

Парамзина Алиса Андреевна
Магистерская программа «Программная инженерия»
Master program «Software Engineering»
Тищенко Евгений Николаевич
Tishchenko Evgenie Nikolaevich,
д.э.н., профессор информационных технологий и защиты информации,
Ростовский государственный экономический университет (РИНХ)
D.Sc. Economics, professor of the department of information technology and
information security, Rostov State University of Economic

ИСПОЛЬЗОВАНИЕ CASE -ИНСТРУМЕНТОВ ПРИ РАЗРАБОТКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

The use of CASE-tools in software development

Аннотация

В работе описан подход к проектированию и разработке программного обеспечения, нацеленной на использование информационных технологий для повышения качества программ, отсутствие ошибок и простоту обслуживания программных продуктах. Как применение case-средств влияет на эффективность системы внутри нее и управлению в целом.

Abstract

The paper describes an approach to the design of software development aimed at using information technology to improve the quality of programs, the absence of errors and ease of maintenance of software products. How the use of case tools affects the effectiveness of the system within it and management as a whole.

Ключевые слова: информационные технологии; case-средства; разработка программного обеспечения, средства программной инженерии

Keywords: information technologies; case-tools; software development, software engineering tools

CASE-инструмент – (Автоматизированное программное обеспечение/Системная инженерия - это инструмент для сложных программных систем для поддержки полного жизненного цикла или его основных этапов. Для высоких технологий существует комплексное использование САПР, ASPR и автоматизированных систем управления. На основе различных методологий системного анализа было создано большое количество кейс-инструментов. В настоящее время стали популярны Case-инструменты для полного программного цикла.

Целью CASE-инструментов является повышение эффективности работ по разработке, а также отсутствие осложнений в деятельности разработчиков программного обеспечения.

Инструмент CASE состоит в основном из:

- методологии - GL (Графический язык), а также набор правил и методов
- графические редакторы - Графический пользовательский интерфейс (Графический пользовательский интерфейс), используемый при построении диаграмм;
- генератор - генерирует исходный код (исходный код) для различных платформ;
- репозиторий - база данных, в которой хранятся результаты работы разработчиков.

Кейс-инструменты в программной инженерии

Инструменты для работы с требованиями (Инструменты требований к программному обеспечению).

Инструменты проектирования (Инструменты проектирования программного обеспечения) — инструменты для создания и проверки дизайна программного обеспечения. (SADT/IDEF, UML, BPMN/BPEL, Microsoft DSL и т.

Инструменты Построения (Инструменты построения программного обеспечения) В соответствии с пониманием "строительства", данным соответствующей областью знаний SWEBOOK. Эти инструменты используются для создания и перевода представления программы (например, исходного кода), которое является достаточно подробным и явным для машинного выполнения.

Редакторы (редакторы программ). Редакторы используются для создания и изменения программ и для взаимодействия с документацией. Компиляторы и генераторы кода. Переводчики команд исходного кода.

Компиляторы и редакторы в интегрированных средах программирования. Этот класс также включает в себя препроцессоры, компоновщики/загрузчики и генераторы кода.

Вы можете комбинировать интерпретаторы с компиляторами и генераторами кода в качестве средства непосредственной подготовки (перевода) исходного кода для выполнения.

Эти инструменты поддерживают процесс разработки программного обеспечения, но в то же время отличаются от редакторов и компиляторов.

Средства Тестирования (Средства тестирования программного обеспечения)

Платформы для выполнения тестов. Эти инструменты предоставляют среду выполнения для сценариев тестирования в контролируемой среде, которая позволяет отслеживать поведение тестируемого объекта.

Инструменты оценки тестов. Эти инструменты поддерживают оценку результатов тестирования, помогая определить, в какой степени и где именно обнаруженное поведение соответствует ожидаемому поведению.

Инструменты управления тестированием.

Инструменты анализа производительности.

Средства технического обслуживания (Средства технического обслуживания программного обеспечения) В этом разделе рассматриваются инструменты, которые особенно важны для поддержки существующего программного обеспечения, которое может быть изменено:

Инструменты для облегчения понимания (инструменты понимания). Эти инструменты помогают человеку в понимании программ. В качестве примеров могут служить различные инструменты визуализации.

Инструменты реинжиниринга. Эти инструменты поддерживают мероприятия по реинжинирингу, описанные в области знаний SWEBOOK "Обслуживание программного обеспечения".

Инструменты управления конфигурацией (Инструменты управления конфигурацией программного обеспечения) Инструменты управления конфигурацией подразделяются на три категории:

Инструменты для отслеживания дефектов, расширений и проблем.

Инструменты контроля версий.

Создавайте и выпускайте инструменты. Эти инструменты предназначены для управления задачами создания и выпуска продуктов, а также включают инструменты установки.

Инструменты управления разработкой (Инструменты управления разработкой программного обеспечения) Инструменты управления разработкой программного обеспечения делятся на три категории:

Инструменты планирования и отслеживания проектов.

Инструменты управления рисками.

Средства количественной оценки.

Средства поддержки Процессов (Средства разработки программного обеспечения):

Инструменты моделирования.

Инструменты управления проектами.

Инструменты управления конфигурацией, которые поддерживают работу с актуальными версиями всего набора артефактов проекта. Платформы разработки программного обеспечения на основе ролей, которые охватывают все этапы жизненного цикла и на сегодняшний день являются разработкой интегрированных инструментов

разработки и CASE-инструментов в направлении поддержки "связанных" функций - управления требованиями, управления конфигурацией с поддержкой управления изменениями, тестирования и оценки качества.

Инструменты обеспечения качества (Инструменты обеспечения качества программного обеспечения) Инструменты обеспечения качества делятся на две категории: Эти инструменты используются для поддержки анализа и аудита.

Средства (статического) анализа. Эти инструменты используются для анализа программных данных, рабочих процессов и зависимостей.

КЕЙС-инструменты, классифицированные по функциональной направленности.

Инструменты для кейсов

Инструменты анализа и проектирования BPwin, КЕЙС. Аналитик, Silverrun, ДИЗАЙНЕР/2000, создатель команды Vantage

Инструменты проектирования баз данных и файлов ERwin, S-Designor, PRO-IV, CASE.

Инструменты программирования SQL для Windows, Power Builder, Delphi

Инструменты технического обслуживания и реинжиниринга PRO-IV, Rational Rose, Команда по оборудованию

Инструменты для защиты окружающей среды PBX, сода

Инструменты управления проектами MS Project, SE Companion

КЕЙС-инструменты, классифицированные по уровням.

Примеры инструментов CASE

Высокий уровень MS Project, SE Компаньон

Средний уровень. Аналитик, Silverrun, ДИЗАЙНЕР/2000

Низкий уровень Power Builder, Delphi, Uniface, JAM

Специалист в области программной инженерии - это в первую очередь не только разработчик прикладного программного обеспечения, но и системного программного обеспечения, организатор и руководитель (руководитель проекта) промышленной разработки надежных высококачественных программных систем. Важнейшей проблемой разработки и применения современных систем является подготовка и переподготовка специалистов в области разработки программного обеспечения, использование международных стандартов, способствующих высокому качеству программного обеспечения и его надежной оценке. Необходимо обучать специалистов умению формализовывать требования и достигать конкретных значений качественных характеристик функционирования и применения сложных программных пакетов с учетом ресурсов, которые необходимы и доступны для обеспечения и повышения этого качества.

ИСТОЧНИКИ:

1. Орлик С., Булуя Ю. «Введение в программную инженерию и управление жизненным циклом» (базируется на SWEBOOK). <http://software-testing.ru/library/around-testing/engineering/267-swebok>
2. Программная инженерия <http://iibs.vvsu.ru/ispi/nar/pi/>
3. Липаев В.В. Программная инженерия в жизненном цикле программных средств <http://citforum.ru/SE/lipaev/>