Принципы и подходы к обеспечению функциональной безопасности компонентов вычислительно-коммуникационных систем.

О.Я. Бежаева

***Аннотация* — В настоящей статье сформулированы принципы и подходы обеспечению функциональной безопасности компонентов вычислительно-коммуникационных систем. Рассмотрены свойства дефектов как разновидности сложных управляемых систем. Концептуальную основу исследований составляет системное сочетание проактивного, активного и реактивного подходов к управлению дефектами разной природы. Рассмотрение дефектов как разновидности сложных систем создает методологическую основу для научно-обоснованной адаптации подходов, методов и моделей, хорошо зарекомендовавших себя при решении задач управления сложными системами иной природы, в область управления функциональной безопасностью. В качестве примера в работе показаны архитектурные модели, характеризующие различные аспекты функциональной безопасности. Архитектурные модели создают основу решения прямых задач обеспечения функциональной безопасности: исследование проблемных ситуаций, обусловленных наличием различных дефектов в организации проектов и в программных продуктах.**

В рамках системного подхода сформулированы концептуальные основы обеспечения функциональной безопасности компонентов вычислительно-коммуникационных систем.

***Ключевые слова* — сложные системы, функциональная безопасность, субъектоцентрические системы, дефекты, вычислительно-коммуникационные системы, аппаратно-программные комплексы.**

I. Введение

Обеспечение функциональной безопасности компонентов вычислительно-коммуникационных систем относится к числу критических факторов успеха реализации положений доктрины Industry 4.0. [1] Качественное возрастание роли систем обработки данных в управлении современными распределенными техническими системами выдвигает данную проблему на первый план.

В работах многих ученых подчеркивается необходимость смещения акцентов в проблематике создания компонентов аппаратно-программных комплексов от вопросов штатной эксплуатации к вопросам их безопасного функционирования.

Изменения роли информационных систем и их трансформация в системообразующий фактор цифровой экосреды делают необходимым приоритетное развитие методологических, теоретических и модельных основ управления функциональной безопасностью компонентов вычислительно-коммуникационных систем.

Под функциональной безопасностью компонентов вычислительно-коммуникационных систем понимается свойство сохранять работоспособность в соответствии со своим целевым назначением при случайных дестабилизирующих воздействиях и отсутствии злоумышленного влияния на программную, аппаратную составляющую.

Системообразующим фактором функциональной безопасности являются дефекты. Источниками дефектов являются ошибки, совершаемые людьми, на различных стадиях жизненного цикла. Дефекты – неотъемлемая составляющая субъектоцентрических систем. Это утверждение обосновывается в работах [2, 3]. В силу того, что аппаратно-программные комплексы (АПК) являются разновидностью субъектоцентрических систем, можно утверждать, что наличие латентных дефектов как в аппаратной, так и в программной компонентах, является неотъемлемой особенностью компонентов вычислительно-коммуникационных систем. Разные дефекты имеют различную природу возникновения. Характер дефектов зависит от того, на какой стадии жизненного цикла объекта они возникли.

Дефекты играют противоречивую роль в обеспечении функциональной безопасности компонентов вычислительно-коммуникационных систем: с одной стороны проявления дефектов негативно сказывается на функциональной безопасности. С другой – стимулирует коэволюционное развитие методологических, теоретических основ, инструментальных средств обеспечения функциональной безопасности соответственно изменениям масштабов и сложности вычислительно-коммуникационных систем.

В целом, к настоящему времени не сформированы теоретические основы построения «сквозной» системы управления дефектами на разных стадиях жизненного цикла компонентов вычислительно-коммуникационных систем объединяющей в себе реактивный и проактивный подходы.

Статья получена 28 декабря 2021.

Работа поддержана грантом Российского Фонда Фундаментальных Исследований №19-08-00177 «Методологические, теоретические и модельные основы управления функциональной безопасностью аппаратно-программных комплексов в составе распределенных сложных технических систем»

О. Я. Бежаева, Уфимский государственный авиационный технический университет, Уфа, Россия (e-mail: obezhaeva@gmail.com)

II. Принципы обеспечения функциональной безопасности

А. Общесистемные принципы

1) Принцип системности.

Согласно этому принципу, система обеспечения функциональной безопасности, с одной стороны, является сложной субъектоцентрической системой. С другой – подсистемой системы информационной поддержки реализации организации.

2) Принцип полиморфизма.

Этот принцип указывает на разнообразие форм элементов системы и связей между ними. Например, на разнообразие причин возникновения и форм латентных дефектов в АПК [4 - 7], а также особенностей причинно-следственных связей между разными формами дефектов на разных стадиях жизненного цикла с точки зрения конечного результата дефекты в постановках задач являются причинами дефектов в алгоритме решения задачи, которые, в свою очередь являются причиной дефектов в кодах). Противоположным принципу полиморфизма является принцип изоморфизма. Суть этого принципа сходится к тому, что единообразие форм описания систем разной природы, схожесть структур и свойств системных характеристик обосновывает возможность использования схожих формальных приемов исследования разных систем посредством знаковых моделей.

Например, дефекты разной природы при исследованиях субъектоцентрических систем в области здравоохранения и реализации инфраструктурных компонентов распределенных вычислительно-коммуникационных систем в рамках известной платформы Swiss Cheese Model [2] могут рассматриваться как источники опасности (hazard), либо как «дыры» (holes) в защитных барьерах. Принцип изоморфизма, в частности, обуславливает возможность использования архитектурного подхода и системных архетипов в задачах, связанных с обеспечением функциональной безопасности АПК.

3) Принцип многообразия.

Данный принцип означает наличие множества форм (морфизмов) ошибок, совершаемых субъектами при разработке АПК на разных стадиях их жизненного цикла, и обусловленных этими ошибками дефектов. Частым проявлением принципа многообразия является принцип многокритериальности, согласно которому в рамках обеспечения функциональной безопасности АПК может оптимизироваться по многим частным критериям (например, затраты на тестирование-возможность проявление латентного дефекта). Одновременно, один и тот же критерий (например, доля не выявленных латентных дефектов [8, 9]) может использоваться при принятии решений о целесообразности продолжения тестирования модулей различного функционального назначения.

4) Принцип декомпозиции.

Применительно к задачам обеспечения функциональной безопасности означает возможность выделения на каждой стадии реализации проекта создания АПК основных факторов (концептов), времени и мест возникновения дефектов; определить ресурсы и способы позволяющие предупредить возникновение/ устранить дефекты.

Декомпозиция по горизонтали позволяет исследовать дефекты в продуктах проекта (в том числе промежуточных). Декомпозиция по вертикали позволяет исследовать дефекты в системе управления проектом.

5) Принцип интеграции (композиции).

Данный принцип заключается в возможности построения когнитивных моделей, использование которых позволяет обеспечить выбор рациональной стратегии обеспечения функциональной безопасности.

6) Принцип эквивалентных путей достижения цели.

В силу неопределенности состояния в каждый момент времени как самого проекта, так и его продукта (см.модель «Конус неопределенности» [11]), невозможна реализация принципа единственности Р.Колмана при построении моделей, ориентированных на решение задач обеспечения функциональной безопасности. Следует исходить из того, что приемлемый уровень функциональной безопасности может быть обеспечен различными способами, реализуемыми в рамках реактивного и проактивного подходов. При этом нужно учитывать различную результативность и эффективность способов.

7) Принцип системной готовности.

Применительно к задачам обеспечения функциональной безопасности это означает, что опасность (латентный дефект) трансформируется в негативные последствия если одновременно в одном месте, возникают сочетание условий, необходимое для проявления дефекта и реализации в последующем «принципа домино». Из этого следует, что базовым принципом парирования дефектов является исключение возможности возникновения сочетания условий, при которых дефект активизируется и при которых возможна реализация «принципа домино».

В. Принципы проектирования систем обеспечения функциональной безопасности.

1) Принцип адекватности.

Означает, что усилия по предотвращению возникновения дефектов/ парирования их проявлений должны быть адекватны тяжести последствий, связанных с появлением дефектов. Так затраты на испытания и технологии испытаний должны быть адекватны негативным последствиям, обусловленным невозможностью получения информационного сервиса (информационного продукта). Кроме того, потенциальность стратегии испытаний на разных стадиях жизненного цикла АПК должна быть адекватно сложности объекта испытаний, по мере роста сложности объекта испытаний сложность условий испытаний должна возрастать.

2) Принцип согласованности.

Согласно этому принципу следует исходить из того, что задачи обеспечения функциональной безопасности имеют равную значимость с иными задачами проекта. Они должны реализовываться на всех стадиях проекта. Одной из задач организации проекта должна быть

связана с созданием системы мер, во-первых, препятствующих возникновению и способствующих выявлению и устранению дефектов на всех стадиях жизненного цикла продукта. Во-вторых, препятствующих миграции дефектов между фазами проекта.

Частными случаями принципа согласованности являются:

3) Принцип оптимальности.

Означает такое распределение ресурсов (материальных, финансовых, интеллектуальных) по задачам развития потенциальности информационной системы и обеспечения функциональной безопасности, при котором достигается наилучший баланс между основными характеристиками проекта: бюджетом, длительностью реализации, потребительскими свойствами продукта.

4) Принцип координации.

Означает синхронизацию процессов увеличения потенциальности информационной системы; обеспечения функциональной безопасности; соответствия потребительских свойств системы- миссии организации.

5) Принцип сбалансированности.

Означает согласованность усилий направленных, с одной стороны на решение задач обеспечения функциональной безопасности в рамках реализации текущих проектов, с другой – на создание новых подходов и совершенствование существующих с учетом изменений в масштабах, сложности объектов управления и парадигм управления сложными системами.

6) Принцип типизации и стандартизации.

Означает необходимость учета при решении задач обеспечения функциональной безопасности требований и ограничений, заложенных в профиле документов, определяющих организацию проекта. Кроме того, учет базовых требований к потребительским свойствам программных продуктов (определенных в [16]), а также программных проектов (базовыми документами в рамках архитектурного подхода в настоящее время являются [12, 13]).

7) Принцип реализуемости.

Означает, что подходы к выбору мер по предупреждению возникновения дефектов и их устранению должны выбираться, во-первых, с учетом назначения АПК, во-вторых, с учетом возможных последствий от проявления латентных дефектов разной природы, в-третьих, с учетом ограничений на ресурсы проекта.

III. Свойства дефектов как разновидности сложных управляемых систем

Общими свойствами дефектов АПК, являющихся компонентами в составе распределенных вычислительно-коммуникационных систем, объединяющих их со сложными системами иной природы, являются следующие:

- Основным фактором возникновения дефектов являются субъективные ошибки. Дефекты разной природы (ментальные, организационные, конструктивные, технологические), наряду с устройством системы являются критическими факторами, определяющими функциональную безопасность АПК.
- В субъектоцентрических системах (к числу которых относятся аппаратно-программные комплексы) неизбежно наличие латентных дефектов различной природы (это утверждение обосновывается, например, в работах J.Reason [2]). Дефекты являются неотъемлемой составляющей всех субъектоцентрических систем (к числу которых относятся АПК) в силу того, что человеку свойственно ошибаться.
- Характер дефектов неразрывно связан со сложностью систем: чем сложнее система, тем больше возможность возникновения в ней дефектов (это утверждение обосновывается, например, в [15].
- Дефекты являются динамическими объектами, их форма и содержание определяются стадией жизненного цикла и изменяются при переходе на иную стадию жизненного цикла.
- Дефекты обладают свойством наследования: дефекты, проявляющиеся на более поздних стадиях жизненного цикла, могут быть производными от дефектов, возникших на более ранних стадиях жизненного цикла. Моменты возникновения дефектов и проявления дефектов разнесены в пространстве и времени. На эту особенность дефектов в субъектоцентрических системах обращается внимание, например, в [2]. Разнесенность в пространстве и во времени моментов возникновения и проявления латентных дефектов затрудняет выявление коренных причин потери функциональности АПК, например, посредством Root Cause Analysys (RCA).
- Дефекты являются объектами, допускающими управление в рамках реактивного (устранение последствий проявившихся) и проактивного (препятствие распространения последствий) управления:
- Дефекты внутреннего устройства систем проявляются в их поведение в условиях динамически изменяющегося состояния окружающей среды, включая неспособность сохранять важные потребительские свойства при нерасчетных воздействиях.

Дефекты - сложные (многомерные) объекты управления. Для их предотвращения на разных стадиях жизненного цикла используются разные стратегии, подходы, модели и методы для достижения поставленных целей.

Метаморфоза дефектов показана на рисунке 1. Концептуальную основу модели составляют следующие положения:

- качество организации проекта определяет качество конечного продукта;
- дефекты, в продукте не выявленные на предыдущей фазе реализации проекта, являются причиной части дефектов на последующих фазах проекта.

Свойствами дефектов, отличающих их от сложных систем иной природы, являются:

- Принципиальная невозможность установления факта наличия всех дефектов в программных системах посредством специально организованных испытаний (на это обстоятельство обращалось внимание в [8]).

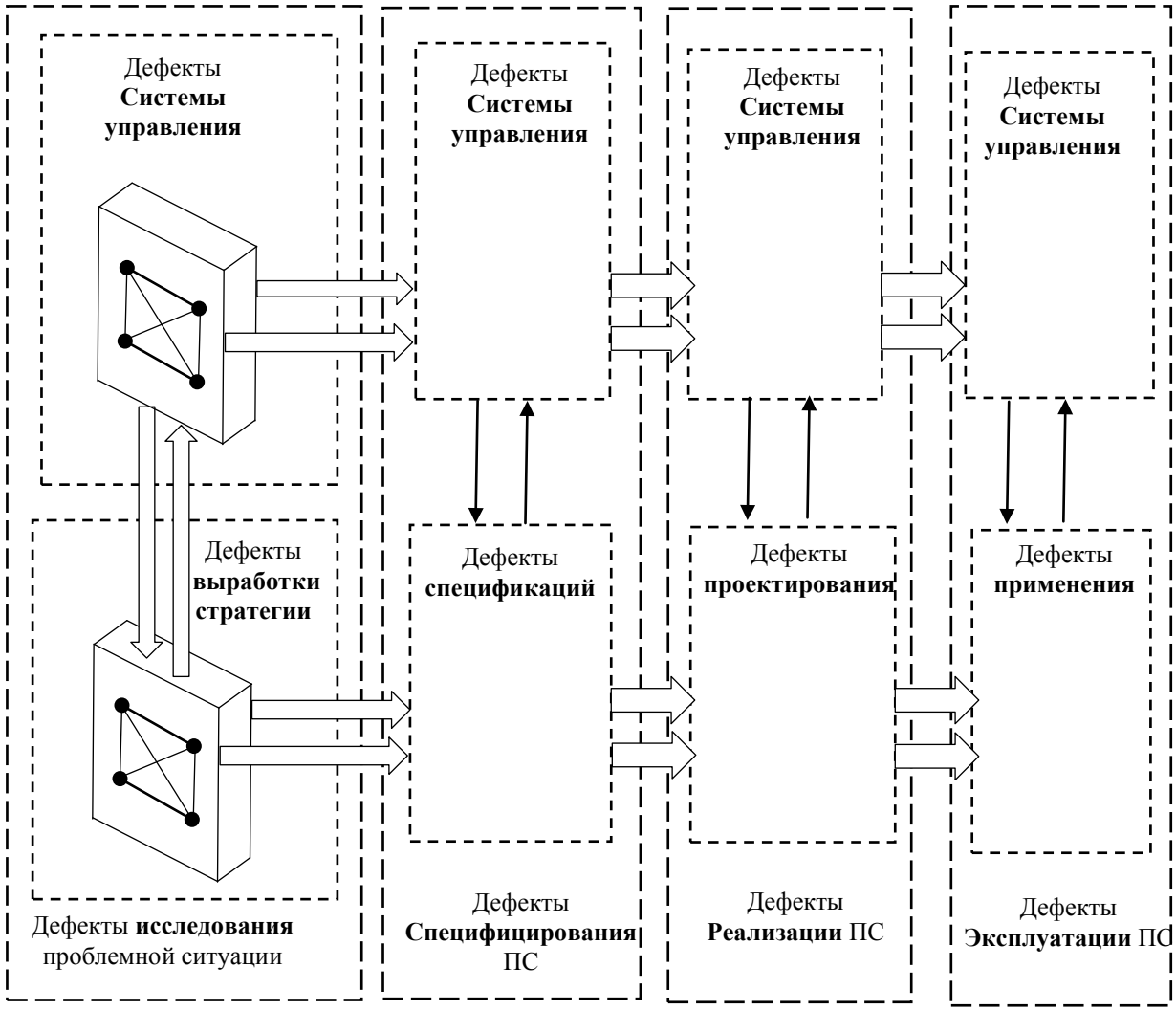

Рис.1 Метаморфоза дефектов.

- Неопределенность требований к свойствам АПК и неопределенность состояния внешней среды АПК в составе динамических распределённых вычислительно-коммуникационных систем препятствует вынесению заключения о функциональной безопасности АПК по результатам специально организованных испытаний.
- Отсутствие/малое число измерительных данных, характеризующих свойство функциональной безопасности в разных условиях использования, что обосновывает расширенное использование при исследовании функциональной безопасности методов структурного анализа, а также математико-статистических методов обработки малых выборок и экспертно-статистических методов.

IV. Архитектурные модели систем функциональной безопасности

Архитектурные модели систем функциональной безопасности являются интегральной характеристикой инфраструктурных компонентов цифровой экосреды. Факторами функциональной безопасности являются дефекты в организации проектов, результатом которых являются дефекты в промежуточных и конечных продуктах. Фундаментальной причиной возникновения дефектов являются ошибки, совершаемые людьми.

Согласно [12, 13], архитектура - фундаментальная организация системы, выражающаяся в её компонентах, их взаимоотношениях между собою и окружающей средой, а также руководящих принципах проектирования и преобразования системы.

Согласно принципу полиморфизма для описания одной и той же системы могут использоваться разные архитектурные модели, соответствующие, например, различным подходам к обеспечению функциональной безопасности (тестирование, сценарный подход, «барьерное мышление» [9, 10, 14]).

Архитектурные модели создают основу решения прямых задач обеспечения функциональной безопасности: исследование проблемных ситуаций, обусловленных наличием различных дефектов в организации проектов и в продуктах.

Архитектурные модели должны характеризовать следующие аспекты функциональной безопасности:

- *Методический:* понятия, содержание, факторы возникновения ошибок и дефектов, обоснование подходов к исследованию дефектов и построению их моделей, методические указания, руководства, стандарты.
- *Теоретический*: модели и методы исследования ошибок и дефектов как неотъемлемых компонентов субъектоцентрических систем.
- *Управленческий:* формирование альтернатив обеспечения функциональной безопасности в условиях неопределенности окружающей среды и внутреннего состояния информационных систем, а также ограничений на доступные ресурсы.
- *Технологический:* алгоритмы, реализованные в рамках парадигм тестирования, сценарного и барьерного подходов.
- *Инструментальный:* инструментальные средства, аппаратно-программные испытательные комплексы.

Целью моделирования является выработка обоснованных решений по урегулированию проблемных ситуаций функциональной безопасности на основе сопоставления альтернатив обращения с дефектами в организации проектов, их промежуточных и конечных результатах.

Исходными данными для решения прямых задач являются:

а) Симптомы проблемных ситуаций как обоснование актуальности исследований.

б) Выделенные целевые группы заинтересованных лиц (заказчик, спонсор, руководитель проекта, постановщик задач, проектировщики, испытатели, пользователи).

в) Границы проблемной ситуации (объекту управления по критериям функциональной безопасности) и окружающая среда.

г) Множество архитектурных точек зрения заинтересованных лиц на источники дефектов и их роль в формировании потребительских свойств АПК.

Результаты моделирования создают информационную основу выработки консолидированного решения правообладателей по подходам к урегулированию проблемной ситуации (содержание понятия «правообладатель», понимается в смысле, определенном в [12]: «…индивидуум, команда, организация или их группы, имеющие интерес в системе»).

Приведем в качестве примеров несколько архитектурных моделей, соответствующих различным архитектурным точкам зрения и различным уровням детализации компонент концептуальной модели описания архитектуры, представленной в [12 - 13]. Основу приводимых моделей составляют вышеизложенные принципы обеспечения функциональной безопасности

На рисунке 2 представлена модель «Треугольник функциональной безопасности». Назначение этой модели – выделение основных характеристик функциональной безопасности.

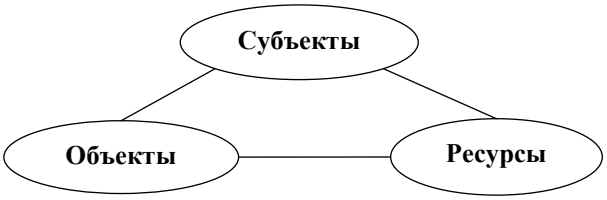

Рис.2 «Треугольник» функциональной безопасности.

Субъектами функциональной безопасности являются люди, совершающие по различным причинам разные ошибки на всех стадиях жизненного цикла компонентов вычислительно-коммуникационных систем.

Объектами функциональной безопасности являются артефакты, в которых присутствуют дефекты, обусловленные совершаемыми ошибками.

Ресурсами функциональной безопасности являются доступные материальные и нематериальные активы, которые могут быть использованы для предотвращения возникновения дефектов, их выявления и устранения, сокращение масштабов негативных последствий в результате проявления латентных дефектов.

Модель, представленная на рисунке 3 соответствует методическому аспекту функциональной безопасности.

Назначение данной модели - выделение базовых компонент непрерывного процесса совершенствования обеспечения функциональной безопасности.

Содержание этой модели состоит в том, чтобы по мере возрастания роли цифровой экосреды в задачах центрического управления, изменения масштабов и сложности инфраструктуры цифровой экосреды, постоянно вырастают требования к функциональной безопасности компонентов информационных систем, а также инфраструктуры в целом.

Рис.3 Цикл обеспечения функциональной безопасности.

Это требует постоянного развития теоретических

основ, технологических и инструментальных средств обеспечения функциональной безопасности.

V. Заключение

Концептуальную основу обеспечения безопасности составляет следующая система взглядов, создающих основу для адаптации известных системных принципов, подходов, системных паттернов и шаблонов в область обеспечения функциональной безопасности:

- Обеспечение безопасности АПК в составе систем поддержки сетецентрического управления распределенными сложными системами относится к классу задач управления в условиях нечетких целей управления и неопределенности, в ограничениях на управленческие решения.
- АПК являются разновидностью сложных субъектоцентрических систем, что является основанием для научно обоснованной адаптации известных подходов к модельному описанию проблемных ситуаций, возникающих при управлении сложными системами разной природы в область функциональной безопасности компонентов вычислительно-коммуникационных систем. Модельное описание проблемных ситуаций служит основанием для информационной поддержки принятия рациональных решений по их урегулированию.
- Основу обеспечения функциональной безопасности составляет комплексное использование сведений, получаемых из разных источников: структурный анализ архитектур систем обеспечения функциональной безопасности и внутреннего устройства АПК; результаты обработки исторических данных (включая метрические характеристики), связанные с проявлением дефектов разной природы; экспертные оценки субъектов, причастных к созданию и использованию АПК.
- Функциональная безопасность определяется количеством дефектов (проявившихся и латентных), возникших на разных стадиях жизненного цикла, начиная с осознания наличия проблемной ситуации, для урегулирования которой необходимо использование аппаратно-программных комплексов. Количество латентных дефектов определяется эффективностью урегулирования проблемных ситуаций, возникающих при управлении потребительскими свойствами АПК на разных стадиях их жизненного цикла. Моменты проявления латентных дефектов и моменты их возникновения распределены в пространстве и во времени.

Библиография

[1] Schuh, G. Industrie 4.0 Maturity Index Managing the Digital Transformation of Companies / G. Schuh, R. Anderl, J. Gausemeier, M. Hompel, W. Wahlster // Acatech STUDY, 2018. ‑ 60 p.

[2] J. Reason, E. Hollnagel, J. Paries, “Revisiting the “Swiss Cheese” Model of Accidents”, EEC Note No. 13/06. European Organization for the Safety of Air Navigation, October 2006, 25 p.

[3] Brooks, Frederick P., “No Silver Bullet: Essence and Accidents of Software Engineering”. Computer, Vol. 20, No. 4 (April 1987) pp. 10-19. (DOI: 10.1109/MC.1987.1663532)

[4] Huang F, Liu B. Software defect prevention based on human error theories. Chinese Journal of Aeronautics, 2017; 30 (3): 1054-1070. (DOI:10.1016/J.CJA.2017.03.005)

[5] Shappell SA, Weigmann DA. The Human Factors Analysis and Classification System – HFACS. Final Report, U.S. Department of Transportation, Federal Aviation Administration; 2000.

[6] Carver, J.C. Defect prevention in requirements using human error information: An empirical study. / J.C. Carver, W. Hu, V. Anu, G. Walia, G. Bradshaw // Requirements Engineering: Foundation for Software Quality - 23rd International Working Conference, REFSQ 2017. – P. 61-76. – https://doi.org/10.1007/978-3-319-54045-0_5.

[7] Hu, Wenhua & Carver, Jeffrey & Anu, Vaibhav & Walia, Gursimran & Bradshaw, Gary. (2016). Detection of Requirement Errors and Faults via a Human Error Taxonomy: A Feasibility Study. (DOI:10.1145/2961111.2962596)

[8] Майерс Г.Дж. Надежность программного обеспечения. М: Издательство Мир, 1980. – 359 с.

[9] Липаев В.В. “Надежность программных средств”. М: Синтег, 1998, 232с.

[10] Липаев В.В. “Надежность и функциональная безопасность комплексов программ реального времени”. М: Институт системного программирования Российской академии наук. 2013, 176с.

[11] Макконнелл, С. Сколько стоит программный проект / С. Макконнелл. – СПб.: Питер, 2007. – 296 с.

[12] ГОСТ Р 57100-2016. Системная и программная инженерия. Описание архитектуры. 36с.

[13] IEEE 1471-2000 - IEEE Recommended Practice for Architectural Description for Software-Intensive Systems

[14] Гвоздев В.Е. Элементы системной инженерии: методологические основы разработки программных систем на основе V-модели жизненного цикла: монография / М.Б. Гузаиров, Б.Г. Ильясов, О.Я. Бежаева. – М.: Машиностроение, 2013. – 180 с.

[15] ESA PSS-05-11. Guide to software quality assurance, 1995. – 55 p.

[16] ГОСТ Р ИСО/МЭК 12207-2010. Национальный стандарт Российской Федерации. Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных систем. - 105с.

Principles and Approaches to Ensuring the Functional Safety of Computing and Communication Systems Components.

O.Ya. Bezhaeva

***Abstract* - In this paper, the principles and approaches to ensuring the functional safety of computing and communication systems components are formulated. The properties of defects as varieties of complex controlled systems are considered. The conceptual basis of the research is a systematic combination of proactive, active and reactive approaches to the management of defects of different nature. The consideration of defects as a variety of complex systems creates a methodological basis for the scientifically based adaptation of approaches, methods and models that have proven themselves in solving problems of complex systems management of a different nature in the field of functional safety management. As an example, the paper shows architectural models that characterize various aspects of functional safety. Architectural models form the basis for solving direct tasks of ensuring functional safety: the study of problem situations, caused by the presence of various defects in the organization of projects and in software products. Within the system approach, the conceptual foundations of ensuring the functional safety of computing and communication systems components are formulated.**

***Keywords* — complex systems, functional safety, subject-centric systems, defects, computing and communication systems, hardware and software complexes.**

REFERENCES

[1] Schuh, G. Industrie 4.0 Maturity Index Managing the Digital Transformation of Companies / G. Schuh, R. Anderl, J. Gausemeier, M. Hompel, W. Wahlster // Acatech STUDY, 2018. ‑ 60 p.

[2] J. Reason, E. Hollnagel, J. Paries, “Revisiting the “Swiss Cheese” Model of Accidents”, EEC Note No. 13/06. European Organization for the Safety of Air Navigation, October 2006, 25 p.

[3] Brooks, Frederick P., “No Silver Bullet: Essence and Accidents of Software Engineering”. Computer, Vol. 20, No. 4 (April 1987) pp. 10-19. (DOI: 10.1109/MC.1987.1663532)

[4] Huang F, Liu B. Software defect prevention based on human error theories. Chinese Journal of Aeronautics, 2017; 30 (3): 1054-1070. (DOI:10.1016/J.CJA.2017.03.005)

[5] Shappell SA, Weigmann DA. The Human Factors Analysis and Classification System – HFACS. Final Report, U.S. Department of Transportation, Federal Aviation Administration; 2000.

[6] Carver, J.C. Defect prevention in requirements using human error information: An empirical study. / J.C. Carver, W. Hu, V. Anu, G. Walia, G. Bradshaw // Requirements Engineering: Foundation for Software Quality - 23rd International Working Conference, REFSQ 2017. – P. 61-76. – https://doi.org/10.1007/978-3-319-54045-0_5.

[7] Hu, Wenhua & Carver, Jeffrey & Anu, Vaibhav & Walia, Gursimran & Bradshaw, Gary. (2016). Detection of Requirement Errors and Faults via a Human Error Taxonomy: A Feasibility Study. (DOI:10.1145/2961111.2962596)

[8] Myers G. J. Software reliability. Moscow: Mir Publishing House, 1980. - 359 p. (in Russian)

[9] Lipaev V. V. "Reliability of software tools". M: Sinteg, 1998, 232p. (in Russian)

[10] Lipaev V. V. "Reliability and functional safety of real-time program complexes". M: Institute of System Programming of the Russian Academy of Sciences. 2013, 176p. (in Russian)

[11] McConnell, S. How much does a software project cost / S. McConnell. - St. Petersburg: Peter, 2007. - 296 p. (in Russian)

[12] GOST R 57100-2016. Systems and software engineering. Architecture description. 36p.

[13] IEEE 1471-2000 - IEEE Recommended Practice for Architectural Description for Software-Intensive Systems.

[14] Gvozdev V.E. Elements of system engineering: methodological foundations for the development of software systems based on the V-model of the life cycle: monograph / M.B. Guzairov, B.G. Ilyasov, O.Ya. Bezhaeva. - M.: Mechanical Engineering, 2013. - 180 p. (in Russian)

[15] ESA PSS-05-11. Guide to software quality assurance, 1995. – 55 p.

[16] GOST R ISO/IEC 12207-2010. National Standard of the Russian Federation Information technology. System and software engineering. Software life cycle processes. 105p.