Компонентный анализ программного обеспечения

Software Component Analysis (SCA)

ОЦЕНКА КАЧЕСТВА ПРОГРАММНОГО КОДА
ФУНДАМЕНТ БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ

webcontrol

Компании сегодня участвуют в **«гонке инноваций»** или соревновании за потребителя в условиях высокой **неопределенности** бизнес окружения и переменчивости (волатильности) рынка.

И в этой гонке важно иметь возможность вовремя **перестроиться** и **удержать** при этом траекторию движения. Традиционные компании с иерархической системой управления не всегда способны вписать в **крутой поворот**.

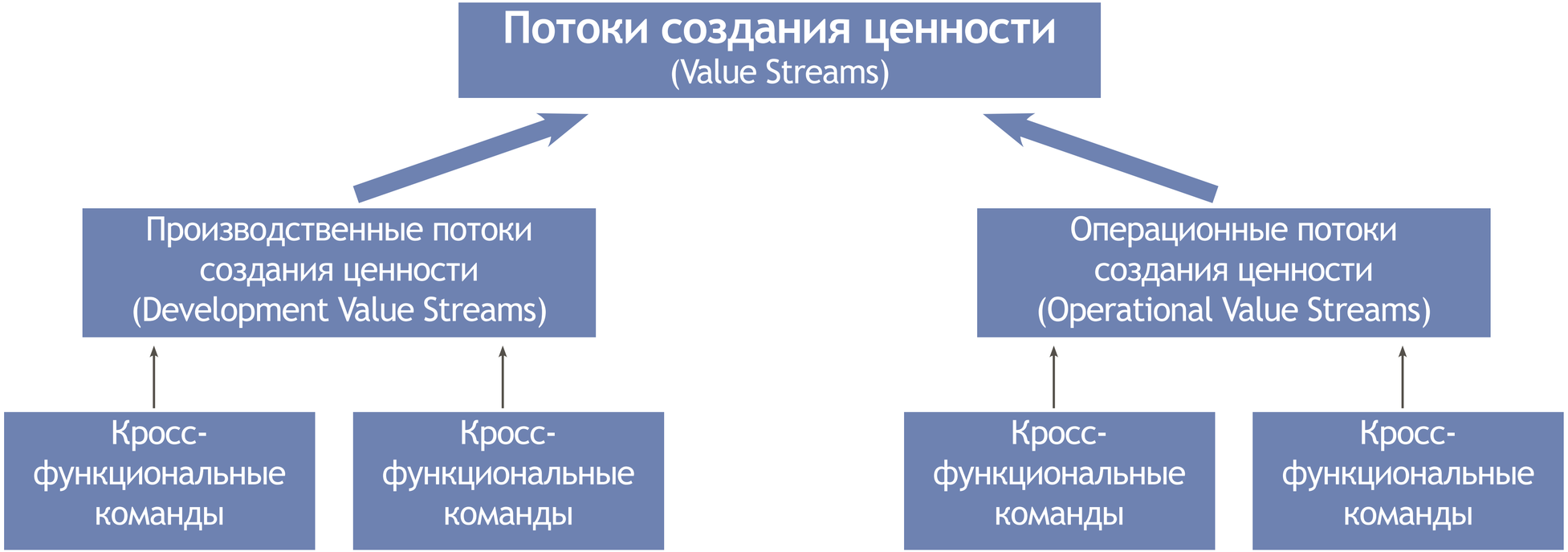
Agile → **Scaled Agile** → **Business Agility**

Business agility – это новый подход к управлению компаний на основе Agile, Lean, DevOps и клиенто-ориентированности, не противоречащий традиционному иерархическому подходу, но **сильно меняющий** ситуацию в компании.

Какое место занимает безопасность в Business Agility?

Business agility предполагает переход от проектной разработки к созданию ИТ-продуктов и организации компании на основе потоков создания ценности

Поток создания ценности - все действия, люди и потоки данных и материалов, необходимые для поставки потребительской ценности.



Множественность команд и ускорение разработки ведет к потере целостности контроля за процессом

Скорость или безопасность?

Успех продукта зависит от

- Скорости его выхода на рынок
- Стоимости его производства
- Качества (включая безопасность)

Ручные инструменты обеспечения безопасности кода

- Неудобны для разработчиков
- Встречают противодействие
- Увеличивают сроки разработки

Безопасность – опция или стандарт?

*Ремень безопасности стал частью **культуры** использования автомобиля только после изобретения динамических трехточечных ремней, которые стали **удобными**.*

Подход **«безопасность по природе»** безболезненно встраивается в конвейер разработки при использовании передовых практик DevOps и инструментов автоматизации

*Использование подхода **«безопасность по природе»** становится частью **культуры** организации, помогает **избежать дополнительных затрат** на безопасность, связанных с задержкой поставки ПО, переделкой кода и устранением уязвимостей.*

«Безопасность по природе»

требует внимания на каждом этапе конвейера DevOps

Continuous Exploration

формирование модели угроз.

Continuous Integration –

SCA / Security IDE плагины
Code review / парная работа
статический и динамический анализ кода
фазинг-тестирование / подписание кода
Сканирование инфраструктуры / антивирус

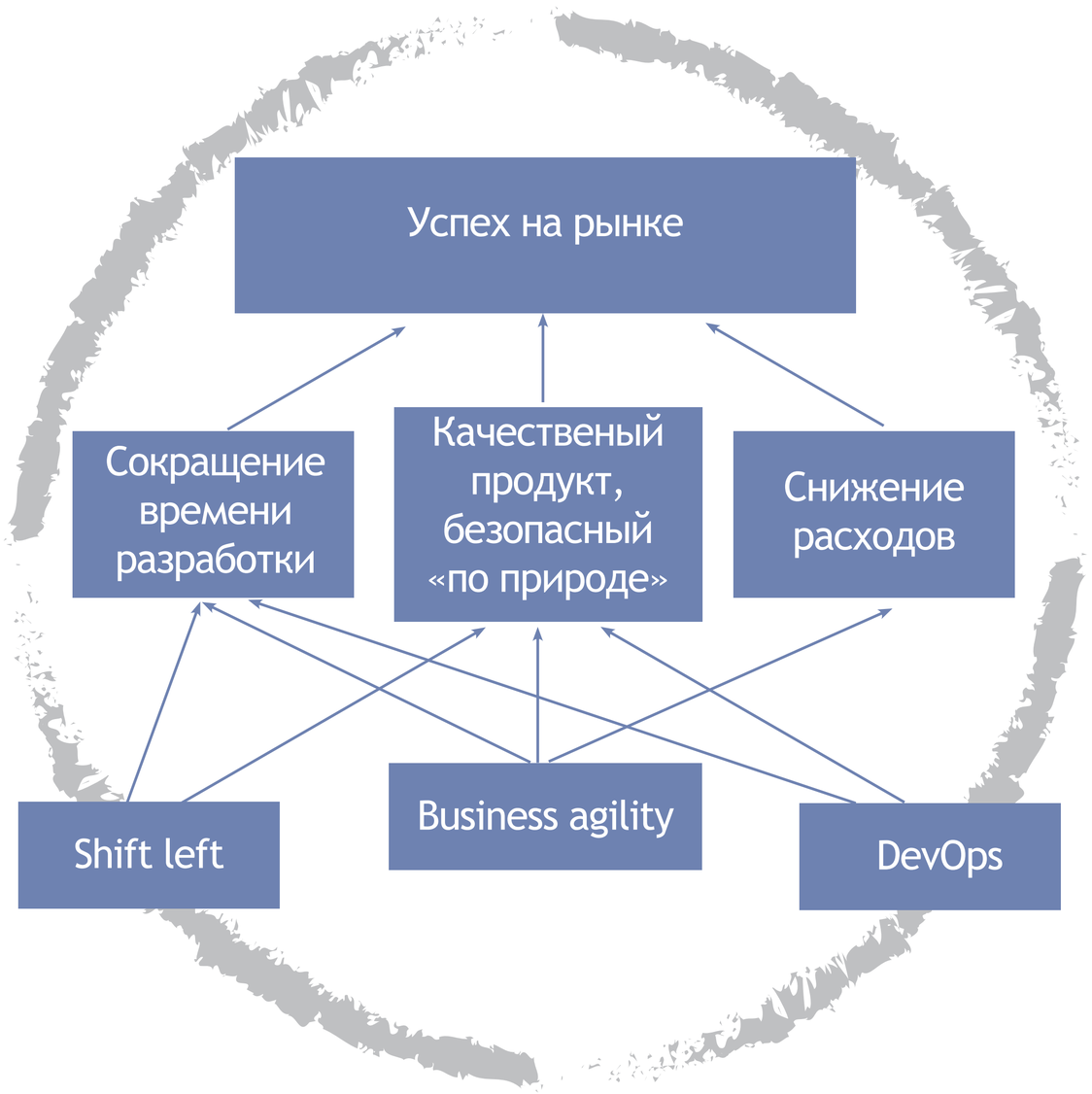
Continuous Deployment

Penetration test

Release on Demand -

SOC / Реагирование на инциденты
Непрерывный мониторинг угроз
Внешний удит уязвимостей

Источник: https://www.scaledagileframework.com/devops/



Для успеха разрабатываемого продукта на рынке необходим качественный продукт, разработанный в сжатые сроки и с минимально необходимыми расходами.

Обеспечение устойчивого развития (Sustainability)

CodeScoring = Гибкий SCA + Качество + Контроль команд

Обеспечивает управление интеллектуальной собственностью компании через автоматическое отслеживание использования программных компонент и оценку качества кода в разрезе команд и потоков создания ценности.

Компонентный анализ и безопасная разработка

CODE SCORING

Всегда максимально слева, работа с кодовой базой.

Компонентный анализ обеспечивает:

- анализ исходных кодов и бинарных сборок
- выявление актуальных уязвимостей
- выявление лицензионного состава
- оценку совместимости лицензий
- анализ переиспользования кода
- политики управления событиями
- интеграцию в жизненный цикл ПО

Компонентный анализ и безопасная разработка

CODE SCORING

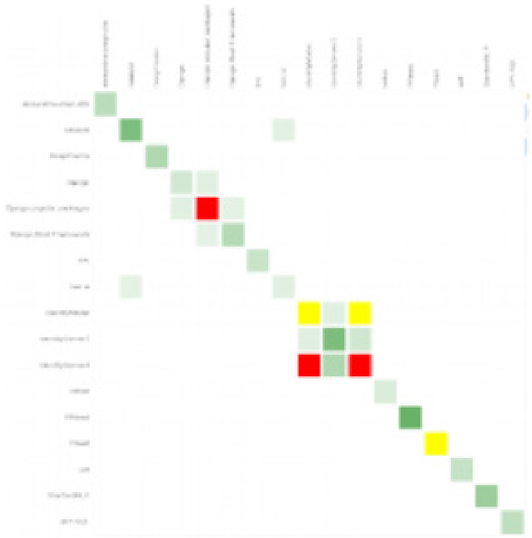
Всегда максимально слева, работа с кодовой базой.

Компонентный анализ обеспечивает:

- анализ исходных кодов и бинарных сборок
- выявление актуальных уязвимостей
- выявление лицензионного состава
- оценку совместимости лицензий
- анализ переиспользования кода
- политики управления событиями
- интеграцию в жизненный цикл ПО

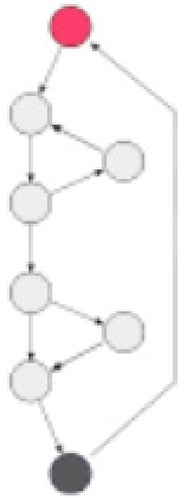
Непрерывная оценка
параметров качества

CODE SCORING



Поиск **дубликатов** как важнеший критерий качества разработки:

Определяется переиспользование кода не только **внутри** проекта,но и **между** проектами, с учетом переименований переменных, методов и классов.



Расчет **цикломатической сложности** дает понимание не только **близости** рефакторинга проекта, но **качества** требований и исполнителей.

Дополнительные параметры качества

Помимо основных характристик качества, оцениваются также следующие параметры:

- уровень комментирования в программных кодах;
- полнота комментариев;
- некачественное форматирование;
- качество именования переменных и функций;
- длинные блоки кода: функции и классы;
- объемы логирования (журналирования);
- выявление нереализованных участков программных кодов (заглушек).

Команда как совокупность авторов компонент

CodeScoring дает прозрачное раскрытие авторского состава анализируемого ПО в разрезе авторства:

- отслеживание истории работы (ретроспективные оценки);
- оценка применяемых технологий;
- отслеживание использования OpenSource
- отслеживание качественных параметров.

Всегда ясно кто автор элемента кода

По отслеживаемым параметрам в **CodeScoring** можно сформировать политику оповещения о событии, например: «Создать задачу в Jira на автора, если в библиотеке которую он добавил и использовал возникла уязвимость»

Отслеживание Команды

Ищем профессиональную работу авторов в **OpenSource** проектах, которые могут являться **каналом утечки** корпоративного кода.

На основании знаний о компетенциях авторов и истории участия авторов в проекте, **CodeScoring** может предложить наиболее близких авторов на **замену** уходящим или при необходимости **расширения команды**, что также облегчает работу руководителя и hr-службы.

Интеграция скоринга кода
в жизненный цикл

Возможные интеграции с системами контроля разработки:
Контроль версий, репозитории, трекинг задач и пр.

CE — CI — CD — RoD

Планирование и адаптация
- контроль технического долга
- анализ состояния команды

Контроль R&D
- лицензионные нарушения;
- качества заимствованого кода

Контроль эксплуатации
- новые уязвимости
- контроль сложность решения

Встраивание процессов обеспечения безопасности в конвейер DevOps и поток создания ценности ValueStream с помощью автоматизированных инструментов для обеспечения безопасности не только не замедляет процесс разработки и не создает неудобств разработчикам, а скорее наоборот - только ускоряет поставку качественных продуктов потребителю и экономит ресурсы.

webcontrol

Андрей Акинин

aakinin@web-control.ru
+7 (495) 925-77-94