

розвитку як клієнтів, так і підприємства. Особливої уваги описаний метод набуває внаслідок динамічної складової процесу закріплення та можливості побудови декількох умовно оптимальних планів.

Додаткових досліджень потребують задачі пошуку залежностей, які описують зміну всіх показників від часу. Аналізу вимагає також і залежність ефективності методики від кількості інтервалів часу при розбитті. Необхідно провести дослідження можливості використання методики з декількома істотно різними клієнтами та різними автомобілями.

Література

1. Бекетов О.Ю. Розрахунок раціональної структури парку рухомого складу [Текст] / Бекетов О.Ю., Наумов В.С. // Вісник ХНАДУ - 2003. - №22 - С.65 - 68.

2. Нагорний Є.В. Методика вибору споживачем центру транспортного сервісу при нечіткому представленні переваг на множині альтернатив [Текст] / Нагорний Є.В., Андросенко В.В. // Автомобільний транспорт - 2004. - №14 - С.72 - 74.
3. Панов С.А. Совершенствование перевозок на автомобильном транспорте [Текст] / Панов С.А. - М.: Наука, 1973. - 153 с.
4. Кузин Л.Т. Основы кибернетики. Том 1. Математические основы кибернетики. Учеб. пособие для студентов вузов. [Текст] / Кузин Л.Т. - М.: «Энергия», 1973 - С. 316 - 332.
5. Глушков В.М. Энциклопедия кибернетики. Том 1. [Текст] / Глушков В.М. - К.: «Главная редакция Украинской советской энциклопедии», 1974. - С. 152 - 153.

Рассмотрены проблемы использования адаптивных подходов разработки программных систем (ПС), предложена их классификация. Сформулирована задача структурной адаптации ПС, приведены существующие методы её решения, предложен кибернетический подход на платформе пост объектно-ориентированных технологий

Ключевые слова: программная система, адаптация, кибернетический подход

Розглянуто проблеми застосування адаптивних підходів розробки програмних систем (ПС), запропоновано їх класифікацію. Сформульовано задачу структурної адаптації ПС, наведено існуючі методи її вирішення, запропоновано кибернетичний підхід на платформі пост об'єктно-орієнтованих технологій

Ключові слова: програмна система, адаптація, кибернетичний підхід

Adaptive approaches for software design were reviewed and classified. Problem of structure adaptation was formulated, existing methods of it's solution were cited, cybernetic approach based on post object-oriented technologies was proposed

Key words: software system, adaptation, cybernetic approach

1. Актуальность проблем использования адаптивных технологий разработки и сопровождения ПС

Анализ источников по методологическим проблемам программной инженерии показывает, что в

последние несколько лет в этой научно-прикладной области наметилась тенденция разработки программных систем (ПС) с повышенной функцией живучести (см., напр., в [1-2]), т.е. таких ПС, которые обладали бы способностью к эволюции в процессе их эксплуа-

УДК 681.518:658.512

СТРУКТУРНАЯ АДАПТАЦИЯ ПРОГРАММНЫХ СИСТЕМ: АНАЛИЗ СОСТОЯНИЯ ПРОБЛЕМЫ И НЕКОТОРЫЕ ПОДХОДЫ К ЕЕ РЕШЕНИЮ

Н. В. Ткачук

Доктор технических наук, профессор*

Контактный тел.: (057) 707-60-86

E-mail: tka@kpi.kharkov.ua

К. А. Нагорный

Ассистент*

*Кафедра автоматизированных систем управления

НТУ «Харьковский политехнический институт»,

ул. Фрунзе, 21, г. Харьков, 61002

Контактный тел.: (057) 707-60-86

E-mail: k.nagorny@gmail.com

тации и заданными свойствами устойчивости и «выживаемости» в условиях непрерывных изменений в среде их функционирования. Эта же проблема, причем еще несколько раньше, начала активно обсуждаться в зарубежных публикациях, и, прежде всего, термин «жизнеспособное программное обеспечение» (*viable software*) появился в работах австралийского ученого Ч. Херринга (С. Herring) [3].

Эти и многие другие публикации положили начало этапу активного внедрения в концептуальный инструментарий современной программной инженерии междисциплинарных подходов, и, в частности, методов теории управления и кибернетических моделей для анализа и синтеза сложных ПС. Прямым отражением этой тенденции стало появление специального термина «программная кибернетика» (*software cybernetics*) (см. напр., в [4]). Под ним понимается подход к разработке ПС, в котором применяются такие базовые концепции вышеназванных дисциплин, как использование различных схем систем управления с обратной связью, разработка и применение для управления компонентами ПС различных моделей знаний о предметной области, использование количественных характеристик и соответствующих метрик для оценки качества их функционирования.

Осознание необходимости и целесообразности применения кибернетических подходов при создании сложного ПО, одним из которых является принцип адаптивности управления сложным объектом, естественным образом приводит к появлению адаптивных технологий в процессах разработки новых и управлению уже существующими ПС.

представляется весьма сложной и неоднозначной. Вместе с тем, если рассмотреть ее с позиций соотношения задач адаптации разрабатываемой ПС с основными фазами ее жизненного цикла (например, в соответствии со стандартом ISO / IEC90003, который опубликован на официальном ресурсе международного консорциума по программной инженерии SWEBOOK [5]), то можно достаточно мотивированно утверждать, что существуют следующие типы задач адаптации ПО, а именно:

- 1) адаптивность *процессов разработки* ПС;
- 2) адаптивность *архитектуры* ПС;
- 3) адаптивность *механизмов управления* функционированием уже существующей ПС.

Тогда можно предложить следующую обобщенную классификацию адаптивных технологий программной инженерии, которая показана на рис. 1. Для решения первого типа задач (1) применяются так называемые гибкие (*agile*) методологии организации процессов разработки (см., например, в [6]): унифицированный процесс разработки ПО (Rational Unified Process - RUP), метод экстремального программирования (*eXtreme Programming* – XP), адаптивный подход к разработке ПО (*adaptive software development* - ASD) и некоторые др.



Рис. 1. Обобщенная классификация адаптивных технологий

2. Классификация адаптивных технологий в современной программной инженерии

На основе анализа уже приведенных источников [1-4] можно утверждать, что все адаптивные технологии разработки ПС направлены на достижение главной цели: получить такие проектные решения, а затем - и программные компоненты, которые в максимальной степени были бы настраиваемыми на те постоянные изменения, которые происходят как в предметной области разработки данной системы, так и в операционной среде ее функционирования. При этом существует множество модельно-инструментальных средств, которые применяются в тех или иных концептуальных подходах и соответствующих технологиях реализации, и проблема их классификации

Общими для них является тенденция стирания жесткого разграничения между фазой проектирования и фазой конструирования в жизненном цикле ПС и переход от статической схемы: «Планирование – Проектирование – Конструирование» к итерационной динамической схеме управления программным проектом: «Обдумывание – Взаимодействие – Обучение».

Группа задач (2) решается путем применения таких методов разработки архитектуры ПС как: архитектурное проектирование, управляемое моделями (*Model-driven Architecture* - MDA), построение сервисно-ориентированных архитектур (*Service-oriented Architecture* - SOA) и т.п. Каждый из этих подходов обеспечивает независимость получаемых архитектурных решений от конкретных особенностей технологической платформы реализации ПС.

Третий тип задач (3) направлен на разработку таких моделей и соответствующих программных компонентов, которые позволяют адаптировать уже существующие ПС к изменяющимся условиям их функционирования (см., например в [7]). Эти задачи, в свою очередь, могут быть разделены на подтипы в соответствии с одним из трех возможных подходов к их решению, которые хорошо известны из общей теории адаптивных систем [8], а именно: а) *параметрическая*; б) *алгоритмическая* и в) *структурная* адаптация.

В дальнейшем в данной статье более подробно рассматриваются вопросы именно структурной адаптации ПС.

3. Задача структурной адаптации ПС и некоторые традиционные подходы к ее решению

Задача структурной адаптации ПС в неформализованном виде может быть сформулирована следующим образом: как при изменении воздействий определенных факторов среды применения некоторой ПС необходимо проводить ее реструктуризацию с целью обеспечить требуемые значения определенного критерия качества функционирования этой ПС? При этом факторами воздействия со стороны внешней среды, как правило, являются следующие [7]:

- а) колебания уровня вычислительной нагрузки на ПС;
- б) изменения в объеме доступной системе памяти (оперативной или вторичной);
- в) появление новой функциональности в тех прикладных задачах, которые решаются с помощью данной ПС.

При этом в качестве критериев качества функционирования ПС обычно рассматриваются показатели как:

- д) производительность (performance);
- е) надежность (reliability).

Это характерно, в частности, для ПС управления различными технологическими процессами, включая системы реального времени и некоторые другие.

Следует также отметить, что технологической платформой для проведения структурной адаптации ПС до последнего времени все еще остается объектно-ориентированное программирование (ООП). Это позволяет эффективно решать проблему усовершенствования унаследованных ПС, которые в свое время были разработаны с использованием технологий структурно-функционального программирования [9], но этот инструментарий уже не достаточен для того, чтобы вести реструктуризацию ПС, созданных на платформе ООП, а тем более – новых перспективных систем. С другой стороны, в последние несколько лет в программной инженерии появился ряд новых подходов к разработке ПС, которые можно объединить под одним названием *пост объектно-ориентированных технологий* (ПООТ). К ним относятся, например, такие технологии как [10-11]: *аспектно-ориентированная разработка* (*aspect-oriented programming* - АОР), программирование, ориентированное на достижение определенных заданных свойств системы (*feature-oriented programming* – FOP), контекстно-ориентированное программирование (*context-oriented programming* –

СОР). Они позволяют рассматривать компоненты ПС с точки зрения предоставляемых ими сервисов и более гибко модифицировать её, распределяя их по новым логическим уровням. Эти уровни затем могут расширять функциональность базовых компонентов ПС без необходимости изменения их самих как на этапе компиляции, так и на этапе выполнения программы.

Таким образом, проведенный анализ некоторых традиционных подходов к структурной адаптации ПС показывает их ограниченность и мотивирует необходимость разработки нового модельно-технологического инструментария для решения этой задачи.

4. Постановка инвариантной задачи обеспечения структурной адаптации ПС и кибернетический подход к ее решению на основе пост объектно-ориентированных технологий

Для того, чтобы сформулировать задачу структурной адаптации ПС в виде, максимально инвариантном к специфике той или иной предметной области и обобщить группы факторов влияния (а)-(с) и критериев (д)-(е), представляется целесообразным предложить следующую ее качественную постановку:

(А) наиболее унифицированным типом возмущений, которые влияют на любую ПС в среде ее функционирования, можно считать те *системные требования* (system requirements) - СТ, которые представляют собой требуемые характеристики (свойства) целевой ПС [9];

(В) общую схему реализации процедур построения адаптивной структуры уже существующей ПС, которая реализована на платформе ООП, следует рассматривать как *схему системы управления с обратной связью* [8], т.е. применить *кибернетический подход* к структурной адаптации ПС.

(С) в качестве непосредственного *управляющего воздействия* для структурной адаптации ПС могут быть использованы такие ПООТ, как АОД, FOD и СОД.

(D) критерием качества управления процессом структурной адаптации ПС рационально выбрать такой универсальный показатель, как *сопровождаемость программного обеспечения* (software maintainability) [5,6,9], поскольку он одинаково важен как для информационно-управляющих систем реального времени, так и информационных систем организационного управления, систем аналитической обработки данных и т.д.

Следуя общим положениям теории адаптивных систем [8], в более формализованном виде предложенную качественную постановку задачи (А)-(D) можно представить в следующем виде (см. рис. 2). На этом рисунке: X - множество СТ, Y - множество адаптивных структур ПС, M - множество метрик качества ПО, U - множество доступных ПООТ. Таким образом, объект управления можно представить как преобразователь F :

$$Y = F(X, U) , \quad (1)$$

где F - оператор связи входа X , учитывающий наличие фактора управляющего воздействия U и выхода Y .

В свою очередь, управляющее воздействие U формируется на основании сигналов X', Y' , приходящих от некоторых датчиков, измеряющих состояние входа и выхода соответственно:



Рис. 2 Кибернетический подход к структурной адаптации ПС

$$U = M(X', Y'), \quad (2)$$

$$X' = D_x(X), \quad (3)$$

$$Y' = D_y(Y) \quad (4)$$

В дополнение к уже сделанной интерпретации объектов X , Y , M и U в системе выражений (1)-(4) следует добавить, что датчики D_x и D_y , а также F - это соответствующие CASE-средства, которые обеспечивают автоматизацию процедур количественной оценки входных и выходных данных на схеме рис. 2.

Для полноты концептуальной постановки задачи структурной адаптации ПС (1) – (4), а в дальнейшем – и для ее корректного практического решения, необходимо учитывать следующее обстоятельство: выбор той или иной ПООТ в качестве эффективного управляющего воздействия в кибернетической схеме адаптации структуры ПС на рис. 2 будет зависеть от двух следующих факторов:

1) какова начальная *структурная сложность* рассматриваемой ПС,

2) *насколько часто* в процессе сопровождения этой ПС изменяются системные требования, предъявляемые к ней.

Другими словами, каждую из рассматриваемых ПС можно позиционировать в 2-х мерной системе координат «Структурная сложность» - «Частота изменения системных требований» путем выполнения экспертных процедур оценки этих параметров (см. аналогичный подход в работе [7]).

5. Выводы и направления дальнейших исследований

В данной научной статье: 1) показана актуальность проблемы структурной адаптации ПС; 2) введена классификация адаптивных технологий и проанализированы некоторые уже существующие подходы к

решению задачи структурной адаптации ПС; 3) предложен новый перспективный подход к повышению эффективности процедур структурной адаптации ПС, который использует кибернетический принцип

построения соответствующей модели и применения пост-объектно-ориентированных технологий программирования для ее реализации.

Полученные результаты могут быть использованы для разработки соответствующего CASE-средства поддержки процессов структурной адаптации существующих ПС в условиях изменения системных требований в

их среде функционирования.

Литература

1. Ігнатенко П.П., Стрелов І.А. та інш. Концепція створення моделі прикладної програмної системи з розвинутою функцією життєздатності // Проблеми програмування. - 2004. № 2-3. С. 163-172.
2. Ігнатенко П.П. Проблеми забезпечення життєздатності програмних систем та підходи до їх вирішення // Проблеми програмування. - 2002. - № 3-4. - С. 58-73.
3. Herring, C. and S. Kaplan. The Viable System Model for Software. in 4th World Multi-conference on Systems, Cybernetics and Informatics (SCI'2000). 2000. Orlando, Florida.
4. Miller, S.D., DeCario, R.A., Mathur A.P. A Software Cybernetic Approach to Control of Software Systems // Proceedings of the 29th Annual International Computer Software and Applications Conference, 2005.
5. Официальный интернет-ресурс международного консорциума по программной инженерии SWEBOK: URL: www.swebok.org
6. Фаулер М. Архитектура корпоративных программ и приложений. - М.: Изд. Дом «Вильямс», 2004.
7. Ткачук М.В., Земляний А.О. Про один підхід до оцінки ефективності технологій забезпечення якості програмних систем // Східно-Европейський журнал передових технологій, № 1 (37). - 2009. – с. с.33-38.
8. Расстригин Л.А. Адаптация сложных систем. – Рига, «Зинатне», 1981.
9. Соммервилл И. Инженерия программного обеспечения : Пер. с англ. – М.: Издательский дом “Вильямс”, 2002.
10. Apel S. The Role of Features and Aspects in Software Development - Dissertation, Otto-von-Guericke-Universit t, Magdeburg, Germany, 2006.
11. Hirschfeld R., Costanza P., Nierstrasz O., Context-oriented Programming // Journal of Object Technology, 2008 - vol. 7, no. 3, pages 125-151.