КРПС

Управление изменениями и конфигурациями

Лекция №5

(версия 1.0)

на основе SWEBOK

Конфигурация

Система может быть определена как *коллекция компонент, организованных для выполнения заданных функций или реализации комплекса функциональности* (IEEE 610.12-90, Standard Glossary for Software Engineering Terminology).

Конфигурация системы – функциональные и/или физические характеристики аппаратного, программно-аппаратного, программного обеспечения или их комбинации, сформулированные в технической документации и реализованные в продукте. Конфигурация также может восприниматься как сочетание конкретных версий аппаратных, программно-аппаратных или программных элементов, объединенных вместе, в соответствии с заданными процедурами сборки и отвечающих определенному назначению.

Управление конфигурацией

Конфигурационное управление (CM - Configuration Management) - дисциплина идентификации конфигурации системы в определенные (заданные) моменты времени, с целью систематического контроля изменений конфигурации, а также поддержки и сопровождения целостной и отслеживаемой (трассируемой) конфигурации на протяжении всего жизненного цикла системы.

Управление конфигурацией

Конфигурационное управление формально определяется глоссарием IEEE 610 как “дисциплина приложения технических и административных указаний (инструкций) и контроля (надзора) для: идентификации и документирования функциональных и физических характеристик элементов конфигураций, контроля (управления) изменений этих характеристик, записи (сохранения) и ведения отчетности по обработке изменений и статусу их реализации, а также проверки (верификации) соответствия заданным требованиям.”

SCM

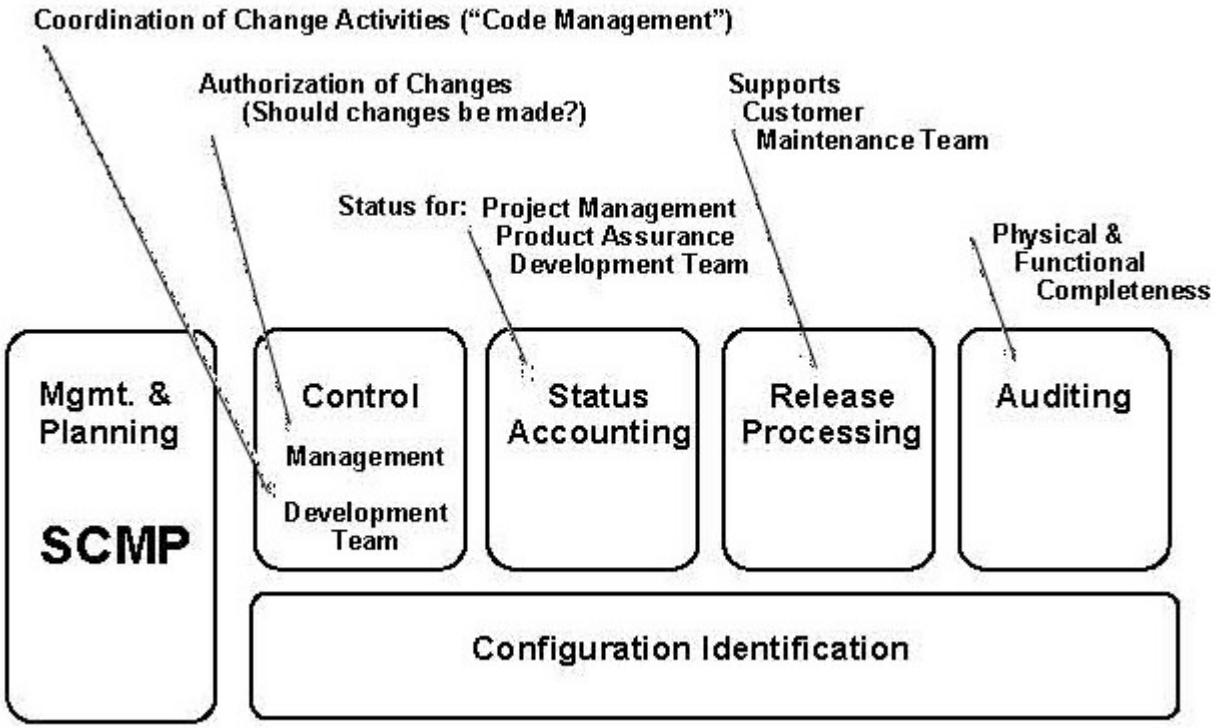
Конфигурационное управление (англ. software configuration management, SCM) в программной инженерии — *комплекс методов, направленных на систематический учёт изменений, вносимых разработчиками в программный продукт в процессе его разработки и сопровождения, сохранение целостности системы после изменений, предотвращение нежелательных и непредсказуемых эффектов, формализацию процесса внесения изменений.*

SCCM

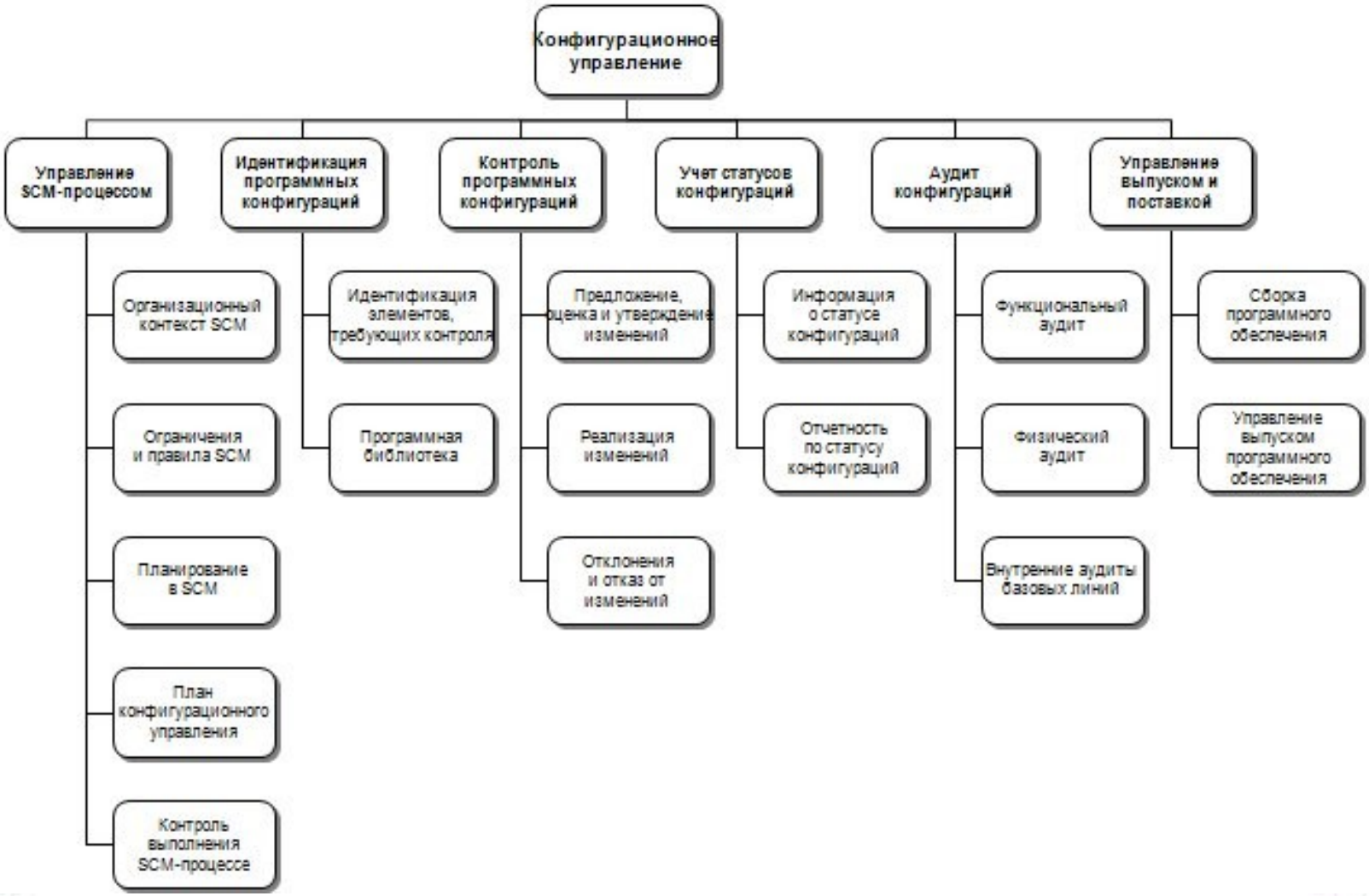
В ряде источников можно увидеть аббревиатуру SCCM – Software Configuration and Change Management. При том, что в понимании SWEBOK и соответствующих стандартов, содержание SCM и SCCM тождественно, термин SCCM иногда используется для того, чтобы подчеркнуть принципиальную значимость управления изменениями как составной части конфигурационного управления.

Виды работ

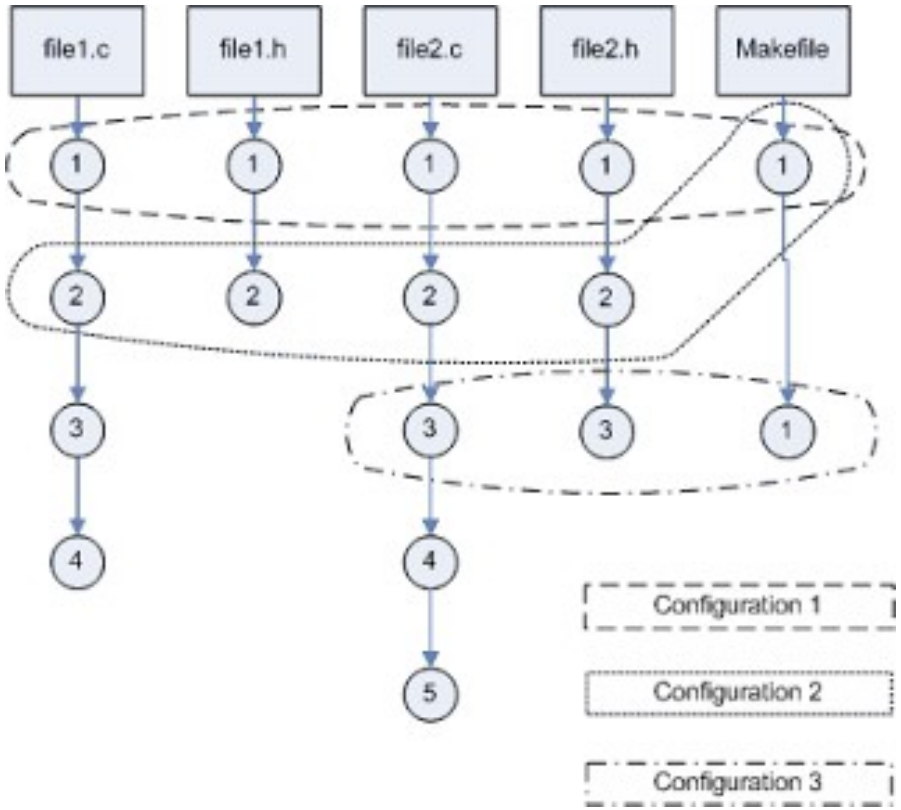
Работы по конфигурационному управлению <программного обеспечения> включают: управление и планирование SCM-процессов, идентификацию программных конфигураций, контроль конфигураций, учет статусов конфигураций, аудит, а также управление выпуском (release management) и поставкой (delivery).



Область знаний



Упрощенное представление



Минимальная конфигурация ПС

1. Системная спецификация
2. План программного проекта
3. Спецификация требований к ПС
4. Предварительное руководство пользователя
5. Спецификация проектирования
6. Листинги исходного кода
7. План и методика тестирования, тестовые варианты
8. Руководства по работе и установке
9. Исполняемый код
10. Описание БД
11. Руководство пользователя по настройке
12. Документы сопровождения, отчеты о проблемах ПС
13. Стандарты и методики конструирования ПС

ГОСТ Р ИСО/МЭК 12207

В соответствии с этим стандартом, процесс состоит из следующих работ:

1) подготовка процесса;

2) определение конфигурации;

3) контроль конфигурации;

4) учет состояний конфигурации;

5) оценка конфигурации;

6) управление выпуском и поставка.

Подготовка процесса

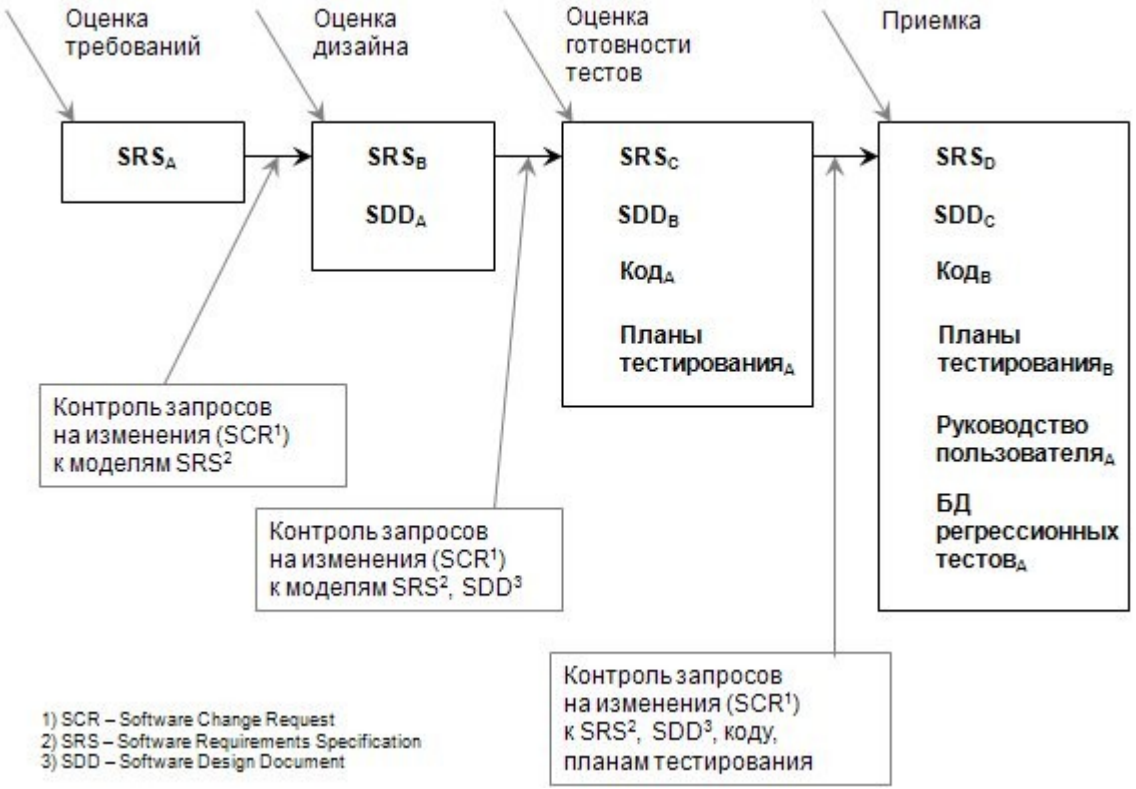
Должен быть разработан план управления конфигурацией. План должен определять: работы по управлению конфигурацией; процедуры и график выполнения данных работ; организацию(и), ответственную(ые) за выполнение данных работ; связь данной организации(й) с другими организациями, например, по разработке и сопровождению программных средств. План должен быть документально оформлен и выполнен (план может быть частью плана управления конфигурацией системы).

Определение конфигурации

Должна быть определена схема обозначения программных объектов и их версий (объектов программной конфигурации), которые контролируются при реализации проекта. Для каждого программного объекта и его версий должны быть определены: документация, в которой фиксируется состояние его конфигурации; эталонные версии и другие элементы обозначения.

Item, configuration, configuration item, relationships, version, baseline

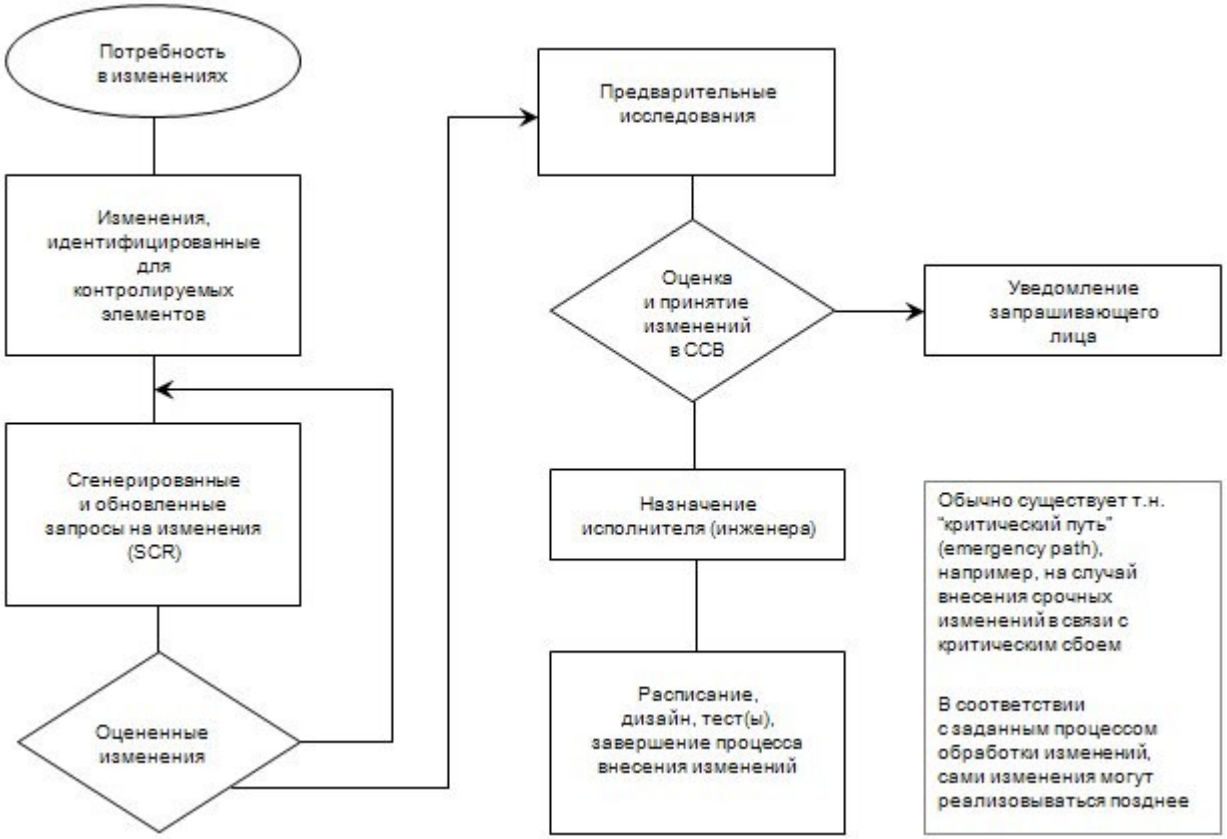
Включение элемента в конфигурацию



Контроль конфигурации

Анализ и оценка изменений; принятие или непринятие заявки; реализация, верификация и выпуск измененного программного объекта. Для каждого изменения должны отслеживаться проводимые аудиторские проверки, посредством которых анализируется каждое изменение, его причина и разрешение на его внесение. Должны быть выполнены контроль и аудиторская проверка всех доступных контролю программных объектов, которые связаны с критическими функциями безопасности или защиты.

Поток процесса контроля изменений



Учет состояний конфигурации

Должны быть подготовлены протоколы управления и отчеты о состоянии, которые отражают состояние и хронологию изменения контролируемых программных объектов, включая состояние их конфигурации. Отчеты о состоянии должны включать количество изменений в данном проекте, последние версии программных объектов, обозначения выпущенных версий, количество выпусков и сравнения программных объектов различных выпусков.

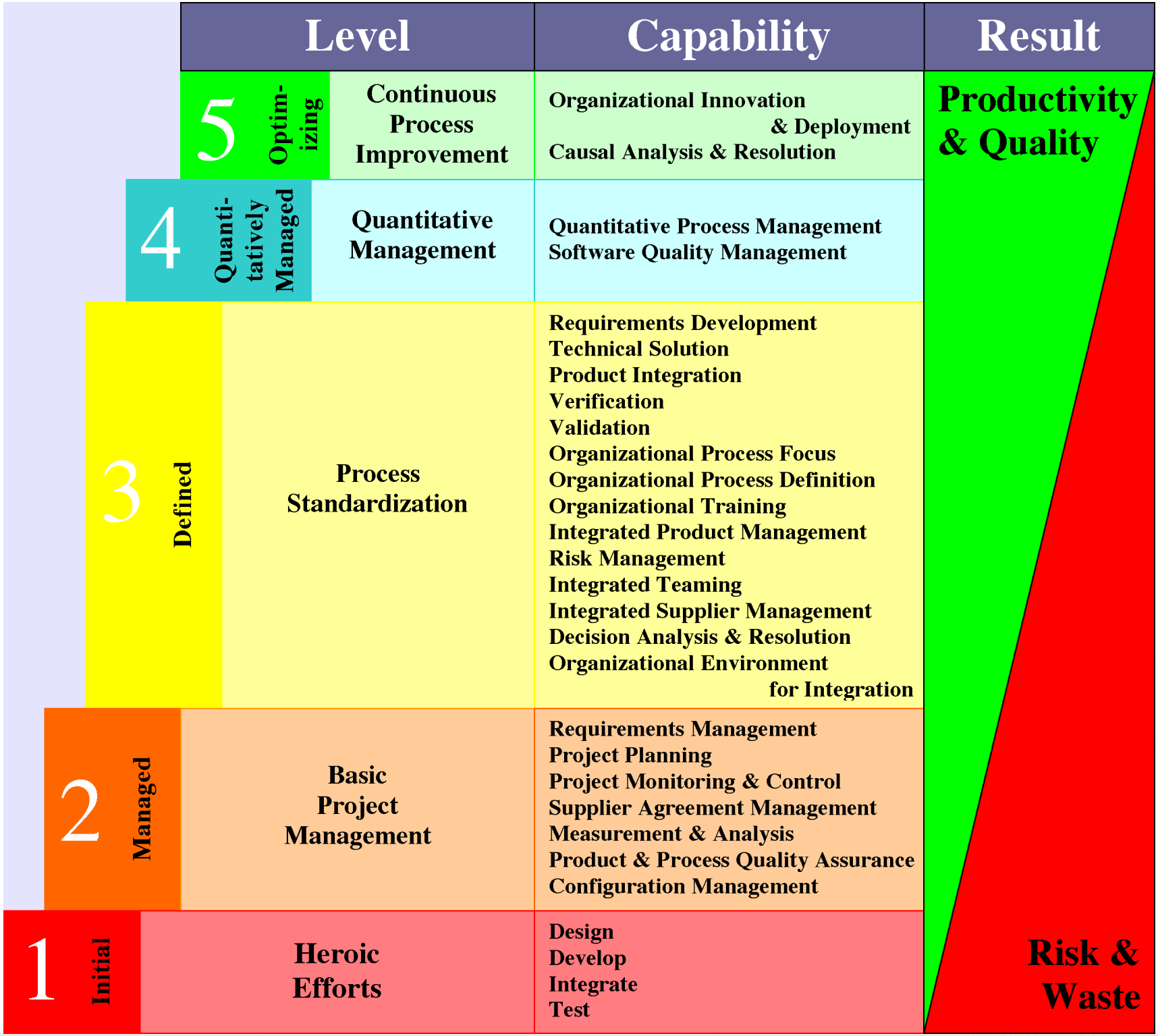
Оценка конфигурации

Должны быть определены и обеспечены: функциональная законченность программных объектов с точки зрения реализации установленных к ним требований; физическая завершенность программных объектов с точки зрения реализации в проекте и программах всех внесенных изменений.

Управление выпуском и поставка

Должны официально контролироваться выпуск и поставка программных продуктов вместе с соответствующей документацией. Оригиналы программ и документации должны сопровождаться в жизненном цикле. Программы и документация, связанные с обеспечением критических функций безопасности или защиты, должны обрабатываться, храниться, упаковываться и поставляться в соответствии с установленными правилами.

CMM



Факторы, влияющие на УК

Фактор	Возможные значения	Воздействие, описание
Тип проекта	Разработка модели (прототипа) Проект сопровождения ПС Коммерческий (с сопровождением) Коммерческий без сопровождения Субподрядный	
Наличие нескольких офисов (регионально распределенная разработка)	Один офис Более одного	Наличие нескольких офисов усложняет план, дополняя его регламентами взаимодействия между офисами. Также дополнительные офисы влияют на общую архитектуру проекта. На такие ключевые факторы как количество ответвлений на проектном дереве (как правило, добавление нового региона, приводит к добавлению минимум одной ветви для каждого региона). Увеличение числа регионов воздействует на уровень формализма плана. Уровень – высокий.
Относительный размер проекта	Малый Средний Большой	Воздействует на количество регламентов и их проработанность и детальность. Фазы, взаимодействие между группами, прохождение запросов на изменения описываются более детально. Чем больше проект, тем более формализованным должен быть план.

Факторы, влияющие на УК

Количество конфигурационных элементов		Число конфигурационных элементов влияет только на более глубокую проработку идентификации элементов. В некоторых случаях полезно определить в плане все типы конфигурационных элементов на основании шаблонов (например, по расширениям файлов)
Количество компонентов и подсистем		Число компонентов и подсистем могут влиять на выборку элементов из репозитория (способ выборки и обращения). Также влияет на глубину изложения раздела, описывающего структуру проектного каталога
Фаза жизненного цикла		План УК обычно описывает все фазы жизненного цикла ПС. Иногда при работе с субподрядчиками бывает необходимо более четко выделить фазу, на которой подключается субподрядная организация. Также к плану УК может выпускаться дополнение, отражающее фазу жизненного цикла ПС.
Модель разработки		В зависимости от того какая модель разработки принята за основу (каскад, итерации, спираль), необходимо откорректировать план УК в части состава фаз ЖЦ ПС, глубины их описания, способа идентификации базовых версий, выпуска релизов.

Факторы, влияющие на УК

Доступность (наличие) средств УК и иных смежных средств	**Базовые** Основные системы УК (как правило, только отслеживание версий) Генераторы отчетов (обычно встроенные) Средства управления библиотеками	Проект может строиться вообще без средств автоматизации (например, управление конфигурацией сборки макета печатной платы). На ход проекта и на план оказывают существенное воздействие такие факторы как используемые средства разработки, платформа разработки (возможно разработка на нескольких платформах и для нескольких платформ одновременно). Также большое значение имеют тип и количество средств реализации (автоматизации УК), их принадлежность одному или нескольким вендорам. Например, в проекте можно использовать средство управления версиями от одного производителя, а средство управления изменениями от другого. Можно иметь интеграцию средства управления со средствами управления проектами а можно и не иметь. Тип интеграции между средствами, архитектура интеграции должны быть детально рассмотрены в плане.
	Продвинутые, интегрированные Тоже что и выше. Плюс средства управления изменениями Встроенные средства сборки и аудита	
	Разрозненные	

Факторы, влияющие на УК

Уровень формализации (как процессов организации, так и тип контроля плана)	Высокий Средний Низкий	Уровень формализации можно варьировать в зависимости от многих факторов, в том числе отраженных в данной таблице. Выбирая уровень формальности и глубины изложения необходимо руководствоваться исходящими задачами и целями. Такие факторы, как сложность проекта, региональная разбросанность, тип проекта, наличие субподрядчиков должны автоматически подвигнуть к написанию высоко формализованного плана УК. Средний и низкий уровень может применяться в относительно краткосрочных проектах, проектах, в которых задействовано небольшое количество разработчиков. С ростом команды, разделением ролей план УК должен быть пересмотрен, уровень формализации поднят.

Источники

- SWEBOK (http://swebok.sorlik.ru/6_software_configuration_management.html)
- http://citforum.ru/SE/quality/configuration_management/
- Зачем нам нужен план управления конфигурациями? Основные понятия и концепции документа http://cmcons.com/articles/CC_CQ/paln_cm/
- Орлов С.А. «Технологии разработки программного обеспечения»