

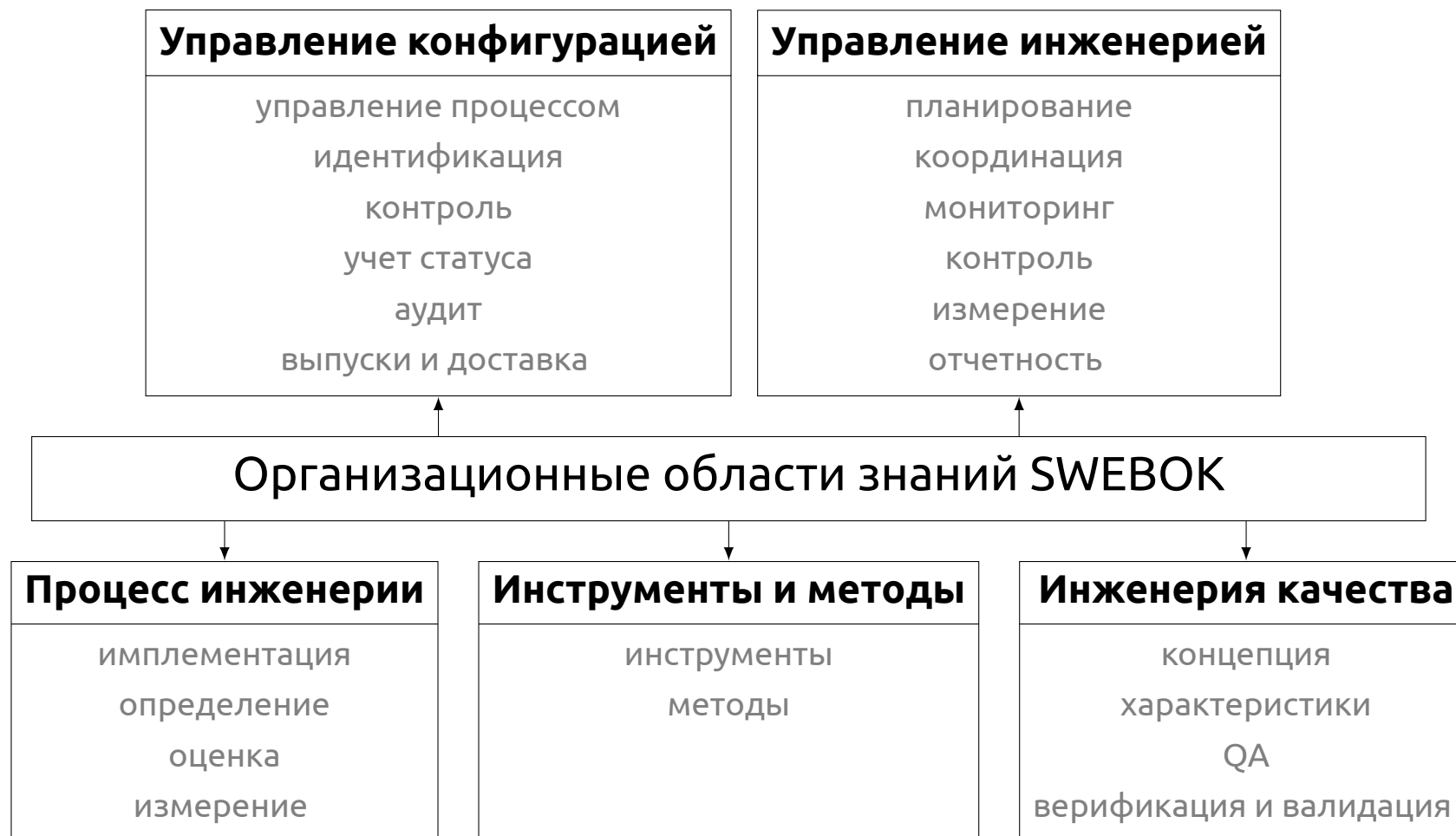
SWEBOOK. Организационные области знаний

Алексей Островский

Физико-технический учебно-научный центр НАН Украины

10 октября 2014 г.

Организационные области знаний SWEBOOK



Управление конфигурацией ПО

Определение

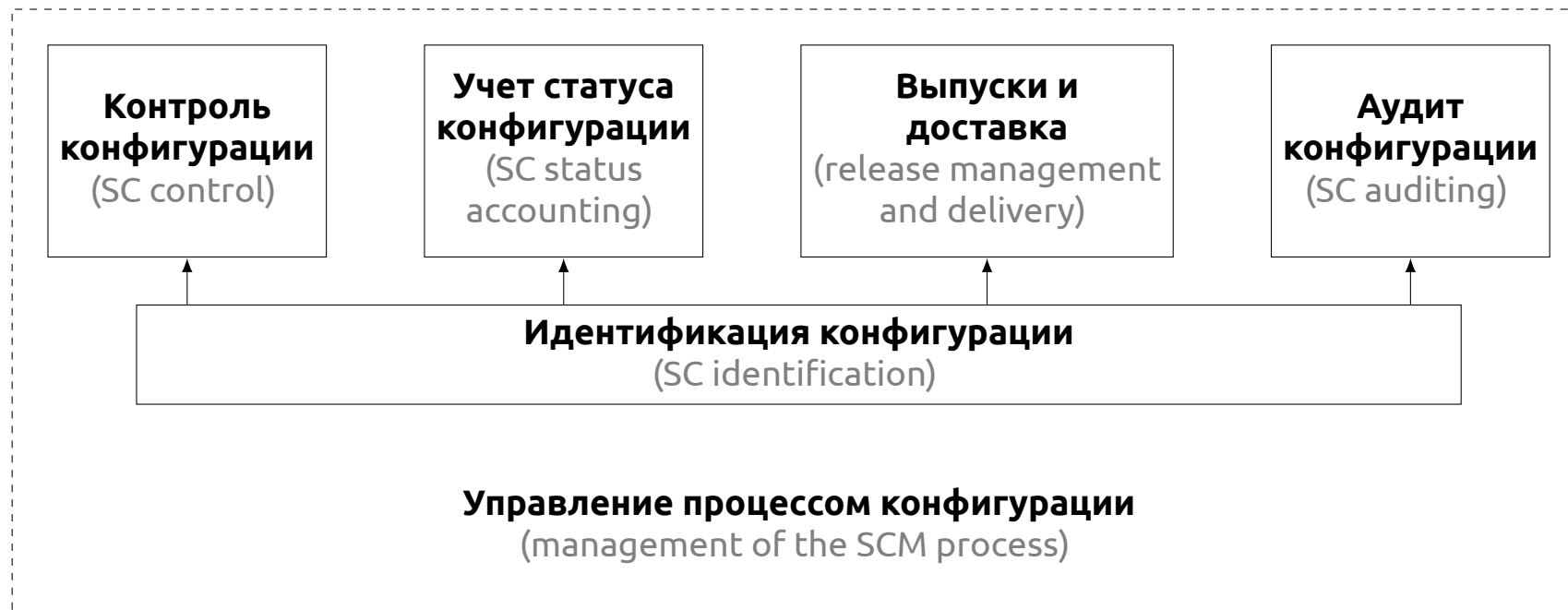
Конфигурация — совокупность функциональных и/или физических характеристик оборудования (*hardware*), прошивок (*firmware*) и ПО (*software*), определенных в технической документации и реализованных программным продуктом.

Определение

Управление конфигурацией (англ. *software control management*) — процесс идентификации конфигурации программной системы в заданные моменты времени с целью:

- ▶ систематического контроля изменений конфигурации;
- ▶ поддержки целостности (*integrity*) и отслеживаемости (*traceability*) конфигурации на протяжении жизненного цикла ПО.

Составляющие управления конфигурацией



Сокращения: SC = software configuration; SCM = software configuration management.

Управление процессом конфигурации

Определение

Управление процессом конфигурации (англ. *management of the SCM process*) — деятельность по контролю эволюции и целостности программного продукта следующими методами:

- ▶ идентификация элементов ПО;
- ▶ управление и контроль изменений, вносимых в продукт;
- ▶ верификация, сохранение и генерация отчетов по конфигурации ПО.

Цель: составление *плана управления* процессом конфигурации, который включает:

- ▶ описание *деятельности* по конфигурации (идентификация, контроль, аудит, ...);
- ▶ *расписание* процессов конфигурации;
- ▶ используемые *ресурсы* (инструменты, ответственные люди, ...).

Идентификация конфигурации ПО

Этапы идентификации конфигурации:

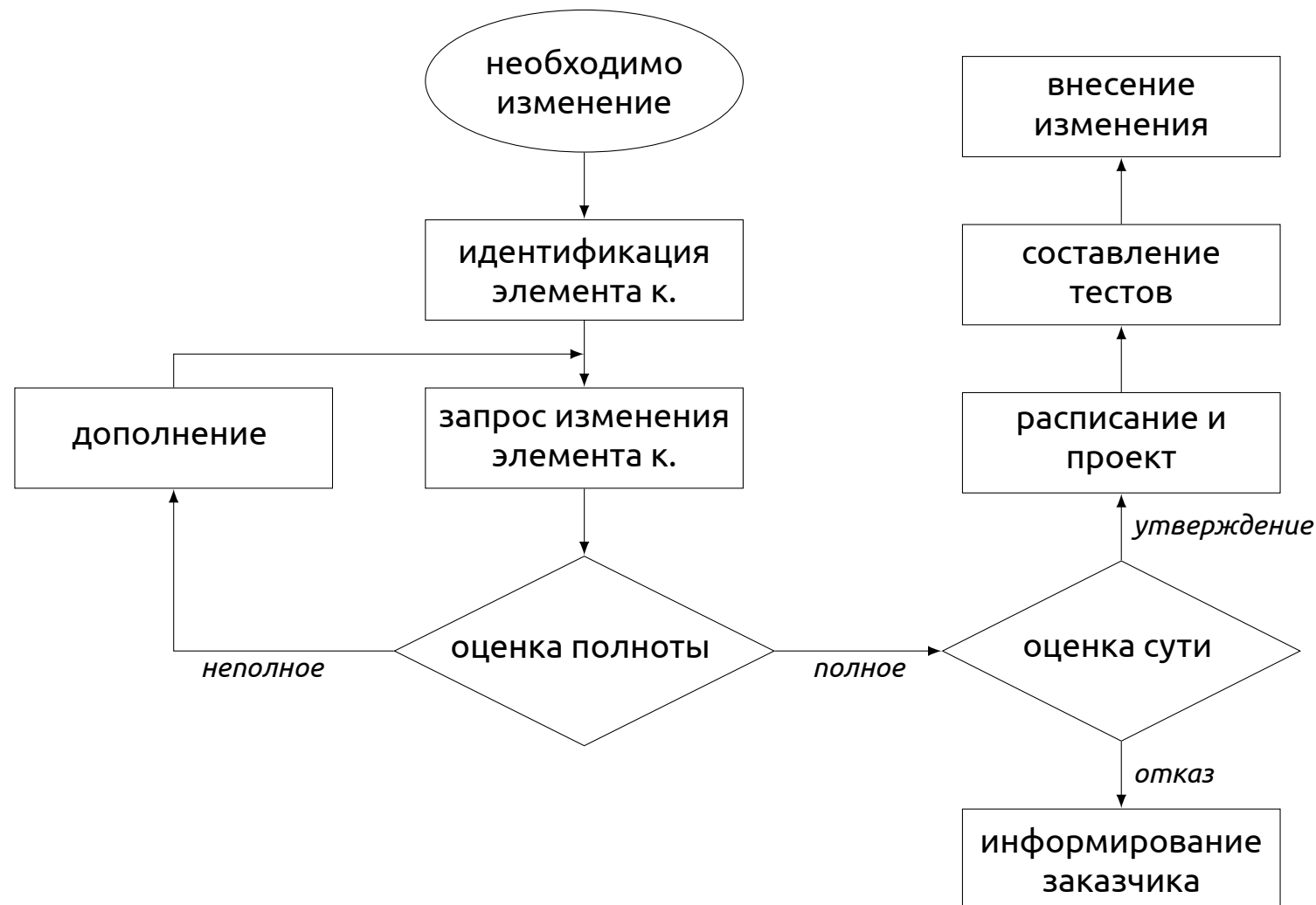
1. идентификация контролируемых элементов;
2. создание схем идентификации для объектов и их версий;
3. определение инструментов и методов для получения и управления контролируемыми элементами.

Элементы конфигурации:

- ▶ исполняемый и исходный код;
- ▶ планы;
- ▶ спецификации и проектная документация;
- ▶ система тестирования;
- ▶ программные инструменты;
- ▶ сторонние библиотеки;
- ▶ справочная документация.

Базис (англ. *baseline*) — набор элементов конфигурации, формально описанных и зафиксированных в конкретный момент времени.

Контроль конфигурации ПО



Контроль конфигурации ПО — деятельность по координации и оценке реализованных изменений элементов конфигурации, а также их утверждению или отбрасыванию.

Учет статуса и аудит конфигурации

Определение

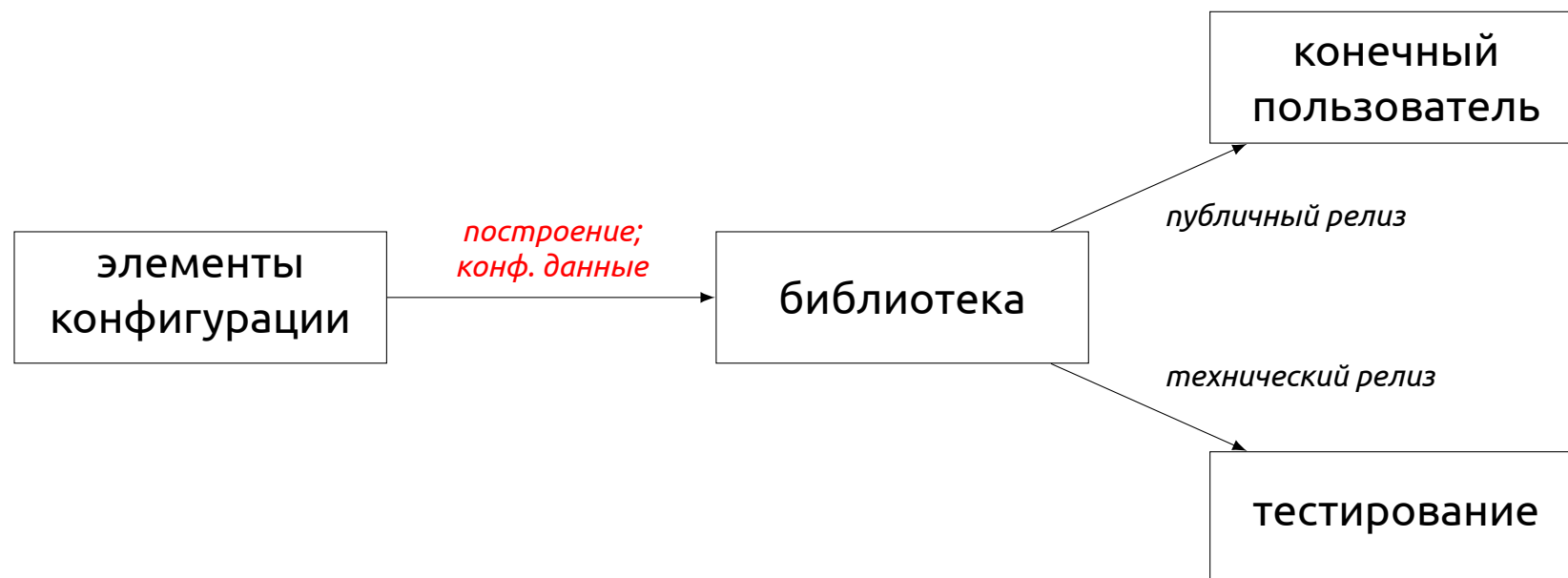
Учет статуса конфигурации — это запоминание информации и составление отчетов для определения степени изменения конфигурации и правильности вносимых в ПО изменений.

Определение

Аудит конфигурации — деятельность для оценки соответствия ПО и процессов разработки регламентированным стандартам, инструкциям, планам и процедурам:

- ▶ *функциональный* аудит (проверка элемента ПО на соответствие спецификации);
- ▶ *физический* аудит (проверка соответствия элемента проектной и справочной документации).

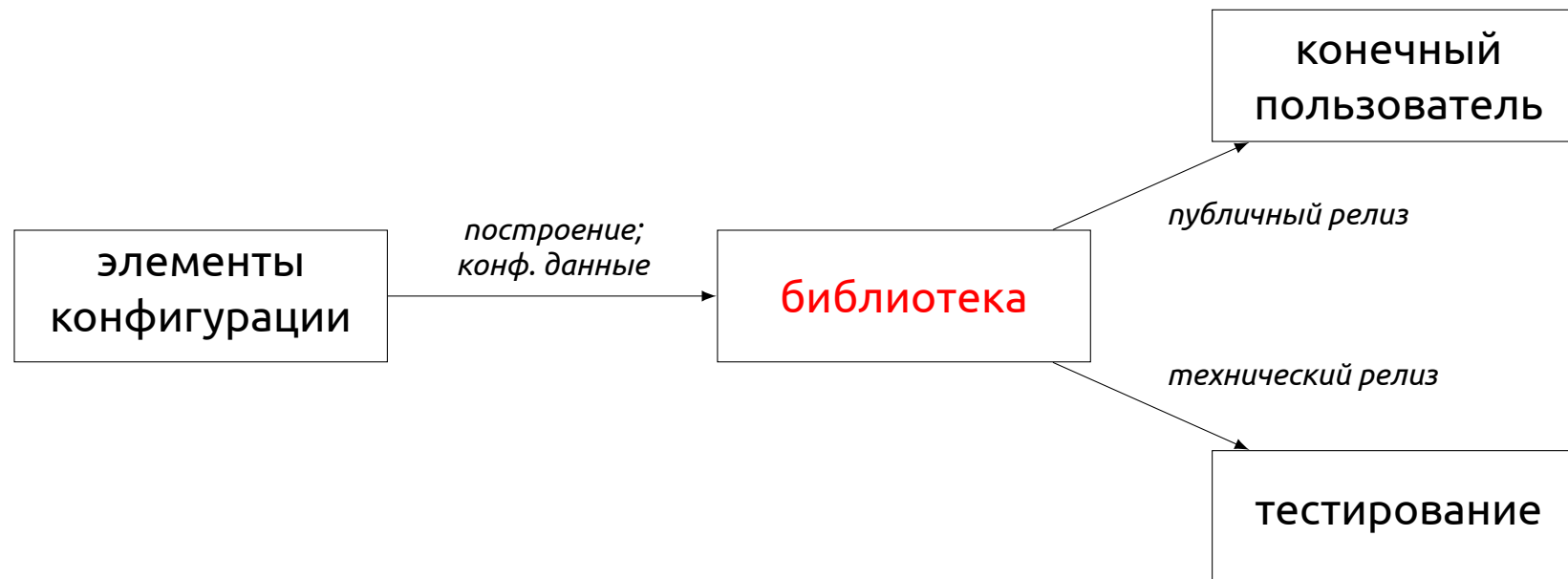
Управление выпусками и доставкой



Определение

Построение ПО — объединение корректных версий элементов конфигурации с использованием конфигурационных данных в исполняемую программу для доставки заказчику или другому потребителю (напр., отделу тестирования).

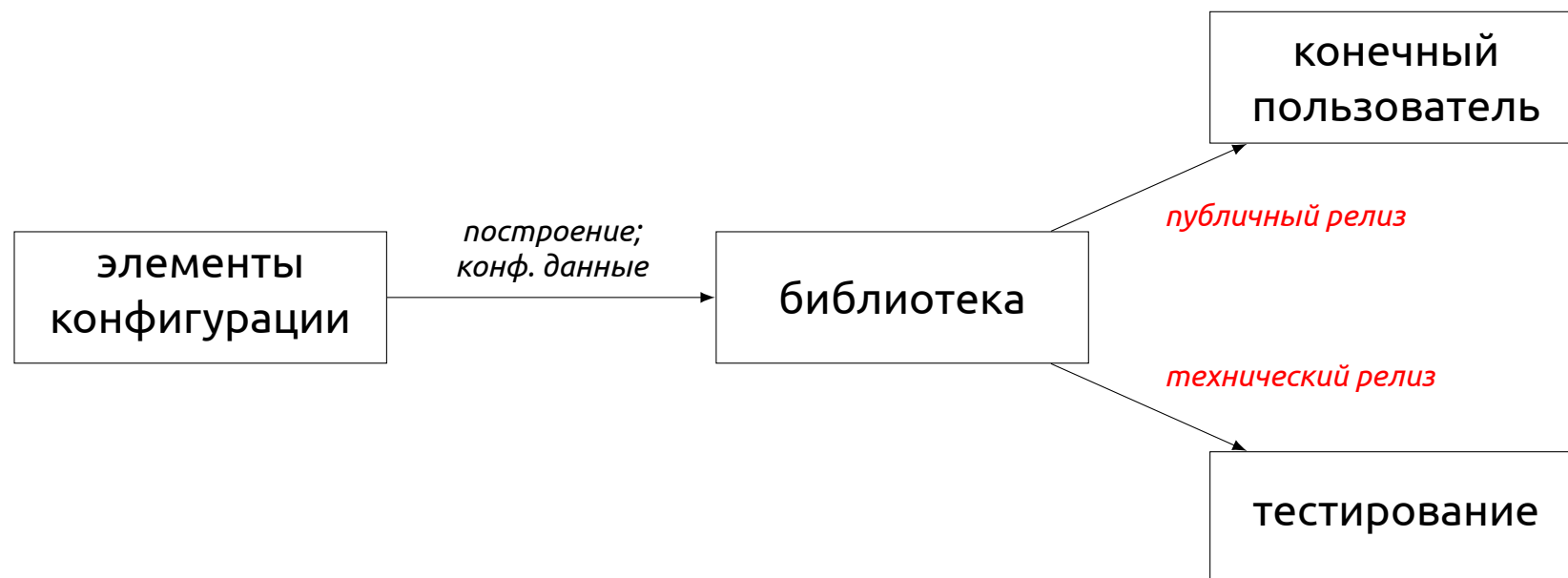
Управление выпусками и доставкой



Определение

Программная библиотека — контролируемая коллекция программ и соответствующей документации, предназначенная для упрощения разработки, использования и сопровождения ПО.

Управление выпусками и доставкой



Определение

Выпуск (релиз) ПО — деятельность по определению, подготовке и доставке элементов программного продукта: исполняемой программы, документации, конфигурационных данных и т. п.

Управление инженерией ПО

Определение

Управление инженерией ПО (англ. *software engineering management*) — управление работами команды разработки ПО в процессе выполнения плана проекта, определение критериев эффективности работы этой команды и оценка процессов и продуктов проекта с использованием общих методов планирования и контроля работ.

Составляющие: планирование, координация, мониторинг, контроль, измерение и отчетность.

Уровни управления:

- ▶ организационное и инфраструктурное управление;
- ▶ управление проектом;
- ▶ планирование и контроль измерений.

Организационное управление

Составляющие организационного управления:

- ▶ планирование и составление графика работ, связанных с проектом;
- ▶ подбор и управление персоналом;
- ▶ контроль выполнения проекта и оценка качества работ в соответствии со стандартами и договором с заказчиком.

Объекты управления:

- ▶ персонал (обучение, мотивация, карьерный рост, ...);
- ▶ коммуникации между сотрудниками (встречи, презентации, ...);
- ▶ портфель (англ. *portfolio*) (организация повторного использования кода).

Управление программным проектом

1. Инициирование проекта и определение его рамок.

Составляющие: определение и согласование требований; анализ выполнимости (технической, операционной, финансовой, социальной); пересмотр требований.

2. Планирование проекта.

Составляющие: планирование процессов ЖЦ; определение промежуточных артефактов; оценка затрат; распределение ресурсов; управление риском; управление качеством; мониторинг проекта.

3. Реализация проекта.

Составляющие: реализация планов; управление контрактами с поставщиками; имплементация измерительных процессов; мониторинг и контроль; отчетность.

4. Рецензирование и оценка проекта.

Составляющие: определение соответствие проекта требованиям; оценка эффективности выполненных работ.

5. Завершение проекта.

Составляющие: определение степени завершенности проекта; деятельность по завершению (напр., *post-mortem*).

Инженерия измерения ПО

Этапы инженерии измерения



Цели инженерии измерения:

- ▶ усовершенствование процессов управления проектом;
- ▶ оценка и регулировка временных затрат и стоимости ПО;
- ▶ определение категорий рисков и отслеживание факторов для расчета вероятностей их возникновения;
- ▶ проверка заданных в требованиях показателей качества продуктов и проекта в целом.

Процесс программной инженерии

Определение

Процесс программной инженерии (англ. *software engineering process*) — деятельность по определению, имплементации, оценке, измерению, управлению, изменению и совершенствованию процессов жизненного цикла ПО.

Составляющие области знаний:

- ▶ имплементация и изменение процессов ЖЦ;
(инфраструктура, деятельность, модели и практические соображения, касающиеся имплементации и модификации процессов жизненного цикла)
- ▶ определение процессов;
- ▶ оценка процессов;
- ▶ измерение процессов ЖЦ и программного продукта.

Определение процессов ЖЦ

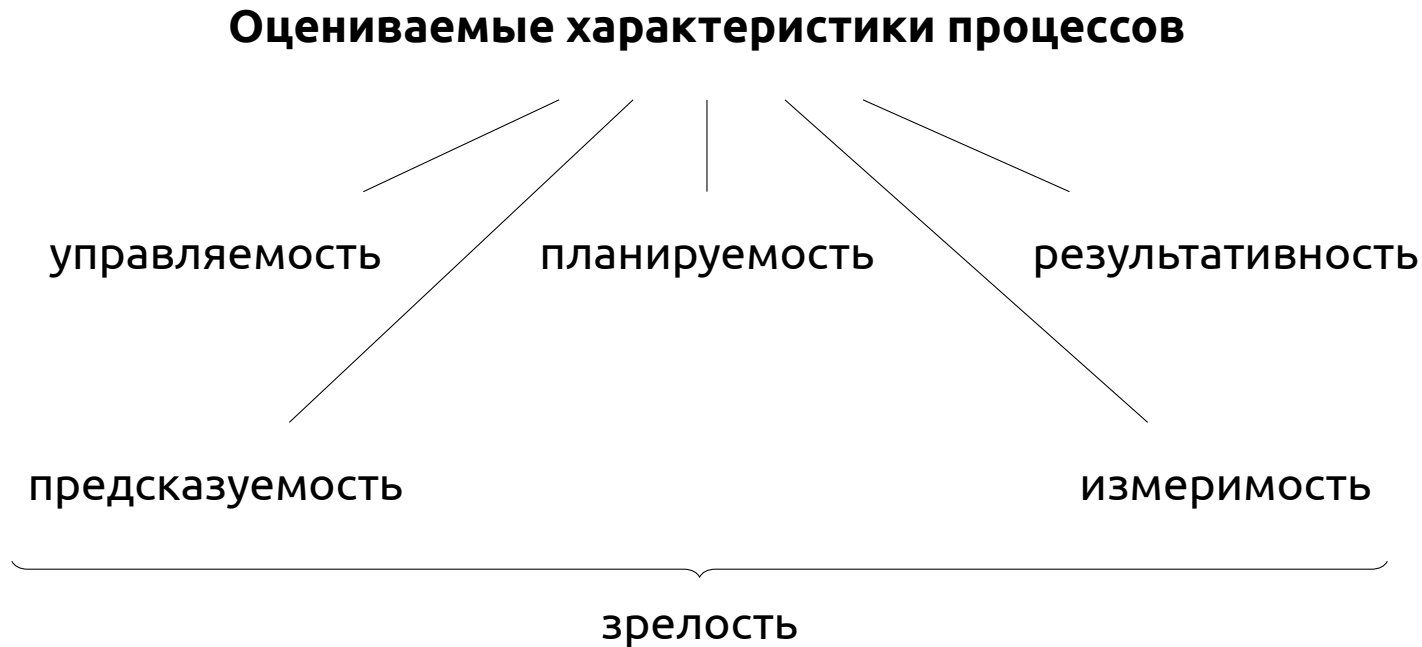
Цель определения процессов:

- ▶ повышение качества программного продукта;
- ▶ облегчение взаимопонимания и коммуникации между разработчиками;
- ▶ поддержка совершенствования процессов и управления ними;
- ▶ автоматизация руководства процессами.

Инструменты определения:

- ▶ модель жизненного цикла (каскадная, спиральная, итеративная, ...);
- ▶ стандарты ЖЦ ПО (ISO/IEC 12207, ISO/IEC 15504, IEEE 1074, IEEE 1219, ...);
- ▶ нотации представления процессов (диаграммы потоков данных, сети Петри, IDEF0, ...);
- ▶ инструменты автоматизации.

Оценка процессов ЖЦ



Модели оценки процессов:

- ▶ SW-CMM (software capability maturity model);
- ▶ CMMI (capability maturity model integration);
- ▶ Bootstrap;
- ▶ стандарты ISO/IEC 15504, ISO 9001.

Измерение процессов и продукта

Определение

Измерение процессов жизненного цикла — сбор, анализ и интерпретация количественных характеристик процессов с целью обнаружить их сильные и слабые стороны, а также оценить качество имплементации и/или внесенных изменений.

Примеры характеристик:

- ▶ качество продукта (напр., число дефектов на 1 строку кода);
- ▶ простота сопровождения (затраты на проведение определенного изменения);
- ▶ продуктивность (напр., в строках кода на человеко-месяц);
- ▶ время разработки;
- ▶ удовлетворенность пользователей (напр., измеренная анкетированием).

Методы и инструменты ПИ

Определение

Инструменты разработки ПО (англ. *software development tools*) — это компьютерные инструменты, предназначенные для упрощения процессов ЖЦ путем автоматизации формализованных повторяющихся действий.

Определение

Методы ПИ (англ. *software engineering methods*) упорядочивают деятельность, связанную с программной инженерией, чтобы систематизировать действия по разработке ПО и увеличить их продуктивность.

Методы ПИ

- ▶ **Эвристические методы** — методы, основанные на неформальном подходе к ПИ.
Примеры: структурные, объектно-ориентированные, информационные (*data-oriented*).
- ▶ **Формальные** методы — основанные на математических моделях.
Примеры: языки спецификации; методы уточнения спецификации (*specification refinement*)
— ее приближение к форме конечного продукта; методы доказательства и верификации.
- ▶ Методы **прототипирования** — техники создания прототипов ПО.

Инструменты ПИ

- ▶ Инструменты работы с **требованиями**: моделирование и отслеживание (*traceability*) требований;
- ▶ инструменты **проектирования**;
- ▶ инструменты **конструирования**: редакторы программ; компиляторы и генераторы кода; интерпретаторы; отладчики;
- ▶ инструменты **тестирования**: генераторы тестов; системы тестирования (*test execution framework*); инструменты оценки тестов; инструменты управления тестированием; инструменты анализа производительности;
- ▶ инструменты **сопровождения** ПО: улучшение понимания (напр., визуализация); средства реинженерии;

Инструменты ПИ (продолжение)

- ▶ инструменты управления **конфигурацией**: средства отслеживания дефектов (напр., баг-трекеры); системы управления версиями; инструменты сборки, выпуска и инсталляции;
- ▶ инструменты **управления ПИ**: средства для планирования и отслеживания проектов; инструменты управления рисками; инструменты измерения характеристик ПО;
- ▶ средства **поддержки процессов** ПИ: инструменты моделирования процессов; инструменты управления процессами;
- ▶ инструменты обеспечения **качества**: средства для аудита и рецензирования; инструменты статического анализа (проверка артефактов на соответствие требованиям).

Качество ПО

Определение

Качество программного обеспечения (англ. *software quality*) — набор свойств продукта, характеризующих его способность удовлетворить явно заданные или подразумеваемые требования заказчика.

Категории характеристик качества (стандарт ISO 9126:01, [25010:11](#)):

- ▶ внутренние — соответствие промежуточных артефактов внутренним стандартам;
- ▶ внешние — требования к функциональности продукта;
- ▶ эксплуатационные (англ. *quality in use*) — характеристики качества, интересующие конечного пользователя.

Основные характеристики качества

- ▶ **Функциональность** (англ. *functional suitability*) — степень соответствия продукта явным и подразумеваемым требованиям при использовании в определенных условиях.
- ▶ **Эффективность** (англ. *performance efficiency*) — эффективность использования предоставленных ресурсов (напр., оборудования, ОС, других программ, расходных материалов).
- ▶ **Совместимость** (англ. *compatibility*) — возможности по обмену информацией с другими программами и совместного оперирования в одной среде.
- ▶ **Удобство применения** (англ. *usability*) — простота обучения, легкость управления системой, доступность пользовательского интерфейса и т. п.

Основные характеристики качества (продолжение)

- ▶ **Надежность** (англ. *reliability*) — отказоустойчивость, доступность (*availability*), возможности по восстановлению после сбоев (*recoverability*).
- ▶ **Безопасность** (англ. *security*) — степень защиты данных пользователей.
- ▶ **Простота сопровождения** (англ. *maintainability*) — эффективность модификации системы, возможности по повторному использованию (*reusability*), модульность и т. п.
- ▶ **Переносимость** (англ. *portability*) — эффективность переноса ПО в новую среду исполнения.

Процессы инженерии качества

- ▶ **Обеспечение** качества (англ. *quality assurance*) — деятельность для гарантирования характеристик качества в программном продукте (напр., четкая формулировка требований и проблем, составление планов, ...);
- ▶ **верификация** — обеспечение корректной реализации ПО согласно спецификациям (правильно ли создается система?);
- ▶ **валидация** — соответствие системы требованиям (удовлетворяет ли система заказчика?);
- ▶ **инспекции** — выявление аномалий в ПО независимыми экспертами;
- ▶ **аудит** — независимая оценка продукта на соответствие регламентирующим документам (планам, стандартам и т. п.).

Выводы

1. Ядро SWEBOK содержит пять вспомогательных областей знаний (управление конфигурацией, управление инженерией, процесс инженерии, инженерия качества, а также методы и инструменты ПИ). Они соответствуют организационным и управленческим аспектам производства ПО.
2. Основу организационных областей составляют практические рекомендации.
3. Организационные области SWEBOK связаны между собой, а также с основными областями знаний, для которых они регламентируют деятельность.

Материалы

📄 Лавріщева К. М.

Програмна інженерія (підручник).

К., 2008. — 319 с.

📄 IEEE Computer Society

Описание стандарта SWEBOOK.

<http://www.computer.org/portal/web/swebok/html/contents>

📄 International Standartization Organization

Стандарты ISO.

<http://www.iso.org/iso/home.html>

Спасибо за внимание!