Конфигурационное управление

П.Н. Советов

2021

Table of Contents

# Введение

## О чем этот курс

“Программная инженерия — это то, что происходит с программированием при добавлении времени и других программистов” (Russ Cox).

Отслеживание и управление изменениями в ПО — важная задача, которая возникает даже при работе над небольшими индивидуальными проектами. В этом курсе предлагается практико-ориентированный взгляд на конфигурационное управление. Изучаемые темы:

* менеджеры пакетов,
* конфигурационные языки,
* системы автоматизации сборки,
* системы контроля версий,
* контейнеризация приложений,
* непрерывная интеграция,
* системы управления проектами и задачами,
* генераторы документации.

Изначально управление конфигурацией применялось не в программировании. Под конфигурацией понимался состав деталей конечного продукта и «взаимное расположение частей» физического изделия. Таким образом, конфигурацией можно управлять, контролируя документы, описывающие конечный продукт, требования к нему, всю его проектную и технологическую документацию.

В связи с высокой динамичностью сферы разработки ПО, в ней конфигурационное управление особенно полезно. К процедурам можно отнести:

* управление версиями проекта,
* управление сборками проекта,
* контроль требований проекта,
* контроль документации
* и т.д.

Степень формальности выполнения данных процедур зависит от размеров проекта, и при правильном подходе данная концепция может быть очень полезна.

## Основные определения

**Конфигурационное управление** — дисциплина идентификации компонентов системы, определения функциональных и физических характеристик аппаратного и программного обеспечения для проведения контроля внесения изменений и отследживания конфигурации на протяжении ЖЦ, см. рис. 1. Это управление соответствует одному из вспомогательных процессов ЖЦ (ISO/IEC 12207), выполняется техническим и административным руководством проекта и заключается в контроле указанных характеристик конфигурации системы и их изменении; составления отчета о внесенных изменениях в конфигурацию и статус их реализации; проверки соответствия внесенных изменений заданным требованиям.



Рис. 1. Цели и задачи конфигурационного управления

**Конфигурация системы** — состав функций, программных и физических характеристик программ или их комбинаций, аппаратного обеспечения, обозначенные в технической документации системы и реализованные в продукте.

**Элемент программной конфигурации** — фрагмент программного обеспечения, вовлеченный в процесс конфигурационного управления и рассматриваемый как одна (атомарная) сущность.

К элементам конфигурации относятся такие объекты, как:

* рабочие документы;
* файлы исходных кодов;
* файлы ресурсов;
* файлы, создаваемые в результате сборки (исполняемые файлы, библиотеки и так далее);
* инструменты, используемые для разработки (их мы тоже должны учитывать для стандартизации и упрощения взаимодействия в команде);

**Версия** — это состояние элемента программной конфигурации, которое может быть восстановлено в любой момент времени независимо от истории изменения.

Элементы системы, их версии и конфигурации показаны на рис. 2.

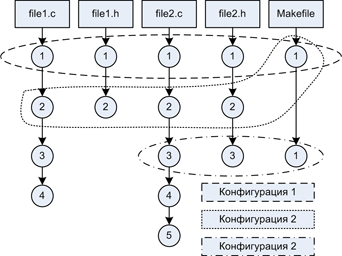


Рис. 2. Элементы, их версии и срезы-конфигурации

Конфигурация ПО включает набор функциональных и физических характеристик ПО, заданных в технической документации и достигнутых в готовом продукте. Т.е это сочетание разных элементов продукта вместе с заданными процедурами сборки и отвечающие определенному назначению. Элемент конфигурации — график разработки, проектная документация, исходный и исполняемый код, библиотека компонентов, инструкции по установке системы и др.

Область знаний «Управление конфигурацией ПО» состоит из следующих разделов:

* управление процессом конфигурацией (Management of SMC Process),
* идентификация конфигурации ПО (Software Configuration Identification),
* контроль конфигурации ПО (Software Configuration Control),
* учет статуса конфигурации ПО (Software Configuration Status Accounting),
* аудит конфигурации ПО (Software Configuration Auditing),
* управление релизами (версиями) ПО и доставкой (Software Release Management and Delivery).

Управление процессом конфигурации — это деятельность по контролю эволюции и целостности продукта при идентификации, контроле изменений и обеспечении отчетности информации, касающейся конфигурации. Включает:

* систематическое отслеживание вносимых изменений в отдельные составные части конфигурации и проведение аудита изменений и автоматизированного контроля за внесением изменений в конфигурацию системы или ПО;
* поддержка целостности конфигурации, ее аудит и обеспечение внесения изменений в один объект конфигурации, а также в связанный с ним другой объект;
* ревизия конфигурации на предмет проверки разработки необходимых программных или аппаратных элементов и согласованности версии конфигурации с требованиями;
* трассировка изменений в конфигурацию на этапах сопровождения и эксплуатации ПО.

**Идентификация конфигурации ПО** проводится путем выбора элемента конфигурации ПО и документирования его функциональных и физических характеристик, а также оформления технической документация на элементы конфигурации ПО.

**Контроль конфигурации ПО** состоит в проведении работ по координации, утверждению или отбрасыванию реализованных изменений в элементы конфигурации после формальной ее идентификации, а также оценке результатов.

**Учет статуса конфигурации ПО** проводится в виде комплекса мероприятий для определения уровня изменений в конфигурацию, аудита конфигурации в виде комплекса мероприятий по проверке правильности внесения изменений в конфигурацию ПО. Информация и количественные показатели накапливается в соответствующей БД и используются при управлении конфигурацией, составлении отчетности, оценке качества и выполнении других процессов ЖЦ.

**Аудит конфигурации** — это деятельность, которая выполняется для оценки продукта и процессов на соответствие стандартам, инструкциям, планам и процедурам. Аудит определяет степень удовлетворения элемента конфигурации заданным функциональным и физическим характеристикам системы. Различают функциональный и физический аудит конфигурации, который завершается фиксацией базовой линии (базиса) продукта. Сборка ПО — объединение корректных элементов ПО и конфигурационных данных в единую исполняемую программу.

Процедуру сборки проекта часто автоматизируют, то есть выполняют не из среды разработки, а из специального скрипта — **build-скрипта**. Этот скрипт используется тогда, когда разработчику требуется полная сборка всего проекта. А также он используется в процедуре непрерывной интеграции (continuous integration) — то есть регулярной сборке всего проекта (как правило — каждую ночь). Во многих случаях процедура непрерывной интеграции включает в себя и регрессионное тестирование, и часто — создание инсталяционных пакетов. Общая схема автоматизированной сборки представлена на рис. 3.

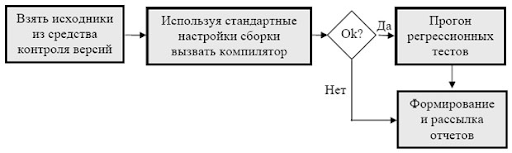


Рис. 3. Схема автоматизированной сборки ПО

**Стабилизация конфигурации** — это процесс получения новой конфигурации из имеющихся промежуточных конфигураций. Для этого процесса также используются также термины «выпуск» или «релиз». Результат стабилизации также может быть назван, в свою очередь, релизом или выпуском.

**Управление релизами (версиями) ПО** это: отслеживание имеющейся версии элемента конфигурации; сборка компонентов; создание новых версий системы на основе существующей путем внесения изменений в конфигурацию; согласование версии продукта с требованиями и проведенными изменениями на этапах ЖЦ; обеспечение оперативного доступа к информации относительно элементов конфигурации и системы, к которым они относятся. Управление выпуском охватывает идентификацию, упаковку и передачу элементов продукта и документации заказчику.

**Базис (baseline)** — формально обозначенный набор элементов ПО, зафиксированный на этапах ЖЦ ПО. Это конфигурация, выбранная и закрепленная на любом этапе жизненного цикла разработки как основа для дальнейшей работы.

Hа рис. 4 показан пример появления конфигураций во времени.

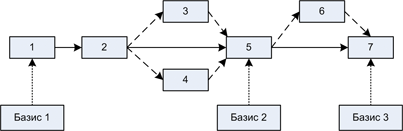


Рис. 4. Пример появления конфигураций во времени

Начальное состояние проекта — конфигурация 1. Она же является первым базисом, от которого будет идти дальнейшая разработка. Предположим, проект на начальной стадии. Через какое-то время появляется обновленная конфигурация 2. Разработка только началась и был выпущен релиз, чтобы дать команде какую-то основу для дальнейшей работы. В ходе проверки выяснилось, что базой для работы этот выпуск служить не может — есть непонятные и противоречивые места.

Для их устранения группы разработки делают доработки. В результате них появляются конфигурации 3 и 4 — обе они разработаны на основе 2, но друг с другом они не согласуются, поскольку не включают изменения друг от друга. CM-инженер создает конфигурацию 5, сделанную на основе 2, 3 и 4. После проверки менеджмент объявляет конфигурацию базовой. По этому сигналу CM-команда выпускает этот релиз как официальную базовую конфигурацию и разработчики берут уже ее за основу. Далее история повторяется, группа разработки вносит изменения — появляется конфигурация 5. Ее, в свою очередь, интегрирует CM-инженер и она получает номер 7. Он также становится официальной базой для разработки.

# Работа в командной строке

Текст.

# Менеджеры пакетов

Текст.

# Конфигурационные языки

Текст.

# Системы автоматизации сборки

Текст.

# Системы контроля версий

Текст.

# Генераторы документации

Текст.

# Виртуализация и контейнеризация

Текст.