

ИНЖЕНЕРНЫЙ DEVOPS



ИНЖЕНЕРНЫЙ DEVOPS

«Инженерный DevOps» (EngOps) – это управленческие практики и инфраструктура для построения прозрачных процессов командной разработки сложных технических систем.

Термин «DevOps» заимствован из мира IT-проектов. Он связан с идеологией непрерывной связи между процессами разработки подсистем (development) и их развертывания «в проде» для эксплуатации (operations). Для успешной разработки сложной технической системы – **прозрачность** и **автоматизация** – две ключевые идеи, о которых должны думать все участники проекта: программисты, инженеры, тестировщики, администраторы, управленцы. В последние десятилетия IT-компании вырабатывали процессы и методы управления сложностью программных проектов.

Инженерные компании также столкнулись со взрывным ростом применения и сложности алгоритмов в разрабатываемых системах. Современные автомобили, ракеты, самолеты, спутники, роботы и станки стали похожи на продвинутые гаджеты с автономным интеллектом. Например, по количеству строк кода беспилотный электромобиль сопоставим с объемом кода всех сервисов Google или подобных IT-гигантов.

EngOps переосмысливает многолетние практики прозрачности и автоматизации кодоцентричного IT-DevOps применительно к инженерным проектам. При проектировании инженерных систем программный код – это тоже важная, но не единственная часть, поскольку алгоритмы неразрывно связаны с аппаратурой: вычислителем и поведением инженерной системы в окружающей среде. В инженерных компаниях роль моделирования гораздо выше, а цена ошибки из-за **непрозрачности** процессов, **разрозненности** инженерных групп, недостатка **централизации** или **автоматизации** существенно больше.

«ИНЖЕНЕРНЫЙ DEVOPS» ДЛЯ РУКОВОДИТЕЛЯ ИНЖЕНЕРНЫХ КОМАНД

Три столпа инженерного DevOps



Культура разработки

Внедрение «инженерного DevOps» – это не разовый проект, а управляемый вывод культуры командной работы инженеров на новый уровень одновременно с выполнением текущих проектов. Технические и экономические преимущества «инженерных DevOps» подходов заметно улучшают работу команд с самого начала внедрения и постепенно эффект наращивается и закрепляется через такие элементы инфраструктуры, как серверы централизации, автоматизации и инструментирования.

Централизация

Серверы централизации обеспечивают высокий уровень прозрачности как для инженерных руководителей, так и для линейных инженеров. Прозрачность позволяет сотрудникам любого уровня принимать адекватные и своевременные корректирующие решения, избегать тупиков и переделок. Единообразие артефактов, понимание откуда, когда, кем и что введено в проект, определенность типовых процедур (например таких, как сборка проекта) исключает риски срывов испытаний из-за человеческого фактора, например, инженер внес изменения в последний момент, но не смог откатиться на рабочую версию. Сервер централизации также дает понимание темпа работ по проекту, степени его готовности и покрытия тестами, и того, на что реально уходит внимание инженерного коллектива в моменте. Все это в разы повышает адекватность управленческих решений.

Культура разработки

Античное, но актуальное выражение «Ex Nihilo Nihil Fit» – «Из ничего происходит ничего» в обсуждаемом контексте говорит о том, что зачастую движение вперед – это результат внешнего импульса. Серверы сами по себе ничего не дадут, нужна поэтапная трансформация правил работы, которая делается людьми. Повышение культуры прозрачности и автоматизации требует специалистов с опытом «инженерного DevOps», наделения их полномочиями стимулировать коллектив воспринимать новое, преодолевать первичное отторжение. Наиболее результативное и безопасное для коллектива освоение практик EngOps проходит в совместной проектной работе с опытными инженерами ЦИТМ Экспонента. Процесс трансформации культуры и передачи компетенций в зависимости от размера коллектива и разнообразия проектов варьируется по нашему опыту от 9 месяцев до 2–3 лет.

Сервер автоматизаций и инструментирования

Исходя из нашего опыта, без автоматизаций сопротивление коллектива новой культуре разработки будет непреодолимо, а попытка преодоления «сверху» – разрушительна. Повышение качества требует огромных дополнительных трудозатрат, которых ни у одного коллектива в свободном доступе нет. Но мероприятия обеспечения качества могут быть автоматизированы и запускаться без участия разработчика. На любом этапе работы инженера сложные (и монотонные) инженерные процессы автоматизируются, освобождая его ресурсы.

Например, генерация и сборка кода для крупных проектов может занять 4–5 часов, а при наличии сервера инструментирования экономится половина дня инженера на каждой итерации.

Еще пример: автоматический запуск тестовых сценариев, который не только экономит время и внимание разработчика, но и оперативно выявляет ошибки проектирования, пока инженер погружен в задачу, а не спустя неделю, когда потребуются повторное погружение в проект.

Существует ряд проверок и отчетов, о пользе которых инженеры не будут спорить до тех пор, пока их самих не попросят делать эти проверки вручную на ежедневной основе.

Автоматизация экономит не только трудозатраты и моральный дух коллектива, но и прямые издержки на полезные и дорогие инструменты, которые не должны простаивать. Централизация дорогих инструментов на сервере с автоматизацией типовых рутин ускоряет окупаемость вложений в современные средства разработки.

ЗАЧЕМ ИНЖЕНЕРУ ENGOPS?

а. Управляемый творческий... порядок. Мы видели на практике ответственный стенд, на котором для отработки любого сценария надо было заново просмотреть исходные коды на языке Си в несколько тысяч строк. Без этого невозможно было понять, тот ли вариант моделей там, или уже кто-то менял, пока нас у стенда не было. Приходилось тратить дни на отладку и поиск причин неожиданного поведения отлаживаемых подсистем. Потребовалось переделать все, что было отлажено на неправильных моделях объекта и окружения. Удивительно, но это факт: в инженерных компаниях стандартная (для программистов) практика использования централизации артефактов, системы управления конфигурацией и систем контроля версий не используется как должно. Либо этого нет вообще, либо используется только в некоторых отделах или в инициативном порядке отдельными разработчиками. Какой бы ни был крутой и современный проект, но отсутствие в компании хотя бы основ «инженерного DevOps» деморализует. В инфраструктуре EngOps у обычного инженера гораздо больше прозрачности и видения общего проекта, но при этом права на изменения соответствуют области ответственности.

б. Исключение технических конфликтов. Почему эта часть системы перестала работать? Когда, кем или чем была сломана система? При работающей инфраструктуре EngOps эти ответы находятся моментально и точно. Можно посмотреть историю изменений, историю прохождения автоматических тестов после каждого изменения, сравнить и вывести разницу между несколькими версиями моделей, задать конкретный вопрос автору изменений или дать предложение по улучшению, откатить нежелательные изменения, или без общения с коллегами понять на моделях, почему такое изменение было внесено (например уточнение целевых ТТХ системы).

с. Автоматизация тестов. Когда есть готовая инфраструктура автотестов, создание автоматизированных тестовых сценариев перестает казаться лишними трудозатратами. Во-первых, тестовые сценарии запускаются автоматически на удаленном сервере по настраиваемым событиям в проекте (например при фиксации модели в системе контроля версий) и не нужно помнить про их запуск. Во-вторых, автотесты помогают узнать о проблеме сразу же, пока инженер еще погружен в решаемую задачу, а не спустя неделю, когда уже забыл почему именно так решил смоделировать подсистему. В-третьих, обычно исправлять проблему должен тот, кто ее создал, поэтому в интересах разработчика покрыть свою часть автоматическими тестами, чтобы они срабатывали на изменения, вносимые другими инженерами, и давали другим сигнал о необходимости исправления вносимых ими изменений. В-четвертых, когда есть инфраструктура автотестов, то и руководство выделяет время на создание тестовых сценариев, а это снижает уровень стресса при сдаче этапов. Наконец, возможность увидеть сводную информацию о метриках и «здоровье» проекта позволяет снизить потери на коммуникацию с руководством или заказчиком.


д. Автоматизация инженерных и не инженерных рутин. Инфраструктура «инженерного DevOps» позволяет поручить рутину роботам, и это не только автоматизация тестирования (хотя и это очень много), но и создание (обновление) отчетов и документации, автоматическая генерация исходных кодов, сборка проектов, уведомления по событиям в проекте и так далее. Как пример: один наш проект по системам связи на ПЛИС

состоит примерно из четверти миллиона строк кода. Код генерировался из моделей, а затем синтезировался на ПЛИС около четырех часов, в течение которых разработчик ничего не мог сделать на ПК. На финальном этапе приходилось создавать много вариантов прошивок с разными параметрами и каждый раз заново проходить процесс синтеза. Мы увидели существенную выгоду от переноса инженерных рутин на сервер генерации и сборки кода. Там была создана очередь генерации прошивок с возможностью запуска нескольких параллельных процессов. За ночь создавались прошивки, а утром инженер подключал стенд и прогонял автоматические тесты. Это сильно ускорило выявление проблем, которые можно отловить только в железе. Месяцы отладки превратились в дни.

е. Болоту – нет. Когда руководство согласно внедрять современную культуру разработки, выделять в каждом проекте время на создание автоматизаций и изучение современных подходов к разработке, существенно ниже вероятность очутиться в застойном неконкурентном болоте, даже посвятив родному предприятию много лет. Нет смысла сопротивляться и не перенимать успешный многолетний опыт IT-программистов, адаптированный и переосмысленный в ЦИТМ Экспонента для инженерных компаний.

Можно помечтать и продолжить пилить текущие проекты не заточенной пилой или самостоятельно набивать шишки и собирать все подводные камни. А можно посмотреть на конкретные примеры инженерной инфраструктуры DevOps. Лучше один раз увидеть. Напишите нам о желании узнать из первых уст о подробностях работы в инфраструктуре «инженерного DevOps».



 +7 (495) 009 65 85

 <https://exponenta.ru>

 info@exponenta.ru