

РЕАЛИЗАЦИЯ ПОДХОДА ОБРАБОТКИ ИСКЛЮЧИТЕЛЬНЫХ СИТУАЦИЙ В JAVA НА ОСНОВЕ НОТИФИКАЦИЙ И ВОЗВРАТА РАСШИРЕННОЙ ИНФОРМАЦИИ ОБ ОШИБКАХ, СОДЕРЖАЩИХСЯ В ОБЪЕКТЕ РЕЗУЛЬТАТА

Куприянов А.А. Национальный исследовательский университет ИТМО
Научный руководитель – преподаватель, Николаев В.В.
Национальный исследовательский университет ИТМО

В работе представлена реализация подхода обработки ошибок, который может быть использован как замена стандартного механизма обработки ожидаемых исключений в языке Java

Введение. Во время выполнения программы, написанной на Java, могут возникать различные исключительные ситуации, которые могут повлиять на нормальное поведение программы. Эти исключительные ситуации можно разделить на два типа: ожидаемые и непредвиденные исключения.

Как правило, если возникает ожидаемое исключительное состояние, детали исключения менее важны, чем сам факт его возникновения. В таком случае, использование стандартного механизма исключений, предоставляемого языком программирования для обработки ожидаемых ошибок, может быть чрезмерным или даже неправильным, что приводит к увеличению сложности программы и дополнительным затратам времени

Основная часть. В 2014 году Мартин Фаулер предложил подход нотификаций. Данный подход основан на том, чтобы не бросать исключение в случае ошибки, а возвращать объект, который содержит описание возникшей ошибки. В статье М. Фаулера предлагалось использовать строковое значение для описания ошибки, а сама нотификация не была унифицированной. Она была полезной при валидации, о чем и была основная тема статьи.

Этот подход был улучшен путем добавления функциональных возможностей для объекта-обертки, которые позволяют манипулировать информацией об ошибке и исполнением кода. В объект-обертку добавлены методы, через которые можно передать обработчик ошибки и получить сам возвращаемый объект, если ошибок не было.

Для реализации описанного подхода и его улучшений разработана библиотека NoTex (Not Exception). В NotEx оберткой является класс Result, который содержит возвращаемое значение, а также уведомление в случае возникновения ошибки

Для получения значения из Result нужно вызвать метод, который принимает в качестве аргумента обработчик ошибки. В таком случае, пользователю необходимо будет явно обрабатывать ошибку, что исключает возможность пропуска необработанных ошибок, что может произойти при исключениях.

Библиотека предлагает не только альтернативу исключениям, но и изменение стиля написания кода, а также улучшения его читаемости, так как код становится линейным. Также основной класс-обертка библиотеки Result имеет несколько вспомогательных методов, которые упрощают работу с последовательностью исполнения кода.

Выводы. Было проведено сравнение использования библиотеки и механизма исключений через бенчмарк JMH. При исполнении простого метода мы получаем скорость при

использовании исключений 958048 операций в секунду, а при использовании библиотеки 274030504 операций в секунду, что больше в 286 раз.

Библиотека NotEx может использоваться для валидации. В отличие от исключений, она не прерывает исполнение кода, благодаря чему можно валидировать все поля объекта даже с интеграцией с JSR 303

Таким образом, разработанная библиотека имеет обширный функционал, позволяющий улучшить код благодаря линейному выполнению и явной обработке ошибок. Тем не менее, речь идет не о полной замене исключений, а частичного замещения в тех местах, где исключения чрезмерны или используются неправильно.

Куприянов А.А. (автор)

Подпись

Николаев В.В. (научный руководитель)

Подпись