# О чем моя работа?

Прежде всего хотелось бы сказать, что моя работа однозначно не претендует на оптимистичный взгляд на будущее IT-специалистов. Несмотря на заметный каждому думающему человеку рост отрасли в целом, в ней кроется серьёзная проблема, связанная с людьми, работающими в ней.

Последующие главы помогут раскрыть мое текущее отношение к специалистам данной отрасли, некоторые тенденции, а как следствие и более полно раскрыть заявленную тему "как я вижу программиста в 2015 году".

Все последующее является моим личным субъективным мнением, никаким образом не претендует на безоговорочную истину, а тем более вряд ли понравится самим программистам.

## Почему я считаю, что со специалистами проблемы

Проблем в отрасли много, и я попытаюсь обозначить некоторую часть из них. Условно их можно разделить на проблемы, связанные с самими специалистами и их окружении. Некоторые из них, пожалуй, общечеловеческие, но это не вытесняет программистов, как частное от общего.

## Особая каста

Во-первых, хотелось бы начать разговор с того, что многие программисты говорят, что они "элитарная группа", они самые умные... Они движут мир вперед и всё подобное. Однако если брать "среднюю температуру по больнице" и сказать, что средняя зарплата 50 т.р в месяц, то возникают некоторые сомнения в дешевизне всего мира.

## Перфекционисты

Программисты, дизайнеры, верстальщики и прочие представители вышеупомянутого ремесла постоянно срываю все сроки выхода релизов. Для них это уже норма, это прикольно. При постановке задачи (о которых будет сказано позже) часто оказывается, что программист занят вообще другим - это может быть совершенно неизвестно что. Однако если что-то и пишут, то чаще всего проявляются две крайности:

* Откровенный "говнокод" - непонятно как связанная, работающая, а также крайне плохо читаемая последовательность букв.
* Попытка сделать универсальное решение. Если стоит задача сделать, чтобы код выполнят функции "A" и "B", то не надо пытаться сделать, во-первых, никому ненужный функционал, о котором никто не знает (а это баги+уязвимости), а во-вторых это чрезмерная куча абстракций в коде и чересчур разделение на малые части.

Все это срывает сроки. А это потери денег для бизнеса. А при чём тут вообще бизнес, если программисты "деятели искусства”? Ответ прост, программистам платят как обычным работникам как уборщице, например. Это не более чем объект в бизнес процессах. Держатся они порой только из-за проблем на рынке, о которых и начинается речь.

## Проблема новых знаний

Почти всегда говорят, что IT-специалисты чаще всего меняют свои методики/инструменты/языки, однако хотелось бы сказать, что это далеко не так. Есть отдельные показательные специалисты, которые действительно совершенствуют свои навыки - о них пойдет речь в следующей части (однако и там не без проблем).

Последнее время всё чаще замечаю лично, что около 80% программистов просто оставили свои навыки на определенном уровне и совершенно не хотят совершенствоваться, куда-то расти. Их совершенно устраивают полученные знания лет 5, а то и 10 назад. Одно дело, когда создаются продукты на не самых новых версиях (но хотя бы в рамках их поддержки). Иногда это связанно с требованиями заказчика, которым никто не может объяснить ценовые преимущества перехода на что-то новое. Проблема своего языка у ITшников - это отдельный разговор. Другое дело - это то, что программисты, не читающие ничего нового реально уверены в более традиционных подходах и технологиях. В этом случае мягко говоря всё работает не через лучшее место, что порождает ряд проблем для всех и нежеланием связываться вообще с IT отраслью лишний раз у потенциального заказчика.

Прежде всего это связано с тем, что IT-отрасль стала для многих людей, далеко не самых умных, местом где они могут одноразово изучив какие-то технологии (например, в 2000х годах) получать деньги необходимые для жизни. Специалистов вообще мало т.к отрасль новая. Так что это вполне работает.

У многих таких людей такое мировоззрение, что они просто хотят стабильности. А многие из нас вообще интересуются выходящими новинками? Например, пару недель назад была презентация Apple WWDC. Но многие ли вообще из разработчиков смотрели её? Потратить 2 часа своего вечернего времени ради того, чтобы узнать, что планирует одна из самых крупных IT-компаний на ближайшие полмесяца-год. Я не думаю, что разработчики должны упускать такой возможности.

## Ограниченность рамок познания

Разработчики крайне редко представляют что-то не связанное с другими языками и технологиями. Я даже не про то, что разработчик PHP не знает, например, Python.

Я даже про инструменты разработки. Как бы это печально не звучало, но не так много разработчиков, кто банально пользуется, например, Vagrant, Docker. Контейнеры и виртуализация - это что-то неслыханное. За их использование не платили; это сложно; не надо нам этого.

Многие разработчики вообще не представляют куда их код идёт дальше. Например, frontend разработчик на JavaScript вряд ли знает о процессе деплоя на сервера; тестирование клиентской части, если, например, используется AngularJS, часто так же не производится, ибо работает же здесь и сейчас. Как данные создаются на backend, например, по договоренности REST так же загадка.

Вообще во frontend особенно шаг влево - это незнание. Например, вы вряд ли убедите frontend разработчика использовать что-то на подобие Protobuf. А как это??? Где JSON ???

С 2010х годах появилось, пожалуй, непонимание с тем, что же такое jQuery. Это всего лишь framework для JavaScript. Но люди говорили и говорят (что еще ужасно), что jQuery лучше JS. Они вводят в Google:"Как сравнивать строки на jQuery".

Хорошо, что хоть начинают внедряться современные системы контроля версий.

Если же говорить о не технических знаниях, а об каких-то гуманитарных, то ситуация еще хуже. Показательно, что пытаются объяснить многие процессы в бизнесах именно с технической стороны. Легко манипулируемы.

Меня умиляет то, как все любят ставить дистрибутивы Linux, включая Gentoo, например, как основную ОС. Говорящих О ее превосходстве над всем остальным. Что уж говорить о кидании различных веществ в сторону Microsoft. Все гики разом высказались о презентации Surface pro 3 как об очень плохом продукте. Хотя сами никогда не смогут купить его из-за цены. Для них все, что не для них и куда нельзя установить Linux - плохая вещь. Слова «целевая аудитория» мало что говорит для них.

## Проблемы Facebook'a или нет

Не так давно Facebook переводил все свои проекты с SVN в систему контроля версий git, руководствуясь тем, что у них все медленно работает. Вот они перешли и оказалось, что commit приходится ждать по 40минут. Вот веселье то у них было. А ситуация была в том, что у них в был всего один главный репозиторий в более 8гб (+ 46-служебная информация системы контроля версий кода = 54gb), куда они пробовали слать изменения. Вопрос вроде очевидный: в git ты качаешь весь репозиторий и не можешь отдельно папку, так зачем и сервер грузить и качать лишнее: например, зачем Android разработчикам серверный код; создать бы десяток другой репозиториев и все.

Но это скучно как оказалось даже такой компании. Первое время после переход с SVN у них было все хорошо, потом из-за истории все стало притормаживать, и они решили перейти еще раз. На этот раз в Mercurial. Через некоторое время у них опять все стало тормозить (вот удивительно то). Конечно это проблема в системе контроля версий, решили они и начали изменять код Mercurial, начали урезать некоторые функции. Да еще и выложили свой порезанный Mercurial вместе со своей историй. Вопрос то простой: если даже в таких компания делают что-то странное, то что же во многих других?

Как бонус, недавно выяснил, что у многих проблемы вообще с пониманием многих технологий. Так, сказать, что же такое Continuous Integration, не сказав, что внутри там репозиторий кода и подобное крайне сложно.

## Проблемы гиков и менеджеров

Другой стороной может быть следующий случай. Человек, как правило асоциальный, применяет кучу новых средств для более быстрой разработки. Для него написать что-то новое и серьёзное на языке, который вышел неделю назад и никаким образом не production-ready - совершенно норма. Такие люди метались между Ruby on Rails к NodeJS в 2010х годах.

Вот только проблема в том, что такие системы, во-первых, далеко не стабильные и если что-то упадет, то лично для них это новый challenge, то вот если они уйдут вопрос последующей поддержки проекта компанией остается крайне открытым.

Ошибкой менеджеров в данном случае допуск разработки на языках, имеющий резкий всплеск - в данном случае надо разделять что-то революционное, что поможет сократить сроки и затраты от желания программистов покодить на чём-то новом. Этим людям неплохо бы находить что-то новое в жизни, а не перебирать все языки программирование по алфавиту.

## Облачные технологии - маркетинг

Многих не самые смышлёные программисты убедили многих именно в этом. Всё это маркетинг, всё это зло, данные на твоих виртуальных машинах (instance в EC2 AWS и droplet в DigitalOcean , например) не защищены, их все могут читать. Вот только люди, делающие находят решения а не сразу кидают нечто в сторону чего-то нового. Так, например, финансовые стартапы, сотрудничающие с крупнейшими банками в целях экономии хранят все данные в облаке (AWS - лидер всех облачных решений), обезличивают данные хранящиеся там, и понятно обрабатывают данные тоже там, а в своём дата-центре хранят персональную и прочую sensitive информацию о своих клиентах.

Преимущества в цене — это реально огромный плюс AWS по сравнению с затратами на собственными решениями (которые включают еще и обслуживание парка серверов). Это достаточно проверить, просто посмотрев на цены и клиентов облачных решений. Это крупнейшие видеохостинги (Netflix с AWS, например). Это многими используемый Dropbox - он хранит все данные на AWS, а конкретно на Amazon S3 (сервис в рамках AWS по хранению файлов). И понятно, что у них высокие нагрузки и высокая отказоустойчивость т.к в один клик можно разместить "свои" сервера на всех континентах/зонах.

Облачный трафик около года назад достиг трети от общего трафика в сети Интернет. Из этого нетрудно сделать некоторые выводы.

## Личные качества IT-специалистов

Традиционно считается, что "те люди перед монитором" крайне необщительны, закрыты; с ними вообще сложно говорить, а тем более получить необходимую информацию из разговора.

Лично я полностью согласен с этим мнением. Конечно из этого правила есть куча исключений: тут есть в основном две группы:

* IT-специалисты, успешные в жизни и хорошо зарабатывающие (c з/п около 100тыс.$ в год). Тут всё достаточно понятно: им нравится эта работа, они раскрепощены и просто радуются жизнью
* Молодые и нетребовательные IT-специалисты. Они скорее попали в IT случайно и им просто оказалось тут проще существовать и зарабатывать деньги необходимые для жизни.

## Подвижки в IT отрасли

Из-за того, то толковых программистов мало, то всё еще существуют профессии вроде "администратор баз данных", "системный администратор”. ПО-мне облачные технологии должны были уже заменить во многих компаниях подобное, но подобные евангелисты и просто опытные программисты сейчас чаще идут в стартапы, нежели в крупные компании, где люди ищут только виновных и ответственных.

Хотя профессия "администратор баз данных" по статистике становится всё менее и менее популярной. Сказать, что теперь всё настраивают программисты - сказать сложно. Скорее всего просто отдаётся всё на аутсорс какой-то крупной компании. Как и вопросы администрирования. Например, часто покупают RedHat дистрибутив Linux из-за корпоративной поддержки. А вместо сисадмина сидит просто некоторый эникейщик, который и сервер какой-нибудь перезагрузит и принтер заправит.

Количество умных людей именно среди программистов я б сказал между стагнацией и падением. Есть группа людей, движущая всю отрасль, а есть все остальные - лишь бы работало сейчас и ничего больше не интересует.

Вообще часто можно услышать, что логичным продолжением программиста является менеджер. А дальше куда? Это всё или как? Лично меня не очень впечатляет. Многие опытные программисты, побывавшие на конференция Java One думали, что вот оно. Вот же тысячи успешных программистов. Но после бизнес конференции Oracle OpenWorld, многие начинали понимать, что деньги где-то далеки от джавы. Совсем далеко...

Я бы сказал, что сегодня тенденция у опытных программистов - это переход в стартапы - небольшие компании с интересным продуктом и уникальным торговым предложением; командой в 6 человек, без рутины и с интересными задачами, не только техническими. А многие остальные так и остаются на своих многолетних местах синиорами или тимлидами (если вообще есть какая-то адекватная структура в организации/группе), с весьма уже устаревшими знаниями.

# Вместо заключения

Говорить плохо о работниках целой отрасли было бы неправильно.

Сегодня я разделил бы всех IT-шников на несколько групп по тому где они работают:

* **Программисты в стартапах**. Это реально очень смышлёные люди. Связывая программистов и стартапы, я невольно вспоминаю то, что тух новых технологий, бизнес идей, намного шире в MIT чем в классических бизнес-школах вроде Harvard. Эти программисты реально двигают мир вперед. Особенно последние несколько лет — это хорошо заметно. Это люди использующие новые технологии очень быстро после релиза самих технологий. Бизнес и сами программисты крайне гибки в таких условиях. Эти люди не страдают проблемами с общением. Они коммуникабельны.
* **Программисты в крупных компаниях**. Здесь установлены жёсткие бизнес-процессы, как сотрудники обязаны работатать друг с другом, какие технологии использовать. Здесь очень важно, то, что называется "среднее по больнице", например, в такой крупной корпорации, как Microsoft явно есть проблемы с менеджментом. Переписывать несколько раз их мобильную OS Windows Phone странное решение. Так и не могут получить приличную долю рынка. Но в других направлениях получше. Они начали интересоваться компаниями, связанными с их языком C#, открыли исходники компилятора, начали выпускать некоторые opensource проекты. Сложно говорить, как отдельные команды ограничены общекорпоративными рамками, но отдельные подразделения очень даже современные. Не могу не сказать об их дизайне - мне крайне импонируют идеи Windows 8. Люди креативят, но часто упираются в то, что уже база клиентов этой компании не готова к серьёзным переменам. Просто к переменам. Даже благоприятным. Чёткие бизнес-процессы и стабильность не даёт часто революционным идеям выйти на свет. И хорошие программисты часто зачахают попросту тут. Это не программисты в Google и Apple. Это программисты в GM, например.
* **Фрилансеры** - если это работники в какой-то компании, то как правило им разрешают работать удалённо из-за самоконтроля и того, что они вкладываются в свой проект. Если же это фрилансеры на биржах, то это низкооплачиваемые работники, которые как правило делают недорогие landing pages, редактируют что-то немного или делают небольшие проекты. Но в отличии от работников в больших компаниях, где как говориться десятку людей нетрудно затеряться, фрилансеры тут просто находятся в своей низкооплачиваемой нише, что тоже неплохо для рынка.
* **Эникейщики** - это фактически разнорабочие в IT. Это некий человек в компании, работающий со всеми отделами, исправляющих проблемы с их Windows XP и прочее. С помощью методологии google driven development может и что-то на сервера настроить простое.
* **Инженеры**. Это люди решающие фундаментальные проблемы IT. Это фактически математики только с практической направленностью. Сейчас популярно заниматься искусственным интеллектом, нейронными сетями, machine learning, что в общем то одно и тоже. Распознавание изображений и многое другое. В России хорошо заметно выращивание Яндексом таких специалистов, например, ШАД (плюс сотрудничество с ВШЭ в Москве). Сложно найти человека, который смог бы взглянуть с другой стороны на проблему, да еще и за малые деньги. Но Яндекс очень даже неплохо смог создать в сознании студентов, что он хороший работодатель. А технические люди часто не понимают, что они реально делают. Это вообще их главная беда. Мир и деньги - это не двоичный код и кучка процессоров с материнскими платами и множеством сетей между дата-центрами.
* **Программисты-программисты**. Сюда я внёс бы чисто разработчиков в средних IT компаниях. Они, как правило, интересуются отраслью и хорошо знают современные технологии. Иногда есть pet-проджекты.
* **Программисты в силу времени**. Здесь те люди, то любит стабильность. Они как правило ничего не знают из новых технологий, просто что-то делают. В случае увольнения - неизвестность т.к на рынке он никому не нужен.

Собственно говоря, хотелось бы, чтобы программистов из первой категории было больше всего. Я уверен, что смышлёную программисты наконец то начнут уходить из дурдома крупных и средних компаний

Хотя с другой стороны, понятно, что люди не хотят учиться. Это я слышу от многих окружающих: “а, ну, самому изучать что-то, время тратить". Это проблема людей, а не отрасли. Так что я не верю в то, что профессиональных IT специалистов станет так уж много - людей способных на самообучение мало, а сейчас ни один ВУЗ не может подготовить способного IT специалисты т.к программы обучения, где работают чисто теоретики - это программы 15-летней давности, а где старается сотрудничать бизнес, то во-первых сложно найти людей, а во-вторых делают это слишком малыми шагами и не систематично.

Так что я не вижу, что хотя бы половина из существующих программистов в новом 2015 году начнёт использование более эффективных средств, технологий, уйдёт со своего чисто технического мировосприятия и станут другими людьми.

Но всё-таки как последнюю фразу хотелось бы сказать многим специалистам, что они всегда способны изменить свою жизнь и стать, например, одним из разработчиков, self-driving cars в компании Google. Таким я хотел бы видеть программиста в 2015 году, программиста, уходящего от legacy code и смежных с ним технологиями, занимающегося саморазвитием и вносящим новые технологии в реальные consumer решения.