

## НАДЕЖНОСТЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Теория надежности как наука получила развитие применительно к сложным техническим системам. Необходимость контроля параметров компонентов систем и систем в целом доказана практикой. В этом направлении выполнено значительное количество работ по надежности применительно к техническим системам, разработано множество моделей.

Применение моделей в ряде случаев позволяет не только оценивать показатели надежности и готовности технических систем и их компонентов, но и дают возможность предсказания этих показателей на основе накопленного опыта.

Под системой в теории надежности понимается совокупность подсистем или элементов, функционально объединенных в соответствии с некоторым алгоритмом взаимодействия при выполнении заданной задачи в процессе применения по назначению. Под это определение системы полностью подходит определение программного обеспечения.

Чтобы перейти к определению понятия надежности программного обеспечения, необходимо определиться с понятием ошибки в программе.

Рассмотрим подходы к определению понятия ошибки в программном обеспечении.

В программном обеспечении содержится ошибка, если его поведение не соответствует спецификациям. В этом определении делается предположение, что спецификации абсолютно корректны. Но создание спецификаций – это сложный процесс перевода неформализованных и зачастую неполных требований пользователя к программному обеспечению. Этот перевод может приводить к внесению ошибок в саму спецификацию. В итоге может возникнуть ситуация, когда пользователь говорит, что обнаружил в системе ошибку, на что разработчик отвечает, что система так и задумывалась. Ярким примером может служить система раннего обнаружения баллистических снарядов, разрабатываемая в США. Эта система должна была наблюдать за объектами, движущимися по направлению к США, и проводить ряд защитных мероприятий. Ранняя версия этой системы по ошибке принимала поднимающуюся луну за снаряд, движущийся над северным полушарием.

Другое определение: в программном обеспечении содержится ошибка, если ПО ведет себя не по спецификации при использовании системы в установленных при разработке пределах. Главным изъяном этого определения является то, что в случае использования системы в непредвиденных ситуациях поведение ПО должно быть разумным. Например, диспетчерская авиационная система, рассчитанная на слежение за 200 самолетами, не должна выходить из строя при появлении в ее секторе 201-го самолета.

Согласно еще одному определению ошибка заключается в том, что ПО ведет себя не в соответствии с официальной документацией на ПО. Здесь возникает вопрос в корректности документации на ПО. Но корректная

документация описывает ожидаемую и планируемую работу, но не описывает поведение ПО в случае ошибки пользователя.

Самым корректным определением ошибки является такое поведение, когда ПО не выполняет того, что пользователю разумно от него ожидать. А отказ является проявлением ошибки.

Главной особенностью надежности ПО является то, что ошибка не является внутренним свойством программы. Именно по этому Дейкстра говорит, что тестированием можно определить только наличие ошибки. Тестированием невозможно доказать отсутствие ошибок в программе.

Из практики известно, что ошибки бывают различных типов и приводят к различным последствиям. Ошибки могут быть в орфографии текста, выводимого на экран, а могут быть в логике программы и приводить к катастрофическому отказу системы с потерей всех данных. Поэтому при определении надежности программ необходимо учитывать последствия для пользователя из-за проявления ошибок в ПО.

Надежность ПО – вероятность работы без отказов в течение определенного периода времени, рассчитанная с учетом стоимости для пользователя каждого отказа.

Главной проблемой в обеспечении надежности ПО является внутренняя сложность программной системы.

При разработке аппаратуры используются отработанные проверенные блоки, в то время как при создании программы используются более примитивные блоки: операторы языка программирования, ранее написанные модули. Объектно-ориентированный подход в программировании помогает значительно облегчить эту особенность создания программ. При объектно-ориентированном программировании ранее созданный, отработанный и проверенный код гораздо легче использовать в новых проектах; при этом значительно ослабляется проблема повторного использования кода и упрощается разработка программ.

Другим отличием проектирования программ от проектирования аппаратуры является то, что работа создаваемой аппаратуры не зависит от входной информации, в то время как работа программ целиком зависит от нее. Входной информационный поток по желанию пользователя может быть сколь угодно сложным.

Еще одним отличием надежности программ от надежности аппаратуры является то, что компоненты программ не изнашиваются и не подвержены процессу старения, и возникновение дефектов программ в процессе производства можно легко обнаружить и исключить.

Надежность программ в первую очередь зависит от процесса проектирования и программирования. Последующие процессы в жизненном цикле программной системы (тестирование, сопровождение) направлены на повышение надежности.

Главным препятствием в создании надежных программ является бюджет разработки, так как повышение надежности требует большого количества финансовых средств.

В настоящее время в теории надежности программного обеспечения развиваются следующие группы методов повышения надежности:

**1. Предупреждение ошибок:** эти методы направлены на снижение уровня ошибок в процессе проектирования программных средств и кодирования. Они включают в себя: использование хороших проверенных методов разработки, инженерные подходы к процессу спецификаций требований к ПО, создание простого и понятного кода программ и др. В настоящее время в этой области большое внимание уделяется формализации задания спецификаций к ПО (разработка Computer-Aided Software Engineering средств) и повторному использованию кода (объектно-ориентированное программирование).

**2. Удаление ошибок:** эти методы направлены на обнаружение и исключение ошибок из программ. Используются методы доказательства корректности программ, методы тестирования и верификации, создаются средства для автоматизации генерации тестовых данных для эффективной проверки (CASE). Главный вопрос в этой области: как измерить качество тестов и обеспечить взаимосвязь тестов с надежностью?

**3. Повышение устойчивости программ к ошибкам:** эти методы направлены на создание ПО, способного предоставлять сервис пользователям при наличии ошибок в системе, оставшихся после разработки. Оставшиеся ошибки могут проявиться в процессе функционирования ПО, а могут и не проявиться, но методы этой группы направлены на создание механизмов, предотвращающих отказ системы в случае проявления ошибки. Методы этой группы основаны на введении в программную систему информационной и программной избыточности и широко зарекомендовали себя в аэрокосмической отрасли, в ядерной энергетической отрасли и медицине.

**4. Предсказание ошибок:** численная оценка наличия ошибок и потока отказов, моделирование программной надежности. Эти методы направлены на создание моделей надежности, сбор статистики отказов, применение моделей надежности, обоснование выбора подходящей модели, анализ и интерпретацию результатов, принятие и обоснование управленческих решений.