# Понятие

Управление конфигурацией – это процесс разработки и применения стандартов и правил по управлению эволюцией программных продуктов. Эволюционирующие системы нуждаются в управлении по той простой причине, что в процессе их эволюции создается несколько версий одних и тех же программ. В эти версии обязательно вносятся некоторые изменения, исправляются ошибки предыдущих версий; кроме того, версии могут адаптироваться к новым аппаратным средствам и операционным системам. При этом в разработке и эксплуатации могут одновременно находиться сразу несколько версий. Поэтому нужно четко отслеживать все вносимые в систему изменения.

Процедуры управления конфигурацией регулируют процессы регистрации и внесения изменений в систему с указанием измененных компонентов, а также способы идентификации различных версий системы. Средства управления конфигурацией применяют для хранения всех версий системных компонентов, для компоновки из этих компонентов системы и для отслеживания поставки заказчикам разных версий системы.

Управление конфигурацией нередко рассматривается как часть общего процесса управления качеством. Поэтому иногда одно и то же лицо может отвечать как за управление качеством, так и за управление конфигурацией. Но обычно разрабатываемая программная система сначала контролируется командой по управлению качеством, которая проверяет ПО на соответствие определенным стандартам качества. Далее ПО передается команде по управлению конфигурацией, которая контролирует изменения, вносимые в систему.

# Цель

Существует много причин, объясняющих наличие разных конфигураций одной и той же системы. Различные версии создаются для разных компьютеров или операционных систем, включающих специальные функции, нужные заказчикам, и т.д. Менеджеры по управлению конфигурацией обязаны следить за различиями между разными версиями, чтобы обеспечить возможность выпуска следующих вариантов системы и своевременную поставку нужных версий соответствующим заказчикам.

# Технология

Процесс управления конфигурацией и связанная с ним документация должны подчиняться определенным стандартам. В качестве примера можно привести стандарт IEEE 828-1983, определяющий составление планов управления конфигурацией. Каждая организация должна иметь справочник, в котором указаны эти стандарты, либо они должны входить в общий справочник стандартов качества. Общенациональные или международные стандарты могут быть также использованы как основа для разработки детализированных специальных норм и стандартов для конкретных организаций. За основу можно взять любой тип стандарта, поскольку все они содержат описания однотипных процессов. Для сертификации качества своих программных продуктов организация должна придерживаться официальных стандартов управления конфигурацией, которые приведены в стандартах ISO 9000 и в модели оценки уровня развития SEI. При традиционной разработке ПО в соответствии с каскадной моделью разрабатываемая система попадает в группу по управлению конфигурацией уже после полного завершения разработки и тестирования ПО. Именно такой подход лежит в основе стандартов управления конфигурацией, которые, в свою очередь, обусловливают необходимость использования для разработки систем моделей, подобных каскадной. Поэтому упомянутые стандарты не в полной мере подходят при использовании таких методов разработки ПО, как эволюционное прототипирование и пошаговая разработка. В этой ситуации некоторые организации изменили подход к управлению конфигурацией, сделав возможным параллельную разработку и тестирование системы. Такой подход основан на регулярной (иногда ежедневной) сборке системы из ее компонентов.

1. Устанавливается время, к которому должна быть завершена поставка компонентов системы (например, к 14.00). Программисты, работающие над новыми версиями компонентов, должны предоставить их к указанному времени. Работу над компонентами не обязательно завершать, достаточно представить основные рабочие функции для проведения тестирования.

2. Создается новая версия системы с новыми компонентами, которые компилируются и связываются в единую систему.

3. После этого система попадает к группе тестирования. В то же время разработчики продолжают работу над компонентами, добавляя новые функции и исправляя ошибки, обнаруженные в ходе предыдущего тестирования.

4. Дефекты, замеченные при тестировании, регистрируются, соответствующий документ пересылается разработчикам. В следующей версии компонента эти дефекты будут учтены и исправлены.

Основным преимуществом ежедневной сборки системы является возможность выявления ошибок во взаимодействиях между компонентами, которые в противном случае могут накапливаться. Более того, ежедневная сборка системы поощряет тщательную проверку компонентов. Разработчики работают под давлением: нельзя прерывать сборку систем и поставлять неисправные версии компонентов. Поэтому программисты неохотно поставляют новые версии компонентов, если они не были предварительно тщательно проверены. Таким образом, на тестирование и исправление ошибок ПО уходит меньше времени.

Для ежедневных сборок системы требуется достаточно строгое управление процессом изменений, позволяющее отслеживать проблемы, которые выявляются и исправляются в ходе тестирования. Кроме того, в результате возникает множество версий компонентов системы, для управления которыми необходимы средства управления конфигурацией.