



**10 языков программирования,
которые могут перевернуть мир IT**

**11 вещей, которые
нужно знать
исполнительным
директорам стартапов**

**MoSync SDK:
мобильная разработка
на C/C++**

**История успеха:
EffectiveSoft**

Репортаж с ФИТУ БГУИР

ФИТУ БГУИР: хорошие связи



Беседовала Инна РЫКУНИНА

Помните, мы рассказывали вам о Факультете компьютерных сетей и систем БГУИР? А теперь несколько слов о его “главном конкуренте” скажет декан ФИТУ, Леонид Юрьевич Шилин. А начался наш разговор именно с ФКСиС.

— Правда ли, что ваши факультеты неустанно соперничают?

— А Вы знаете, что раньше мы были одним факультетом, а потом разделились? Что касается соперничества, то, на мой взгляд, сопер-

— Это точно. Они у нас не только программисты, но также хорошо разбираются в электронике.

Мне недавно звонили из Германии и спрашивали, могли бы мы дать им специалистов. Мы говорим: “Нет, не можем. Мы готовим для своей страны!”. Тогда мы перешли к обсуждению совместного обучения: пусть они приезжают к нам, а мы к ним. Мы дружим с разными университетами, в том числе и с западными. Это выгодно отличает наш факультет. Мы ездим к ним в гости. Не так давно ездили в Германию. Смотрели производство Volkswagen, также нам показывали студенческие аудитории, лекции, мы общались с профессорами. У них практически нет платного образования. Но Германия знает, куда вкладывает деньги. Самое приятное даже не то, что наши студенты там увидели, а то, что они вернулись оттуда с внутренней мыслью: “Да мы же не хуже”.

— А как вы смогли наладить

такие хорошие связи с разными странами?

— Иногда они нам звонят. Иногда

и получают деньги.

— Куда идут ваши выпускники после окончания вуза?

Факультет информационных технологий и управления БГУИР был открыт в 1964 году, только тогда он именовался факультетом автоматизированной вычислительной техники.

Сейчас на факультете можно выучиться по следующим специальностям:

- “Информационные технологии и управление в технических системах”;
- “Автоматизированные системы обработки информации”;
- “Искусственный интеллект”;
- “Промышленная электроника”.

да мы звоним. Бывает, что встречаемся на конференциях. Мы проводим две международные конференции на базе факультета. Бывает, после конференций нам говорят: “Мы думали, что у вас умерла электроника, а у вас, оказывается, всё так здорово, давайте вместе что-то делать”.

Недавно приезжали гости из Прибалтики, у них ведь, в самом деле, в стране уровень образования существенно упал. Они и средней, и высшей школой крайне недовольны. Поэтому и сделали нашим преподавателям предложение: “Было бы хорошо, если бы вы приезжали к нам и за хорошие деньги читали лекции”. Я не против. Пусть они в свободное время заключают договора, едут, читают

— Больше всего идут в EPAM Systems. Чуть меньше идут в Itransition, Artox, а также в “Атлант-М”.

Как вы понимаете, наши студенты не крутят там гайки. Мы готовим людей, которые работают в белых рубашках и галстуках.

На факультете существует пять совместных лабораторий с EPAM, а также лаборатории с Itransition, HTC, IVA, Omron и другими компаниями. Я считаю, что в этом плане среди факультетов БГУИР мы лучше всех укомплектованы.

— Кто проводит занятия в этих лабораториях?

— Наши преподаватели, которые обучаются в этих компаниях. Сначала сотрудники компаний сами пытались про-



Леонид Юрьевич Шилин

ничество молодых людей — это своеобразное их совершенствование.

— Я слышала, что ФИТУ готовит более универсальных специалистов.



ФИТУ БГУИР: хорошие связи

↑ дить занятия, но посчитали, что наши преподаватели лучше справляются с этой задачей: они легко вникают в предмет и умеют подать информацию. Оказывается, дано это далеко не всем.

— Кстати, большая у вас те-

в крупную компанию, получает там, как минимум, пять миллионов. В этом явлении есть и положительный момент. В преподавателях остаются те, кто действительно чувствует от этого удовольствие. Конечно, часто они устают от маленькой зарплаты и уходят.



кучка кадров в преподавательском составе?

— Что есть, то есть. Это легко объясняется. У нас преподаватель получает небольшую зарплату. А ведь он, как правило, в прошлом студент-отличник, который потом ещё и закончил аспирантуру. А выпускник, который, может, не так хорошо учился, но пошёл работать

Но самые крепкие защищают диссертации. Кроме того, по вечерам преподаватели могут вести курсы, о которых я говорил, там несколько иная зарплата.

— **Много ли за год вылетает студентов?**

— Отчисляем много, но чаще всего они куда не деваются. Они переходят на дистанционное обу-

чение, или одумываются и восстанавливаются на следующий год. Многие начинают учиться, а многие опять вылетают.

— **Насколько большой популярностью в университете пользуется дистанционная форма обучения?**

— Часто на дистанционно форме обучаются люди, у которых уже есть высшее образование. Они дисциплинированы и могут учиться самостоятельно. А вот тому, кто пришёл сразу после школы — очень тяжело так учиться.

— **У вас часто отчисляют за прогулы?**

— Да. В кодексе образования указано количество часов, после которого мы обязаны отчислить. За студентами, которые “висят на волоске” мы следим особенно.

— **Отчисленные за прогулы студенты могут восстановиться в этот же год?**

— Могут, но только на курс младше. Ведь мы отчислили их за то, что они не ходили на занятия, а значит, не усвоили материал.

— **Насколько тяжело учиться на ФИТУ?**

— Если большая часть университетов существенно снижают требования к студентам, то наши преподаватели сопытом крайне строгие. Но студентам, которые одержи-

мы учёбой, не тяжело, им всё крайне интересно и в удовольствие.

Чем хороши наши студенты, так это тем, что зачастую они везде успевают. Лет пять тому назад у нас была группа представителей из Philips. Они сказали примерно так: “Мы смотрели ваше оборудование. Ну, приличное оборудование, кое-где можно было бы и лучше, — тогда у нас действительно была ещё не очень хорошая материальная

но уже такой интерес пропал”.

— **Считается, что из-за довольно высокой стоимости обучения студентам БГУИР приходится рано идти работать. Так ли это?**

— Да, мы сами поощряем работу студентов, начиная с третьего курса. Мы считаем, что если у студентов средний балл выше семи, то работа пойдёт только на пользу их образованию. Иногда они могут даже принести что-то



база. — Но вот чем вы нас удивили: у вас на занятиях студенты хотят знать. Они пристают к преподавателям с вопросами. У нас дав-

новое для других студентов, рассказав, к примеру, как решают конкретную проблему на их предприятии. ↓

ФИТУ БГУИР: хорошие связи

↑ — Нужны ли студенту научные публикации?

— Наверное, есть шансы поступить в магистратуру и без них. Думаю, многие поступают без работ. Но мне нравятся студенты, которые уже со второго-третьего курса начинают участвовать в науке. Сейчас у меня четыре аспиранта, два магистранта и два дипломника, а также студенты, с которыми я занимаюсь.

— В какой форме чаще всего

задачи. Спустя час все свои работы студенты оставляют на партах и выходят в коридор. Затем заходят по одному, берут свои работы, и мы с ними разговариваем.

— А как Вы относитесь к экзаменам в форме тестирования?

— Такие экзамены мне не очень нравятся. Я стараюсь, чтобы такая форма оценки знаний не сильно распространялась. Почему? Личное общение позволяет избежать

решать, я могу подсказать ему метод. Его задача — реализовать этот метод.

— Сколько лет Вы работаете деканом факультета?

— Третий год.

— Тяжело?

— Откровенно говоря, я всё-таки больше научный работник, чем администратор. И мне многое мешает. Но по жизни я человек, желающий нового, мне нравятся студенты, наше образование, университет, факультет.

— А часто к Вам лично обращаются студенты по каким-то вопросам? Или эту работу берут на себя Ваши заместители?

— Да, многие вопросы студенты решают с моими заместителями, но если они сами не могут решить какой-то вопрос — они идут ко мне.

— А какие это могут быть вопросы?

— Например, студент хочет устроиться на работу, но у него чуть-чуть не хватает до семи баллов. Заместитель объясняет ему, что по закону положено семь. Тогда студент идёт ко мне и просит сделать исключение. И мы с ним начинаем думать. Мы смотрим конкретно, по каким дисциплинам, какие оценки у него стоят. Берём во внимание, участвует ли студент

ещё в чём-то, кроме не посредственного учебного процесса. Я могу понять, что студент не успевает, потому что занят в спортивной сфере или в самодеятельности. Эти вещи дают плюс факультета.

руководство вуза не против. Но тот, кто берёт студента неофициально, он и платить будет меньше, а, возможно, и ущемлять во многом.

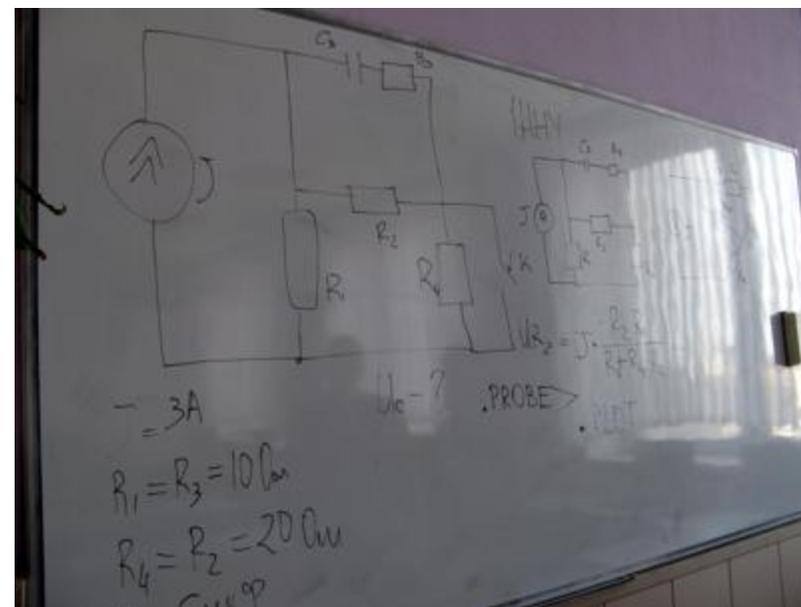
— Когда Вам хочется пререк-



проводятся экзамены?

— Всё зависит от преподавателя. У меня на экзамене первая половина письменная: студент приходит, вытягивает билет и решает

недопонимания. Студент всегда может что-то уточнить. Я ведь стараюсь не быть оппонентом, а, наоборот, быть помощником. Если студенту непонятно, как что-то



тету, поэтому можно пойти на встречу. Мы делаем так: смотрим, как он учится ближайшие полгода. Если успеваемость улучшается, то делаем поощрение. Но если студент бездельник и много пропускает, увы, ничего не могу сделать.

Беда в том, что многие фирмы берут студентов без всякого разрешения. Это нарушение. Фирмы обязаны требовать у работника справку с места учёбы о том, что

ламирывать свой факультет, что Вы, в первую очередь, говорите о нём?

— Я говорю, что мы даём не только замечательное образование, мы готовим успешных людей. В самом деле, очень много людей, состоявшихся после окончания факультета. Это и депутаты, и дипломаты, и академики, и наш ректор, и наш проректор. Многие выпускники открывают



ФИТУ БГУИР: хорошие связи

↑ свой бизнес. Трудно сказать, где нет наших студентов.

— А какие виды спорта вы поддерживаете на факультете?

— Футбол, например. Мы не-

давно организовали свою факультетскую лигу.

— А преподаватели в чём-нибудь участвуют?

— Конечно, хоть мы и не на-

столько успешны, как студенты.

— Тоже футбол?

— Нет. Мы отлично играем в шахматы и шашки. И тоже занимаем призовые места. Кроме того, участвуем в соревнованиях по настольному теннису и волейболу. Чуть хуже у нас с дартсом и бильярдом.

— Как Вы считаете, нужно ли человеку технической специальности второе высшее образование?

— Я считаю, что если у человека есть хорошее первое образование, то нужно идти в магистратуру, в аспирантуру, становиться кандидатом наук, доктором, академиком. Вот когда ты станешь академиком — получишь второе высшее и проходи все пункты заново. А так... Никто ведь не хвалится тем, что у него три средних образования: “Представляете, он три средних школы закончил!”. А вот три института это почему-то хорошо. Не кажется ли Вам, что в таком случае, вероятнее всего, не всё хорошо с первым образованием? Может, человек неправильно выбрал профессию, или он не очень хороший специалист.

— Составляет ли БГУ вам конкуренцию?

— Конечно, составляет. Другое дело, что в конкретной нашей об-

ласти, наверное, мы всё-таки самые сильные.

— А что Вы можете сказать о коммерческих вузах?

— Ну, Вы же прекрасно видите: в мировых соревнованиях по про-

— Смогут ли когда-нибудь догнать до вашего уровня коммерческие вузы?

— Наверное. Но обеспечить коммерческому вузу свои лаборатории такой аппаратурой, как у



граммированию от Беларуси участвует БГУИР и БГУ. Больше никого нет. БГУ в этом плане проще: они в качестве абитуриентов имели победителей школьных олимпиад. А мы просто хорошо подготовили обычных студентов. Среди участников соревнований по программированию преимущественно два факультета: ФКСиС и ФИТУ. Но всех их готовили наши преподаватели.

нас, — нереально, либо нужно, чтобы владельцем вуза был миллиардер, который не побоится вложить в это дело огромные деньги. Кроме того, достаточно сравнить состав наших преподавателей: сколько докторов наук, сколько доцентов. Сразу понятно, что мы университеты разных уровней.

[Обсудить](#)

Большой кофейный конкурс



Призы ждут тебя



История успеха: EffectiveSoft

Беседовала Инна РЫКУНИНА

Очередным героем нашей рубрики “История успеха” стал Александр Курковский и его детище — компания EffectiveSoft.

Моё общение с Александром Ивановичем началось с чашки кофе, шоколадки и вопроса о том, как возникла компания.

— Более пяти лет я работал в США в софтверной компании. Затем вернулся в Беларусь. К этому времени в моём багаже уже был определённый опыт и несколько бизнес-контактов, которые я приобрел, работая там. У моих партнёров уже имелись идеи, которые они хотели реализовать с моим участием. Фактически, EffectiveSoft стартовал не с нуля, не на голом месте, а уже имея портфель заказов на конкретные продукты. Заказы оказались большими, ёмкими, а клиенты — долгосрочными. Например, первый проект у нас длился около пяти лет. В компании тогда работало 10-15 человек. Кстати говоря, мы до сих пор работаем с некоторыми из этих старых клиентов.

— В чём была суть первого проекта?

— Он был ориентирован на инженеров, работающих на промышленных предприятиях. Основная

идея заключалась в том, чтобы помочь инженерам и конструкторам решать практические технические проблемы, с которыми они сталкиваются в своей повседневной деятельности, например: как уменьшить энергопотребление устройства, как уменьшить его вес, не жертвуя прочностью.

Программный продукт, который мы создали, содержал огромную мультимедийную базу патентов США, на сколько я знаю, самую большую в мире и в своем роде уникальную. В процессе работы над этим продуктом мы также привлекали белорусских учёных и инженеров. В результате для нашего клиента было разработано несколько версий этого программного продукта. Он оказался довольно-таки сложным, поэтому впоследствии мы его дополнили обучающим компьютерным курсом для инженеров.

— С какого года вы в этом

офисе (компания располагается в Минске на углу площади Я. Коласа — ред.)?

— Шестой год. Проблема роста компаний, как известно, похожа на проблему роста детей. Если ребёнок быстро растёт, то даже новые вещи становятся ненужными, ребёнок “вырастает из штанишек”. Так случилось и у нас. Компания росла очень быстро и “штанишки стали трещать по швам”. Был ещё один минус помещения, которое мы снимали до этого: оно было фрагментировано. Сотрудники, хотя и сидели на одном этаже, но работали в изолированных комнатах, рядом с офисами других компаний. Тесное общение команды разработчиков было затруднено. Мы, конечно, провели сеть и сделали всё возможное для упрощения процесса коммуникации. Но хождение по коридорам из комнаты в комнату усложняло жизнь. Кроме того, возникла ещё одна проблема, характерная для растущих ИТ-компаний — мы потребляли много электроэнергии. Кроме персональных компьютеров, для разработчиков для поддержания процесса разработки необходимы сервера, резервные источники питания и другое сетевое оборудование, необходимо их

кондиционирование. Как выяснилось, нам нужны сотни киловатт электроэнергии. Мы начали перегружать местную систему энергоснабжения.

— Новое помещение Вам нравится?

— Здесь есть большой плюс — единое помещение в центре города. Кроме комнат, где работают сотрудники, имеется несколько конференц-комнат для совещаний, уютная кухня-бар со спутниковым телевидением, места для отдыха и оздоровления сотрудников. Люди могут свободно общаться как между собой, так и с приезжающими к нам клиентами. У нас около 3000 квадратных метров. Если у нас в будущем будет два офиса, то хотелось бы второй ещё большего размера. Я считаю, что людям нужно работать в комфортных условиях.

— А были ли ещё, кроме необходимости сменить место расположения компании, какие-то “подводные камни” по мере развития?

— Рост компании вызвал естественные изменения её структуры управления и организацию новых процессов, а это, конечно же, связано с дополнительными затратами. Если, например, компания со-

стоит из шести человек, то команда разработчиков может сидеть в одной комнате и вести обсуждение возникающих рабочих вопросов, просто повернувшись друг к другу. По мере роста компании происходят структурные изменения, возникают новые подразделения и связи между ними. Известно, что менеджер может эффективно управлять пятью-шестью людьми одновременно — контролировать исполнение заданий и обсуждать с командой запланированные работы. А вот чтобы эффективно управлять командой из 20 человек, одному менеджеру уже не хватит и целого рабочего дня. При этом пропадает связь с некоторыми членами команды, они как бы выпадают из поля зрения менеджера. Поэтому нам пришлось дополнительно готовить и руководителей среднего звена. Рост компании также ведёт к расширению спектра ИТ-услуг для клиентов, что требует особого внимания в программе подготовки и обучения разработчиков. Но самая большая проблема не только нашей компании, но всего ИТ-сектора в республике — это нехватка квалифицированных ресурсов. Причина проста — маленькая страна, это следует признать. Стра-

История успеха: EffectiveSoft

на талантливая, страна трудолюбивых и умных людей, которые многое умеют и гордятся своими результатами. Но... маленькая. Поскольку ИТ-бизнес становится популярным, ИТ-компании в республике в какой-то степени стали конкурировать и за людские ресурсы. Компаниям приходится работать спотенциальными сотрудниками ещё с их студенческой скамьи. Классический подход в бизнесе построен так: сначала нужно найти клиентов, а потом под проекты этих клиентов надо подобрать сплоченный, грамотный коллектив. Бывает, что приходит серьёзный клиент, а разработчиков для его проекта не хватает — они все заняты. Начинается поиск специалистов. А это — процесс небыстрый. Получается, что проблемы поиска проектов и процесс подготовки кадров надо решать параллельно.

— **А чем занимается команда, которая в данный момент не занята на каком-либо проекте?**

— Сначала скажу, что перед началом работы над проектом всегда есть предпродажная или подготовительная стадия. Это стадия предварительных переговоров с клиентом, в течение которой производится ознакомление потенциального клиента с

компанией, с командой разработчиков, а команды — с деталями проекта. Оговариваются техни-

ческие детали проекта, выстраивается планирование. Не бывает так, что клиент приходит — и за-

тем же мы подписываем с ним контракт. Так редко вообще бывает в любом бизнесе. Представь-

те, что вы заказали кухню. Процесс её непосредственной установки может занять неделю, но перед этим вы должны обсудить с изготовителем, что конкретно вы хотите видеть, что может предложить дизайнер. И хорошо, если в вашей голове есть представление, как это должно выглядеть. Чаще всего уточняются конфигурация, материалы, цвет... Это — общепринятый подход и, конечно же, что-то похожее делаем и мы.

Кроме того, как правило, не занятые в данный момент на конкретных проектах специалисты осваивают новые технологии и поднимают свою квалификацию. Мы поощряем сертификацию наших сотрудников. Ведь может оказаться, что завтра новые знания пригодятся на новом проекте. Сотрудники также работают и на внутренних проектах, необходимых для поддержания процессов разработки в компании, например, проектах управления продажами, управления ресурсами компании, а также проектами рекламного характера. Занимаясь внутренними разработками и компании, сотрудники, естественно, повышают свою квалификацию. Иногда мы приглашаем молодых людей, которые находятся на первых ступенях своей профессиональной



История успеха: EffectiveSoft

↑ карьеры. Их не сразу можно поставить на коммерческий проект, поскольку ожидание клиента намного выше, чем то, что они знают и умеют. Это как на «Фабрике звёзд»: перед тем, как выйти на сцену, ребята много учатся и репетируют.

— **А кто придумывает и подаёт идею для внутренних проектов?**

— Во-первых, жизнь, во-вторых, отдел маркетинга. Он эти идеи тоже берет из жизни. Мы иногда делаем бесплатные игры или приложения просто для того, чтобы заявить о себе. Впоследствии, когда идут разговоры с клиентами, в качестве примеров готовых проектов мы показываем эти разработки. Это своего рода образцы, которые можно посмотреть, подержать в руках. Опять обращаюсь к примеру. Вы пошли к портнихе, чтобы заказать свадебное платье. Скорее всего, она покажет вам фотографии невест в её платьях. Да, это — не именно то, что вы хотите, но, при этом, вы понимаете, что вам предлагают качественный сервис.

— **Как в Вашем понимании выглядит идеальный сотрудник? Какого человека Вы возьмёте на работу?**

— Требования, на самом деле,

стандартные. Самое главное, чтобы он был равнодушным, чтобы у него был искренний интерес к своему делу и стимул самообучаться и расти профессионально. В этом плане такой сотрудник очень интересен компании, поскольку его отношение к работе положительно сказывается на отношении с клиентом. Среди общечеловеческих качеств могу выделить коммуникабельность, доброжелательность в общении с коллегами, умение ужиться и работать в команде, ведь работа над проектами — это коллективный труд, выручка и взаимопомощь. Это здесь не просто слова.

Авообще, люди работают у нас разные. Очень многие из них — люди знающие и с серьёзными профессиональными амбициями. Это нужно учитывать при составлении команд. Например, два лидера могут не ужиться вместе на одном проекте. Поэтому мы просто предлагаем им разные проекты. А есть сотрудники из категории откровенных интровертов. Они хорошо чувствуют себя в любом коллективе, потому что больше общаются с компьютером, чем с людьми. Но, как правило, это умные и интересные люди. И я не могу сказать, кто из них лучше. Они просто разные. Если предла-

гать сотруднику проект с учётом его личных особенностей, то он проявит свои лучшие качества. Один, например, очень быстро решает все задачи, другой медленней, зато — основательно. Если этого не учитывать, то они не будут иметь возможности комфортно работать в команде. Спросите у десяти мужчин, какой должна быть идеальная жена. Их требования будут сильно различаться, но каждый живёт со своей и считает её самой лучшей. Просто совпали характеры и им комфортно вместе. А идеальных-то не бывает. С подбором команды на проект — тоже самое.

— **Как менялась с течением времени стратегия компании?**

— Сначала расскажу о том, что у нас НЕ поменялось. Не менялась установка на то, чтобы максимально удовлетворить клиента. Довольного результатом клиента легче удержать, чем найти нового. Мы любим наших клиентов, они наша ценность и залог процветания компании.

Перед тем, как новый клиент подпишет контракт, должен вырасти маленький цветок доверия, которого ещё нет на первых порах. Мы стараемся «удобрить грунт», показать, что мы достойны доверия, предоставить рекомендации

других клиентов. Как только этот цветок доверия вырос, как только обе компании понимают ожидания друг друга, всё упрощается и становится менее формальным. Клиенты уже воспринимают нас как часть своей команды. Некоторые заказчики даже приглашают нас в гости. И мы не остаемся в долгу.

Клиент всегда хочет быть уверенным в том, что, во-первых, его проект будет сделан, во-вторых, что проект будет сделан вовремя, а в-третьих, что он будет сделан в рамках определённого бюджета. У заказчика не должно возникнуть даже мысли, чтобы искать другого партнёра.

Теперь о том, что менялось. По мере роста компании, по мере роста количества проектов, которые мы выполняли, у нас расширялся опыт в разных областях. Постепенно вместо предложения на рынке общих аутсорсинговых услуг, в которых конкурировать последнее время становится всё труднее, мы выходим на нужные сектора рынка и предлагаем специфические услуги. Мы понимаем, что клиенты хотят найти не просто программистов, а людей, которые знакомы с характером и особенностями бизнеса, в котором они работают. Сейчас наши сотрудники готовы разговаривать на том же

бизнес-языке, на котором разговаривают клиенты. Это уже внутри компании мы разговариваем на языках программирования, а с клиентом мы говорим на языке бизнеса.

— **Прогнозируете ли Вы в ближайшем будущем какие-то проблемы рынка?**

— Конечно. Проблемы западного рынка встают в полный рост, например, проблема европейского кризиса, который нарастает в последнее время. А там находится существенная часть наших клиентов. И как аутсорсинговую компанию, нас это не может не волновать. Финансовая стабильность наших клиентов — это наша финансовая стабильность. Но, с другой стороны, напрямую на отрасль аутсорсингового программирования эти кризисные явления могут и не повлиять. У западных компаний в такой период нарастает потребность сэкономить. Для некоторых из них одной из возможностей для сокращения затрат станет перевод части работ в более дешёвые страны, в том числе и в нашу. Получается, что интерес на Западе к нашим компаниям из-за кризиса может только усилиться.

Популярность Беларуси как аутсорсинговой страны растёт. Во-первых, мы геогра-

История успеха: EffectiveSoft

↑ физически близко к Европе. Клиентов привлекает минимальная разница во времени. Например, с Индией им в этом плане гораздо сложнее. Во-вторых, менталитет белорусских разработчиков очень близок к западному. Они, как правило, выполняют то, что обещают. Уровень образования наших специалистов достаточно хорош для того, чтобы общаться на английском языке. Клиентам нравится, что можно обойтись без переводчика. Они общаются с нашими сотрудниками, как на профессиональные, так и на непрофессиональные темы. Беларусь традиционно славится хорошим образованием, несмотря на существующие сегодня проблемы. Большинство выпускников ВУЗов, работающих у нас, не являются исключительно программистами. Практически у всех из них есть ещё и дополнительный бэкграунд: инженерный, научный, экономический. В той же Индии это иногда бывает проблемой, потому что там люди оканчивают программистские курсы, и, к примеру, какие-то экономические или инженерные проблемы им обсуждать тяжелее.

Есть ещё одна проблема — низкая рождаемость в Беларуси. Мы по мере сил пытаемся увеличить

благополучие наших сотрудников, дабы материальный вопрос не стал препятствием для появления нового ребёнка в семье. Не секрет, что уровень зарплат в ИТ-области существенно превышает средний уровень зарплат по Республике.

— Много в EffectiveSoft многодетных сотрудников?

— Есть сотрудник, у которого трое детей. У многих — по двое. Для большинства наших сотрудников вопрос о большом количестве детей ещё ставить рано. У нас довольно молодой коллектив. Средний возраст около 26 лет. Некоторые ещё технически не успели обзавестись ребёнком.

— Есть ли у Вас подготовленные “спасательные круги” на случай кризиса? Может быть, Вы рассматриваете возможность заняться какой-то другой деятельностью?

— Честно говоря, я не представляю себя в другом бизнесе. Делать один бизнес с оглядкой на другой, такое, наверное, возможно, если не любишь дело, которым ты занимаешься, или, если тебе абсолютно всё равно, чем ты занимаешься. В принципе, я могу, например, и столярничать, и слесарничать. Но я знаю, что я это буду делать хуже и медленнее, чем профессионал. Для меня не

всё равно, чем заниматься. Хороший заработок — это хороший допуск к любимому делу. Если бы мне предложили другой прибыльный бизнес, я бы чувствовал себя в нём не очень компетентным, не очень уверенным. Мне пришлось бы положиться на мнение более авторитетных людей, что уверенности в успехе, как вы понимаете, не придаёт.

— На каком расстоянии, как начальник, Вы держитесь со своими сотрудниками?

— Дверь моего кабинета обычно открыта. Я иногда закрываю её, чтобы не мешали шаги по коридору. Кстати, в США, как правило, двери в офисы всегда открыты. А во многих компаниях вообще нет дверей. Мы ведь часть коллектива. Постоянно открытые двери снимают барьер и упрощают/ускоряют общение с моими коллегами.

— Так в чём же секрет успеха EffectiveSoft?

— Надо работать. Поскольку мы являемся коммерческим предприятием, все планы в жизни компании основаны на том, что мы сами заработали. Люди, которые с нами работают, и клиенты, которые нас любят, — это основная составляющая нашего успеха.

[Обсудить](#)

KOSHT.com
выбирай всегда

- Ноутбуки
- Жесткие диски (винчестеры)
- Видеокарты и графические ускорители
- Мониторы
- Мобильные телефоны
- Процессоры
- Системные платы (motherboard)
- Принтеры
- USB Flash накопители
- Цифровые фотоаппараты
- Телевизоры

ABAP-разработчик (SAP CRM)

Полная занятость
Атлантконсалт

Консультант SAP BW

Полная занятость
Атлантконсалт

Ведущий SAP BI консультант

Полная занятость, гибкий график
EPAM Systems

ABAP-разработчик

Полная занятость
Атлантконсалт

Архитектор Oracle

Полная занятость, гибкий график
EPAM Systems

Microsoft BI разработчик

Полная занятость, гибкий график
EPAM Systems

C++-программист

Полная занятость
Applied Systems

C/C++-разработчик

Полный рабочий день, для студентов — от 30 часов/неделю
ltransition

Разработчик C++

Полная занятость. Могилев
Awem-Studio

C++ Developer/ Team Lead

Полная занятость, гибкий график
EPAM Systems

C#-, ASP.NET-/WINFORMS-программист

Полная занятость
BelHard

Middle/Senior .NET Developer

Полная занятость, гибкий график
EPAM Systems

ASP.NET C# программист

Полная занятость
Открытый контакт

.NET-разработчик

Полный рабочий день, для студентов — от 30 часов/неделю
ltransition

Лидер ASP.NET команды

Полный рабочий день, гибкий график
ltransition

ASP.NET C# Developer

Полная занятость
Открытый контакт

ASP.NET(C#)-разработчик

Полный рабочий день, для студентов — от 30 часов/неделю
ltransition

JAVA(J2 EE)-программист

Полная занятость
BelHard

Java-разработчик

Полный рабочий день, для студентов — от 30 часов/неделю
ltransition

Ведущий Java-разработчик

Полный рабочий день, для студентов — от 30 часов/неделю
ltransition

Разработчик Java

Полная занятость, гибкий график
EPAM Systems

Сборщик и установщик версий разрабатываемой ИС (JAVA программист с навыками системного администратора)

Полная занятость.
BelHard

PHP-разработчик

Полный рабочий день, для студентов — от 30 часов/неделю
ltransition

Ruby on Rails-разработчик

Полный рабочий день, для студентов — от 30 часов/неделю
ltransition

PHP-программист

Полная занятость
BelHard

PYTHON-программист

Полная занятость
BelHard

Mobile-разработчик

Полный рабочий день, для студентов — от 30 часов/неделю
ltransition

Гейм-дизайнер

Полная занятость. Могилев
Awem-Studio

Web-дизайнер

Полная занятость. Могилев
Awem-Studio

Web-дизайнер

Полная занятость
Открытый контакт

Ведущий дизайнер (от \$1500)

Гибкий график
ltransition

Художник компьютерной графики

Полная занятость
Applied Systems

UI разработчик

Полная занятость, гибкий график
EPAM Systems

Специалист по тестированию ПО

Полная занятость. Могилев
Awem-Studio

Специалист по тестированию ПО

Полная занятость
BelHard

Специалист по тестированию

Полная занятость, гибкий график
EPAM Systems

Тестировщик ПО

Полная занятость
Applied Systems

Функциональный тестировщик (QA-инженер)

Полный рабочий день, для студентов — от 30 часов/неделю
ltransition

Senior QA Engineer, ведущий специалист по тестированию ПО

Полная занятость
BelHard

Senior QA Functional Tester/Team Lead

Полная занятость, гибкий график
EPAM Systems

Специалист по функциональному тестированию ПО (SQL)

Полная занятость, гибкий график
EPAM Systems

Специалист по нагрузочному тестированию

Полный рабочий день, для студентов — от 30 часов/неделю
ltransition

Senior Automation Testing Engineer

Полная занятость, гибкий график
EPAM Systems

Специалист по автоматизированному тестированию веб-приложений

Полная занятость, гибкий график
EPAM Systems

Специалист по тестированию программного обеспечения

Полная занятость
Открытый контакт

Инженер по интеграции ПО

Полная занятость
Applied Systems

Support Engineer

Полная занятость, гибкий график
EPAM Systems

ИТ-специалист, сисадмин

Полная занятость
BelHard

Ведущий специалист по автоматизированному тестированию ПО (in Java)

Полная занятость, гибкий график
EPAM Systems

Специалист службы внедрения ERP и CRM систем

Полная занятость
BelHard

Бизнес-аналитик — специалист по анализу требований и созданию ТЗ

Полная занятость
BelHard

Marketing and Sales Assistant

Полная занятость
Applied Systems

.NET тренер

Полная занятость, гибкий график
EPAM Systems

Копирайтер

Полная занятость. Могилев
Awem-Studio

Копирайтер

Полная занятость
Applied Systems

Специалист по работе с клиентами

Полная занятость
Открытый контакт

Специалист по маркетингу и рекламе

Полная занятость
Открытый контакт

Менеджер по рекламе

Полная занятость
Открытый контакт



MoSync SDK: мобильная разработка на C/C++

Александр СТРЮКОВ

Введение

Цель статьи — познакомить читателя, знакомого с языками C/C++, с одним из средств разработки для мобильных платформ — MoSync SDK.

На данный момент доступна масса фреймворков, позволяющих разрабатывать приложения для мобильных платформ. Для подавляющего большинства из них основными языками являются Java или связка HTML + CSS + JavaScript. Использование языков C/C++ в сфере мобильной разработки в некотором смысле экзотика. Однако некоторые средства позволяют создать такие мобильные приложения. Так что если у разработчика есть знания C/C++ и большое желание создать своё собственное мобильное приложение, существует возможность поэкспериментировать на новом поле.

Введение в MoSync

MoSync — это средство разработки для мобильных платформ. Состоит из двух независимых продуктов — MoSync SDK (C/C++) и

MoSync Reload (HTML + CSS + JavaScript). В данной статье основное внимание уделено первому продукту.

Отличительные особенности данного фреймворка:

– Обилие поддерживаемых мобильных платформ

На сегодняшний день заявлено о поддержке следующих платформ: Android, Blackberry, iOS, JavaME, Moblin, Symbian, Windows Mobile, Windows Phone. В лучшем случае пересборка под другую платформу происходит без изменения исходного кода приложения. Но так бывает не всегда, конечно.

– Свободная лицензия

Для личного использования MoSync бесплатен. Если есть желание продавать создаваемые продукты, можно использовать для них GPLv2 лицензию, или придётся приобретать коммерческую лицензию.

– Подробная документация

Детально описан API продуктов. Кроме того, в наличии масса дополнительной документации: обзоры, установка, примеры кода и приложений. Документация рассчитана на различный уровень подготовки читателя, как новичок,

так и искушённый разработчик сможет найти для себя полезное в этой хорошо структурированной библиотеке. Отдельного упоминания заслуживает раздел “Руководства разработчика” (“Coding guides”). В нём представлено большое количество руководств, связанных со спецификой реализации различных аспектов мобильных платформ средствами MoSync: камера, bluetooth, gps, карты, звуки, шрифты, рекламные баннеры и многое другое.

– Активное сообщество разработчиков

Активность проявляется как в частоте обновлений исходного кода на github, так и в частоте сообщений на форуме. Общаться с разработчиками приятно, реагируют оперативно и добродушно.

– Открытый исходный код

Кодовая база находится в свободном доступе на github.

– Активная разработка проекта

Последняя версия датируется мартом 2013 года. Обновления установщиков появляются в среднем в течение полугода.

– Низкий порог вхождения

Благодаря обилию документа-

ции и примеров приложений для начала работы потребуются минимум усилий.

Относительные недостатки:

– Отсутствие документации на русском языке



ТВОЙ ПЕРВЫЙ ШАГ К ВЫСОКОЙ ЗАРПЛАТЕ

Компания **BELCHARD**®

приглашает всех желающих на бесплатную лекцию по теме:

"Знакомство с IT специальностями"

Вы узнаете Проект IT-Страна

О наиболее востребованных IT специальностях. Как стать IT специалистом за короткое время!?

Число мест ограничено

Предварительная запись по тел. **8(044) 744-75-54**

Научиться можно всему, даже начиная с "0"

MoSync SDK: мобильная разработка на C/C++

Крайне желательно иметь хотя бы средний уровень технического английского.

– Нет поддержки Linux

Хотя при большом желании можно попробовать собрать части **MoSync** из исходного кода, на сайте разработчика есть инструкции.

– Более узкие возможности по сравнению с нативной разработкой

MoSync SDK

MoSync SDK позволяет создать полноценное мобильное приложение с использованием языков C/C++. Актуальная версия 3.2.1 выпущена в марте текущего года. Основная среда разработки основана на Eclipse (рис. 1).

Установка фреймворка весьма проста и подробно освещена в документации на его сайте. Самый быстрый способ начать разработку—

импортировать один из примеров приложений и заняться его исследованием. Исходный код примеров стремится быть лаконичным и содержит обстоятельные комментарии, позволяющие понять происходящее.

Возможности

Рассмотрим наиболее востребованные возможности, доступные пользователю MoSync SDK.

Для каждой из них приведены названия примеров приложений.

– Сенсоры

AccelerometerOpenGLES, ScreenOrientation, SensorTest

Поддерживается солидный список сенсоров: акселерометр, гироскоп, датчик положения и т.д.

– Адресная книга

PIMExample

На данный момент работа с адресной книгой устройства находится на начальном уровне разработки. Однако ряд основных функциональностей уже реализован: доступ к контактам, добавление, изменение и удаление нового контакта.

– Рекламный API

NativeUIDemo

Позволяет выводить рекламу в мобильном приложении.

– API Покупок

PurchaseExample

Позволяет осуществлять покупки внутри приложения с помощью библиотеки MoSync Purchase C++.

– Звук

MoSound

Различные аспекты работы со звуком: загрузка, воспроизведение, повтор.

– Bluetooth

BluetoothServer, MoTooth

Различные аспекты работы с Bluetooth: создание сервера и клиента, обнаружение устройств, ус-

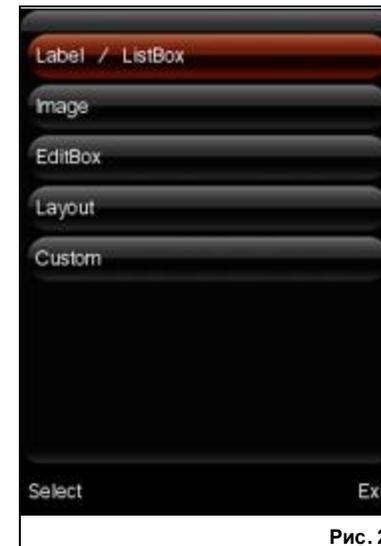


Рис. 2

тановка соединения.

– Базы данных

DatabaseTest

Пример работы с MoSync Database API.

– Шрифты

DeviceFonts, DeviceFontsNativeUI

Различные способы доступа к шрифтам устройства.

– GPS, карты

HelloMap, Location, MapDemo, NativeUIMap

Определение координат, отображение карт.

– Графика

MoSketch, Stylus

Различные аспекты работы с графикой: фрейм буфер, обрезка, графические прими-

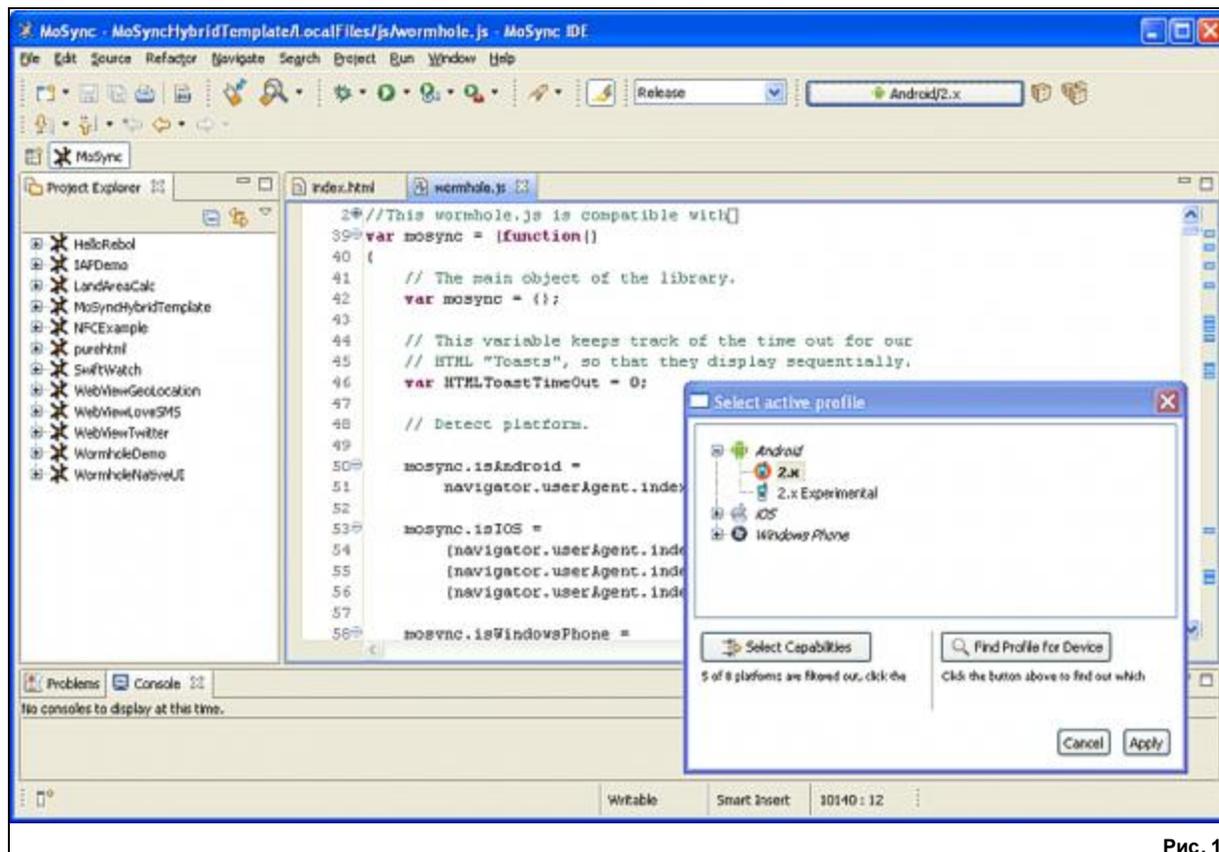


Рис. 1

MoSync SDK: мобильная разработка на C/C++

↑ ТИПЫ.
 – **Трёхмерная графика**
3DLines, AccelerometerOpenGL ES, HelloOpenGL ES, GLMOblet-

следующие:
 – **C Basic project**
 Проект, основанный на стандартном ANSI C. Содержит один

библиотеку.
 – **C++ OpenGL Project**
 Основа для реализации OpenGL проектов.

на старых аппаратах (рис. 2).
 – **Библиотека NativeUI**
 Нативный вид платформы, подходит для современных аппаратов. Пример пользовательского

зволяет запускать приложения, использующие NativeUI библиотеку — основную библиотеку для смартфонов (рис. 5).

– “Родной” эмулятор выбранной



Рис. 3

OpenGL ES 1/2
 Работа с OpenGL ES.

Полный список можно отыскать в разделах документации на сайте разработчика.

Типы проектов
 MoSync SDK позволяет создавать несколько типов C/C++ проектов.
 Наиболее характерные из них

файл с небольшим примером.
 – **C++ Moblet Project**
 Проект использует Moblet — объект для событийной модели взаимодействия.
 – **C NativeUI Project**
 Содержит цикл обработки событий и использует WidgetAPI для создания виджетов.
 – **C++ NativeUI Project**
 Реализует Moblet и использует для создания виджетов NativeUI

Пользовательский интерфейс
 MoSync SDK позволяет создавать несколько типов пользовательского интерфейса:

– **Widget C API**
 Сложен в реализации, представляет из себя низкоуровневый программный интерфейс системных вызовов.

– **Библиотека MAUI**
 Приложение выглядит одинаково на всех платформах; работает

интерфейс с использованием NativeUI библиотеки для платформ Android и Windows Phone 7 (рис. 3).
 Кроме того, приложение может иметь вид консольного приложения (что более актуально для аппаратов с физической клавиатурой) или просто выводить двух- или трёхмерную графику (рис. 4).

Эмуляция устройств
 MoSync даёт возможность заниматься разработкой или исследованиями даже без физического устройства. Существует два варианта:
 – Встроенный эмулятор MoRE.
 В составе SDK присутствует эмулятор MoRE, позволяющий создавать виртуальное устройство из обширной встроенной базы устройств/платформ. Вещь удобная, так как имеет следующие преимущества:

- интегрирован в среду разработки,
- удобный интерфейс,
- функция мультитач,
- низкое время загрузки.

Существенный недостаток видится в том, что MoRE, в силу архитектурных особенностей, не по-

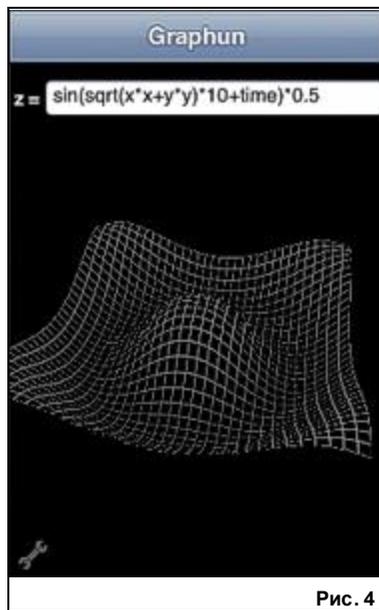


Рис. 4

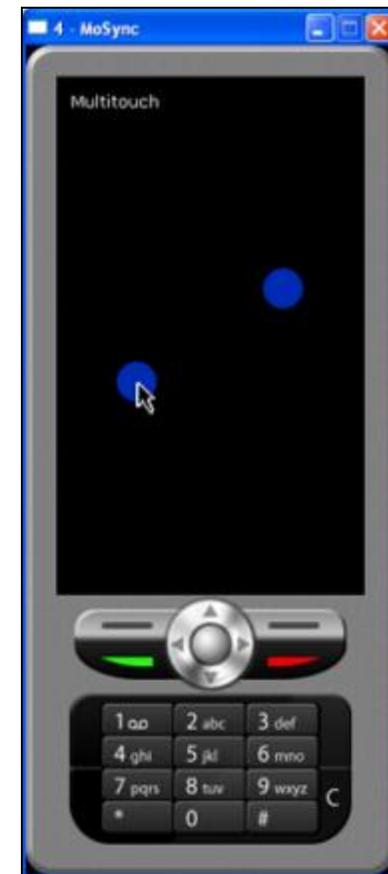


Рис. 5

платформы. При желании можно использовать Android Emulator и/или iPhone/iOS Simulator (рис. 6). Они устанавливаются и настраиваются отдельно от



MoSync SDK: мобильная разработка на C/C++

MoSync, а затем связываются со средой разработки. При выборе платформы и запуске приложения на исполнение будет

Внутренности MoSync SDK
Магия кросс-платформенной мобильной разработки реализуется внутренним инструментарием



Рис. 6

вызван соответствующий эмулятор. К сожалению, добиться стабильного выполнения приложений в среде Android Emulator'a удастся не всегда. Судя по всему, это связано с тем, что он довольно требователен к ресурсам компьютера. Конечно, наиболее надёжным способом получения рабочего приложения является отладка на реальном устройстве. Настройка связи телефона со средой разработки обычно происходит без проблем, приложение загружается на аппарат быстро, всё работает стабильно.

MoSync. В общих чертах, процесс выглядит так (рис. 7). MoSync SDK использует свой собственный вариант широко известного инструмента GCC, который генерирует код промежуточного языка MoSync IL. Результаты работы передаются следующему инструменту — Pipe-Tool. На данном этапе происходит преобразование кодовой формы байт-кода. Полученный байт-код передаётся последней в цепи инструментарию MoSync — утилите-упаковщику, которая создаёт пакеты, готовые для установки на устройстве или

эмуляторе.

– **GCC**

Реализация GCC от MoSync компилирует пользовательский исходный код в промежуточный язык MoSync IL, который содержит различную метainформацию о приложении.

– **Pipe-Tool**

Эта утилита комбинирует в себе функции компилятора ресурсов, линкера и оптимизатора кода. Происходит связывание кода MoSync IL с библиотеками MoSync.

На выходе получается один из форматов целевой платформы, включая байт-код MoSync, байт-код Java или исходный код C/C++. Также проводится верификация и оптимизация кода, компиляция внешних ресурсов.

– **Упаковщик**

Цель утилиты — создание пакета, готового к установке на целевой платформе или создание проекта для внешнего инструментария (например, создание Xcode проекта для платформы iOS или

VisualStudio проекта для платформы Windows Phone).

Заключение

Таким образом, MoSync SDK представляется разумным выбором для экспериментов C/C++ программистов на мобильных платформах. Он даёт возможность превратить практически любой мобильный телефон в среду исполнения вашего C/C++ кода.

[Обсудить](#)

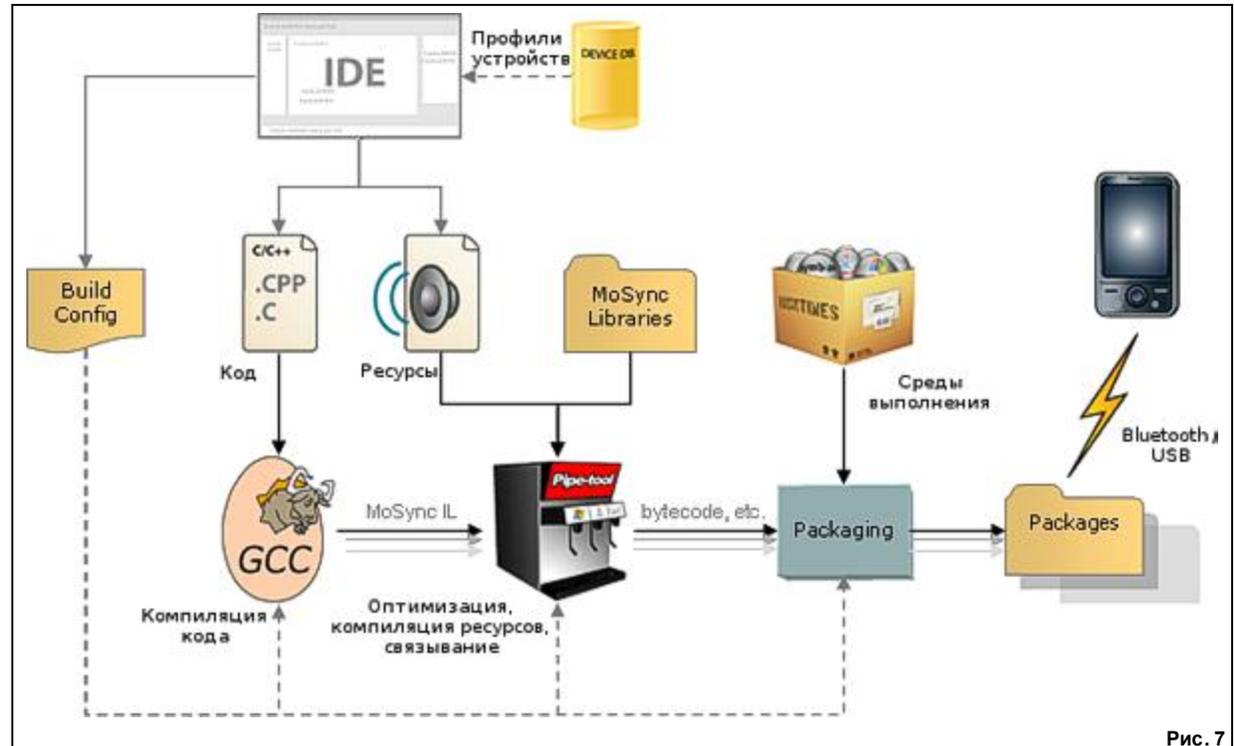


Рис. 7

Менеджерам

Financial Management for Project Managers

2 учебных дня, 16 часов
Учебный центр ИВА

Managing Projects in Software Development

2 учебных дня, 16 часов
Учебный центр ИВА

Project Cost and Schedule Management

3 учебных дня, 24 часа
Учебный центр ИВА

Комплексный Интернет-маркетинг. Как использовать интернет для пользы дела

60 часов
УЦ "БелХард Групп"

Принципы управления проектами

5 учебных дней, 36 часов
Учебный центр ИВА

Управление проектами при разработке ПО

Диплом Международного института качества. 3 уровня по 60 часов
ЦОТ "БелХард Групп"

Основы бизнес-анализа при разработке программного обеспечения

Диплом Международного института качества. 3 уровня по 48 часов
ЦОТ "БелХард Групп"

Построение системы менеджмента качества при помощи Business Studio

3 учебных дня
ЧП "СТАМ Софт"

Проектирование системы управления компанией в Business Studio

3 учебных дня
ЧП "СТАМ Софт"

Управление рисками в ИТ-проектах (риск-менеджмент при разработке и внедрении ПО)

ЦОТ "БелХард Групп"

Специалистам

Администрирование Oracle

40 часов
УЦ "БелХард Групп"

Авторизованные курсы Microsoft

Более 30 курсов по разным направлениям
Учебный центр ИВА

Администрирование Linux

Сертификат. 1,5 месяца (18 занятий)
ОЦ "Айти Эксперт"

Использование сетевого оборудования Cisco. Части 1-2

5 учебных дней, 40 часов
Учебный центр ИВА

Основы поддержки Mac OS X 10.6

3 учебных дня, 24 часа
Учебный центр ИВА

Академия 1С-Битрикс

Серия курсов "1С-Битрикс: Управление сайтом". 6 курсов для разных специалистов
От 1 до 4 учебных дня, 8-32 часа
Компания "Новый сайт"

Приложения для iPhone: разработка и программирование

56 часов
УЦ "БелХард Групп"

Windows Phone 7: проектирование и разработка приложений

60 часов
УЦ "БелХард Групп"

SEO-продвижение

2 месяца, 60 часов
ООО "Шипалекс"

Система информационной безопасности: организационные аспекты и техническая защита информации

5 учебных дней, 36 часов
НПП "БелСофт"

Разработка приложений для платформы Android

60 часов
УЦ "БелХард Групп"

Продвинутый SEO-курс (для бизнес-специалистов)

30 часов
Институт Бизнес-Технологий

Промышленная разработка ПО

2 месяца, бесплатно
ITransition

Профессиональный курс 3D-моделирования. Уровень 1

40 часов
ЦКО "Freedom"

Начинающим

1С: Бухгалтерия: Программирование

42 часа
Институт Бизнес-Технологий

Автоматизация тестирования ПО

Бесплатно
ITransition

Тестирование ПО

Бесплатно
ITransition

Java-программирование для новичков

1.5 месяца, 40 часов
ООО "Шипалекс"

Основы программирования на Java

60 часов
УЦ "БелХард Групп"

Видеомонтаж для начинающих

40 часов
Институт Бизнес-Технологий

Web-дизайн (комплексный курс)

72 часа
УЦ "БелХард Групп"

Web-дизайн: HTML, CSS, Flash, ActionScript

96 часов
ОЦ "Юниверсум"

Базовый SEO-курс (от новичка до специалиста)

58 часов
Институт Бизнес-Технологий

Графический дизайн в Photoshop и CorelDraw

1.5 месяца, 45 часов
ООО "Шипалекс"

Дизайнер-аниматор: 3DS Max, Macromedia Flash

70 часов
Институт Бизнес-Технологий

Компьютерное проектирование (САПР AutoCAD)

36 часов
ОЦ "Юниверсум"

Увлечения

Ди-джей на радио

70 часов
Институт Бизнес-Технологий

Студия танцев ODEON

Капоэйра. Набор в новые группы

Курсы по дайвингу

Курсы по стрельбе в стрелковом клубе Shoot.by

Фехтование для начинающих

Школа фотографии

Экстремальное вождение

Школа Н. Овчинникова



11 вещей, которые нужно знать исполнительным директорам стартапов

Nick SHCHETKO, [Minyanville](#)

“Питч-семинар инвесторов похож на свидание вслепую”, — утверждает Джей Сэммит (Jay Samit), инвестор, президент ooVoo, а также известный новатор в сфере цифровых технологий, который начал распространять технологии CD-ROM и LaserDisc. — “В большинстве случаев вы очень быстро понимаете, придёт ли вам провести всю свою оставшуюся жизнь с человеком, сидящим рядом с вами”.

На семинаре WSJ Tech Cafе, который прошёл недавно в Нью-Йорке, Сэммит и его коллеги обсуждали, что же должны осознавать предприниматели перед тем, как назначать “свидание вслепую” с инвестором, и как сделать его успешным.

Кроме Сэммита, на семинаре также выступал бизнес-ангел Брайан Коэн (Brian Cohen), председатель организации NYAngels, а также рискующий предприниматель Стив Кейс (Steve Case), председатель и исполнительный директор инвестиционной компании Revolution LLC и соучредитель холдинговой компании AOL.

Тройка (на фото вместе с руководителем семинара Ванессой О’Коннелл (Vanessa O’Connell), редактором в Wall Street Journal)

дала кучу полезных советов предпринимателям, находившимся в тесном помещении. Вот некоторые главные аспекты:

1. Хорошо подготовьтесь к своему выступлению. Нет, это не значит, что нужно создать бесконечную презентацию в PowerPoint. Добавьте чуточку индивидуальности и харизмы, откройтесь, установите личный контакт, даже “будьте забавным и”, как предлагает Коэн. Знайте свой проект, и, что ещё более важно, своего клиента. Каков показатель стоимости привлечения клиента? Почему ваши клиенты пользуются вашими продуктами? Будьте готовы, что потенциальный инвестор может позвонить вашим клиентам, но не раньше, чем они спросят вас о том, кто был вашим

клиентом и почему им понравился ваш продукт.

2. Стремитесь заинтересовать инвестора в первые 10–15 секунд. Эксперты единодушно согласны в том, что в большинстве случаев инвестору требуется 30 секунд, или даже меньше, чтобы

экономить время, обе стороны должны быть откровенными в своих намерениях и выводах.

4. Изучите своих инвесторов не спеша. Узнайте об их интересах, приоритетах и чертах характера. Вы должны быть на одной волне — когда дело дойдёт до

стихии и верят ли они в вас. Реагируйте на критику. Будьте готовы слушать и слышать. Как утверждает Коэн, важно также “поддаваться обучению”.

5. Не пере- или недооценивайте важность больших идей. Кейс говорит, что он больше стре-



сказать “Нет”, как минимум, усебя в голове.

3. Не бойтесь слова “Нет”. “Это следующее слово, которое вы хотите услышать”, подчёркивает Коэн. “Стремитесь к нему изо всех сил”, советует он. Чтобы сэ-

финансирования, вполне вероятно, что вы будете проводить больше времени с инвесторами, а не со своими родными и близкими. Спросите инвесторов об их стратегиях инвестирования, увлечениях, почему они вкладывают инве-

стируются инвестировать идеи, а не люди, и, однако признаёт, что самой трудной задачей является претворение идей в жизнь. Учитывая то, что на сегодняшний день создать стартап технически не так уж и сложно, ↓

11 вещей, которые нужно знать исполнительным директорам стартапов

вам лучше иметь что-то уже работающее, когда вы выступаете перед инвестором. Держите в уме знаменитую фразу Эдисона “Невоплощенные мечты есть галлюцинации”. То есть, даже самая невероятная, но не воплощенная идея может быть отвергнута.

6. Найти нужного, идеально-го инвестора очень проблематично. Если кто-то вам отказывает, это может значить всё, что угодно — возможно, ему/ей не нравится сегмент рынка, в котором вы работаете. Например, инвестиционная компания Digital Sky Technologies предпочитает современные быстроразвивающиеся Интернет-компании и поэтому вложила капитал в Facebook, Groupon и Zynga как только каждая из этих компаний прочно обосновалась на рынке и стала готовой к выходу на IPO. Некоторые инвесторы следуют моде и охотно вкладывают деньги в крупные информационные компании, или в стартапы, занимающиеся коммуникациями ближнего поля (NFC стартапы), или во что-нибудь ещё, что, по их мнению, является модным на данный момент. Другие смотрят шире. Существуют десятки целевых инвесторов, и вполне возможно, что

человек, который отклонил ваше предложение, порекомендует вам, в какую дверь можно постучать.

7. Друзья и семья также должны делать свой вклад. В бизнесе важными являются как деньги, так и вера в то, что вы делаете, следовательно, те, кто верят в успех вашего проекта, тоже должны быть готовы сделать вклад. По мнению инвесторов, хорошо, когда в первую очередь ваш проект поддерживают ваши друзья и родственники, а уже потом в дело вступает сторонний инвестор.

8. Значимость работы в команде не может быть недооценена. Как говорит Кейс, “Если вы хотите идти быстро, идите один, но если вы хотите идти далеко, то вам нужна компания”. Коэн говорит, что при оценивании стартапов он уделяет особое внимание командному компоненту. “В сущности, они должны уметь заканчивать предложения друг друга”, утверждает он.

9. Добросовестность: сделайте это сами и не пугайтесь, если это сделает ваш потенциальный инвестор. Коэн уверен, что добросовестность принесёт только пользу начинающей компании.

10. ВСЕГДА (да, именно заглавными буквами) помните о

конкуренции. Проводите тщательные исследования и откровенно докладывайте инвестору о том, что вы узнали. Говоря “У нас нет конкурентов”, вы тем самым сразу же подаёте инвестору тревожный сигнал. Вы должны знать обстановку на рынке, на который вы выходите, и быть готовым указать, чем именно вы лучше своих конкурентов.

11. Да, вы слышали об этом уже много раз, но провалы имеют значение. Не сходите с ума, если вы потерпели неудачу. Сэммит говорит, что он предпочитает инвестировать компании, которые уже терпели крах, а Коэн упрявлялся “успешные провалы”. Возможность быстро становиться банкротом в бизнесе, который слабеет, является необходимой. Легко говорить, но трудно сделать. Однако примите неудачу и извлеките из неё урок, превратите её в положительный опыт. Коэн считает, что настоящие неудачи случаются тогда, когда “вы перестаёте пытаться”.

Чтобы подвести всему вышесказанному итог, приведу общеизвестное утверждение, процитированное Сэммитом: “Из его собственного опыта: а учредители, которые больше фокусировались на том,

чтобы заработать больше денег, редко преуспевали. Вместо этого, целью должно быть “Я собираюсь изменить весь мир”. Обращайтесь

к инвесторам с чувствами, идущими из глубины души”.

[Обсудить](#)

ООО "Открытый контакт", лицензия Минсвязи РБ № 02140, от 23.03.2009 до 26.04.2014
УНП 100008738

Курсы валют

OPENBY

\$ € ₪ = ? ? ? ? Курс НБРБ

- Актуальные курсы банков
- Конвертер валют
- Кросс-курсы
- Архив курсов валют

www.open.by/finance

Будьте в курсе!

Мы с удовольствием опубликуем статьи по следующим тематикам:

- Всемирная паутина
 - софт
 - мобильные устройства
 - программирование и разработка ПО
 - администрирование
 - наука
 - технологии
- и по многим другим!

Знаешь что-то
интересное?
Поделись этим
с читателями
“КВ”!

Нам не важно, студент ты или матерый профессионал.

**Главное - интересная тема и горячее
желание рассказать о ней!**

Нет опыта в написании статей? Наши редакторы помогут довести твои мысли до публикации.

Пиши: info@kv.by

Звони: (017) 203-90-10



10 языков программирования, которые могут перевернуть мир IT

Эти передовые языки программирования обладают уникальной способностью изменить будущее разработки ПО

Neil McALLISTER, [InfoWorld](#)

Действительно ли нам нужны новые языки программирования? Безусловно, на данный момент их вполне достаточно. Среди разнообразия императивных, функциональных, объектно-ориентированных, динамических, компилируемых, интерпретируемых и скриптовых языков ни один разработчик не сможет познать все доступные на сегодняшний день возможности.

И всё же возникновение новых языков — явление довольно частое. Некоторые из них создаются студентами или любителями в качестве индивидуальных проектов, другие являются продуктами крупных производителей программного обеспечения. Даже небольшие и средние компании принимают участие в этом процессе, создавая языки для нужд своих отраслей. Так почему же люди продолжают изобретать велосипед снова и снова?

Дело в том, что, несмотря на мощь и многофункциональность популярных на данный момент языков, ни один синтаксис не является идеально универсальным. Более того, само программирование постоянно развивается.

Распространение многоядерных процессоров, облачного программирования, мобильности, а также распределённых архитектур создали новые проблемы для разработчиков. Добавление поддержки самых последних функций, парадигм и шаблонов к уже существующим языкам, особенно наиболее популярным, может быть чрезмерно сложным. Иногда лучшим решением является начать с нуля.

Таким образом, здесь представлены 10 передовых языков программирования, каждый из которых рассматривает искусство разработки ПО с новой стороны, решая определённую проблему либо

специфический недостаток языков, наиболее популярных на сегодняшний день. Некоторые из них являются уже законченными проектами, тогда как другие находятся лишь на ранних стадиях своего развития. Вполне вероятно, что некоторые из них так и не обретут популярность, но любой из них может стать революционным достижением, которое окончательно изменит программирование — по крайней мере до тех пор, пока не будут созданы новые языки.

Исследуемый язык программирования №1: Dart



JavaScript хорош для добавления базовых элементов интерактивности веб-страницам, но когда код ваших веб-приложений состоит из тысяч строк, слабые места этого языка становятся заметны. Вот почему Google создала Dart —

язык, который, как полагает компания, станет новым “родным” языком веб-программирования.

Как и в JavaScript, в Dart используются синтаксисы и ключевые слова, похожие на те, которые используются в языке C. Однако одним существенным различием является то, что в то время как JavaScript основывается на прототипах, объекты в Dart определяются с помощью классов и интерфейсов, как в C++ или Java. Также Dart позволяет программистам дополнительно задавать переменные статическими типами. Идея заключается в том, чтобы сделать Dart таким же привычным, динамичным и гибким языком, как и JavaScript, который в то же время позволяет разработчикам писать коды, быстрые и лёгкие в выполнении, и в которых сложно сделать труднонаходимые ошибки.

Сегодня Dart мало где можно использовать. Он разработан для запуска либо на клиенте, либо на сервере (а-ля Node.js), но единственным способом запустить клиентскую версию Dart-кода является его кросскомпиляция в JavaScript. Однако и после этого он не будет запускаться во всех браузерах. Но так как Dart выпускается по бесплатной лицензии типа BSD, то любой продавец, который согласен с условиями Google, может свободно использовать этот язык в своих продуктах. Всё, что осталось сделать Google, — это убедить всю индустрию.

Исследуемый язык программирования №2: Ceylon



Гэвин Кинг (Gavin King) отрицает тот факт, что Ceylon, разрабатываемый им язык в Red Hat, должен стать “убийцей Java”. Кинг хорошо известен как создатель фреймворка объектно-



10 языков программирования, которые могут перевернуть мир IT

↑ реляционного отображения Hibernate для Java. Ему нравится Java, но он считает, что в нём ещё многое нужно улучшить.

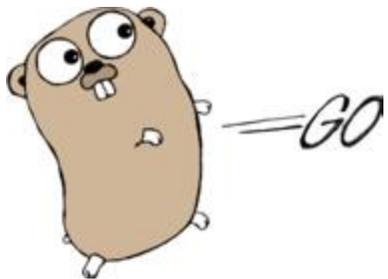
Среди тех вещей, которые не устраивают Кинга в Java, находятся подробный синтаксис языка, отсутствие первоклассных функций и функций высшего порядка, а также недостаточная поддержка метапрограммирования. Особенно его огорчает отсутствие декларативного синтаксиса для определения структурированных данных, что, по словам Кинга, “приравнивает Java к XML”. Seylon нацелен решить все эти проблемы.

Кинг и его команда не планируют заново изобретать колесо. Никакой виртуальной Seylon-машины не будет, Seylon-компилятор будет выдавать java-байткод, который можно запустить на JVM. Но Seylon будет больше, чем просто компилятор. Стратегической целью данного проекта является создание нового комплекта разработчика для Seylon, который заменит Java, являющегося, по словам Кинга, “разбухшим”, неуклюжим и до сих пор тщательно не модернизированным.

Это трудная задача, и пока что Red Hat не выпустил никаких утилит для Seylon. Кинг говорит, что компилятор должен выйти в этом

году. Но в ближайшее время не ждите какого-либо ПО, написанного на “100% чистом Seylon”.

Исследуемый язык программирования №3: Go



Интерпретаторы, виртуальные машины, а также управляемые коды сегодня очень популярны. Неужели нам действительно нужен ещё один старомодный язык, который компилирует в “родные” двоичные коды? Команда инженеров Google — под руководством Роберта Гризмера (Robert Griesemer) и легенд Bell Labs Кена Томпсона (Ken Thompson) и Роба Пайка (Rob Pike) — говорит “да”.

Go — язык программирования общего назначения, который подходит для всего: начиная от разработки приложений и заканчивая программированием систем. В этом смысле он больше похож на C или C++, чем на Java или C#. Однако как и последние языки, Go включает в себя такие современные функции, как очистка памяти

от ненужных данных, отражение текущих процессов выполнения и поддержку параллельности.

Также важно то, что Go был создан для лёгкого программирования. Его основной синтаксис похож на C, однако Go удаляет ненужный синтаксис и повторения во время оптимизации таких программ, как, к примеру, определение объектов. Целью команды Go было создание языка, на котором было бы удобно писать коды, как и на динамическом языке сценариев, но который в то же время обладал мощностью компилируемого языка.

Над Go ещё работают, и спецификация языка ещё может быть изменена. Тем не менее, работать с ним можно уже сейчас. Google создала утилиты и компиляторы, к которым прилагается подробная документация. К примеру, в учебном руководстве Effective Go хорошо показаны отличия Go от предыдущих языков.

Исследуемый язык программирования №4: F#



Функциональное программирование давно пользуется популярностью у программистов и академиков, но чисто функциональные языки, такие как Lisp и Haskell, часто считаются непригодными для разработки реального ПО. Одной из наиболее распространённых причин является то, что код функционального вида очень сложно сочетать с кодом и библиотеками, написанными на императивных языках, к примеру, на C++ или Java.

Знакомьтесь, F# (произносится “Эф-шарп”) — язык, разработанный Microsoft с целью объединения в себе функциональности и практичности. Так как F# является языком первого класса на виртуальной машине .Net CLR, он может поддерживать такие же библиотеки и функции, как и другие CLR языки, например, как C# и Visual Basic.

Код F# чем-то похож на код OCaml, но содержит свой собственный интересный синтаксис. Например, для облегчения проведения научных расчётов числовыми типами данных в F# могут являться единицы измерения. Также в F# имеются конструкции для облегчения асинхронных вводов/выводов, параллелизации ЦПУ и вывода процессов на графический

процессор.

После длительного периода созревания в Microsoft Research, на сегодняшний день F# поставляется вместе с Visual Studio 2010. Что ещё лучше, но не характерно для корпорации, Microsoft создала F# компилятор и корневую библиотеку, доступную по открытой лицензии Apache. Вы можете начать работу с ним бесплатно и даже использовать его на системах Mac и Linux (с помощью Mono runtime).

Исследуемый язык программирования №5: Ора



Веб-программирование — дело очень сложное. Даже самые простые веб-приложения состоят из бесконечных строк, написанных на нескольких языках: HTML и JavaScript для клиента, Java или PHP для сервера, SQL для базы данных и т.д.

Ора не заменяет ни один из этих языков по отдельности. Более того, он пытается заме-



10 языков программирования, которые могут перевернуть мир IT

▲ нить их все сразу, представляя собой совершенно новую парадигму в веб-программировании. В Ора-приложении интерфейс пользователя клиента, логика сервера и база данных ввода-вывода написаны на одном языке, Ора.

Это достигается с помощью комбинации фреймворков со стороны клиента и сервера. Компилятор Ора решает, где должна быть запущена данная программа (на клиенте, сервере либо же и там, и там), и затем выполняет её код. Для клиентских программ он переводит Ора на соответствующий JavaScript код, в том числе и AJAX вызовы.

Естественно, с такого рода интегрированной системой в конце должно получиться что-то волшебное. Среда выполнения Ора объединяет собственный веб-сервер и систему управления базой данных, которые не могут быть заменены самостоятельными альтернативами. Как бы то ни было, это может быть не так уж и важно, учитывая возможность разработки современных веб-приложений, управляемых данными, с помощью всего лишь нескольких десятков строк кода. Ора поставляется бесплатно и на данный момент доступен для 64-х битных Linux и Mac

OS X платформ, другие же порты пока разрабатываются.

Исследуемый язык программирования №6: Fantom



Хотите создавать приложения для Java или .Net? Если Вы пишете на Fantom, то можете сделать выбор, а потом даже поменять платформы. Всё потому, что Fantom с самого начала создавался для кросс-платформенной мобильности. Проект Fantom состоит не только из компилятора, который может выводить байткод как для JVM или .Net CLI, но также из набора API, которые отделяют API Java и .Net и создают дополнительный уровень мобильности.

Планируется ещё больше увеличить мобильность Fantom. Уже доступен компилятор из Fantom в JavaScript, а в следующие цели могут входить проект компилятора LLVM, Parrot VM и Objective-C для iOS.

Однако мобильность — не вся суть Fantom. Оставаясь по своей

природе похожим на C, он также должен улучшить языки, которые вдохновили разработчиков на его создание. Он пытается занять промежуточную позицию в некоторых наиболее спорных вопросах синтаксиса, таких как строгая или динамическая типизация, либо же привиборе между интерфейсами и классами. Он обладает дополнительным лёгким синтаксисом для объявления структур данных и сериализации объектов. Также он поддерживает встроенное функциональное программирование и параллелизм.

Fantom распространяется бесплатно в соответствии с Academic Free License 3.0 и доступен для Windows и Unix-подобных платформ (в том числе и Mac OS X).

Исследуемый язык программирования №7: Zimbu



Большинство языков программирования заимствуют функции и синтаксис из ранее существовавших языков. Zimbu взял по кусочку почти из каждого. Детище Брэ-

ма Мооленаара (Bram Moolenaar), создателя текстового редактора Vim, Zimbu намерен стать быстрым, точным, портативным и лёгким для чтения языком, который может быть использован для разработки чего угодно: от GUI-приложений до ядер ОС.

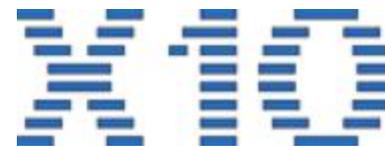
Благодаря своему смешанному характеру, синтаксис Zimbu уникален и специфичен, но в то же время обладает большим количеством функций. Он использует выражения и операторы, похожие на те, которые используются в C, но со своими ключевыми словами, типами данных и блочными структурами. Он поддерживает управление памятью, потоки и конвейеры.

Единственной проблемой является портативность. Хотя Zimbu и является компилируемым языком, его компилятор выдаёт ANSI C код, а двоичные файлы могут быть созданы лишь на платформах со встроенным C-компилятором.

К сожалению, проект Zimbu находится на стадии разработки. Компилятор и несколько программ-примеров могут быть созданы самостоятельно, но не весь действительный Zimbu-код будет компилироваться и выполняться должным образом. Не все заявленные функции ещё разра-

ботаны, а некоторые из уже представленных работают некорректно. Спецификация языка также, возможно, со временем изменится: по мере необходимости добавятся ключевые слова, типы и синтаксис. Следовательно, документация также ещё не полная. Однако если Вы хотите попробовать, предварительные утилиты уже доступны по лицензии Apache.

Исследуемый язык программирования №8: X10



Когда-то параллельная обработка данных была специализированной нишей разработки ПО, но с распространением многоядерных процессоров и распределённых вычислений, параллелизм обрёл популярность. К сожалению, нынешние языки программирования не успевают за этой тенденцией. Именно поэтому IBM Research создаёт X10 — язык, созданный специально для современных параллельных архитектур, который нацелен на увеличение производительности разработчиков “вдесять раз”.

10 языков программирования, которые могут перевернуть мир IT

Параллелизм в X10 возможен благодаря PGAS модели программирования (модели разделённого глобального адресного пространства). Код и данные выделяются в блоки и распределяются по разным “пространствам”, тем самым облегчая шкалирование программы от однопоточкового прототипа (одно пространство) до многопоточкового, выполняемого на одном или более многоядерном процессоре (несколько пространств) в высокопроизводительном кластере.

Код X10 больше всего похож на Java. По сути, среда выполнения X10 доступна как в качестве встроенных исполнимых файлов, так и как классовые файлы для JVM. Компилятор X10 может выдавать исходные коды либо на C++, либо на Java. В будущем планируется разработать прямую совместимость с Java.

А пока язык развивается, хотя он уже довольно разработан. Компилятор и среда выполнения доступны для различных платформ, в том числе Linux, Mac OS X и Windows. В качестве дополнительных утилит выступают интерактивная среда разработки (IDE), основанная на Eclipse, и отладчик, которые распространяются по лицензии Eclipse Public License.

Исследуемый язык программирования №9: haXe



Многие языки можно использовать для написания переносимого кода. С-компиляторы доступны практически для всех ЦПУ архитектур, а Java-байткод будет выполняться везде, где есть JVM. Но haXe (произносится как “хекс”) является более чем просто переносимым. Это мультиплатформенный язык, который может использоваться в различных операционных средах, начиная от встроенных двоичных систем до интерпретаторов и виртуальных машин.

На данный момент разработчики могут писать программы на haXe, затем компилировать их в объектный код, JavaScript, PHP, Flash/ActionScript или байткод Neko VM. Дополнительные модули для компилирования C# и Java находятся в разработке. Дополнением к базовому языку является стандартная библиотека haXe, ко-

торая одинаково работает с любым объектом, а также специализированные библиотеки для выполнения характерных функций каждой из платформ.

Синтаксис haXe похож на C и обладает богатым набором функций. Главным преимуществом является способность устранения проблем, свойственных каждой из платформ. К примеру, в haXe строгая типизация используется там, где она не используется в JavaScript; к ActionScript добавляются параметры настройки и выводение типа; также полностью исключается плохо разработанный, непродуманный синтаксис PHP.

Несмотря на то, что haXe всё ещё находится в разработке, он уже используется в коммерческих целях своим создателем, игровой студией Motion Twin, так что всё серьёзно. Он доступен для Linux, Mac OS X и Windows по комбинации бесплатных лицензий.

Исследуемый язык программирования №10: Chapel



В мире высокопроизводительных вычислений редкие имена звучат громче, чем Cray. Никого не должно удивлять то, что Chapel, первый язык программирования, созданный Cray, был разработан с учётом суперкомпьютерных вычислений и кластеризации.

Chapel является частью программы Cray Cascade Program, масштабного проекта по высокопроизводительным вычислениям, частично профинансированного американским Управлением перспективных исследовательских программ в области обороны (DARPA). Среди целей данного проекта находятся отделение алгоритмов параллельной обработки от используемого оборудования, улучшение их производительности на архитектурах и увеличение портативности параллельных программ.

Синтаксис Chapel составлен из синтаксисов нескольких языков. В дополнение к наиболее часто используемым (C, C++, Java), в нём заимствованы понятия из таких языков научного программирования, как Fortran и Matlab. На его функции параллельной обработки данных повлияли ZPL и High-Performance Fortran, а также предыдущие проекты Cray.

Одной из наиболее притяга-

тельных функций Chapel является его поддержка “программирования с переменным разрешением”, что позволяет разработчикам создавать приложения с очень абстрактным кодом и делать дополнения по мере того, как реализация становится более определённой.

Работа над Chapel продолжается. На данный момент он может быть запущен на суперкомпьютерах Cray и различных высокопроизводительных кластерах, но его можно портировать на большинство систем типа Unix (в том числе на Mac OS X и Windows с помощью Cygwin). Исходный код доступен по бесплатной лицензии типа BSD.

[Обсудить](#)

KV: А КОМПЬЮТЕРНЫЕ ВЕСТИ

Издатель: ООО “РГ “Компьютерные Вести”

Адрес: Минск, ул. Мельникайте, 2, оф. 710.

Для писем: 220004, г. Минск, а/я 57.

Телефон/факс: (017) 203-90-10

E-mail: inf@kv.by

Редакция может публиковать в порядке обсуждения материалы, отражающие точку зрения автора. За достоверность приведенной информации ответственность несут авторы.

При перепечатке материалов ссылка на “KV” обязательна.

За достоверность рекламной информации ответственность несет рекламодатель.

Группа компаний "БелХард" приглашает на работу

В связи с ростом масштабов деятельности и открытием новых направлений требуются **специалисты высокой квалификации** в международные проекты на полную занятость:

- **Программисты прикладных систем** J2EE, C#, C++, Delphi, Python,
- **Web-программисты** ASP.NET, PHP, Ruby, Flash и Web-дизайнеры,
- **Программисты мобильных приложений** iOS, J2ME,
- **Руководители проектов, бизнес-аналитики** (разработка ТЗ для АСУП),
- **Системные интеграторы** (сисадмины со знанием Java),
- **Функциональные тестировщики, тест-разработчики.**

Наши ценности — это сильная команда, постоянное профессиональное совершенствование.

Предлагаемые нами условия: достойные вознаграждения, премии за достижения, широкие карьерные перспективы, соц. пакет с льготами от резидента ПВТ, эффективные процессы (ISO, CMMI) и современный инструментарий, разнообразие творческих задач, благоприятная атмосфера в команде.

С нами Вы сможете реализовать себя в актуальных, интересных проектах!

Специальное предложение студентам ИТ-специальностей со знанием английского языка:

- Проводим набор на стажировку с последующим трудоустройством, направления: SW Tester и SW Developer (PHP, Java, C#, iPhone),
- Гибкий график и сокращенная до 30 часов рабочая неделя,
- Стажеры могут быть направлены к нам на преддипломную и производственную практику,
- Наши сотрудники-выпускники вузов получают возможность оформиться на работу в качестве молодых специалистов (по распределению).

Подробная информация о вакансиях, об интенсивно растущих секторах корпорации,

бланк резюме: www.job.belhard.com.

E-mail для резюме: job@belhard.com.