

## Реферат

1. *Какую систему Вы изучаете в Вашей диссертационной работе? Что является объектом исследования.*

Объектом моего диссертационного исследования являются плагинные системы. В своем диссертационном исследовании я определяю плагинную систему как способ реализации приложения. Возможности приложения при таком способе организации ограничиваются объемом возможностей установленных расширений приложения - плагинов.

Пример плагинной системы - Eclipse IDE. Это выполненная в виде плагинной системы среда разработки ПО. Система образуется за счет того, что плагины могут взаимодействовать друг с другом и расширять уже имеющийся функционал других плагинов.

2. *Каково назначение Вашей системы?*

В моей диссертационной работе плагинные системы рассматриваются с позиции их использования в качестве среды разработки ПО.

Тогда назначением плагинной системы является формирование такой инструментальной среды, которая позволяла бы оператору применять свои навыки и умения в рамках имеющейся квалификации для решения бизнес задач в заданной предметной области.

3. *Какова цель исследования системы? Где она может быть использована?*

Исследование плагинных систем производится с целью разработки правил построения инструментальных средств конфигурирования.

Разработанные правила должны позволить разработчикам инструментальных средств конфигурирования, выполненными в виде плагинов, строить свои решения так, чтобы минимизировать затраты на сопровождение проекта и максимизировать число возможных комплектаций с реализацией требований на систему.

Это позволит сократить объем невостребованного заказчиком в поставке функционала, что снизит стоимость поставляемого функционала, следовательно поднимет конкурентоспособность программного решения на рынке.

Использование плагинной системы в качестве среды разработки подразумевает ее использование для сокращения временных и стоимостных издержек на разработку ПО, а так же отслеживание и информирование о потенциальных неисправностях или уязвимостях разрабатываемого ПО.

Она может быть использована в каждом из процессов разработки ПО. При этом важно, чтобы для каждого из процессов она имела вид наиболее точно характеризующий ее назначение для решаемых в рамках процесса задач.

4. *Частью какой надсистемы является изучаемая система?*

Для плагинной системы как среды разработки ПО надсистемой является процесс разработки ПО.

Процесс разработки ПО включает следующие процессы:

- (a) процесс проектирования;
- (b) процесс разработки требований;
- (c) процесс кодирования;
- (d) процесс интеграции.

5. *Из каких подсистем состоит система?*

Плагинная система состоит из плагинов, состав которых определяет ее функциональные возможности.

Для использования плагинной системы как инструментальной среды разработки ПО считаю, что она должна включать такие плагины как:

- плагин, предоставляющий редактор с подсветкой синтаксиса и возможностями автодополнения для используемого в разрабатываемом проекте языка программирования;
- плагин навигации и поиска компонент в проекте разрабатываемого ПО;
- плагин управления составом подключаемых компиляторов и правилами сборки проекта разрабатываемого ПО;
- плагин взаимодействия с системой контроля версий (СКВ) и др.

6. *Какие задачи решают подсистемы в составе Вашей системы?*

Плагин, предоставляющий редактор с подсветкой синтаксиса и возможностями автодополнения для используемого в разрабатываемом проекте языка программирования решает следующие задачи:

- обеспечение оператора текстовым редактором;
- сокращение времени написания исходного кода за счет предоставления оператору вариантов автодополнения;
- уменьшение когнитивной сложности за счет подсветки ключевых слов в тексте исходного кода.

Плагин навигации и поиска компонент в проекте разрабатываемого ПО решает следующие задачи:

- уменьшение когнитивной сложности навигации по структуре проекта за счет отображения структуры проекта, например, в виде дерева;
- уменьшение когнитивной сложности оценки объема функционала реализованного в файле исходного кода за счет предоставления списка описанных функций или классов;
- сокращение времени поиска файла исходного кода в проекте разрабатываемого ПО по имени или содержанию.

Плагин управления составом подключаемых компиляторов и правилами сборки проекта разрабатываемого ПО решает следующие задачи:

- уменьшение сложности управления составом ключей компиляторов;
- информирование оператора об используемом в процессе сборки компилятора;
- уменьшение когнитивной сложности анализа результата работы компилятора при возникновении ошибок компиляции.

Плагин взаимодействия с СКВ решает следующие задачи:

- обеспечение пользовательским интерфейсом для взаимодействия с СКВ;
- информирование оператора о применяемой базовой версии для внесения изменений в проект разрабатываемого ПО;
- автоматизация внесения изменений в проект разрабатываемого ПО в соответствии с действующей на проекте дисциплиной процесса управления конфигурацией.

*7. Сформулируйте кратко сценарий функционирования системы.*

Для среды разработки ПО выделяю следующий сценарий функционирования:

- (a) установка на базовую версию проекта разрабатываемого ПО;
- (b) внесение изменений в один или несколько файлов исходного кода;
- (c) выполнение формирования файлов исполняемого объектного кода и сборка исполняемого образа с формированием файлов компонентов параметрических данных;
- (d) загрузка исполняемого образа и файлов компонентов параметрических данных на целевой вычислитель;
- (e) фиксация внесенных изменений в соответствии с дисциплиной процесса управления конфигурацией.

*8. Какие факторы внешней среды Вы учитываете при анализе функционирования системы?*

При анализе функционирования системы учитываются следующие факторы внешней среды:

- программно-аппаратные характеристики вычислительной системы, в которой выполняется работа среды разработки ПО:
  - (a) операционная система;
  - (b) тип, количество ядер и тактовая частота процессора;
  - (c) объем постоянной и оперативной памяти.
- частота и количество воздействий оператора на элементы управления среды разработки ПО;
- факторы сетевого обмена:

- (a) пропускная способность сети;
- (b) количество абонентов;
- (c) используемые протоколы сетевого обмена.

9. *Какой информацией о факторах внешней среды Вы располагаете (детерминированные, случайные, интервально неопределенные, активное противодействие «противника» или конкурента)?*

К детерминированным факторам внешней среды отношу:

- (a) количество одновременно взаимодействующих со средой операторов - оператор всегда один;
- (b) количество элементов управления, через которое оператор взаимодействует со средой разработки - исключено взаимодействие одновременно с несколькими элементами управления;
- (c) среда разработки осуществляет свою работу под управлением одной операционной системы.

К случайным факторам внешней среды отношу:

- (a) на какой следующий элемент управления будет осуществлено воздействие оператора;
- (b) количество ресурсов вычислительной системы, доступных для работы:
  - количество процессного времени и объем оперативной памяти влияет на быстродействие среды разработки;
  - политика безопасности определяет характер взаимодействия с ресурсами операционной системы, например, может не выполняться операция записи в файл журнала.

Конкурентная борьба моей проектируемой системы от существующих аналогов на рынке осуществляется по критериям:

- (a) объем функционала в его среде разработки;
- (b) стоимость поставляемого функционала;
- (c) время формирования поставки.

10. *Какие показатели эффективности системы в целом и ее подсистем Вы рассматриваете?*

Считаю, что эффективность системы обратно пропорциональна числу невостребованного заказчику функционала в поставке программного решения.

Так, выделяю коэффициент эффективности поставки:

$$k = v^* - (v_u/v^*)$$

где:

$k$  - коэффициент эффективности;

$v^*$  - количество требований к ПО, реализованных в рамках поставки программного решения;

$v_u$  - количество требований к ПО, реализованных в рамках поставки программного решения и не востребуемых для заказчика.

Как видно из формулы коэффициент эффективности максимален и равен 1 тогда, и только тогда, когда число  $v_u$  равно 0, т.е. в поставке отсутствует реализация не востребуемых для заказчика требований. И минимален, равен 0 тогда, и только тогда, когда число  $v_u$  равно  $v^*$ , т.е. все требования, реализованные в рамках поставки, не востребуемы для заказчика.

*Примечание: считается, что поставка содержит реализацию всех востребуемых для заказчика требований. Таким образом  $k$  всегда больше 0.*

К показателям эффективности плагинов отношу:

- (а) время отклика компонента на воздействие оператора на элемент управления, относящегося к нему.
- (б) отказоустойчивость компонента. Например, если в программном решении не предусмотрен обработчик ситуации при которой невозможно осуществить взаимодействие с ресурсом вычислительной системы, то невозможно гарантировать дальнейшую корректную работу всего программного решения.