаны загрузчик, примитивная файловая система и редактор, а через некоторое время и планировщик процессов. Благодаря ему к "Кроносу" подключили второй терминал, и заработала двухпользовательская операционная система. Название слишком громкое, но тем не менее на двух терминалах одновременно можно было писать программы на ассемблере и компилировать их.

"Теперь для меня все это выглядит как-то дико, — вспоминает Недоря. — Сейчас я работаю за слабенькой по нынешним меркам машиной (Pentium-133/32Мбайт/2Гбайт). Тогда же мы работали вдвоем на машине с 96 Кбайт оперативной памяти, а единственной внешней памятью были флопи-диски (аж 480 Кбайт)".

На ассемблере программировать дальше не хотелось, теперь требовалось написать компилятор. Чтобы избежать многих сложностей, для его предварительной версии язык Modula-2 был несколько упрощен и сведен к подмножеству (Modula-0). При этом модули нового диалекта должны были корректно транслироваться будущим компилятором с Modula-2.

Новосибирцы подготовили проект, распределили работу и втроем (Недорья, Кузнецов и Тарасов) написали компилятор за две недели. Его размер составлял 3 тыс. строк на ассемблере и состоял примерно из 30 модулей. Через две недели после начала разработки (где-то в середине сентября) ассемблер уже был почти не нужен (за исключением улучшения компилятора) и удалось приступить к созданию нормальной операционной системы, которая получила название Excelsior.

Написав вполне добротную ОС, новосибирцы реализовали нормальный компилятор Modula-2 (m2). Он был готов к лету 1986-го. К тому времени новосибирская четверка вошла уже в штат ВЦ СО АН СССР. Правда, ушел Васекин, его в качестве основного разработчика аппаратуры заменил Владимир Филиппов (теперь он директор xTech Ltd.). С 1985 г. начало приходить пополнение, в основном из студентов. К 1988 г. в Kronos Research Group насчитывалось уже около 20 человек (см. [www.xds.ru/krg.html](http://www.xds.ru/krg.html)), а на "Кроносе" работали вполне приличная ОС Excelsior iV и очень быстрый и компактный компилятор mx. "Не знаю, как остальные, — подчеркивает Недоря, — но я до сих пор считаю тот компилятор для "Кроноса" одной из лучших своих разработок".

### **Рабочие станции "Кронос"**

История "Кроноса" на этом не закончилась — проект достиг той стадии, когда уже стоило всерьез подумать о промышленном внедрении.

К 1987 г. был налажен выпуск модификаций "Кронос-2.5" (две платы, которые вставлялись в Labtam, шины Multibus-I/II) и "Кронос-2.6" (отдельная рабочая станция на микропроцессорных секциях серии 1802, шины Q-bus, Q-22, И-41, Multibus-I/II, 8 Мбайт ОЗУ, винчестер 40 Мбайт, растровый дисплей 480 x 360 точек, 16 цветов). При одинаковой частоте (3 МГц) производительность "Кроноса-2.5" составляла 1 млн команд/с, "Кроноса-2.6" — 1,5 млн команд/с.

Выпуском занялись СКБ Института кибернетики АН Эстонии, радиозавод в Ижевске, завод в Ленинграде, несколько предприятий в Новосибирске, Бердске, Кишиневе, Перми. Модернизировались также финские 16-разрядные компьютеры серии ELORG-DATA. Киевское НПО "Микропроцессор" приступило к разработке топологии и подготовке производства процессора "Кронос" в однокристальном варианте. Создавался и 64-разрядный арифметический сопроцессор. Были изготовлены лабораторные образцы кристалла, однако из-за низкого качества оборудования процент брака был слишком велик, и производство так и не было налажено.

Что касается программного обеспечения, то помимо Modula-2 были реализованы компиляторы для языков Fortran-77 и Си (а также менее известных НУТ, БАРС, ПОЛЯР), разные графические пакеты (например, bCAD), САПР печатных плат и многое другое. Причем это ПО создавалось большей частью не для "демонстрации мускулов", а для решения неотложных задач. В частности, САПР печатных плат, заработавшая на "Кроносе-2.5", позволила заметно ускорить разработку "Кроносе-2.6". А идеи пакета bCAD оказались столь удачными, что уже упоминавшаяся новосибирская фирма ProPro Group теперь создала его новое поколение (с поддержкой OpenGL,

механизмов морфинга и сетевого фотореалистичного тонирования, с использованием анимированных текстур и органичного моделирования, основанного на B-сплайнах и NURBS-технологии). Кстати говоря, со временем в Kronos Research Group был реализован и программный эмулятор "Кроноса" для платформы Intel x86.

Предметом особой гордости стала ОС Excelsior. При ее проектировании у разработчиков возникло сильно желание упростить обучение пользователей, а потому за прототип была взята популярная тогда ОС UNIX. Да и окончательная версия, полностью написанная на Modula-2, получила имя Excelsior iV, что означало "мнимая UNIX System V". Система была многопользовательская, многозадачная, построенная на принципе динамической загрузки (каждый модуль — аналог нынешних DLL-библиотек). Архитектуру ее разработали Дмитрий Кузнецов (файловая система) и Алексей Недоря (ядро). В общей сложности ОС Excelsior создавали около 15 человек.

## Проект MAPC

Имея за плечами такой задел, можно было ставить перед собой куда более масштабные задачи. Настало время рассказать о проекте MAPC и о работе временного научно-технического коллектива (ВНТК) "Старт".

Исследовательский проект MAPC (Модульные Асинхронные Развиваемые Системы) являлся одним из ответных шагов Советского Союза на амбициозный японский проект ЭВМ пятого поколения (как известно, соответствующие шаги тогда предприняли США и Великобритания). Он проводился в три этапа. Вначале в ВЦ СО АН СССР под руководством В. Е. Котова и А. Г. Марчука были сформулированы общие положения концепции MAPC. На втором этапе (1985 - 1988 гг.) был образован ВНТК "Старт", объединивший научные и инженерные группы из ВЦ АН СССР, СКБ вычислительной техники СО АН СССР, ИК АН ЭССР, НПО "Импульс" и др. Особенно здесь выделялись два коллектива — новосибирский, формирующий базовый (системный) уровень MAPC, и эстонский, отвечавший за интеллектуализацию (средства представления знаний). Эстонцы реализовали на "Кроносе" систему объектно-ориентированного программирования, обеспечивающую в рамках языка НУТ работу с классами, объектами и производными правилами. Язык НУТ разрабатывался с прицелом на решение задач искусственного интеллекта. На третьем этапе проекта MAPC (1988—1990 гг.) планировалось построить экспериментальный образец системы на новой элементной базе.

С точки зрения аппаратуры фактически были иницированы работы, которые велись параллельно с развитием линии "Эльбрус", но уже не на базе суперкомпьютеров, а на уровне кластеризации процессоров среднего класса. В основу было положено мультипроцессорное ядро на базе процессоров "Кронос" с использованием транспьютерной организации. При этом транспьютерные узлы, обладая собственной локальной памятью, соединялись между собой асинхронными каналами и на основе системной шины комплектовались в транспьютерные модули. Это позволяло достигать гибкой масштабируемости и смещать акцент на аппаратную поддержку распределенных потоковых вычислений.

В отличие от английской концепции транспьютера, разработанной в компании Inmos, в случае MAPC вычислительный элемент имел не четыре, а произвольное число соседей. Основным строительным блоком служил процессор "Кронос", точнее говоря, MAPC-T.

Цель проекта MAPC-T сводилась к тому, чтобы построить сеть процессоров, связанных быстрыми каналами. Память в системе была только локальная. Скорость передачи по каналам довольно высокая — порядка 2 Мслов, что примерно в два раза хуже, чем у транспьютера Inmos T800. Толчок идеям дал язык Оссат, но в случае MAPC все было сделано по-другому.

Макет MAPC-T ("Т" — от "транспьютер") состоял из четырех процессоров "Кронос-2.6", связанных каналами. В процессоры были добавлены дополнительные команды для приема/передачи данных из каналов. На одну машину (головную) загружалась ОС Excelsior iV с модифицированным ядром, на все остальные — небольшое специализированное ядро.

Подсистема ввода/вывода работала только на головном процессоре. Все остальные могли лишь передавать данные по каналам. Программы передавались тоже по каналам. В ядро входила система поддержки виртуальных каналов. Любую программу можно было запустить на одном

процессоре (при этом использовались виртуальные каналы) или на нескольких. Правда, система автоматического распределения нагрузки так и не была доведена до конца.

Для демонстрации возможностей MAPC-T команда Kronos Group написала несколько тестовых примеров. Один пример был графический. На экране вращалась надпись, причем один или несколько процессоров ее обсчитывали, а головной отрисовывал. Как вспоминает Алексей Недоря, "еще была какая-то игра, которая очень удачно распараллеливалась на любое число процессоров и показывала линейный рост производительности — обмен был довольно маленький, а счета достаточно много".

К сожалению, приемная комиссия лишь благосклонно посмотрела демонстрации возможностей архитектуры, на том все и закончилось.

## Технологии XDS

После завершения проекта MAPC группа специалистов Института систем информатики СО РАН, которую возглавлял И. В. Поттосин и куда входили некоторые члены Kronos Research Group, занялась реализацией другого проекта — СОКРАТ. Он предусматривал создание средств кросс-разработки (в частности, для языка Modula-2), которые были предназначены для бортового ПО российских спутников. В течение 1988—1991 гг. в рамках этого проекта закладывались основы новой технологии генерации объектного кода, которая была использована для создания в 1991—1994 гг. целой серии компиляторов для языков Modula-2 и Oberon-2. Тогда же из членов Kronos Group была образована первая компания — xTech Ltd.

В 1994 г. была спроектирована инструментальная многоплатформная среда кросс-разработки, получившая название XDS (eXtensible Development System) и давшая впоследствии имя новой фирме. Характерным отличием ее было наличие переносимой исполняющей системы (RTS), реализующей для Oberon-2 эффективный механизм метапрограммирования (трактовка данных как кода) и сборки мусора. Участие новосибирцев в работе международных комитетов по стандартизации Modula-2 (ISO) и Oberon-2 оказало большую помощь в получении качественных реализаций соответствующих компиляторов. Компиляторы серии XDS стали главным тиражируемым продуктом компании.

Но о том, как они используются в настоящее время у нас в стране и за рубежом, а также о воздействии проекта "Кронос" на современные технологии кросс-разработки речь пойдет уже в следующей статье.

---

**Об авторе.** Руслан Богатырев — научный редактор журнала "Мир ПК", гл. редактор приложения "Мир ПК — диск", bogatyrev@pcworld.ru.