

ИССЛЕДОВАНИЕ ПРОЦЕССА РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ИНФОТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМ

Рассмотрены основные механизмы процесса разработки программного обеспечения инфотелекоммуникационной системы и их состав. Рассмотрено понятие жизненного цикла программного обеспечения. Исследованы модели разработки программного обеспечения инфотелекоммуникационных систем, их преимущества и недостатки.

Ключевые слова: инфотелекоммуникационная система, программное обеспечение, жизненный цикл программного обеспечения, каскадная модель, V-образная модель.

1. Введение

Непосредственно создание инфотелекоммуникационных систем (ИТКС) состоит из цепи взаимосвязанных процессов, включающих в себя проектирование, как системы в целом, так и всех видов ее обеспечения (программного, математического, информационного и т.д.), а также программирование контроллеров, диагностирование, диспетчеризацию и т.д.

Одним из основных видов обеспечения инфотелекоммуникационных систем является программное обеспечение (ПО), так как именно оно обеспечивает реализацию функций системы, заданное функционирование комплекса технических средств инфотелекоммуникационных систем и их предполагаемое развитие [1-5].

Совокупность программ, разрабатываемых при создании системы и включающая программы реализации непосредственно ее функций, называется специальным программным обеспечением инфотелекоммуникационной системы.

При этом выделяют две основных функции инфотелекоммуникационных систем: информационная и управляющая.

Целью информационной функции является сбор, преобразование, хранение информации о состоянии технологического объекта управления, представление этой информации оперативному персоналу и передача ее для последующей реализации.

Целью управляющей функции является непосредственная выработка решений и осуществление управляющих воздействий на технологический процесс.

2. Исследование процесса разработки ПО и его моделей

Жизненный цикл ПО инфотелекоммуникационной системы определяется как период времени, который начинается с момента принятия решения о необходимости создания ПО и заканчивается в момент его полного изъятия из эксплуатации [5].

Одним из основных процессов жизненного цикла ПО является процесс разработки. Процесс разработки ПО включает действия и задания разработчика по созданию программного продукта. В общем случае набор действий в процессе разработки ПО ИТКС зависит от специфики, масштаба и сложности ПО, а также специфики условий, в которых создается и функционирует система [6-8]. Тем не менее, согласно [5], в состав любого процесса разработки ПО ИТКС должны быть включены следующие действия разработки:

1. Реализация процесса разработки R_S .
2. Анализ и формирование системных требований A_{ST} .
3. Проектирование архитектуры системы P_{AS} .
4. Анализ и формирование требований к ПО A_{PO} .
5. Проектирование архитектуры ПО P_{APO} .
6. Разработка детального проекта ПО P_{DAP} .
7. Кодирование ПО C_{PO} .
8. Тестирование отдельных модулей (компонентов) ПО T_{PO} (модульное (компонентное) тестирование).
9. Интеграция ПО I_{PO} .

10. Квалификационное тестирование групп модулей (компонентов) ПО T_{KPO} (интеграционное тестирование).

11. Системная интеграция I_S .

12. Квалификационное тестирование ПО и системы в целом T_{KS} (системное тестирование).

13. Инсталляция (установка) ПО $Inst_{PO}$.

14. Обеспечение принятия ПО As_{PO} (приемочное (приемо-сдаточное) тестирование).

Последовательность выполнения и взаимосвязи действий на протяжении разработки ПО, т.е. характер процесса его разработки, определяется соответствующей моделью, которую иногда ошибочно называют моделью всего жизненного цикла ПО. В данном случае будем понимать под соответствующей моделью именно модель разработки ПО, а фазы модели разработки ПО привяжем к действиям на протяжении соответствующего процесса. С целью более компактного представления модели условно «родственные действия» (за исключением действий по тестированию), такие как A_{ST} и A_{POT} или P_{AS} , P_{APO} и P_{DAPO} , могут рассматриваться как единая фаза, например, анализ требований к системе и ПО A_T и проектирование архитектуры системы и ПО P_A . Некоторые действия в рамках фаз модели

разработки не рассматриваются, например, действия R_S и $Inst_{PO}$ как чисто организационные, а интеграционные действия I_{PO} и I_S как достаточно очевидные. При этом с каждым видом тестирования связаны соответствующие уровни тестирования согласно [1, 2], а именно компонентное (модульное) тестирование (component testing or unit testing), интеграционное тестирование (integration testing), системное тестирование (system testing) и приемочное тестирование (acceptance testing). Собственно уровень тестирования определяет то, над чем производятся тесты: над отдельным модулем, группой модулей или ПО, системой в целом.

К настоящему времени наибольшее распространение получили следующие основные модели разработки ПО [1, 3, 4, 6]: каскадная (водопадная) модель, поэтапная модель с промежуточным контролем или V-образная модель, спиральная модель.

Каскадная (водопадная) модель разработки ПО ИТКС основана на постепенном увеличении степени детализации описания всей разрабатываемой системы. Каждое повышение степени детализации определяет переход к следующему состоянию разработки (рис. 1).

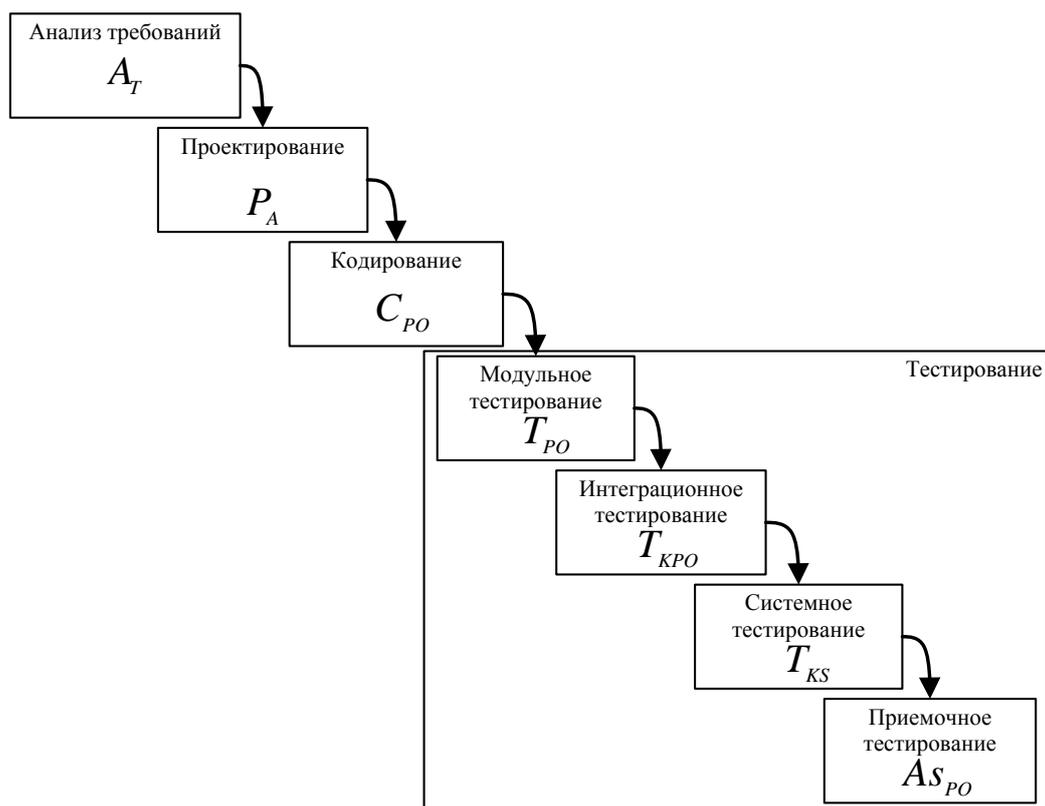


Рис. 1. Каскадная (водопадная) модель разработки ПО АСУ ТП

К достоинствам каскадной модели разработки ПО ИТКС относят то, что переход к следующей фазе происходит только тогда, когда полностью завершены все работы предыдущей. То есть сначала полностью готовятся все требования к системе и ПО, полностью разрабатывается проект системы и ее ПО и так далее до тестирования ПО и системы по соответствующим уровням.

К недостаткам каскадной модели разработки ПО ИТКС относят то, что в случае больших систем работа на каждом этапе занимает значительное время, а внесение изменений в первичные документы либо невозможно, либо вызывает лавинообразные изменения на всех других этапах. При этом, если тестовые наборы обнаружили несоответствие реализации требованиям, то причина дефектов может крыться:

- в неправильном тестовом наборе;
- в дефектах кодирования;
- в неверной архитектуре системы или ПО;
- некорректности требований и т.д.

Все эти случаи требуют анализа для принятия решения о том, на какую фазу каскадной модели надо возвратиться для устранения обнаруженного несоответствия или дефекта.

В качестве одной из основных модификаций классической каскадной модели является модель содержащая фазы двух видов – основные фазы разработки, аналогичные фазам каскадной модели (кроме тестирования) и фазы собственно тестирования и проверки, представляющие собой цепь обратной связи по отношению к основным фазам (рис. 2). Таким образом, в конце каждой фазы модели разработки, а зачастую и в процессе выполнения фазы, осуществляется проверка или тестирование взаимной корректности требований различных уровней в отличие от каскадной, где тестирование рассматривается как завершающий этап разработки ПО.

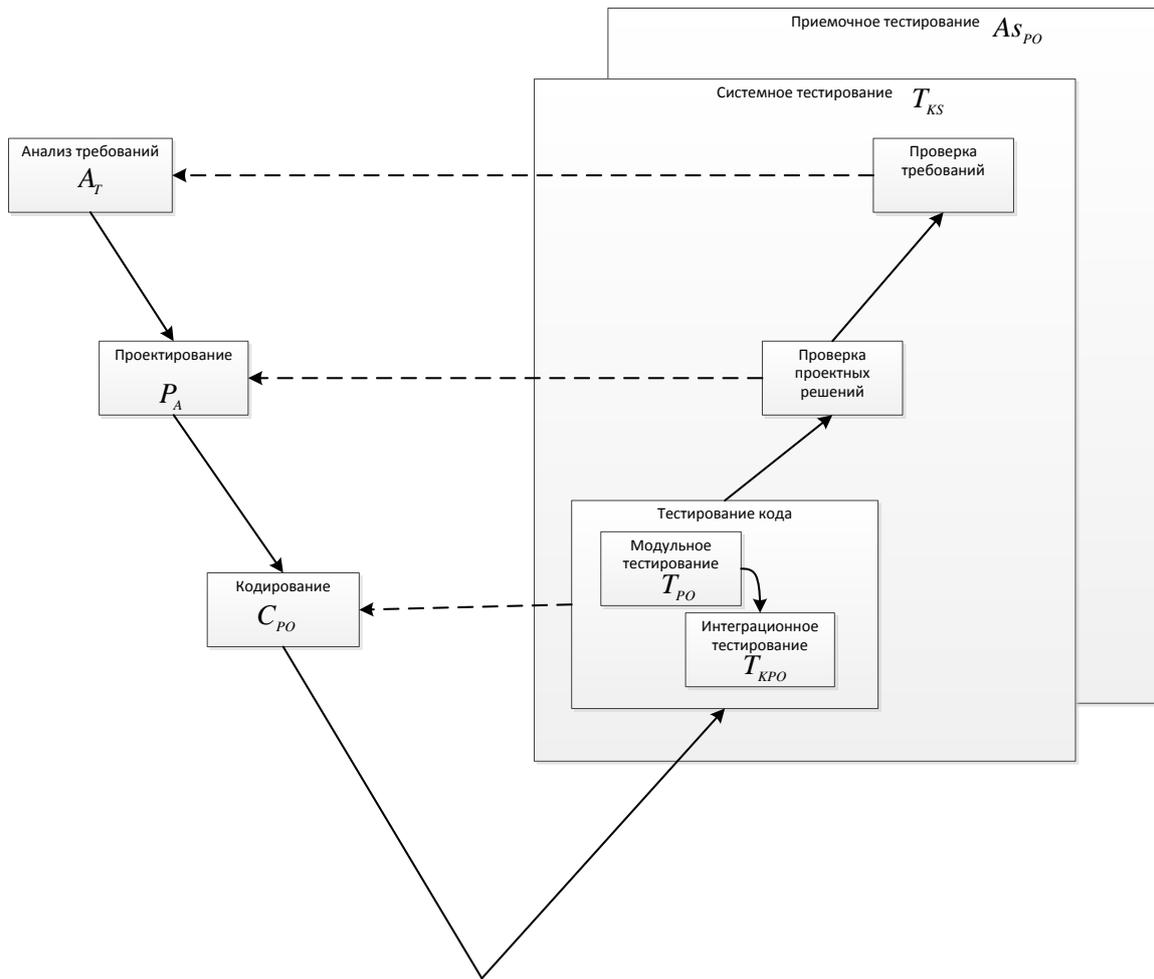


Рис. 2. Поэтапная модель с промежуточным контролем или V-образная модель разработки ПО ИТКС

Основным достоинством данной модели является возможность более оперативной проверки корректность разработки ПО, а недостатком то, что в каскадной модели предполагается разработка на каждой фазе документов, описывающих поведение всей системы в целом.

Оба рассмотренных выше типа жизненных циклов предполагают, что заранее известны и заданы все требования к системе, в целом, и к ПО, в частности, сформированные без прототипа системы.

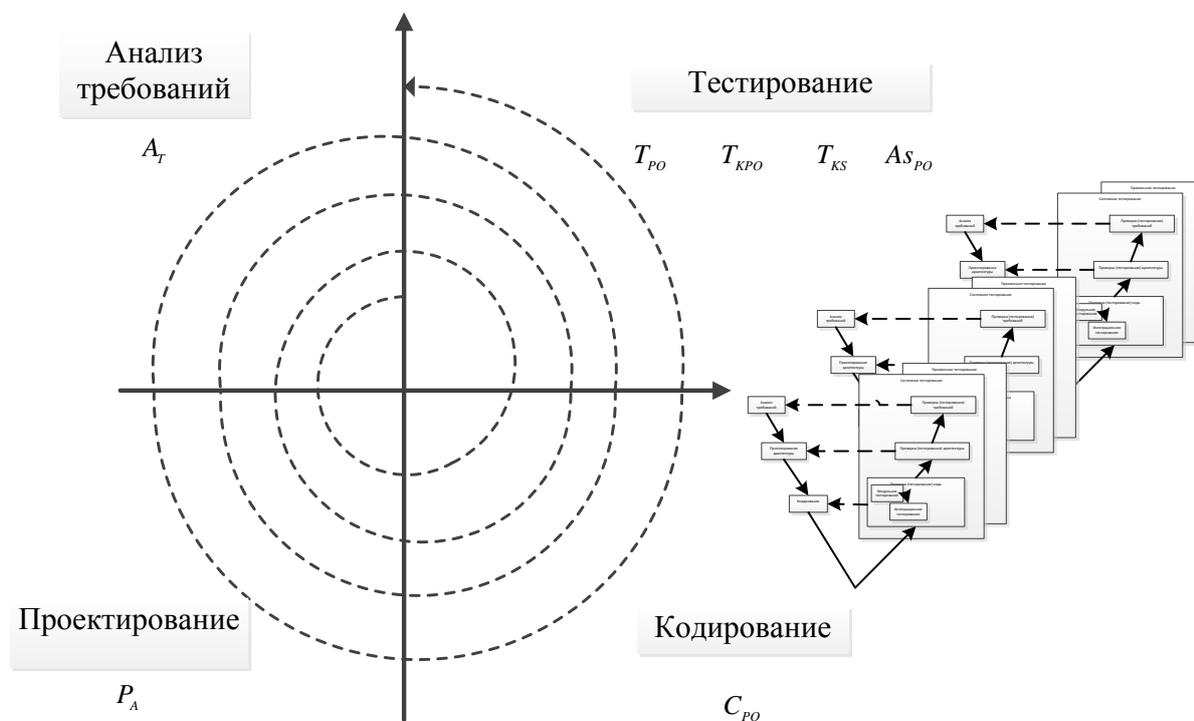


Рис. 3. Спиральная модель разработки ПО ИТКС

В спиральной модели разработка ПО происходит повторяющимися этапами – витками спирали (рис. 3). Каждый виток спирали, как правило, представляет один цикл в соответствии с каскадной или V-образной моделью. В конце каждого витка получается законченная версия системы – прототип, реализующая некоторый набор функций. Затем прототип предъявляется пользователю, на следующий виток переносится вся документация, разработанная на предыдущем витке, и процесс повторяется. Таким образом, система разрабатывается постепенно, проходя постоянные согласования с заказчиком. На каждом витке спирали функциональность ПО ИТКС расширяется постепенно, дорастая до полной функциональности. Таким образом, спиральная модель объединяя достоинства предыдущих моделей, нивелирует их недостатки.

3. Выводы

Таким образом, проведенные исследования показали, что в качестве базовой модели разработки ПО инфотелекоммуникационной системы необходимо использовать спиральную модель, а в рамках выполнения одного цикла –

V-образную модель, т.е. гибридную спирально – V-образную модель разработки ПО ИТКС.

Список литературы

1. Степанченко И.В. Методы тестирования программного обеспечения: Учебное пособие. Волгоград: ВолГТУ, 2006. – 74 с.
2. В.П. Котляров, Т.В. Коликова. Основы тестирования программного обеспечения Учебное пособие М.: Интернет-Университет Информационных технологий. 2006.
3. Вендров А.М. Проектирование программного обеспечения экономических информационных систем. – М.: Финансы и статистика, 2000. – 352 с.
4. Орлов С.А. Технологии разработки программного обеспечения: Учебник. – СПб: Питер, 2002. – 464 с.
5. ДСТУ 3918 – 99 ІТ. Процеси життєвого циклу програмного забезпечення. К.: Держстандарт України, 2000. – 49 с.
6. Милованов И.В., Лоскутов В.И. Основы разработки программного обеспечения вычислительных систем. Учебное пособие. Тамбов. Издательство ТГТУ, 2011. – 88с.
7. Благодатских В.А. и др. Стандартизация разработки программных средств Учеб. пособие / В.А. Благодатских, В.А. Волнин, К.Ф. Посакалов; Под ред.

О.С. Разумова. М.: Финансы и статистика, 2005. — 288 с.

8. Buede D.M. *The Engineering Design of Systems Models and Methods*. John Wiley & Sons, 2009. 516 p.

ЛЫСЕНКО Ирина Анатольевна

Кировоградский национальный технический университет, аспирант кафедры программного обеспечения
Раб. тел. – (0522)561977

СМИРНОВ Алексей Анатольевич

Кировоградский национальный технический университет, доктор технических наук, профессор, заведующий кафедрой программного обеспечения
Раб. тел. – (0522)390449

ПОЛИЩУК Людмила Ивановна

Кировоградский национальный технический университет, старший преподаватель кафедры программного обеспечения
Раб. тел. – (0522)561977

ДОСЛІДЖЕННЯ ПРОЦЕСУ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНФОТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ

І.А. Лисенко, О.А. Смірнов, Л.І. Поліщук

Розглянуто основні механізми процесу розробки програмного забезпечення інфотелекомунікаційної системи та їх склад. Розглянуто поняття життєвого циклу програмного забезпечення. Досліджені моделі розробки програмного забезпечення інфотелекомунікаційних систем, їхні переваги та недоліки.

Ключові слова: *інфотелекомунікаційна система, програмне забезпечення, життєвий цикл програмного забезпечення, каскадна модель, V-образна модель.*

STUDY OF SOFTWARE DEVELOPMENT INFOTELECOMMUNICATION SYSTEMS

I.A. Lysenko, A.A. Smirnov, L.I. Polischuk

The concept mechanisms of the software development process info telecommunication system and their composition. The concept of software life cycle. The model of software development more info telecommunication systems, their advantages and disadvantages.

Keywords: *infotelecommunication system, software, software lifecycle, cascade model, V-shaped pattern.*