

Интернет-журнал «Наукоедение» ISSN 2223-5167 <http://naukovedenie.ru/>

Том 8, №6 (2016) <http://naukovedenie.ru/vol8-6.php>

URL статьи: <http://naukovedenie.ru/PDF/12TVN616.pdf>

Статья опубликована 13.12.2016

**Ссылка для цитирования этой статьи:**

Варлатая С.К., Калужин Е.А., Монастырский Д.С. Методика обнаружения и устранения отказов программного обеспечения // Интернет-журнал «НАУКОВЕДЕНИЕ» Том 8, №6 (2016)

<http://naukovedenie.ru/PDF/12TVN616.pdf> (доступ свободный). Загл. с экрана. Яз. рус., англ.

**УДК 004.492.3**

**Варлатая Светлана Климентьевна**

ФГАОУ ВО «Дальневосточный федеральный университет», Россия, Владивосток<sup>1</sup>  
Школа естественных наук  
Руководитель направления «Информационная безопасность»  
Кандидат технических наук, профессор  
E-mail: Sk-varl@mail.ru

**Калужин Егор Александрович**

ФГАОУ ВО «Дальневосточный федеральный университет», Россия, Владивосток  
Школа естественных наук, направление «информатика и вычислительная техника»  
Магистрант  
E-mail: Workout24@mail.ru

**Монастырский Дмитрий Сергеевич**

ФГАОУ ВО «Дальневосточный федеральный университет», Россия, Владивосток  
Школа естественных наук, направление «информационная безопасность»  
Студент  
E-mail: Egorcool-90@mail.ru

## **Методика обнаружения и устранения отказов программного обеспечения**

**Аннотация.** В настоящее время программное обеспечение является неотъемлемым компонентом функционирования предприятия, с помощью которого производится автоматизация многих процессов, ведется электронный документооборот, обеспечивается связь территориально распределенных филиалов. При отказе ПО необходимо оперативно устранить причину и вернуть ПО в работоспособное состояние. Однако, из-за больших масштабов предприятий поддерживать работоспособность всего программного обеспечения в режиме реального невозможно. Поэтому, на сегодняшний день актуальным вопросом является создание механизма автоматического самовосстановления. В данной статье предлагается методика автоматического поиска и устранения отказов ПО. Основная идея заключается в том, чтобы внедрить механизм самовосстановления на стадии проектирования и реализации ПО. Данный механизм в режиме реального времени производит сканирование файлов программного обеспечения и обнаруживает основные причины отказов: удаление, перемещение и изменения компонентов. Также особенностью предложенного механизма является функция отправки отчета, что позволяет разработчика производить анализ инцидентов для дальнейшего усовершенствования ПО. Для оценки эффективности методика

---

<sup>1</sup> 690922, Россия, Приморский край, г. Владивосток, Аякс-10 корпус D

сравнивается с наиболее распространенным способом устранения отказов – переустановкой ПО.

**Ключевые слова:** отказоустойчивость; самовосстановление; программное обеспечение; сканирование файлов; причины отказов; переустановка; анализ состояния

### Введение

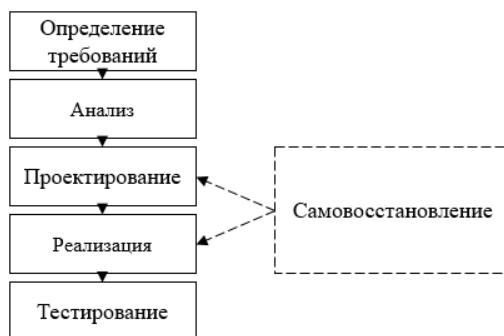
В настоящее время роль программного обеспечения (ПО) в управлении предприятием многократно возросла [1]. Для стабильного функционирования информационной системы применяется огромный спектр ПО: офисное ПО, средства редактирования информации, браузеры, средства автоматизации процессов и т.д. Различные вредоносные программы, несовместимость или воздействие некомпетентного пользователя могут вызвать отказ в работе ПО [2], тем самым нарушить функционирование информационной системы и повлечь материальный ущерб [3]. Отказ программного обеспечения можно определить как прекращение функционирования программы (искажения нормального хода ее выполнения, заикливание) на время, превышающее заданный порог [4]. При больших масштабах и сложной структуре предприятия, администрирование и восстановление ПО в режиме реального времени невозможно. Поэтому вопрос создания программного обеспечения, способного к самовосстановлению, является актуальным на сегодняшний день.

Подобная тема широко обсуждается в Российских и зарубежных научных работах. Авторы статьи [5] разработали механизм самовосстановления ПО на основе иммунитета человека. В другой работе [6] авторами предлагается формальная модель, описывающая работу программного средства как последовательность событий. Данная модель позволяет применить известные методы классификации для автоматического определения класса возникающих ошибок и устранения их последствий за счет выполнения определенных для данного класса действий. В статье [7] авторы предлагают методику обнаружения отказов, основанную на сравнении сценария отказа с уже известными отказами.

Все вышеуказанные методики, безусловно, являются эффективными, однако их реализации довольно сложна и требует разработки специализированного сервиса для поиска и устранения отказов. Целью данной статьи является разработка методики поиска и устранения отказов программного обеспечения, реализация которой проста и не требует создания дополнительных сервисов.

### Описание методики

Процесс создания любого программного обеспечения можно разбить на определенные стадии [8] (рисунок 1).



*Рисунок 1. Процесс создания программного обеспечения с функцией самовосстановления (разработано автором)*

На стадии определения требований разработчики определяют назначение, функционал и способы реализации ПО.

На стадии анализа оцениваются необходимые для создания ПО ресурсы, его название, примерный размер, актуальность и полезность.

На стадии проектирования разрабатывается блок-схемы для написания будущего ПО, определяются наиболее рациональные и целесообразные пути реализации.

На стадии реализации происходит написание программного кода, после чего начинается стадия тестирования и отладки ПО.

Основная идея предлагаемой методики заключается в том, чтобы на стадии проектирования и реализации заложить механизм самовосстановления. Такой подход не требует создания централизованного сервиса и позволяет обеспечить автономность каждого программного продукта т.е. при выходе из строя механизма самовосстановления на одном ПО, остальные будут функционировать стабильно. Механизм самовосстановления должен включать следующий функционал:

1. Анализ состояния: после запуска ПО модуль самовосстановления собирает всю информацию о его файлах: имя, размер, дата создания, дата изменения и хэш-сумма.
2. Диагностика: на основании всей собранной на предыдущем шаге информации данный функционал принимает решение о том, есть ли ошибки в программе. Если какой-либо файл изменен, перемещен или удален, ПО нуждается в восстановлении.
3. Подключение к серверу: данный функционал активируется при условии выявления отказа при диагностике. ПО в автоматическом режиме подключается к серверу, загружает и устанавливает поврежденные файлы.
4. Отправка отчета: данный функционал необходим для того, чтобы разработчик на основе анализа уведомлений мог установить закономерность между различными отказами и их причинами для дальнейшего усовершенствования ПО. Принцип работы предлагаемой системы самовосстановления можно представить в виде блок-схемы (рисунок 2).

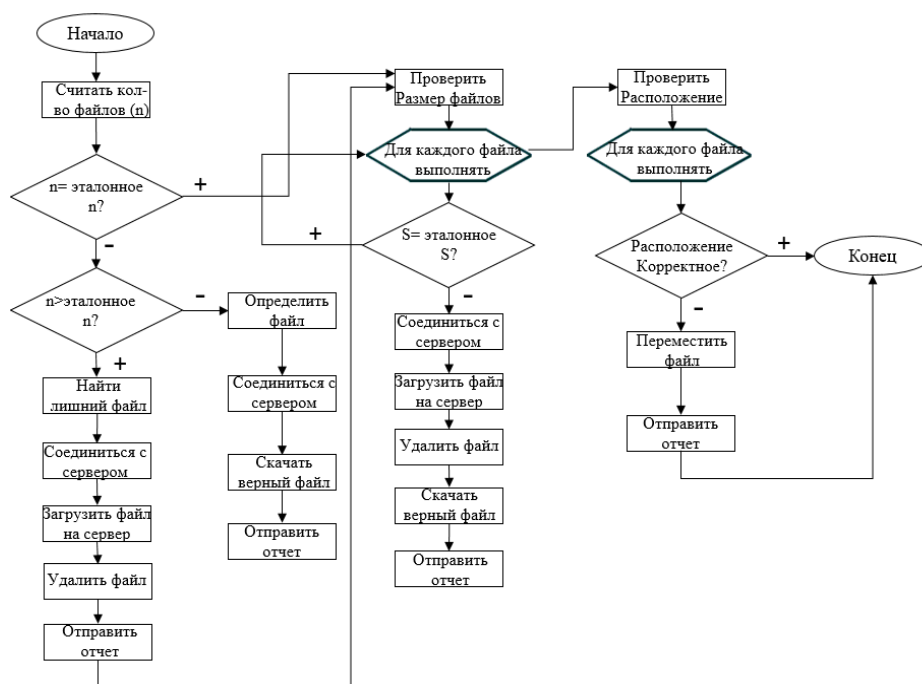


Рисунок 2. Блок-схема функционирования механизма самовосстановления  
(разработано автором)

Такая система восстановления нацелена на обнаружения основных причин отказа ПО: удаление (добавления) файлов, перемещение файлов, модификация файлов. Весь процесс состоит из трех модулей:

- Проверка целостности файлов. Модуль сканирует все файлы с целью поиска лишних или недостающих. Если найден лишний файл, он загружается на сервер для дальнейшего анализа и удаляется. Если модуль обнаружил, что какого-либо файла нет, происходит его загрузка и установка.
- Проверка размеров файлов. Данный модуль сравнивает размеры файлов с эталонными размерами, которые хранятся либо на сервере, либо на ПК. При обнаружении измененного файла происходит его замена и загрузка на сервер.
- Проверка расположения. Для работы многих программ важно расположения файлов в директориях. Поэтому модуль анализирует расположение файлов и при необходимости переносит их в корректную директорию.

### Оценка эффективности предложенной методики

Наиболее распространённым способом устранения отказов является переустановка ПО. При оценке эффективности было принято решение сравнить необходимое для восстановления время при использовании методики и при переустановке ПО.

Время устранения отказа с помощью предложенной методики можно определить как (1).

$$V=O+D+R \quad (1)$$

где:  $V$  – время восстановления ПО,  $O$  – время обнаружения причины отказа,  $D$  – время загрузки корректных файлов,  $R$  – время перемещения файлов.

Время переустановки ПО определяется как (2).

$$T=U+P+I \quad (2)$$

где:  $T$  – время переустановки ПО,  $U$  – время удаления ПО,  $P$  – время загрузки нового дистрибутива,  $I$  – время повторной установки.

Для расчета экспериментального  $V$  и  $T$  использовался ПК со следующими характеристиками: скорость скачки (download): 20 Мбит/с, скорость загрузки (upload): 18 Мбит/с, операционная система: Windows 7 x64, размер оперативной памяти: 4GB, процессор: Intel core™ i3-311M, 2.4 GHz. Результаты эксперимента представлены в таблице 1.

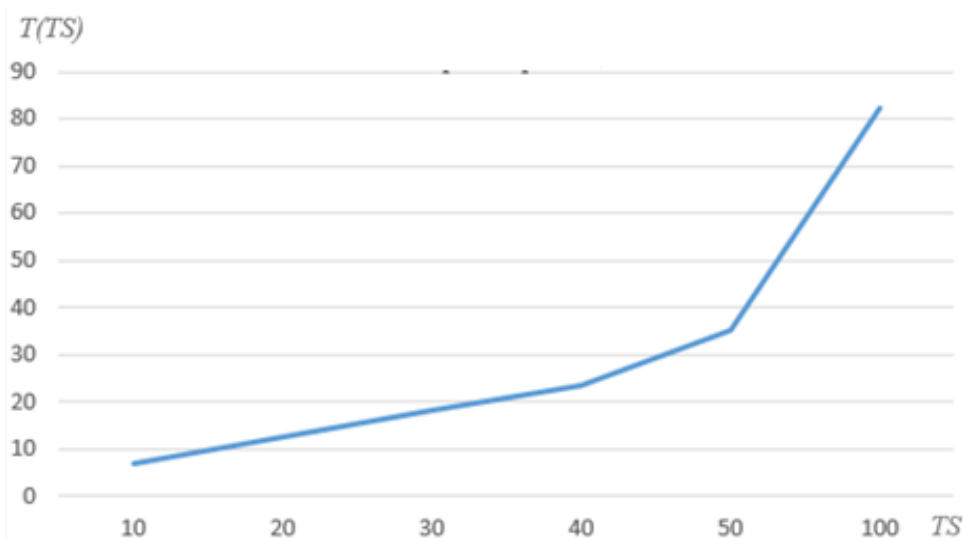
**Таблица 1**

**Экспериментальные данные, отражающие зависимость времени самовосстановления и переустановки ПО от его общего размера и размера поврежденных файлов (разработано автором)**

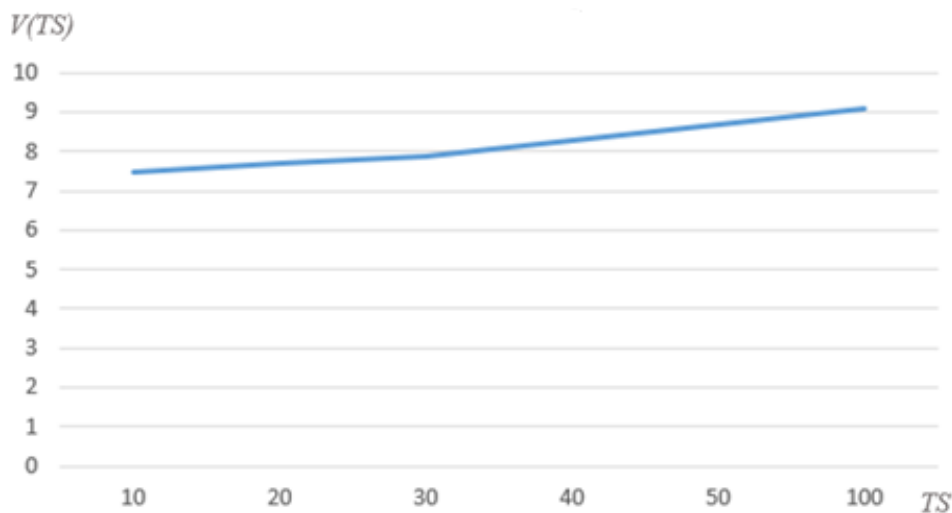
№	Общий размер ПО (МБ)	Размер поврежденного файла (КБ)	T (С)	V (С)
1	10	400	7	7,5
2	15	100	9,2	6
3	20	250	12,4	7,2
4	25	600	15,7	8,3
5	30	620	18,1	8,4
6	35	150	21,3	7,1
7	40	580	23,3	8,2

№	Общий размер ПО (МБ)	Размер поврежденного файла (КБ)	T (С)	V (С)
8	45	180	28,1	7,4
9	50	590	35,2	8,5
10	100	450	82,2	8,9
11	300	480	267,5	12,3
12	500	800	437,2	18,5
13	10	3000	7	13,2

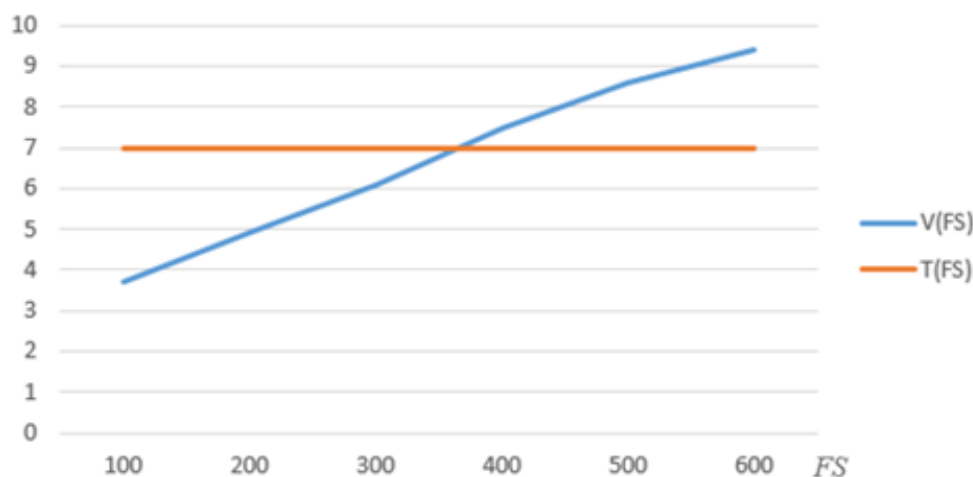
На основе полученных результатов можно составить графики зависимости  $T$  и  $V$  от общего размера ПО (Total Size), которые наглядно демонстрируют различия двух рассматриваемых методов (рис. 3-5).



**Рисунок 3.** График зависимости времени переустановки ПО от его общего размера (разработано автором)



**Рисунок 4.** График зависимости времени самовосстановления ПО от его общего размера (разработано автором)



**Рисунок 5.** График зависимости времени переустановки и самовосстановления ПО от размера поврежденных файлов (разработано автором)

Согласно представленным графикам можно сделать вывод, что время переустановки ПО ( $T$ ) зависит только от общего размера ПО ( $TS$ ), а время восстановления ПО ( $V$ ) зависит как от общего размера ( $TS$ ), так и от размера поврежденных файлов ( $FS$ ). Поэтому предложенный механизм самовосстановления показывает сравнительно меньшее время, если размер поврежденных данных составляет до 15% от общего размера ПО.

### Заключение

Таким образом, была предложена методика обнаружения и устранения отказов ПО. Согласно данной методике, механизм самовосстановления закладывается на стадии проектирования и реализации. Сильными сторонами такого механизма являются:

1. Независимость восстановления ПО от человеческого фактора.
2. Накопление отчетов об отказах ПО для дальнейшего усовершенствования механизма.
3. Отсутствие необходимости переустановки ПО.
4. Высокая скорость восстановления при небольших размерах поврежденных данных.

Недостатками данного механизма являются:

1. Усложнение программного кода.
2. Низкая скорость восстановления при больших размерах поврежденных данных.

Несмотря на приведенные недостатки, предложенная методика является эффективной т.к. позволяет автоматизировать процесс восстановления программного обеспечения, что очень важно на масштабных предприятиях, остановка работы которых недопустима, например, на критически важных объектах [9].

## ЛИТЕРАТУРА

1. Журавлева А.В. Роль программного обеспечения в управлении предприятием / А.В. Журавлева // Сборник научных трудов международной научной конференции «Информационные науки в управлении, социальной сфере и медицине», Томск 2014 г.
2. Ивутин А.Н. К вопросам об отказах программного обеспечения / А.Н. Ивутин, Е.В. Ларкин // Известия тульского государственного университета. Технические науки. – 2014 №3. – С. 203-209.
3. Федеральное агентство по техническому регулированию и метрологии. Национальный стандарт РФ. ГОСТ Р 51901.12-2007 Метод анализа видов и последствий отказов.
4. Дроботун Е.Б. Критичность ошибок в программном обеспечении и анализ их последствий / Е.Б. Дроботун // Фундаментальные исследования. - 2009. - №4.
5. Selvin, G., David, E. and Lance, D. (2002) A Biologically Inspired Programming Model for Self-Healing Systems. WOSS'02 Proceedings of the First Workshop on Self-Healing Systems, Charleston, November 2012, 102-104.
6. Холод И.И. Самовосстановление программных систем на основе автоматической классификации ошибок / И.И. Холод, А.В. Кочетков // Информационные технологии. - 2008. - №6. - С. 16-21.
7. Fuad, M., Deb, D. and Baek, J. (2011) Self-Healing by Means of Runtime Execution Profiling. Proceedings of 14th International Conference on Computer and Information Technology (ICCIT2011), Dhaka, 22-24 December 2011, 202-207.
8. Обзор процесса разработки программного обеспечения // habrahabr.ru: информационно-технологический портал Хабрахабр.2015 URL: <https://habrahabr.ru/post/255991/> (дата обращения 23.10.2016).
9. Министерство Российской Федерации по делам гражданской обороны, чрезвычайным ситуациям ликвидации последствий стихийных бедствий // Методика отнесения объектов государственной и негосударственной собственности к критически важным объектам для национальной безопасности Российской Федерации. - 2012.

**Varlataya Svetlana Klimentyevna**

Far Eastern federal university, Russia, Vladivostok  
E-mail: [Sk-varl@mail.ru](mailto:Sk-varl@mail.ru)

**Kaluzhin Egor Aleksandrovich**

Far Eastern federal university, Russia, Vladivostok  
E-mail: [Workout24@mail.ru](mailto:Workout24@mail.ru)

**Monastirskiy Dmitriy Sergeevich**

Far Eastern federal university, Russia, Vladivostok  
E-mail: [Egorcool-90@mail.ru](mailto:Egorcool-90@mail.ru)

## **The technique of finding and elimination software faults**

**Abstract.** This article provides new approach of finding and elimination software faults. The main idea is that self-recovery technique should be implemented on the stage of designing and realization of software. This technique scans software components in real time and find main reasons of faults: deleting, replacing and reconfiguring. The feature of this technique is reporting, that allows developers to enhance their software. For efficiency estimation, the offered approach is compared with widely spread method of fault elimination – reinstalling of software.

**Keywords:** fault-tolerance; self-recovery; software; scanning of files; fault reasons; reinstalling; condition analyze