

## УПРАВЛЕНИЕ ПРЕЕМСТВЕННОСТЬЮ И ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

© Калмыков А., 2012

Розглянуто питання забезпечення наступності розробки ІТ-систем в багатостадійних програмах розвитку. Досліджуються спадкоємність у проектуванні, реалізації, розгортанні, супроводженні систем згідно з NGOSS концепцією життєвого циклу. Запропонована класифікація задач збереження наступності з урахуванням етапів проектування, розробки та впровадження систем. На основі стратифікації процесів проектування та класифікації відповідних задач запропоновано підхід до управління наступністю розробки інформаційних систем.

**Ключові слова:** наступність розробки, ІТ-систем, життєвий цикл, страт проектування.

The paper is dedicated to questions of the succession granting in multi-stage development programs of IT-systems. It investigates succession in the design, implementation, deployment, maintenance according to the NGOSS life-cycle concept. The paper presents tasks classification for the procuring of the succession in systems development, taking into account the stages of design, development and implementation of systems. The approach to the succession management in the information systems development is proposed on the basis of the stratified design processes and classification of the related tasks.

**Key words:** the succession of development, IT-systems, life cycle, design strata.

### Введение

Современные информационные решения различного назначения, как правило, строятся на принципах «открытых» систем, допускающих их расширение и развитие на основе уже существующих и эксплуатируемых компонентов. Разработка и внедрение новых решений, а особенно миграция на них действующих процессов, хранимых данных, является очень затратной и болезненной процедурой. Иногда стоимость таких мероприятий и связанные с ними неизбежные потери сопоставимы с прогнозируемой выгодой от новшеств. В то же время отложенный переход на новые принципы построения систем потенциально может привести к еще большим издержкам при развитии и внедрении данного решения. Возможности своевременного и эффективного развития системы определяются не только технологическими факторами, заложенными архитектурными принципами, но и организацией процессов разработки и сопровождения.

Одним из направлений решения данной задачи является обеспечение преемственности процедур разработки и сопровождения систем, повторное использование компонентов и частей, имеющих положительный опыт внедрения и эксплуатации.

### Известные подходы

Повторное использование компонентов, проектных решений, процедур разработки и сопровождения является широко распространенной практикой во многих областях техники. Данный подход востребован в высокотехнологичных сферах деятельности: машиностроение, отрасли, связанные с информационными технологиями. В последнее время вопросы обеспечения повторного использования компонентов находятся в фокусе внимания исследований в области управления разработкой программного обеспечения [1]. В ряде работ рассматриваются и предлагаются различные принципы классификации, определения спецификаций [2], формального описания

повторно используемых частей программных решений [3], методы оценки возможностей многократного применения компонентов в различных программных приложениях [5, 6]. Исследуются различные подходы к организации разработки и повторного использования компонентов систем на основе принципов «build-by-reuse», «build-for-reuse».

Также в разработке IT-решений применяется подход SOA (Service Oriented Architecture), в основе которого лежит идея реализации части функциональности системы как независимого элемента. Элемент не имеет каких-либо предварительных связей и зависимостей с использующей его системой и может использовать независимую базовую платформу [1, 7]. Повторное использование данного элемента выглядит как вызов внешней услуги или функции. Дальнейшее развитие этого принципа приводит к идее компонентного программирования, которая предполагает создание новых систем как набора «строительных блоков».

Несмотря на интенсивное исследование данных вопросов, большинство авторов рассматривает только технологические вопросы разработки систем на основе повторного использования частей и подсистем, либо же процессы управления разработкой только с точки зрения исполнителя. Поэтому при всех преимуществах и достоинствах вышеперечисленных методик и подходов, они являются лишь односторонним взглядом на задачу обеспечения повторного использования компонентов и частей программного обеспечения и IT-систем.

Понятие преемственности развития охватывает гораздо больший спектр вопросов. В широком смысле под преемственностью понимается связь между процессами и объектами в развитии систем, при котором сохраняют наиболее удачные и эффективные элементы, подходы, процедуры и процессы. Поэтому фактор преемственности является актуальным не только для разработчиков, но и для пользователей и заказчиков, осуществляющих эксплуатацию и определяющих требования к развитию используемых информационных систем. К сожалению, к настоящему времени не сформирован единый системный взгляд на проблемы обеспечения преемственности в развитии информационных систем, хотя потребность в исследовании таких вопросов возрастает. Например, многие предприятия уже сейчас сталкиваются с необходимостью миграции существующих систем на новые технологии, что приводит к возникновению серьезных вопросов как в сфере разработки, сохранении уже существующей функциональности, так и в области обеспечения преемственности процедур эксплуатации и сопровождения.

### **Постановка задачи**

Текущая ситуация в исследовании задач и вопросов, относящихся к обеспечению преемственности развития информационных систем в течение их жизненного цикла, вероятно, обусловлена пока еще недлительными сроками реальной эксплуатации действительно больших информационных систем и достаточными объемами ресурсов, выделяемыми для качественного сопровождения IT-систем. Однако, с ростом количества внедрений конкретной системы, резко увеличивается объем поддерживаемой функциональности. Это, в свою очередь, приводит к росту сложности и объемов работ по сопровождению и развитию таких систем. Поэтому рациональная организация процесса разработки и сопровождения, обеспечение унификации и преемственности развития IT-систем являются насущной задачей для предприятий–разработчиков и поставщиков и заказчиков-пользователей информационных систем.

Данные вопросы являются особенно актуальными для телекоммуникационных компаний, так как именно в этой отрасли происходит ускоренная смена поколений и технологий информационных систем, качество и сама возможность предоставления услуг напрямую зависит от корректности миграции на новые версии систем, стабильности бизнес-процессов. Подходом, учитывающим особенности разработки и внедрения IT-систем, считается применяемая для управления развитием IT-инфраструктуры телекоммуникационной отрасли методология TMF NGOSS [8, 9], согласно которой развитие инфраструктуры рассматривается с учетом позиций внутренних и внешних сторон, особенностей реализации решений на каждом этапе жизненного цикла.

Предлагается рассмотреть проблематику обеспечения повторного использования компонентов, процедур и процессов с позиций управления проектами и программами развития информационных систем на основе модели жизненного цикла и с учетом влияния заинтересованных сторон, вовлеченных в эти процессы.

## Декомпозиция процессов развития IT-систем на основе положений методологии NGOSS

Рассмотрим декомпозицию процессов разработки систем, их долгосрочного развития и эксплуатации на примере IT-инфраструктуры телекоммуникационного предприятия, как наиболее подробно исследованной в документах TMF. В рамках методологии NGOSS предложен ряд моделей и инструментов проектирования информационных систем поддержки деятельности. Наряду с определением принципов декомпозиции систем, используемых в основной производственной деятельности операторов связи и соответствующих им задач проектирования, предлагается модель жизненного цикла компонентов IT-инфраструктуры и определяются позиции заинтересованных сторон на каждом из его этапов. Рассматриваемая методология также полностью соответствует принципам обеспечения преемственности развития систем, так как, предусматривает декомпозицию сложного объекта (системы) на относительно слабосвязанные функциональные элементы [10, 11], пригодные для повторного применения в других проектах, разработках. Методология NGOSS позволяет рассмотреть развитие системы в течение всего ее жизненного цикла с учетом интересов разработчиков-исполнителей, пользователей-заказчиков, что необходимо для формирования системного подхода к управлению преемственностью разработки IT-систем.

В соответствии с концепцией NGOSS (см. рис. 1) заказчик определяет цели и задачи, бизнес-процессы разрабатываемой системы (бизнес ракурс) и практическую реализацию, используемое программное и аппаратное обеспечение (ракурс развёртывания). Исполнитель определяет архитектуру, структуру, состав, внешние и внутренние связи (системный ракурс), конкретные особенности и детали реализации информационной системы, состав и спецификации интерфейсов, принципы организации и обработки данных (ракурс внедрения).

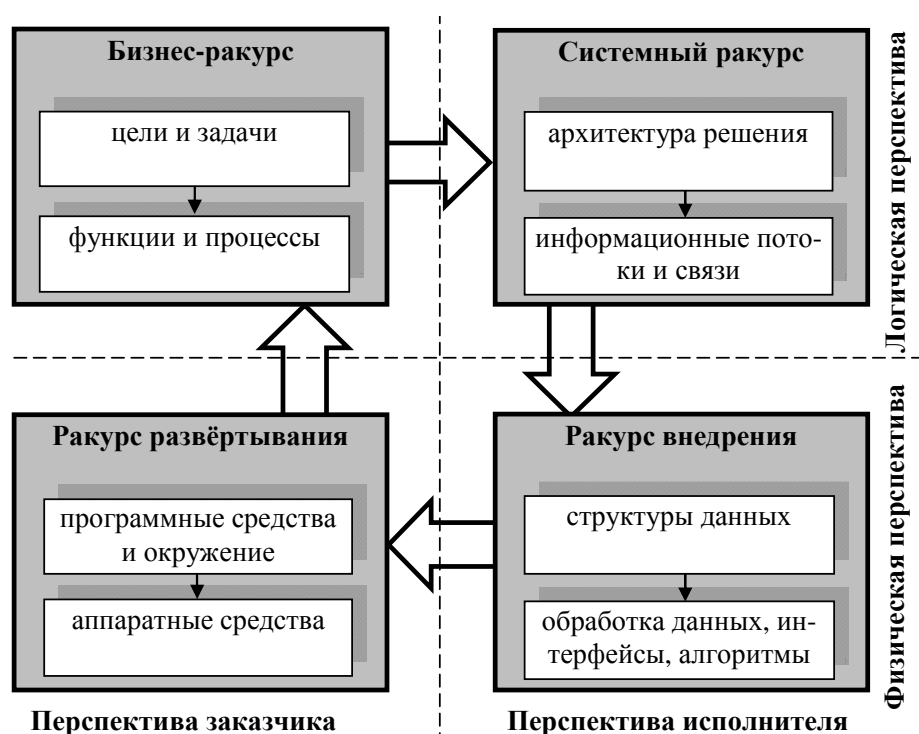


Рис. 1. Жизненный цикл IT-систем в соответствии с TMF NGOSS.

На рис. 1 эти отношения определены как перспектива заказчика и перспектива исполнителя. Кроме того, на основе модели жизненного цикла возможно разделение на технологически независимые аспекты (логическая перспектива) IT-систем и технологически зависимые (физическая перспектива) компоненты. Конечно, нельзя утверждать, что на каждом их этапов жизненного цикла выполняются проектные действия, строго соответствующие описанным перспективам заинтересованных сторон и технологическим зависимостям, но фокус деятельности и разработчика, и заказчика находится именно в данных областях (см. табл. 1).

## Классификация задач обеспечения преемственности на основе положений методологии NGOSS

В работе [12] предложено представление процесса разработки и развития ИТ-системы как последовательность действий в пространстве проектных действий относительно ракурсов жизненного цикла NGOSS. На рис. 2 показано пространство проектных действий в измерениях ракурсов бизнеса, системы, внедрения (реализации) и развертывания ИТ-системы.

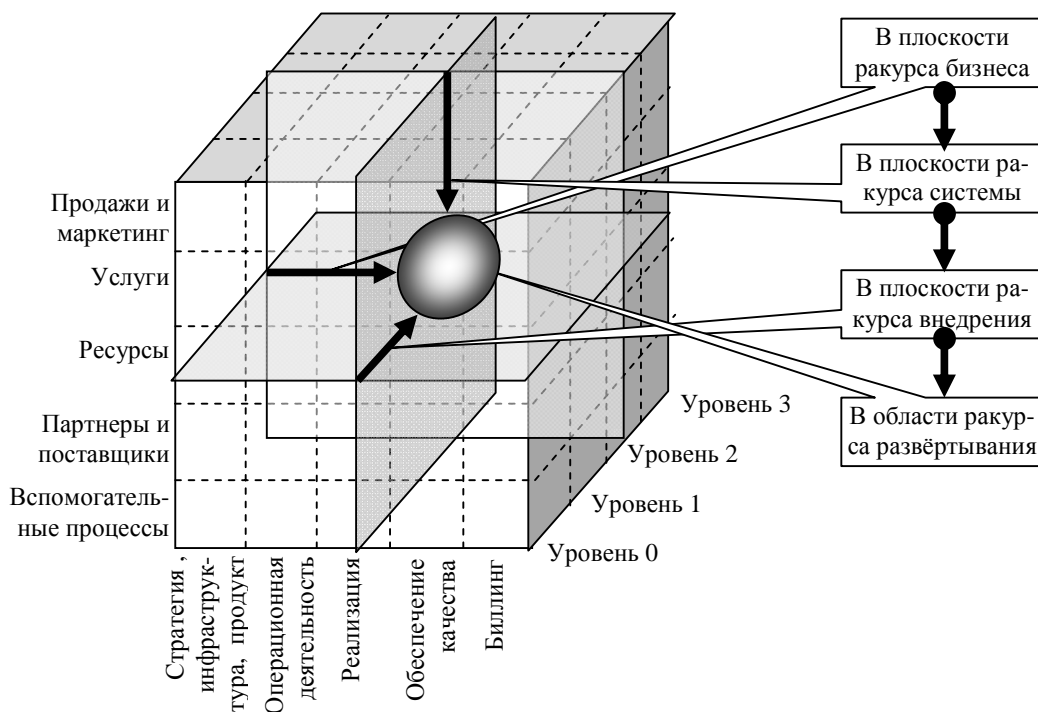


Рис. 2. Действия в пространстве задач проектирования согласно жизненного цикла NGOSS

Базовыми моделями для определения выполняемых действий относительно рассматриваемых измерений (ракурсов) являются известные концепции NGOSS [11] – TAM, eTOM:

1. В ракурсе бизнеса определяется положение выполняемых действий по разработке, внедрению, сопровождению системы относительно цепочки создания ценности согласно модели eTOM с тем, чтобы определить цели, задачи и общую функциональность. При этом определяются требования к преемственности разрабатываемого решения относительно обеспечиваемых бизнес-процессов, в том числе и совместно выполняемых с другими ИТ-системами.

2. В системном ракурсе определяется положение в плоскости уровней деятельности с тем, чтобы согласно модели TAM определить функциональную архитектуру решения, его связь с другими подсистемами, участвующими в решении задачи и достижении поставленных целей, в контексте обеспечения преемственности при взаимодействии с другими компонентами и подсистемами ИТ-инфраструктуры предприятия.

3. В ракурсе внедрения проектные действия выполняются в плоскости уровней детализации, что связано с реализацией системы, программированием, конфигурированием приложений. Требования к преемственности развития ИТ-системы относительно данного ракурса связаны с обеспечением повторного использования уже существующих и успешно используемых компонентов, модулей, интерфейсов, протоколов, средств и методик разработки, тестирования.

4. В результате действий аналитиков, архитекторов, разработчиков относительно ракурсов бизнеса, системы и внедрения создается реальная информационная система, имеющая конкретные спецификации архитектуры, кода, интерфейсов информационного обмена. Такая система, как правило, требует дополнительных действий, связанных с её развертыванием в реальном, действующем ИТ-окружении предприятия. Уточняется положение и взаимодействие разрабатываемой и

внедряемой системы с существующей инфраструктурой и окружением. Преимущество системы в данном случае предполагает использование известных внешних систем и компонентов, стабильность процедур и процессов.

Предлагаемая декомпозиция процессов развития ИТ-инфраструктуры предприятия телекоммуникаций, анализ требований к преемственности, основанные на использовании моделей NGOSS, применимы к любой другой информационной системе. Основными особенностями предлагаемого подхода является следующая последовательность действий:

бизнес-ракурс, применение модели цепочки создания ценности для анализа вопросов преемственности в организации и поддержке производственных процессах;

системный ракурс, «вертикальная» декомпозиция производственной деятельности с целью определения позиции рассматриваемой системы в иерархии производственных систем и выявления проблем преемственности, связанных с организационным и технологическим управлением ИТ-системой и ее взаимодействием с внешним и внутренним окружением;

ракурс внедрения, декомпозиции системы «в глубину» для определения составляющих ее компонентов и соответствующих им процедур, допускающих повторное использование при наличии положительного опыта применения в аналогичных решениях;

ракурс развертывания, уточнение и «сужение» области проектных решений разрабатываемой системы, фактическое определение и решение проблемных моментов в обеспечении преемственности ИТ-системы при ее фактическом развертывании по всем ранее рассмотренным направлениям: бизнес, система, внедрение (реализация).

Основываясь на таком подходе к декомпозиции процессов развития, становится возможным формирование требований к преемственности ИТ-систем, как с учетом интересов различных сторон, так и с учетом динамики развития таких объектов. В Табл. 1 приведена классификация основных задач, решаемых в рамках обеспечения преемственности системы на всех этапах ее жизненного цикла.

Таблица 1

**Классификация задач обеспечения преемственности процессов в развитии ИТ-системы**

	Перспектива заказчика	Перспектива исполнителя
Логическая перспектива	Преемственность бизнес-процессов предприятия Преемственность процедур и политик обслуживания ИТ-системы.	Преемственность процессов анализа требований, согласования изменений.
Физическая перспектива	Преемственность процедур развертывания и внесения изменений	Стабильность процедур и технологий разработки.

Рассматривая преемственность развития системы как набор обеспечивающих действий и процедур, на каждом этапе жизненного цикла выделяются соответствующие процессы. Это помогает в определении причинно-следственных связей в задачах обеспечения преемственности развития на протяжении всего жизненного цикла системы.

Также предлагаемая классификация задач позволяет разделить вопросы преемственности на аспекты (процессы), оперирующие с относительно постоянными физическими компонентами, и на логические аспекты, изменение которых может быть не так болезненно с точки зрения реализации новых требований и новой функциональности.

**Стратификация процессов разработки и внедрения ИТ-систем**

Декомпозиция общей архитектуры информационных систем на постоянную функциональную часть, и полупостоянную, реализующую изменения в требованиях и процессах заказчика позволит определить ключевые моменты в разработке и дальнейшем развитии ИТ-системы. Корректно спроектированные и реализованные постоянные (физические) уровни архитектуры решения обеспечивают эволюционное развитие и многоуровневое использование компонентов системы. Выделение логических уровней архитектуры ИТ-системы позволяет своевременно и гибко реаги-

ровать на изменяющиеся требования заказчиков. Кроме того, декомпозиция архитектуры решения на физические и логические составляющие позволяет применить различные подходы к их разработке и сопровождению, привлекая различные по уровню подготовки и квалификации человеческие ресурсы, что в целом повышает эффективность таких процессов. Подобные идеи достаточно широко применяются в современных производственных системах телекоммуникационных операторов [4, 12]. При этом выделяются:

- уровень приложений (фактически интерфейс конечного пользователя);
- уровень конфигурируемых бизнес-процессов;
- уровень специализированного программного обеспечения, реализующего базовую функциональность для построения бизнес-процессов и логики обработки данных или событий (базы данных, интерфейсы, протоколы, обработчики потоков событий и т.д.);
- системные программно-аппаратные платформы, обеспечивающие функционирование и необходимые физические ресурсы для верхних уровней информационной системы.

Фактически подобная декомпозиция соответствует основным аспектам жизненного цикла согласно NGOSS:

- уровень приложений соответствует внешнему представлению бизнес-процессов, маркетинговым вопросам предоставления услуг, то есть бизнес ракурсу NGOSS;
- уровень бизнес-процессов соответствует системному ракурсу NGOSS, который предполагает определение принципов и логики обработки данных, структуру основных бизнес процессов;
- уровень базового специализированного программного обеспечения соответствует ракурсу внедрения. На этом уровне определяются протоколы, интерфейсы, структуры и архитектуры актуальных баз данных;
- уровень системно-аппаратных платформ соответствует ракурсу развертывания, аспектам физической реализации информационной системы.

Проектирование и разработка сложных систем на основе различных подходов к декомпозиции их свойств и функций рассмотрены во многих работах отечественных и зарубежных авторов [13, 14]. Построение таких процессов основывается на декомпозиции системы на страты, соответствующие разным классам характеристик и свойств: целевая, функциональная, информационная, структурная, данных, алгоритмическая, ПО, КТС (см. рис. 3).

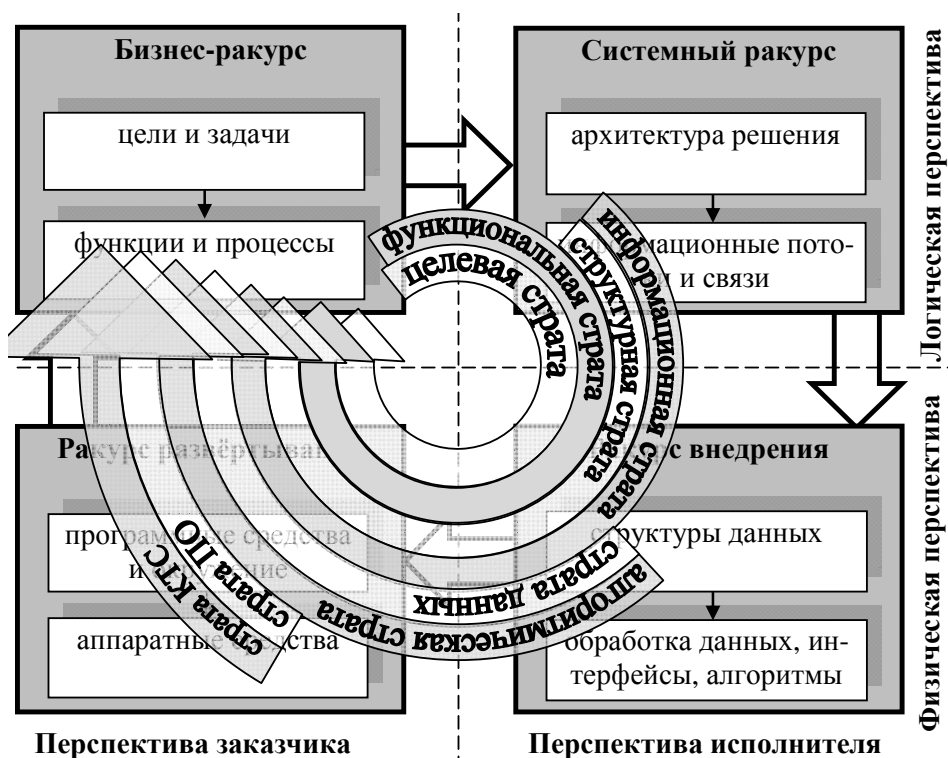


Рис. 3. Стратифицированное представление IT-систем и жизненный цикл NGOSS

На рис. 3 показаны страты (классы свойств и характеристик) ИТ-системы относительно ракурсов жизненного цикла NGOSS. Логическое проектирование выполняется по стратам верхних уровней, и его результаты учитываются на всех ракурсах жизненного цикла ИТ-системы, на каждом последующем этапе происходит уточнение и детализация разрабатываемой системы.

Из рисунка видно, что разработка и развитие ИТ-систем имеют циклический, повторяющийся характер, причем действия по стратам верхнего уровня выполняются на протяжении всего жизненного цикла системы. Это связано с необходимостью постоянно отслеживать возможные изменения целей и функций системы, обусловленные требованиями заказчика, координировать проектные действия и оценивать их результаты. Такие же действия выполняются и на последующих стратах проектирования ИТ-системы, но для соответствующего перечня задействованных ракурсов жизненного цикла системы.

На основании предложенной стратификации ИТ-системы и разделения аспектов системы на логический и физический ракурсы в Табл. 2 предлагается классификация задач обеспечения преемственности свойств, характеристик (артефактов) ИТ-системы.

Таблица 2

**Классификация задач обеспечения преемственности свойств (артефактов)  
в развитии ИТ-системы**

Страты	Логическая перспектива (ракурсы бизнес и системы)	Физическая перспектива (ракурсы внедрения и развёртывания)
Целевая	Сохранение изначальных целей и задач системы	–
Функциональная	Сохранение лучшей существующей функциональности	–
Информационная	Сохранение необходимых данных, предыстории	–
Структурная	Преемственность архитектуры при смене используемых технологий, расширении. Использование принципов TNA.	–
Данные	Преемственность интерфейсов и протоколов взаимодействия	Преемственность структур и спецификаций данных
Алгоритмическая	Преемственность описания бизнес-процессов, повторное использование	Преемственность описания алгоритмов, повторное использование алгоритмов.
Программное обеспечение	Обеспечение преемственности бизнес-задач, архитектуры системы	Использование известных и ранее применявшихся технологий, средств разработки, внешних программных систем и решений.
Технические средства	Обеспечение преемственности бизнес-задач, архитектуры системы	Преемственность используемых аппаратных средств

**Выводы**

Таким образом, предлагается подход к управлению преемственностью развития ИТ-систем в многоэтапных долгосрочных программах, состоящий в следующей последовательности действий:

1. Определение позиции и места разрабатываемой системы в бизнес-процессах предприятия, взаимодействии с внешними компонентами инфраструктуры.

2. Классификация процессов в задачах обеспечения преемственности с позиций исполнителя (поставщика) и заказчика, разделение логических аспектов проектирования ИТ-системы и вопросов, относящихся к физической реализации.

3. Управление решением задач преемственности с учетом стратификации ИТ-систем (разделение на классы свойств, характеристик, артефактов) и относительно этапов жизненного цикла.

Подход, предложенный в данной статье, позволяют классифицировать задачи обеспечения преемственности в многоэтапных программах развития информационных систем. На основе концепции жизненного цикла систем методологии NGOSS рассмотрены позиции заказчика и

исполнителя, возможности влияния на процессы разработки и модификации систем, выделены фазы логического проектирования и физической реализации.

Циклический характер жизненного цикла информационных систем позволяет определить процессы и артефакты, определяющие требования к преемственности. Использование стратификации позволяет декомпозировать процессы и задачи по обеспечению преемственности развития ИТ-систем относительно уровней свойств, характеристик и соответствующих им процессов разработки таких объектов.

Перспективным направлением развития предлагаемых подходов является формирование на их основе практических рекомендаций для использования в управлении реальными проектами разработки ИТ-систем.

1. *Reference Model for Service Oriented Architecture 1.0. Committee Specification 1, 2 August, 2006 / OASIS Open, 2006. – 31 p.*
2. Penix, J. *Classification and Retrieval of Reusable Components Using Semantic Features / John Penix, Phillip Baraona, Perry Alexander – 1068-3062/95 \$4.00 © 1995 IEEE, 1995. – pp.131-138.*
3. Sametinger, J. *Software Engineering with Reusable Components / Johannes Sametinger – Springer-Verlag, 1997. – 272 p.*
4. Mohagheghi, P. *The Impact of Software Reuse and Incremental Development on the Quality of Large Systems: Doctoral Thesis / Parastoo Mohagheghi – Trondhei : Norwegian University of Science and Technology, 2004. – 272 p.*
5. Kontio, J. *Defining Factors, Goals and Criteria for Reusable Components Evaluation /Jyrki Kontio, Gianluigi Caldiera, Victor R. Basili – Cascon Conference, Toronto Canada, November 14-16, 1996. – pp. 211-223.*
6. Bassam, A. *Reusable Software Component Life Cycle / Anas Bassam AL-Badareen, Mohd Hasan Selamat, Marzanah A. Jabar, et.al – NAUN, International Journal of Computers, issue 2, vol.5, 2011. – pp.191-199.*
7. Kruger, I.H. *Applying Service-Oriented Development to Complex Systems: BART case study. / Ingolf H. Kruger, Michael Meisinger, Messimiliano Menarini. – Proceedings of the 12th Monterey conference on Reliable systems on unreliable networked platforms. – Berlin : Springer-Verlag, Heidelberg ©2007. – pp.26–46.*
8. *New Generation Operational Support Systems (“NGOSS”). Architecture Overview. Public Version 1.500 [Электронный ресурс] / The TeleManagement Forum. – 2003. – Режим доступа : \www/ URL : <http://www.tmfforum.org/sdata/documents/TMFC763%20GB920v1.5.pdf> – 10.11.2011 г. – Загл. с экрана.*
9. *GB 927.The NGOSS Lifecycle and methodology. Version 1.1 [Электронный ресурс] / The TeleManagement Forum. – 2004.– Режим доступа : \www/ URL : <http://www.itarchitects.ca/whitepaper/The%20NGOSS%20Lifecycle%20and%20Methodology.pdf> – 15.05.2010 г. – Загл. с экрана .*
10. *GB 921.Enchanced Telecom Operations MAP (eTOM). The Business Process Framework. TMF Approved Version 4.0 [Электронный ресурс] / The TeleManagement Forum. – 2004. – Режим доступа : \www/ URL: [http://www.idef.ru/documents /tmfc2495\\_gb921\\_v4-0-1\\_040318.pdf](http://www.idef.ru/documents /tmfc2495_gb921_v4-0-1_040318.pdf) – 15.05.2011 г. – Загл. с экрана.*
11. Райли, Дж. *NGOSS: Построение эффективных систем поддержки поддержки и эксплуатации сетей оператора связи [текст] / Дж. Райли, М.Кринер. – пер. с англ. – М. : Альпина Бизнес Букс, 2007. – 192 с.*
12. Калмыков А.В., *Системное представление разработки информационных решений поддержки производственной деятельности / А.В. Калмыков, О.Н. Воскобойник // Восточно-Европейский журнал передовых технологий. – 2011. – № 4/2(52). – С. 21-27.*
13. Илюшко, В. М. *Системное моделирование в управлении проектами [текст] : монография / В.М. Илюшко, М.А. Латкин. – Харьков : НАУ «ХАИ», 2010. – 220 с.*
14. Слюсаренко, М.Ю. *Системологический подход к декомпозиции в объектно-ориентированном анализе и проектировании программного обеспечения [Электронный ресурс] / И.М. Слюсаренко, М.Ю. Слюсаренко. – Режим доступа : \www/ URL : [http://citforum.ru/programming/case/ooad\\_systemology](http://citforum.ru/programming/case/ooad_systemology).*