

# **Курс лекций по дисциплине «Программная инженерия»**

## **Лекция 6. Конфигурационное управление**

Ст. преподаватель кафедры ИС  
Важдаев А.Н.

# Проблема

- В программных проектах необходима специальная деятельность по поддержанию файловых активов проекта в порядке. Она называется **конфигурационным управлением**. Выделяют две основные задачи в конфигурационном управлении – **управление версиями** и **управление сборками**.
- **Управление версиями** отвечает за управление версиями файлов и выполняется в проекте на основе специальных программных пакетов – **средств версионного контроля**.
- **Управление сборками** – это автоматизированный процесс трансформации исходных текстов ПО в пакет исполняемых модулей, учитывающий многочисленные настройки проекта, настройки компиляции, и интегрируемый с процессом автоматического тестирования.

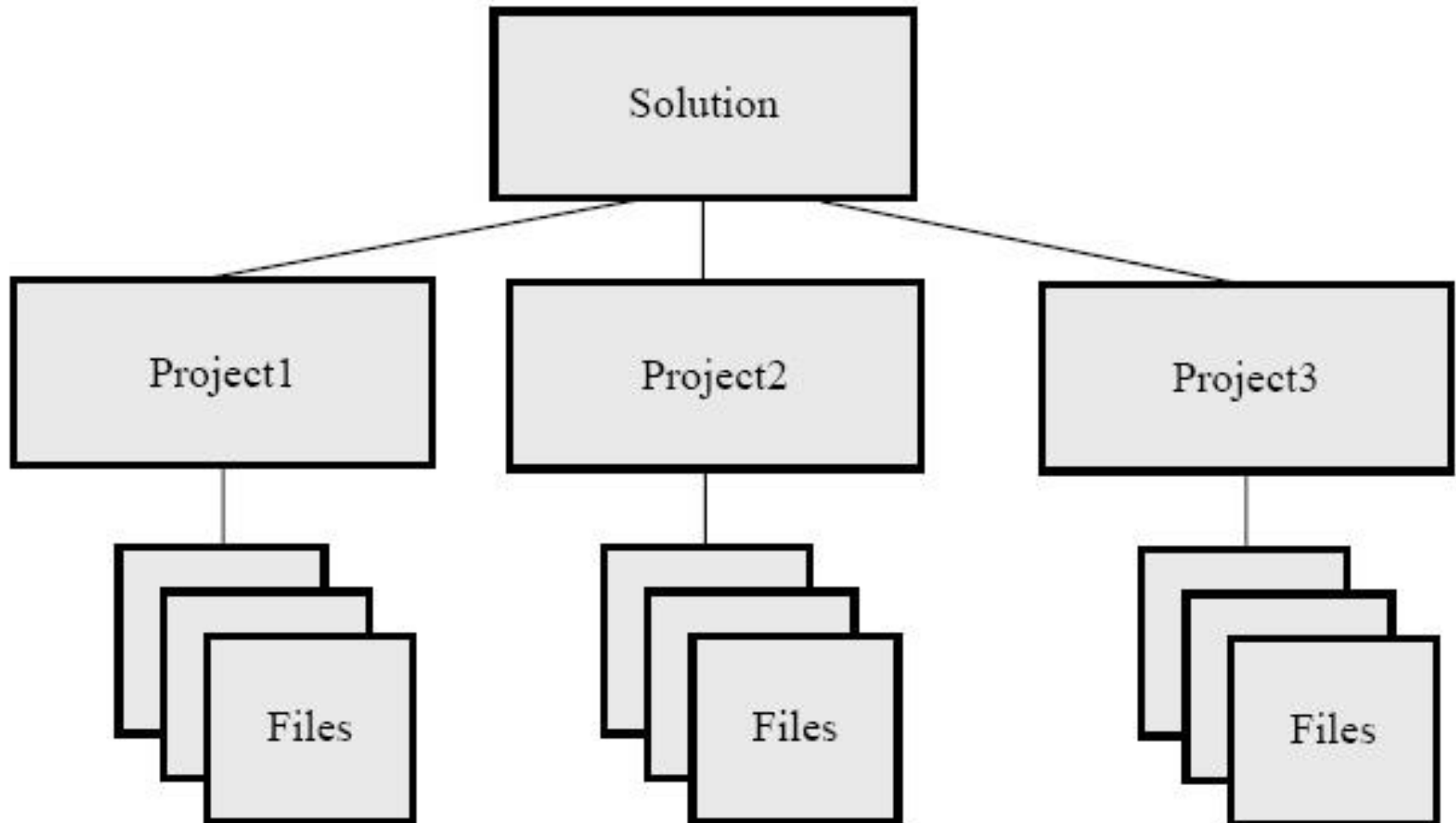
# Единицы конфигурационного управления

- Конфигурационное управление имеет дело с меняющимися в процессе продуктами, состоящими из наборов файлов. Такие продукты принято называть единицами конфигурационного управления (configuration management items).
- Примеры единиц конфигурационного управления:
  - 1. пользовательская документация;
  - 2. проектная документация;
  - 3. исходные тексты ПО;
  - 4. пакеты тестов;
  - 5. инсталляционные пакеты ПО;
  - 6. тестовые отчеты.

# Единицы конфигурационного управления

- Структура – набор файлов.
- Ответственное лицо и, возможно, группу тех, кто их разрабатывает, а также более широкую и менее ответственную группу тех, кто пользуется этой информацией.
- Практика конфигурационного управления – кто и в каком режиме, а также в какое место выкладывает новую версию элемента конфигурационного управления в средство управления версиями, правила именования и комментирования элемента в этой версии, дальнейшие манипуляции с ним.
- Автоматическая процедура контроля целостности элемента.

# Единицы конфигурационного управления



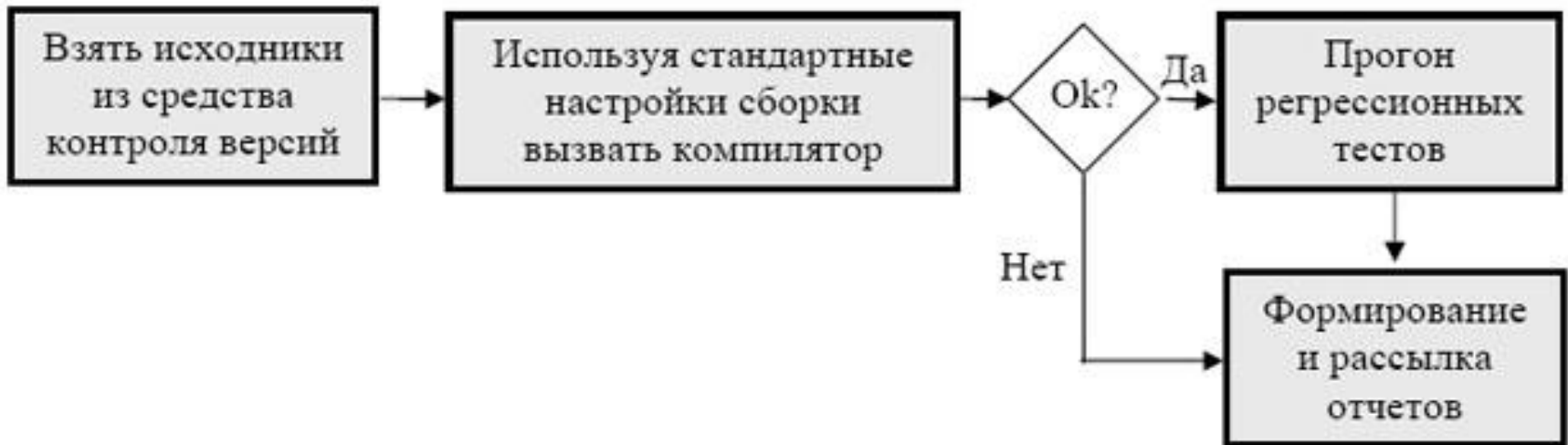
# Управление версиями

- **Управление версиями файлов.** Поскольку программисты имеют дело с огромным количеством файлов, многие файлы в один момент могут быть необходимы нескольким людям и важно, чтобы все они постоянно составляли единую, как минимум, компилируемую версию продукта, необходимо, чтобы была налажена работа с файлами с исходным кодом.
- **Управление версиями составных конфигурационных объектов.** Понятие "ветки" проекта. Одновременно может существовать несколько версий системы – и в смысле для разных заказчиков и пр. (так сказать, в большом, настоящем смысле), и в смысле одного проекта, одного заказчика, но как разный набор исходных текстов.

# Управление версиями

- V1.0 – ветвь, соответствующая выпущенному релизу.
- Fix V1.0.1 – ветвь, соответствующая выпущенному пакету исправлений к определенной версии.
- Upcoming (V1.1) – ветвь, соответствующая релизу, готовящемуся к выпуску и находящемуся в стадии стабилизации.
- Mainline – ветвь, соответствующая основному направлению развития проекта. По мере созревания именно от этой ветви отходят ветви готовящихся релизов.
- WCF Experiment – ветвь, созданная для проверки некоторого технического решения, перехода на новую технологию, или внесения большого пакета изменений, потенциально нарушающих работоспособность кода на длительное время.

# Управление сборками





# Понятие baseline

- Baseline – это базовая, последняя целостная версия некоторого продукта разработки, например, документации, программного кода и т.д.  
Подразумевается, что разработка идет не сплошным потоком, а с фиксацией промежуточных результатов в виде текущей официальной версии разрабатываемого актива. Принятие такой версии сопровождается дополнительными действиями по оформлению, сглаживанию, тестированию, включению только законченных фрагментов и т.д.
- Baseline служит хорошим средством синхронизации групповой работы.
- Baseline может быть совсем простой – веткой в средстве управления версиями, где разработчики хранят текущую версию своих исходных кодов.

# Понятие baseline

