



## КАК ПРИМЕНЯТЬ RUN-CHANGE-DISRUPT

### С чего начать?

Не надо «резать» компанию на части по уровням — это не торт, так модель работать не будет. **В трехмодальной концепции RUN-CHANGE-DISRUPT речь идет о разных подходах к управлению изменениями в бизнесе.** Это не означает, что какой-то из них хуже, а какой-то лучше: каждый незаменим, так как обладает набором полезных инструментов, уместных в своем контексте. Например, зачем изобретать велосипед, если методы классического водопадного управления хорошо зарекомендовали себя, есть наработанные лучшие практики и компетентные эксперты? Но иногда только актуальные agile-инструменты способны оперативно ликвидировать разрывы на любом уровне организации в зависимости от внешних и внутренних вызовов. Поэтому важно обеспечить максимальную проницаемость между модальностями.

**Лучше начать переход на уровень CHANGE с небольшого проекта,** который будет нести определенную ценность и может быть реализован силами команды в 5–12 человек за 3–6 месяцев. Саймон Мингай, создатель бимодальной IT-концепции, предшественницы трехмодальной, советует: «Вам следует начать, прежде чем вы подумаете, что готовы, будь то создание лаборатории инноваций или адаптация политики гибкого поиска к мелким поставщикам. Не размышляйте слишком долго о том, какие возможности вам понадобятся, и не планируйте слишком много. Абсурдность ситуации заключается в том, что в таком случае вы используете подход 1 [то есть применяете RUN к более высокому уровню. — Примеч. ред.]. Найдите бизнес-спонсоров, готовых сотрудничать с вами, начните и развивайтесь с этого момента. Важно начать».



## Как работает?

	RUN	CHANGE	DISRUPT
ПРОЦЕССЫ	<p>Поддержка действующих отлаженных бизнес-процессов. Работа по принятым нормам, алгоритмам и стандартам. Сверка с планом.</p> <p><b>КЛЮЧЕВОЕ СЛОВО:</b> <b>СТАБИЛЬНОСТЬ</b></p>	<p>Эксперименты, проверка гипотез. Оценка Just-in-Time для каждой итерации agile-команды. Сверка с обстоятельствами</p>	<p>Разрыв шаблонов. Определение перспективы «К чему мы хотим прийти?», а затем выстраивание маршрута в условиях внешней неопределенности с применением практик DevOps. Постоянный мониторинг текущей ситуации. Сверка с реальностью в режиме on-air</p>
ПРОДУКТЫ	<p>Поддержка существующей продуктовой линейки</p>		
ПРОЕКТЫ		<p>Запуск новых проектов на основе анализа трендов рынка.</p> <p><b>КЛЮЧЕВОЕ СЛОВО:</b> <b>ПЕРЕМЕНЫ</b></p>	
ИННОВАЦИОННАЯ ДЕЯТЕЛЬНОСТЬ		<p>Современное реагирование на возникающие инновации. Разработка новых продуктов и технологий</p>	<p>Внедрение прорывных идей и инноваций. Поддержка стартапов, формирование новых бизнес-моделей.</p> <p><b>КЛЮЧЕВОЕ СЛОВО:</b> <b>ИННОВАЦИИ</b></p>



## Когда и как применять?

БУДУЩЕЕ	ХАРАКТЕРИСТИКА	RUN	CHANGE	DISRUPT
<b>ЯСНОЕ</b>	<b>Можем предсказать однозначно и с высокой вероятностью</b> (планирование ресурсов предприятия, управление взаимоотношениями с клиентами)	<b>+</b> <b>Планируем задачу</b>		
<b>АЛЬТЕРНАТИВНОЕ</b>	<b>Можем обозначить варианты развития</b> (проектное управление)		<b>+</b> <b>Определяем горизонт планирования. Сценарный анализ ситуации в зависимости от факторов</b>	
<b>ДИАПАЗОН БУДУЩЕГО</b>	<b>Можем представить диапазон возможных исходов</b>			<b>+</b> <b>Тестируем гипотезы для получения информации</b>
<b>НАСТОЯЩАЯ НЕОПРЕДЕЛЕННОСТЬ</b>	<b>Не обладаем базисом для прогнозирования</b> (непредсказуемые ситуации, вопросы, требующие инновационного подхода)			<b>+</b> <b>Проводим стресс-тестирование</b>



## Как управлять рисками:

ФАКТОР РИСКА	СПОСОБЫ ПРЕОДОЛЕНИЯ
Отсутствие диалога между командами	Все системы должны работать в открытом прозрачном режиме сотрудничества. Это в первую очередь касается согласования приоритетов, распределения ресурсов и обмена информацией
Менталитет «кто не с нами, тот против нас»	Поддерживать равенство между командами разных уровней в плане вознаграждения и признания заслуг
Режим полумер («робкая середина»)	Не стараться снизить риски, применяя традиционные подходы к управлению проектами или к реагированию в хаотической системе (по модели «Кеневин»)
Перекрытый доступ к общим системам	Проследить, чтобы сотрудники на всех уровнях имели открытый доступ к информации и равную возможность с ней работать. Также рекомендуется обновить основные системы, чтобы все группы могли создавать или расширять их в сервис-ориентированной архитектуре (SOA) или через интерфейсы прикладного программирования