

переключаться во время работы скрипта между различными вспомогательными программами. Дело в том, что для написания полноценного тестового скрипта в таком случае, необходимо использовать дополнительные программные инструменты, такие как фреймворк ITCL, который позволяет запомнить элементы управления в тестируемом приложении и создать для них статические объекты в среде RFT, которые в дальнейшем можно использовать для создания как тест-скриптов так и специализированных вспомогательных элементов – хелперов.

Далее будет описано, как удалось автоматизировать процесс тестирования ETL последовательностей при обработке данных в GDP-системе. Для извлечения, преобразования и загрузки данных используется множество систем, однако в данном конкретном случае использовалось программное обеспечение IBM InfoSphere DataStage Designer. Интерфейс системы позволил построить объектно-ориентированный проект в среде RFT, который необходим для реализации трехуровневой системы ITCL-фреймворка.

После выбора программных сред была сделана работа по интеграции, необходимая для того, чтобы тестируемое Web-ориентированное приложение было доступно и видимо в пределах проекта, в котором реализовывалось взаимодействие RFT и IBM InfoSphere. Большую помощь оказало наличие специализированных виджетов, которые были включены в фреймворк. Именно они позволили написать небольшой интерфейс, который позволяет переключаться между приложениями в реальном времени.

Довольно большое количество времени пришлось уделить и работе с базой данных тестируемого приложения. В процессе создания комплекса возникла необходимость написания специализированного API, в котором использовались JDBC-коннектор для языка Java. Разработанный API позволил гибко использовать данные из базы данных для валидации различной информации в процессе тестирования.

Н.В. Дементьев (УО «ГГУ им. Ф. Скорины», Гомель)

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

ИСПОЛЬЗОВАНИЕ ФРЕЙМВОРКА ITCL В АВТОМАТИЗАЦИИ ТЕСТИРОВАНИЯ

Фреймворк ITCL представляет собой трехуровневую архитектуру, реализующую разделение объектов, используемых в автоматизированном тестировании от их поведения. Во время исследования предметной области автоматизации тестирования выяснилось, что использования

стандартных библиотек недостаточно ввиду не читаемости записываемых скриптов и малой гибкости написания скриптов вручную.

Интеграция IBM Framework довольно проста, достаточно импортировать библиотеку с классами и виджетами к созданному проекту и реализовать трехуровневую структуру внутри проекта. Далее можно использовать два сценария разработки скриптов. При первом объекты приложения необходимо добавить в карту объектов каждого скрипта и уже исходя из имеющегося списка реализовывать тест-скрипт. При втором можно использовать специальный ClassGenerator, который позволяет выделить все необходимые объекты управления тестируемого приложения в отдельный комплекс объектов, которые можно использовать как для создания специализированных методов, являющихся универсальными для всех будущих скриптов.

В данном исследовании приходилось пользоваться обоими подходами сразу, так как найти необходимый объект в приложении довольно просто, но порой приходилось сталкиваться с тем, что объектов с одним именем или идентификатором оказывалось больше одного, и тогда польза от использования ClassGenerator сводилась к минимуму. Именно для таких случаев был написан специальный метод, который генерирует геттеры для объектов по найденным вручную объектам.

Благодаря ITCL-фреймворку удалось создать специальный API. Происходило это в несколько этапов. Первым этапом стал поиск объектов и создание для них хранилища. Вторым этапом стало написание специальных библиотек с методами, которые будут использоваться при написании скриптов. Примечательным является тот факт, что всю работу можно сделать еще до того, как приложение будет иметь первую рабочую версию. В третьем этапе происходит написание самих тест-скриптов, а также рефакторинг кода первых двух этапов, что повышает его быстродействие. Так, например, в некоторых методах пришлось отказаться от простых структур данных и перейти на хешируемые, такие как Hashtable, HashMap и другие, тем самым увеличив производительность скриптов.

А.С. Демиденко (УО «ГГУ им. Ф. Скорины», Гомель)

Науч. рук. **Е.Е. Пугачёва**, ассистент

СРАВНИТЕЛЬНАЯ ХАРАКТЕРИСТИКА

1С: ПРЕДПРИЯТИЕ 8.2 И 7.7

1С: Предприятие 8.2 является универсальной системой автоматизации деятельности предприятия. За счет своей универсальности система