

§ 2 ПОКАЗАТЕЛИ КАЧЕСТВА И ПОВЫШЕНИЕ НАДЕЖНОСТИ ПРОГРАММНЫХ СИСТЕМ

Поначугин А.В.

ПРОБЛЕМЫ НАДЕЖНОСТИ ФУНКЦИОНИРОВАНИЯ И СОПРОВОЖДЕНИЯ СОВРЕМЕННЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

Аннотация: В середине прошлого века проблема надежности приобрела статус научности. Этому способствовало активное развитие вычислительных систем. Внедрение и сопровождение информационной системы - это долгий, трудный и кропотливый процесс, который влияет на то, как в будущем сложатся отношения пользователя с системой. В работе рассмотрены различные подходы к распределению и использованию времени на сопровождение информационной системы. Проанализирована значимость процессов сопровождения информационной системы на всех стадиях её жизненного цикла. Рассмотрены подходы к обеспечению надёжности функционирования и сопровождения различных классов информационных систем. На разных уровнях управления по решению проблем обеспечения функционирования информационной системы определён перечень рекомендаций. В работе проведен анализ существующих программных и аппаратных средств и инструментальных методов, связанных с сопровождением информационных систем и их надёжным функционированием. В ходе проведенного анализа автором сделан вывод: функция сопровождения вычислительных систем одинаково важна на всех этапах жизненного цикла и занимает значительную долю времени в работе информационно-технического отдела и неразрывно связана с показателями надежности функционирования информационных систем. Также, в работе предложена методика использования программных инструментальных средств для контроля характеристик функционирования информационной системы.

Ключевые слова: устаревание информационной системы, безотказность, вычислительные системы, сопровождение информационной системы, программы-агенты, надёжность функционирования, жизненный цикл, долговечность, проектирование информационных систем, автоматизированные системы управления

Abstract: In the middle of the last century the problem of functioning and maintenance reliability

has acquired the status of scientific. This was facilitated by the active development of computer systems. Implementation and support of information systems are long, difficult and complicated processes, which affect how the future interactions between users and the system. The article discusses various approaches to the allocation and use of time required for supporting information system. Author analyzes the importance of supporting information system at all stages of its life cycle. Author reviews different approaches to ensuring the reliability of the operation and maintenance of the various classes of information systems. The article gives a defined list of recommendations for ensuring functioning of the information system at different levels of management. The study shows an analysis of existing software and hardware tools and methods related to maintenance and reliable functioning of information systems. The author concludes: the maintenance of computer systems is equally important at all stages of the life cycle and occupies a considerable share of time in the work of the information technology department and is inextricably linked with the indicators of reliability of information systems. Also, the author suggests the technique of the use of software tools for the control of characteristics of information system.

Keywords: *durability, life cycle, reliability, soft agents, support of information system, computer systems, infallibility, obsolescence of the information system, design of information systems, automated control systems*

В середине 20-го века проблема надежности приобрела статус научности. Этому способствовало активное развитие вычислительных систем. Интенсивное изучение которых сделало проблему надежности основной технической проблемой государственного значения. Академик А.И. Берг [1] внес большой вклад в развитие исследований проблемы надежности в нашей стране, при этом достаточно четко определив её сущность. Не имеет смысла пытаться решить ее путем только эпизодических мероприятий. Проблема надежности непрерывно связана с развитием технической мысли и как следствие с созданием новой техники, разработкой новых технологий. Решенная проблема на текущем этапе неизбежно будет возникать в будущем и никогда не перестанет существовать.

Грамотное и профессиональное сопровождение информационной системы обеспечивает ее стабильное функционирование и позволяет говорить о надёжной работе.

Надежность функционирования вычислительной системы – это комплексное свойство, которое в зависимости от назначения объекта включает в себя безотказность, ремонтпригодность и долговечность. Понятие «надежность» применяется для описаний общего характера в не количественном выражении.

Середина 20-го века характеризовалась существенным рывком в развитии и аппаратных составляющих и программной части вычислительных систем - особое распространение получили большие и малые автоматизированные системы управления (АСУ) разного назначения. Разработка и использование таких систем без применения мер по обеспечению их надёжности бессмысленны. Опасность заключается как в не работоспособности таких систем, так и в том, что отказ в работе или неправильная работа, может

повлечь за собой катастрофические последствия.

Должное внимание можно уделить понятию «программная надёжность», роль которого в обеспечении надёжности АСУ занимает одно из самых характерных особенностей прикладной теории надёжности АСУ.

Оптимальным путем для достижения надежности программного обеспечения является предупреждение ошибок (рис. 1). Наилучшим способом обеспечения надежности в первую очередь считается не допустимость возникновения ошибок [6,7,8,9].

Методы обеспечения функциональной надежности программ							
Предупреждение ошибок		Обнаружение ошибок		Исправление ошибок		Устойчивость к ошибкам	
Защитное программирование	Много-версионное программирование	Пассивное	Активное	Динамическая избыточность	Рестарт	Изоляция ошибок	Алгоритмическая избыточность

Рис. 1 - Основные аспекты построения архитектуры надежного программного обеспечения

При обеспечении функционирования информационной системы у руководителя информационного отдела постоянно возникает много проблем, не имеющих под собой реальной основы. Например, бытует мнение, что сопровождение - не творческая, скучная работа, которой заняты самые неквалифицированные сотрудники. Одновременно с этим, неквалифицированное сопровождение негативно сказывается даже на хорошо проработанном программном обеспечении. Мнение о постоянном удовлетворении руководителя качеством сопровождения программного обеспечения - ошибочно (сопровождение — это один из этапов всего проекта, и качество программного проекта напрямую зависит от качества проведенного процесса первоначальной разработки).

Имеется ряд принципиальных заблуждений, которые касаются процесса сопровождения. Например, по мнению руководителя данный процесс занимает около 20% рабочего времени, а 80% этого времени сопровождающий специалист тратит на исправление ошибок в программе [2]. На самом деле это не соответствует действительности.

В некоторых литературных источниках, убедительно показано, что процесс сопровождение отнимает от 50 до 80% материальных затрат и времени. Некоторые исследования доказывают то, что исправление ошибок составляет небольшую часть работы сопровождающего специалиста (16-42%). К основным этапам разработки программного обеспечения относятся следующие:

- процесс разработки программного обеспечения,
- управление проектом разработки,
- описание целевого программного продукта,

- проектирование продукта,
- разработка продукта,
- тестирование частей продукта,
- интеграция частей и тестирование продукта в целом,
- сопровождение продукта.

Жизненный цикл характеризует сущность прохождения информационной системы во времени. Информационная система, как и любой созданный продукт, включает в жизненный цикл стадии от начала создания до момента окончания использования [5].

Жизненный цикл вычислительной системы заканчивается, обычно, не по причине физического износа, а в результате её морального устаревания. Моральный износ и устаревание - это прекращение соответствия информационной системы, предъявляемым к ней требованиям.

В данной ситуации дальнейшая модификация информационной системы экономически нецелесообразна или невозможна, что приводит к потребности разработки новой информационной системы.

Надежность информационной системы не зависимо от ее сложности должна быть обеспечена на всех без исключения этапах жизненного цикла: от начальной стадии выполнения проектно-конструкторской разработки до заключительной стадии эксплуатации. Важным фактором обеспечения надежности является строгое исполнение правила, которое называют триада надежности: надежность закладывается при проектировании, обеспечивается при изготовлении и поддерживается в эксплуатации. Без строгого следования этому правилу невозможно выполнить задачу создания высоконадежных вычислительных систем с помощью компенсации недоработок предыдущего этапа на последующем.

Если в процессе проектирования полностью не решены все вопросы создания вычислительных системы с определенным уровнем надежности и не заложены алгоритмические и схематичные решения, обеспечивающие безотказное функционирование всех подсистем, то данные недостатки часто невозможно исправить в процессе производства, последствия которых приведут к снижению уровня надежности функционирования системы. В процессе разработки и внедрения вычислительной системы должны быть полностью реализованы все решения, технические задания и рекомендации ведущего программиста.

Новые информационные системы, внедряемые на предприятии, нуждаются в оценке рисков отставания от конкурентов вследствие ее необратимого устаревания со временем, по причине быстрой смены новыми версиями или видами. Период сменяемости варьируется от нескольких месяцев до одного года.

Если в ходе внедрения новой информационной системы этому не уделять внимания, не исключено, что к концу завершения перевода предприятия на новую информационную технологию она устареет и придется ее модернизировать. Подобные неудачные попытки внедрить информационную технологию обычно связаны с несовершенством технических средств, но при этом основной причиной неудач является отсутствие или

слабо изученная методология использования информационной системы (рис. 2).



Рис. 2 - Жизненный цикл системы и процесс реализации выгод

В 90-х годах прошлого века, в условиях либерализации экономики, начинают появляться и бурно развиваться торговые компании, которые остро нуждались в программном обеспечении, способном поддерживать текущие потребности бизнеса, в том числе обеспечивать работу не только локальных, но и удалённых информационных систем. Многие компании не нашли готового решения, удовлетворяющего их потребностям, на рынке информационных технологий, и поэтому разрабатывали свои программные комплексы самостоятельно. Они явились основой для будущих информационных систем. Бизнес-процессы приходилось «сочинять» непосредственно ИТ специалистам, с чем они хорошо справлялись. Написанные в 90-е годы программы работали и продолжают работать, а некоторые из них явились основой достаточно известных автоматизированных систем [4].

Собственное программное обеспечение имеет преимущество: низкая стоимость при более высоком соответствии потребностям конкретного бизнеса. К недостаткам относится проблема сопровождения. Разработчики при возникновении конфликтной ситуации могут просто уволиться, оставив, а в некоторых случаях сознательно испортив разработанную ими систему. Оставленная в таком виде вычислительная система постепенно «старела»: начинала медленнее работать, давать сбои, а часто совсем выходила из строя.

Обычно, системы собственной разработки слабо документированы. При существенной текучки кадров, «специалист-разработчик» является единственным кто досконально знает созданную им систему. Он единственный кто может предоставить консультацию, выполнить модификацию в программном коде, выявить ошибки, объяснить результаты отчета, в его помощи нуждаются все сотрудники отделов, которые используют систему. Постепенно такой специалист становится «незаменимым», он это осознаёт и начинает диктовать руководству свои правила игры. Он думает, что лучше других понимает назначение системы, знает бизнес-процессы и потребности конечных пользователей. В итоге из-за несоответствия реальным потребностям бизнеса информационная система начинает тормозить развитие компании.

Часто в маленьких розничных магазинах складывается обратная ситуация. Специалист

по информационным технологиям совсем не участвует в процессе сопровождения и развитием информационной системы, а ограничивается регламентными работами по настройке компьютеров и доступа в Интернет. Зачастую не исполняются даже обязательные для сопровождения вычислительной системы действия - создание резервных копий, поддержание целостности и актуальности данных. Информационная система не поддерживается ни собственными специалистами, ни компанией-разработчиком. В такой ситуации ходе работы пользователями нарушается технология, и достоверность информации искажается. В данном случае, например, в базе данных розничной сети могут появляться отрицательные остатки по многим товарным позициям.

Чтобы информационный отдел эффективно работал, необходимо четко определить круг обязанностей, которые на него возлагаются, права и порядок взаимодействия с другими подразделениями. Главной задачей отдела является обеспечение бесперебойной работы информационной системы, а именно: локальной вычислительной сети, коммутационного оборудования, рабочих станций, сервера, программного обеспечения. К функциям информационного отдела также относится: разработка и внедрение собственного программного обеспечения или участие в выборе готового программного комплекса, обучение и консультация персонала, эксплуатация и сопровождение вычислительной системы. Перечень обязанностей отдела зависит от различных факторов и формируется самой организацией. Например, часто информационные отделы обслуживают мини-АТС, хотя, это оборудование не является частью информационной системы. Это должно отражаться в положении об отделе, а в состав отдела входить специалист по поддержке АТС.

Следующий этап включает определение численности сотрудников и закрепление их функциональных обязанностей. Обязанности и права сотрудника прописываются в должностной инструкции. Не менее важная задача - определение графика работы отдела. Долгие задерживания специалистов на работе, не связанные с проведением специальных работ - признак перегруженности специалистов или их низкой квалификации. Руководителю также следует учитывать, что у специалистов информационного отдела существует своя специализация: прикладные программы, сети, операционные системы, технические средства и пр.

В некоторых случаях возможно привлечь сторонние специализированные организации. При этом контроль за выполнением и приемку работ, выполненных другой организацией, обязательно должен осуществлять собственный специалист. Анализ отечественных компаний показал, что численность ИТ отделов может зависеть от таких факторов как: общее число работников, количество используемой вычислительной техники, сложность используемых информационных систем, проводимые исследования и разработки. Количество специалистов варьируется в зависимости от функциональных обязанностей информационного отдела и от их квалификации и универсальности.

Крупные компании могут иметь в штате два информационных отдела. Это может быть обусловлено наличием разных основных программ, например, «складской» и «бухгалтерской». При длительном сроке работы каждая из них начинает формировать собственную инфраструктуру - сервер, локально-вычислительная сеть, пользователи, и

в итоге создается отдельный информационный отдел. В таком случае говорят о наличии двух информационных систем в компании, за счет чего увеличиваются издержки, происходит дублирование работ, возникает необходимость в приобретении дополнительных технических средств, увеличивается количество специалистов.

В компании должен быть один информационный отдел. Если предприятие состоит из нескольких подразделений, принадлежащих разным юридическим лицам, целесообразно выделить информационный отдел, обслуживающий корпоративную информационную систему, в отдельную компанию со своим балансом. При таком раскладе расходы по созданию и сопровождению корпоративной информационной системы организации будут нести солидарно. Кроме использования административного и трудового ресурса, важно уделять внимание и ресурсам инструментальным. В настоящее время развиваются специальные программные средства, предназначенные для обработки сбоев в работе программного обеспечения и уведомления производителя такого обеспечения о возникших неполадках, с предоставлением некоторой информации о режиме работы программы и прочей информации, помогающей установить причины сбоя - программы-агенты.

Использование программ-агентов дает возможность [3]:

- повышать психологический комфорт пользователя (уменьшить стресс и недовольство от возникновения ошибок);
- повышать надежность и качество программного обеспечения за счет участия пользователей в тестировании;
- облегчать работу службы поддержки компании-разработчика и компании-пользователя;
- обнаруживать ошибки, которые не удастся найти при стандартном тестировании программного обеспечения, а которые выявляются в реальной работе системы.

Вычислительная система должна быть «открытой» для внесения изменений в нее конкретным предприятием. Плюсом является возможность формировать собственный отчет, описывать дополнительные характеристики товара или нового вида документа. При таком раскладе можно развивать систему собственными силами. Иногда пользователи системы, предлагая информационному отделу изменить программу, не могут четко сформулировать задачи. Предложения пользователей об изменении существующей системы рационально рассматривать на заседании координационного совета с участием представителей всех заинтересованных сторон. Такая необходимость возникает по причине противоречия идей, поступающих от разных отделов предприятия, или по причине уже реализованных в системе решений. Все изменения должны происходить согласно единой стратегии развития предприятия. Увеличение возможностей информационной системы влечет за собой увеличение затрат, поэтому прежде чем реализовывать изменения системы необходимо проанализировать на сколько это рационально и экономически обосновано. Утвержденные советом изменения составляют базис плана развития информационной системы. Чтобы выполнить этот план необходимо иметь в распоряжении определенные ресурсы. Ответственность за внедрение новых функций должна распределяться между разработчиками и инициаторами работ.

Исходя из этого следует, что информационный отдел - обслуживающий отдел, целью которого является поддержка надёжного функционирования и развитие комплексной вычислительной системы предприятия для снабжения бизнеса всей необходимой информацией. Информационная система не может стать самоцелью и развиваться в отрыве от бизнеса.

Библиография :

1. Аксель Иванович Берг. 1893-1979 / ред.-сост. Я.И. Фет; сост.: Е.В. Маркова, Ю.Н. Ерофеев, Ю.В. Грановский; отв. ред. А.С. Алексеев. М.: Наука, 2007. 518 с.
2. Брауде Э. Технология разработки программного обеспечения. СПб.: Питер, 2004. 655 с.: ил.
3. Гулиев Я.И., Бабаян М.А., Фохт О.А. Внедрение и сопровождение корпоративной информационной системы. Обеспечение обратной связи // Тр. междунар. конф. «Программ-ные системы: теория и приложения», ИПС РАН, Переславль-Залесский, 2004: В 2 т. / Под ред. С.М. Абрамова. М.: Физматлит. Т. 2. С. 271.
4. Девятьярова И. Трудно быть Богом. Проблемы сопровождения и развития инфор-мационной системы торгового предприятия [Электронный ресурс]-URL: <http://www.domino.ru/-to-interesno/trudno-byit--bogom.html> (дата обращения: 23.01.2016).
5. Сенник Ю.С. Жизненный цикл информационных систем / Ю.С. Сенник, И.П. Гребенников // Системный анализ и прикладная информатика. 2015. № 2. С. 4-9.
6. Boehm B.W. and Others (2000), Software Cost Estimation with COCOMO IF, Prentice Hall PTR, ISBN: 0130266922.
7. Fenton N.E. and Ohlsson N. (2000), Quantitative Analysis of Faults and Failures in a Complex Software System, IEEE Transactions on Software Engineering, to appear, Vol. 26, № 8, pp. 797-814.
8. Janaki Ram D. and Raju S.V.G.K. (2000), Object Oriented Software with Predictive Ob-jective Points, Proceeding of First Asia Pacific Conference on Quality Software, pp. 121-126.
9. Peter Tandler (2001), Software Infrastructure for Ubiquitous Computing Environments Supporting Synchronous Collaboration with Multiple Single-and Multi-User Device, Proceedings of UBICOMP Lecture Notes in Computer Science, Heidelberg: Springer.

References:

1. Aksel' Ivanovich Berg. 1893-1979 / red.-sost. Ya.I. Fet; sost.: E.V. Markova, Yu.N. Erofeev, Yu.V. Granovskii; otv. red. A.S. Alekseev. M.: Nauka, 2007. 518 s.
2. Braude E. Tekhnologiya razrabotki programmogo obespecheniya. SPb.: Piter, 2004. 655 s.: il.
3. Guliev Ya.I., Babayan M.A., Fokht O.A. Vnedrenie i soprovozhdenie korporativnoi informatsionnoi sistemy. Obespechenie obratnoi svyazi // Tr. mezhdunar. konf. «Programm-nye sistemy: teoriya i prilozheniya», IPS RAN, Pereslavl'-Zalesskii, 2004: V 2 t. / Pod red. S.M. Abramova. M.: Fizmatlit. T. 2. S. 271.
4. Devet'yarova I. Trudno byt' Bogom. Problemy soprovozhdeniya i razvitiya infor-matsionnoi sistemy torgovogo pred-priyatiya [Elektronnyi resurs]-URL: <http://www.domino.ru/-to-interesno/trudno-byit--bogom.html> (data obrash-cheniya: 23.01.2016).

5. Sennik Yu.S. Zhiznennyi tsikl informatsionnykh sistem / Yu.S. Sennik, I.R. Grebennikov // Sistemnyi analiz i prikladnaya informatika. 2015. № 2. S. 4-9.
6. Boehm B.W. and Others (2000), Software Cost Estimation with COCOMO IF, Prentice Hall PTR, ISBN: 0130266922.
7. Fenton N.E. and Ohlsson N. (2000), Quantitative Analysis of Faults and Failures in a Complex Software System, IEEE Transactions on Software Engineering, to appear, Vol. 26, № 8, pp. 797-814.
8. Janaki Ram D. and Raju S.V.G.K. (2000), Object Oriented Software with Predictive Objective Points, Proceeding of First Asia Pacific Conference on Quality Software, pp. 121-126.
9. Peter Tandler (2001), Software Infrastructure for Ubiquitous Computing Environments Supporting Synchronous Collaboration with Multiple Single-and Multi-User Device, Proceedings of UBICOMP Lecture Notes in Computer Science, Heidelberg: Springer.