

УДК 621.396.253

И.А. Лысенко, А.А. Смирнов, Л.И. Полищук

Кировоградский национальный технический университет, Кировоград

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.

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.

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 программного обеспечения, а также специфики условий, в которых создается и функционирует система [6 – 8]. Тем не менее, согласно [5], в состав любого процесса разработки программного обеспечения 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_{POT} .
 5. Проектирование архитектуры ПО P_{APO} .
 6. Разработка детального проекта ПО P_{DAPO} .
 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. харак-

тер 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 связа-

ны 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).

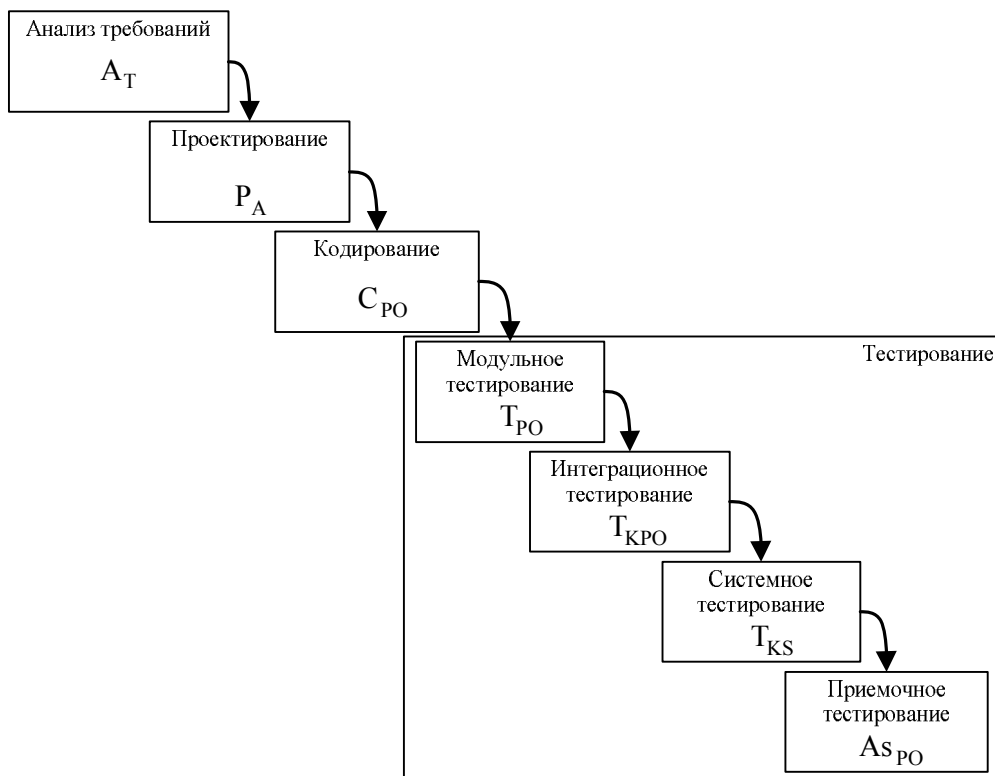


Рис. 1. Каскадная (водопадная) модель разработки ПО АСУ ТП

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;

– некоректності вимог і т.д.

Все ці випадки вимагають аналізу для прийняття рішення про те, на яку фазу каскадної моделі треба повернутися для усунення виявленого невідповідності або дефекту.

В якості однієї з основних модифікацій класическої каскадної моделі є модель, що містить фази двох видів – основні фази розробки, аналогічні фазам каскадної моделі

(крім тестування) і фази власне тестування і перевірки, що представляють собою ланку зворотного зв'язку по відношенню до основних фаз (рис. 2). Таким чином, в кінці кожної фази моделі розробки, а зазвичай і в процесі виконання фази, здійснюється перевірка або тестування взаємної коректності вимог різних рівнів в відмінності від каскадної, де тестування розглядається як завершальний етап розробки ПО.

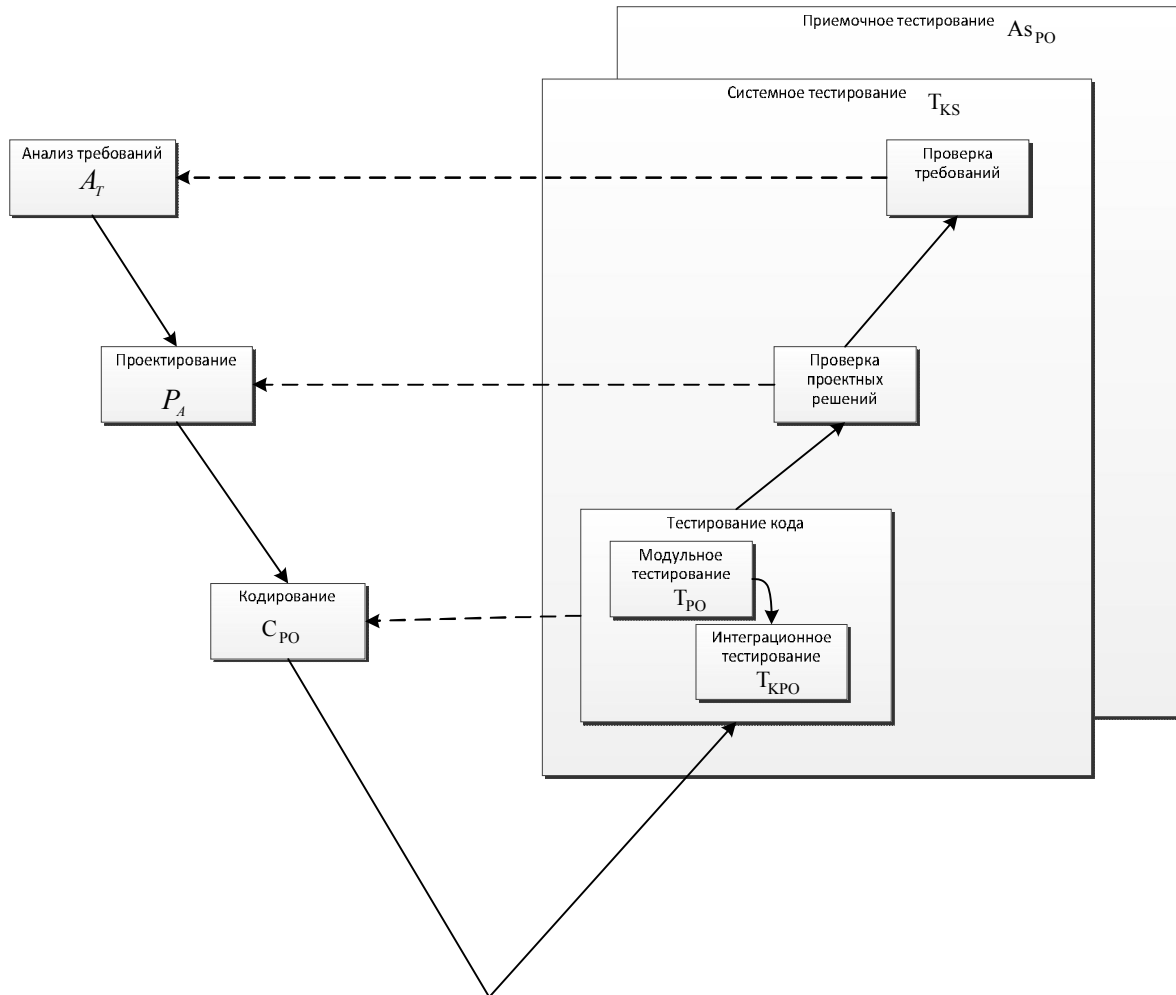


Рис. 2. Поетапна модель з проміжним контролем або V-образна модель розробки ПО ІТКС

Основним достоїнством даної моделі є можливість більш оперативної перевірки коректності розробки ПО, а недоліком те, що в каскадній моделі передбачається розробка на кожній фазі документів, що описують поведінку всієї системи в цілому.

Оба розглянутих вище типу життєвих циклів передбачають, що заздалегідь відомі і задані всі вимоги до системи, в цілому, і до ПО, в частині, сформульовані без прототипу системи.

В спіральній моделі розробка ПО відбувається повторюючися етапами – витками спіралі (рис. 3). Кожний виток спіралі, як правило, представляє один цикл в відповідності до каскадної

або V-образної моделі. В кінці кожного витка отримується закінчена версія системи – прототип, що реалізує деякий набір функцій. Далі прототип представляється користувачеві, на наступний виток переноситься вся документація, розроблена на попередньому витку, і процес повторюється. Таким чином, система розробляється поступово, проходя постійні узгодження з замовником. На кожному витку спіралі функціональність ПО ІТКС розширюється поступово, досягаючи до повної функціональності.

Таким чином, спіральна модель об'єднує достоїнства попередніх моделей, нівелює їх недоліки.

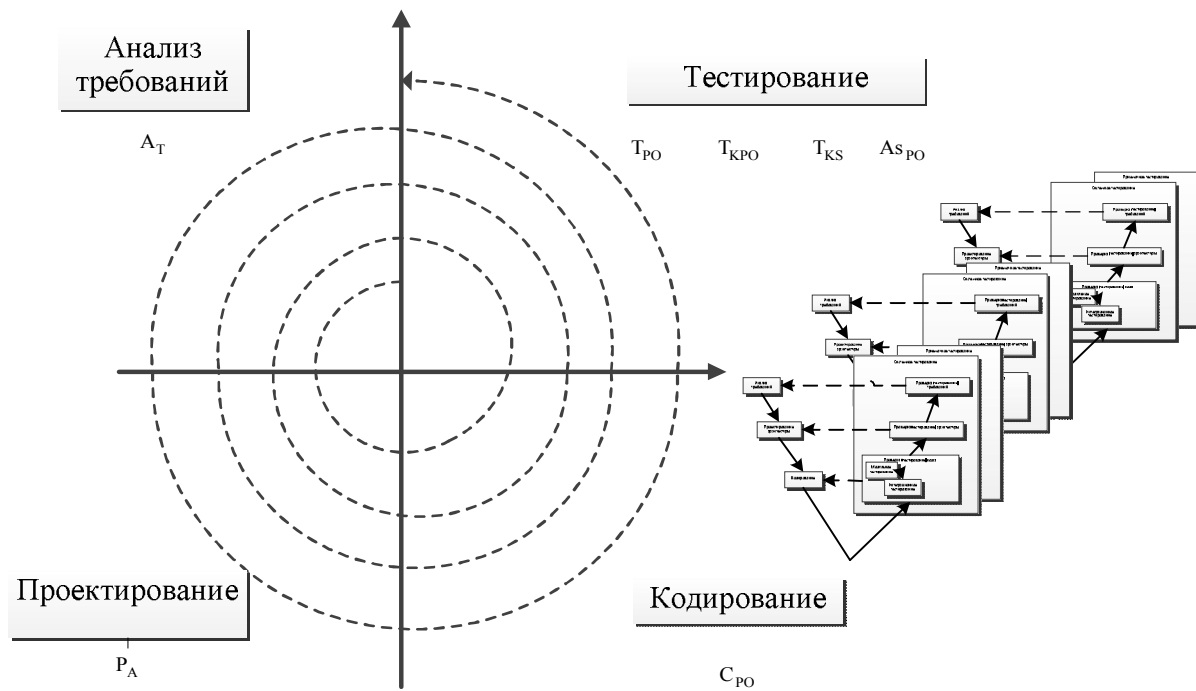


Рис. 3. Спиральная модель разработки ПО ИТКС

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: учеб. пособ. / И.В. Степанченко. – Волгоград: ВолГТУ, 2006. – 74 с.
2. Котляров В.П. Основы тестирования программного обеспечения: учеб. пособ. / В.П. Котляров, Т.В. Коликова. – М.: Интернет-Университет Информационных технологий, 2006.
3. Вендров А.М. Проектирование программного обеспечения экономических информационных систем / А.М. Вендров. – М.: Финансы и статистика, 2000. – 352 с.

4. Орлов С.А. Технологии разработки программного обеспечения: учебн. / С.А. Орлов. – 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: учеб. пособ. / И.В. Милованов, В.И. Лоскутов. – Тамбов: Издательство ТГТУ, 2011. – 88 с.

7. Стандартизация разработки программных средств: учеб. пособ. / В.А. Благодатских, В.А. Волнин, К.Ф. Посакалов и др.; 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.

Поступила в редколлегию 3.10.2014

Рецензент: д-р техн. наук, проф. Г.А. Кучук, Харьковский университет Воздушных Сил им. И. Кожедуба, Харьков.

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.