

Создание дерева требований и связь его с конструкторским составом изделия через структуру функций изделия для конструкторского бюро

Ю.В. Берчун,
ст. преп., y_berchun@mail.ru,
В.А. Бочаров,
студ.-магистр, vovan13111@gmail.com,
Д.М. Жук,
к.т.н., доц, zhuk@bmstu.ru,
В.Б. Маничев,
к.т.н., доц., vladimir.manichev@gmail.com,
МГТУ им Н.Э. Баумана, г. Москва

В докладе рассмотрено создание дерева требований и связь его с конструкторским составом изделия (КСИ) для типового конструкторского бюро через связь с функциональной структурой изделия (ФСИ), описывающая принцип действия возможностей данной продукции. Для PLM системы Teamcenter компании Siemens PLM Software приведены основные характеристики, функционал и рассмотрен основной механизм работы управления требованиями в данной системе. Рассмотрены недостатки этого механизма и варианты решения данных проблем.

The report considers the creation of a tree of requirements and its connection with the design composition of the product for a typical design bureau through the connection with the functional structure of the product, describing the principle of operation of the capabilities of this product. For PLM-system Teamcenter of Siemens PLM Software, the main characteristics, functionality and basic mechanism of work of requirements management in this system are presented. The limitations of this mechanism and the solution of these problems are considered.

Введение

Проектирование наукоёмкой продукции, на основании ГОСТ 2.103-68 [1], состоит из 5 этапов: «Техническое предложение», Аванпроект, Эскизный проект, Технический проект и Рабочий проект.

Для этапа «Техническое предложение» исходными данными являются первичные требования. Техническое предложение — совокупность конструкторских документов, которые должны содержать технико-экономические обоснования целесообразности разработки изделия на основании: 1) анализа технического задания заказчика и различных вариантов возможных конструктивных и технологических решений; 2) сравнительной оценки вариантов решений с учётом конструктивных и эксплуатационных особенностей разрабатываемого изделия и существующих аналогов и др.

На этапе аванпроекта отдел главного конструктора и службы маркетинга преобразуют требования заказчика в структуру требований для конструкторов, составляют ТЗ и создают первичную компоновку изделия, уточняя детали у заказчика.

На этапе эскизного проектирования формируется календарный план работ, выполняются инженерные расчеты для выбранной компоновки изделия, готовится электронный макет верхнего уровня (компоновочный).

На этапе технического проектирования разрабатывается детализированный электронный макет изделия, выполняются необходимые виды инженерного анализа предложенного конструкторского решения, производится оценка его технологичности, формируются списки используемых стандартных и покупных изделий. В ходе технического проектирования осуществляется валидация требований, закреплённых в техническом задании.

На этапе рабочего проектирования разработанное конструкторское решение дополняется технологической информацией (в частности, в виде аннотаций к трёхмерным моделям).

Требования появляются на первом этапе и связаны с изделием на всех этапах. Само понятие требования очень широкое, используем определение, которое чаще применяется, в соответствии с ГОСТ 2.114-95 [2], в машиностроении: требование — это утверждение, которое идентифицирует целевые эксплуатационные, функциональные параметры, характеристики или ограничения проектирования продукта или процесса, которое однозначно проверяемо и измеримо. Так же есть еще одно определение: требования — это документально изложенное ограничение, которое должно быть выполнено, если необходимо соответствие документу, и по которому не разрешены отклонения.

Достижение цели зависит от её формулировки, и первый шаг к успеху дела — правильно сформулированные задачи, и поэтому все требования должны обладать определенными характеристиками. В соответствии с теорией S.M.A.R.T. [3] все требования к изделию должны быть:

- S - Specific (Конкретный). Объясняется, что именно необходимо достигнуть. Например, «Трехступенчатый цилиндрический понижающий редуктор».
- M- Measurable (Измеримый). Объясняется, в чем будет измеряться результат. Если показатель количественный, то необходимо выявить единицы измерения, если качественный, то необходимо выявить эталон отношения. Например, «масса редуктора не должна превышать 100 кг».
- A- Attainable, Achievable (Достижимый). Объясняется за счёт чего планируется достигнуть цели. И возможно ли её достигнуть вообще? Например, «уменьшить диаметр тихоходного вала относительно предыдущей разработки».
- R- Relevant (Актуальный). Определение истинности цели. Действительно ли выполнение данной задачи позволит достичь желаемой цели? Необходимо удостовериться, что выполнение данной задачи действительно необходимо. Например, «при необходимости изменений можно использовать двухступенчатый редуктор».
- T- Time-bound (Ограниченный во времени). Определение временного триггера/промежутка по наступлению/окончанию которого должна быть достигнута цель (выполнена задача). Например, «эскизный проект должен быть полностью готов и утвержден в течение ближайшего месяца».

Актуальность задачи информационной поддержки управления требованиями заключается в том, что от качества процессов управления требованиями при проектировании технически сложной продукции в конечном счёте зависят трудоёмкость и эффективность реализации эскизного и технического проекта, техническая сложность и потенциал развития самого проекта. Для конструкторского бюро формирование дерева требований и связь его с конструкторским составом разрабатываемого изделия является первоочередной задачей после уточнения ТЗ.

Функции информационной поддержки процессов управления проектными требованиями реализуются в ведущих PDM-системах, позиционируемых как основа PLM-решения на предприятии. К числу таких систем можно отнести Teamcenter (Siemens) и SmarTeam (Dassault Systemes).

Проблемы PLM-системы Teamcenter

В PDM-системе Teamcenter функция информационной поддержки процессов для управления проектными требованиями состоит из нескольких этапов:

- Формирование структуры требований
- Построение Функциональной структуры изделия
- Логическая схема взаимодействия
- Трассировка требования
- Связка требований с параметрами
- Валидация требований

Но при использовании управление требованиями в PLM-решении, предложенный компанией Siemens, отсутствуют возможности, которые нужны представителям промышленности. Во-первых, нет типового процесса, как должен формироваться конструкторский состав изделия (КСИ) на основе требований для любой отрасли машиностроения, из-за этого возникает проблема наладки процесса подготовки производства и самого производства. Во-вторых, у представителей промышленности возникает интерес к программным инструментам поддержки проектных процедур на ранних стадиях проектирования (в том числе – с учётом решений, полученных в ранее выполненных проектах).

Способы решения проблем Teamcenter

Первая проблема решается созданием типового процесса, на основе которого формируется КСИ, и быть актуальным на всех видах производства машиностроения, с несерьёзными изменениями. На рис. 1 представлен шаблон процесса формирования конструкторского состава изделия.

Вторая проблема решается разработкой системы, которая будет искать и сравнивать данные и на основе совпадения данных создавать/восстанавливать дерево требований к новому проекту. Возникает вопрос в хранении и эффективном поиске информации. Существуют разные виды БД, но для такой задачи успешней выглядит графовая база данных. Графовая база данных — разновидность баз данных с реализацией сетевой модели в виде графа и его обобщений. Данная БД эффективна для поиска в огромных объёмах данных, но слабо эффективна, когда информации мало, так как главная проблема таких баз данных это скорость доступа к самому графу.

Следующая задача является разработка логики и интерфейса приложения. Для того, чтобы не загружать ПК пользователей, было решено использовать программирование на основе MVC, которая позволяет работать с системой клиенту в браузере. Model–View–Controller (MVC, «Модель–Представление–Контроллер», «Модель–Вид–Контроллер») — схема разделения данных приложения, пользовательского интерфейса и управляющей логики на три отдельных компонента: модель, представление и контроллер — таким образом, что модификация каждого компонента может осуществляться независимо [4]:

Основным действием в структуризации является описание контроллеров приложения. Нужно понять сколько их должно быть, и что каждый будет делать и как они будут взаимодействовать между собой.

Приложение должно работать с тремя структурами: дерево требований, ФСИ, КСИ, уметь связывать элементы этих структур между собой с помощью трассировки и отображать связи между элементами. Так как подразумевается, что проектов может быть много и в каждом хранятся свои структуры, то должно быть реализовано создание проектов и просмотр структур конкретного проекта. Получается должно быть 5 контроллеров и плюс начальный контроллер, с помощью которого отображается начальное окно, и контроллер администрирования, позволяющий удалять данные из базы данных.

Первый контроллер описывает логику работы с проектами, в которых можно создавать и смотреть детально все три структуры. При необходимости просмотра структуры требований для данного проекта, происходит перенаправление с контроллера проекта на контроллер требований, где выстроена логика поиска определенных элементов. Она состоит в сравнении номера проекта каждой строчки с номером проекта, который был открыт, и проверяем в БД значение столбца “IsDeleted”, если значение равно “False”, то элемент отобразится в окне браузера, иначе он будет не виден для пользователя. Это было сделано для того, чтобы, если пользователь случайно удалил элемент из структуры, он не будет удален из самой БД, а присвоено значение “True”, чтобы администратор в последствии восстановил элемент структуры.

В контроллере реализована возможность перемещения элементов в иерархии, если вдруг требование было создано не в той ветви дерева, пользователь может его переместить. Так же реализована возможность добавления требований, редактирования информации и детального просмотра конкретного требования.

Контроллеры, которые реализуют логику работы с ФСИ и КСИ построены на таком же принципе. Для трассировки также был создан отдельный контроллер, где реализована логика отображения элементов, у которых значение столбца “Trassirovka” равно “Да”, это означает, что у элемента есть связи. А также существует возможность просмотра связей на всех уровнях, то есть будет видно не только связь конкретного элемента с другими, но и последующие связи, то есть видны трассировки на всех уровнях. Есть возможность удаления связей, если удаляется самый нижний элемент в иерархии, то это приводит к тому, что значению столбца “Trassirovka” присваивается значение “Нет”, и он исчезает из списка элементов, у которых есть связи.

В контроллере для администратора реализована логика редактирования и удаления элементов из БД, а также авторизация пользователя для того, чтобы войти в меню администрирования, это делается с помощью отдельного контроллера. Для обычных пользователей авторизация не реализована, поскольку это излишне.

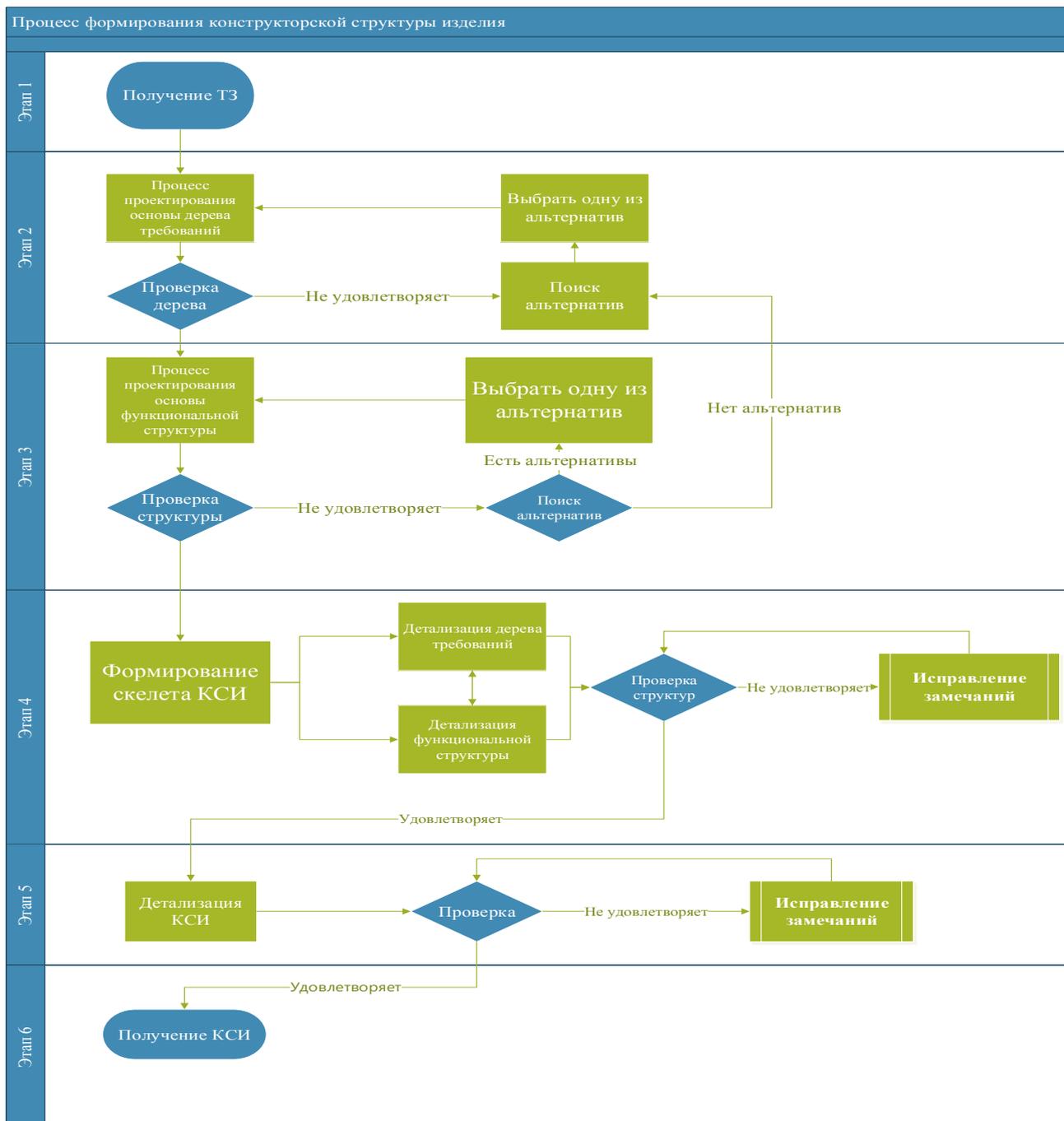


рис. 1. Представление процесса формирования конструкторской структуры изделия

Вывод

Исследуя управление требованиями в системе Teamcenter, можно сделать следующие выводы. Система позволяет отслеживать выполнения требований заказчика на всех уровнях детализации процесса проектирования, а также генерировать отчеты о выполненных проверках, чтобы проектировщики могли видеть, на сколько изделие соответствует ТЗ. Минусом является то, что нет четкого алгоритма проектирования конструкторской структуры изделия, из-за чего возникают проблемы с детализацией структуры требований и ФСИ, а также нет генерации структуры по результатам уже выполненных проектов. Для того, чтобы исправить данные недостатки, были предложены следующие варианты:

- 1) разработать систему по управлению требованиями, способную создавать структуру текущего проекта, на основе уже выполненных;
- 2) создать базовый процесс по проектированию КСИ на основе детализированных требований.

Литература

1. ГОСТ 2.103-68. Стадии проектирования. – М.: издательство стандартов, 1970. – 3 с.
2. ГОСТ 2.114-95. Технические условия. – М.: издательство стандартов, 1995. – 12 с.
3. Doran G. T. (1981). There's a S.M.A.R.T. way to write management's goals and objectives. Management Review, Volume 70, Issue 11(AMA FORUM). – P. 35-36.
4. Adam Freeman. Pro ASP.NET MVC 5 [Текст]. – Apress, 2013. – 812 с.