

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ**

**Федеральное государственное бюджетное образовательное учреждение  
высшего образования «Московский авиационный институт (национальный  
исследовательский университет)»**

Институт № 6 «Аэрокосмический»

Кафедра 604 «Системный анализ и управление»

Реферат

На тему: «Исследование и разработка принципов построения инструментальных  
средств конфигурирования в плагинных системах»

По курсу: «Системный анализ»

Выполнил:  
Аспирант группы:  
Проверила:

Шаблий А.Д.  
М8О-108А-23  
к.т.н. Дьячук А. К.

Москва 2024

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

Следуя своей диссертационной работе сложной технической системой я называю архитектуру инструментального средства конфигурирования, выполненного в виде набора плагинов. Будучи интегрированными в плагиновую среду, они ее дополняют и конечный объем функционала, доступный пользователю, будет перерасчитан. Конечный объем функционала определяется суммарным объемом функционала каждого установленного в среду плагина.

2. *Каково назначение Вашей системы?*

Архитектура предназначена для выработки правил, методов и шаблонов при разработке инструментального средства конфигурирования.

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

Целью исследования архитектуры программного средства как сложной технической системы является формирование паттернов к ее построению, используя которые, компания-производитель снизит затраты на постпродажное обслуживание поставки программного средства, исключая неконтролируемый рост затрат на его разработку.

Использование предполагается для решения следующей задачи: предположим, что по составленному техническому заданию был реализован некоторый функционал, который для возможности поставки в плагиновую среду сгруппирован по плагинам. Потенциальный заказчик потребовал не весь реализованный объем, а только его часть.

В этом случае определяется объем функционала потребный заказчику и сбор функциональных зависимостей, без которых потребный функционал не может быть поставлен.

Результирующий объем функционала поставляется заказчику. Задача заключается в минимизации объема непотребного заказчику функционала в поставке.

4. *Частью какой надсистемы является изучаемая система?*

Для архитектуры программного средства надсистемой является жизненный цикл программного обеспечения. Начиная от получения сформированных требований к программному средству в виде технического задания на разработку и заканчивая формированием поставки, в которую входят потребные заказчику требования.

5. *Из каких подсистем состоит система?*

Подсистемами для архитектуры программного средства являются:

- (a) требования к программному обеспечению;
- (b) файлы исходного кода;
- (c) плагины.

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

В требованиях к программному обеспечению описаны функциональные возможности программного средства.

В файлах исходного кода описана реализация требований к программному обеспечению на языке программирования.

Плагины содержат файлы исходного кода и обеспечивают их выполнение в плагиновой среде.

7. *Сформулируйте кратко сценарий функционирования системы.*

Для определения объема функционала, включенного в поставку, предполагается следующий сценарий:

- (a) заказчик назначает потребный объем функционала;
- (b) определяется соответствующий объем потребных требований;
- (c) определяются файлы исходного кода, которые реализуют потребные требования;
- (d) определяются плагины, которые содержат вышеуказанные файлы исходного кода;
- (e) формируется суммарная выборка файлов исходного кода, которые включены в вышеуказанные плагины;
- (f) формируется суммарная выборка требований, которые реализуются в таких файлах исходного кода и делается вывод о составе поставки.

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

К факторам внешней среды относятся:

- (a) изначальный перечень требований технического задания на разработку;
- (b) потребный заказчику функционал.

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

Изначальный перечень требований технического задания задан изначально. Изменение его состава согласуется со всеми участниками жизненного цикла программного обеспечения и актуальные данные о нем известен потенциальному заказчику.

Потребный заказчику функционал транслируется в требования. Общий перечень требований описан в техническом задании на разработку. Таким образом потребные требования не могут выходить за рамки множества требований описанных в техническом задании.

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

К показателям эффективности относятся:

- (а) возможность формирования различных комплектаций поставки программного средства;
- (б) объем кодовой базы проекта;
- (с) отношение числа невостребованных требований к общему числу реализованных требований в поставке.

11. *Как в этих показателях учитывается информация о внешней среде?*

Благодаря возможности формирования различных комплектаций достигается уменьшение коэффициента бесполезности при

$$\begin{cases} R_n^1 \neq R_n^2 \\ |R_n^1| = |R_n^2| \end{cases}$$

где:  $R_d^1$  и  $R_d^2$  - множества востребованных требований в рамках разных поставок

$$F_p = \begin{cases} 1 & \text{если условие выполняется} \\ 0 & \text{если условие не выполняется} \end{cases}$$

где:  $F_p$  - возможность формирования различных комплектаций

Объем кодовой базы влияет на стоимость разработки и поддержки проекта. Чем он больше - тем дороже разработка и поддержка. Объем кодовой базы зависит от двух факторов:

- (а) чем больше требований технического задания - тем больше объем кодовой базы;
- (б) количество ограничений на наличие функциональных зависимостей. Чем больше ограничений, тем более сложные связи. Чем сложнее связи - тем больше объем кодовой базы. Сложность связей увеличивается, если элементы цепочки зависимостей находятся в разных плагилах.

$$V_c = \begin{cases} 1 & \text{если } |P^*| = 1 \\ 2 & \text{если } |C^*| \gg |P^*|, \\ 3 & \text{если } |C^*| \not\gg |P^*| \end{cases}$$

где:  $V_c$  - объем кодовой базы

$P^*$  - множество плагинов

$C^*$  - множество файлов исходного кода

В формируемую поставку включаются как потребные требования, так и не востребовавшиеся. Исходя из потребных заказчику требований формируется цена поставки, в которую включена цена на постпродажное обслуживание. Коэффициентом бесполезности характеризуется объем выполняемых работ подразделениями постпродажного обслуживания неоплаченного заказчиком и выполняемых за счет компании-поставщика.

$$K_f = \frac{|R_{un}^*|}{|R_d^*|},$$

где:  $K_f$  - коэффициент бесполезности

$R_{un}^*$  - множество требований не востребовавшихся заказчиком в рамках поставки

$R_d^*$  - множество требований реализованных в рамках поставки

12. *Какое множество альтернатив системы (дискретное, непрерывное, дискретно-непрерывное) Вы рассматриваете?*

Рассматриваются две альтернативы системе:

- (а) построение по схеме «1 требование - 1 плагин»;
- (б) построение по схеме «все требования - 1 плагин».

Для альтернативы «1 требование - 1 плагин» характерно:

- (а) один файл исходного кода не может реализовывать более одного требования;
- (б) файл исходного кода не может иметь зависимость на другой файл исходного кода если они реализуют разные требования;
- (с) плагин не может включать файлы исходного кода, которые реализуют разные требования.

Для альтернативы «все требования - 1 плагин» характерно:

- (а) нет ограничений на число требований, которое реализует один файл исходного кода;
- (б) нет ограничений на число и характер зависимостей между файлами исходного кода;
- (с) в проекте один плагин, который включает все файлы исходного кода.

13. *Назовите основные структуры и параметры, характеризующие рассматриваемое Вами множество альтернатив системы.*

Для альтернативы «1 требование - 1 плагин» характерно:

- (а)  $F_p = 1$
- (б)  $V_c = 3$ ;
- (с)  $K_f = 0$ .

Здесь объем кодовой базы максимален из-за невозможности переиспользовать код, который участвует в реализации других требований. Появляется необходимость в создании большого числа сущностей в проекте, что приводит к неконтролируемому росту стоимости разработки и сопровождения. Коэффициент бесполезности минимален и равен 0. Данная схема гарантирует, что в формируемую поставку не войдет функционал, реализующий не востребуемые требования.

Для альтернативы «все требования - 1 плагин» характерно:

- (a)  $F_p = 0$
- (b)  $V_c = 1$ ;
- (c)  $K_f \approx 1$ .

Здесь объем кодовой базы минимален. Нет искусственных ограничений на построение кода, что существенно снижает стоимость разработки и поддержки проекта.

Однако коэффициент бесполезности принимает максимальное значение. Это говорит о том, что практически вся работа служб постпродажного обслуживания производится за счет компании-поставщика.

14. *При каких ограничениях на значения непрерывных параметров системы решается задача?*

Уменьшение числа не востребуемых требований в формируемой поставке целесообразно при выполнении условия:

$$|R_n^*| \ll |R^*|,$$

где:  $R_n^*$  - множество требований потребных заказчиком  
 $R^*$  - множество требований технического задания

Альтернативу «1 требование - 1 плагин» целесообразно применять при  $|R^*| \not\approx 0$ .

Альтернативу «все требования - 1 плагин» целесообразно применять при  $|R_n^*| \approx |R^*|$ .

15. *Составьте функциональную схему и опишите функциональные связи между подсистемами в составе системы, а также между подсистемами и внешней средой, которые Вы предполагаете учитывать при разработке математической модели системы.*

Функциональные связи между подсистемами следующие:

- (a) одно требование может быть реализовано в одном или нескольких файлах исходного кода;
- (b) один файл исходного кода может реализовывать одно или несколько требований;

- (с) один файл исходного кода не может быть включен в несколько плагинов;
- (d) файлы исходного кода могут иметь зависимости друг на друга, в том числе и циклические;
- (е) плагины так же имеют друг на друга зависимости: если первый плагин содержит файл имеющий зависимость на файл из второго плагина, то первый плагин имеет зависимость на второй плагин;
- (f) циклические зависимости между плагинами запрещены.

Функциональная схема связей подсистем друг с другом приведена на рисунке 1.

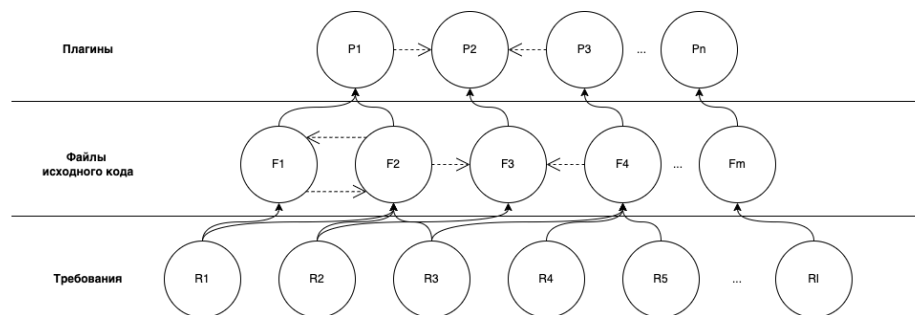


Рис. 1: Схема связей подсистем

Функциональная схема связи системы с внешней средой приведена на рисунке 2.

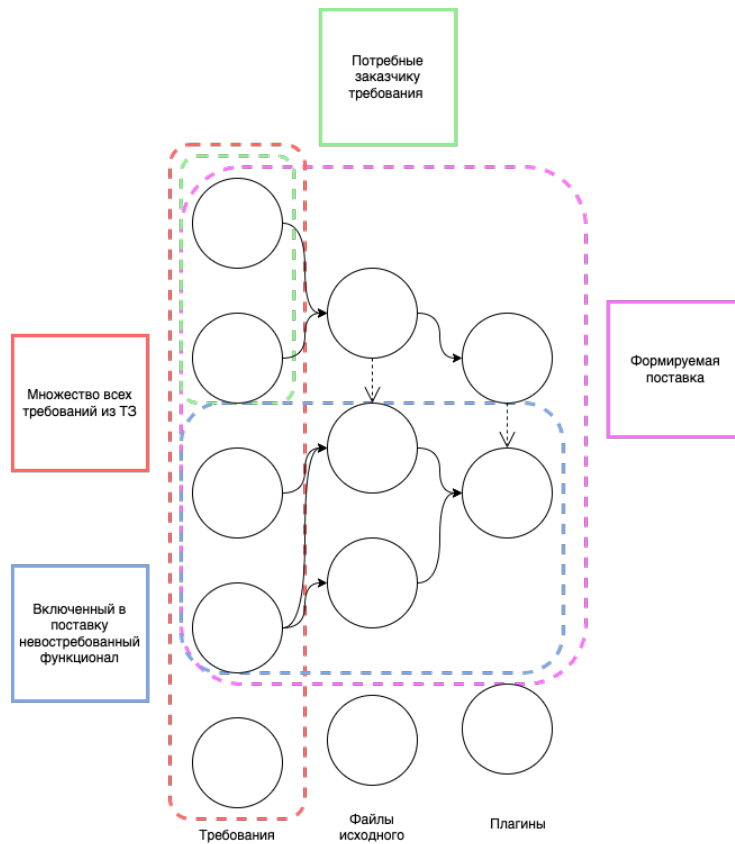


Рис. 2: Связь системы и надсистемы

16. *Какую систему (или системы) Вы готовы рассматривать как актуальный прототип по отношению к системе, рассматриваемой в Вашей работе?*

В качестве актуального прототипа по отношению к рассматриваемой системе я рассматриваю спецификацию OSGI. В частности ее реализацию Equinox в IDE Eclipse. Благодаря ей достигается возможность создавать приложения в виде группы наборов, использующих общие сервисы и инфраструктуру.

17. *Какую задачу – анализа или синтеза системы – Вы решаете? Как решаемая задача связана с целью Вашего исследования?*

Я решаю задачу синтеза системы. Цель моего диссертационного исследования - разработка и формулировка правил построения инструментальных средств конфигурирования в плагинных системах. Решая задачу синтеза системы, я раскрываю тематику построения решений в плагинных средах, а именно:

- (а) выявление элементов предметной области;



- (b) их анализ, сравнение характеристик;
  - (c) описание ограничений;
  - (d) формирование математической модели.
18. *Укажите тип математической модели (аналитическая, основанная на использовании физических законов и/или теории, имитационная, эмпирическая), которую Вы будете использовать для решения задачи анализа системы.*
- Я использую графовую модель.
19. *Охарактеризуйте кратко особенности разрабатываемой Вами модели системы.*
- Вершинами являются элементы подсистем:
- (a) требования;
  - (b) файлы исходного кода;
  - (c) плагины.
- Ребрами являются функциональные связи:
- (a) трассируемость требований к программному обеспечению на файлы исходного кода;
  - (b) распределение файлов исходного кода по плагинам;
  - (c) зависимости между файлами исходного кода;
  - (d) зависимости между плагинами.
20. *В каком состоянии находится разработка модели в настоящее время?*
- В настоящее время разработан модуль, описывающий работу двух слоев:
- (a) требований к программному обеспечению;
  - (b) файлов исходного кода.
21. *В какой среде программирования Вы реализуете модель системы?*
- Я реализую модель системы в среде программирования Java.
22. *Если в работе решается задача синтеза системы, то какой алгоритм оптимизации альтернативы системы Вы используете или предполагаете использовать?*
- В своей дальнейшей работе по реализации модели я предполагаю использовать следующие алгоритмы оптимизации:
- (a) генетический алгоритм;
  - (b) PageRank;
  - (c) случайный лес;

23. Если при синтезе системы рассматриваются несколько показателей ее эффективности, то как решается задача оптимизации системы по векторному критерию?

Ожидается, что выявленная оптимальная декомпозиция функционала по плагинам обеспечит следующие показатели эффективности:

- (a)  $F_p = 1$ ;
- (b)  $V_c = 2$ ;
- (c)  $0 \leq K_f < 1$ .

Сравнение показателей эффективности приведено в таблице 1.

Таблица 1: Сравнение показателей эффективности

	$F_p$	$V_c$	$K_f$
«1 требование - 1 плагин»	1	3	0
«все требования - 1 плагин»	0	1	$\approx 1$
предлагаемое решение	1	2	$\{x \in \mathbb{R} : 0 \leq x < 1\}$

Необходимо, чтобы  $F_p$  решения был равен 1. По этому условию «все требования - 1 плагин» не подходит. Из оставшихся вариантов выбирая между минимизацией  $V_c$  и  $K_f$  отдается предпочтение минимизации  $V_c$ .

24. Какие новые научные и/или практические результаты Вы уже получили (или предполагаете получить) в Вашем исследовании?

Разработанный модуль применяется для проведения экспериментов по разрешению циклических зависимостей между файлами исходного кода. Произведено исследование зависимости времени работы решения от используемых в программе реализаций Java-коллекций.

25. Есть ли у Вас публикации по работе? Выступали ли Вы на научных конференциях по теме Вашей диссертации? Перечислите публикации, укажите место выступления (выступлений).

На конференции в МГТУ им. Баумана я в своем докладе «Исследование правил построения конфигулятора ARINC 653 спецификации в IDE Eclipse» показал результаты применения сформулированной модели для решения задачи путем разбиения файлов исходного кода по плагинам по частотам вызова

функций. В качестве результата продемонстрировал снижение невостребованного функционала до 10% при различных запросах состава требований.

На конференции в Воронеже в ВУНЦ ВВС «ВВА имени профессора Н.Е. Жуковского и Ю.А. Гагарина» в своем докладе «Реализация инструментального средства конфигурирования компонентов БРЭО» показал актуальность применения плагинных систем для решения задач в авиационной сфере.

Подготовил выступление на конференции, организуемой Союзом Машиностроителей России. В качестве материала выступления мною предоставлено средство интеллектуального конфигурирования системы определения состояния воздушного судна. В этой работе я сформировал среду конфигурирования, которая независимо подключается к базе данных и заполняет ее конфигурационными параметрами и их значениями с целью дальнейшей их обработки непосредственно в самой системе.

Подготовлена публикация ВАК, в которой описан алгоритм разрешения циклических зависимостей графовой модели трассируемости требований к ПО на файлы исходного кода. Работа передана в редакцию ИПМ им. М.В.Келдыша РАН.