

Оглавление

Введение.....	1
I. Элеполю и применение универсальной системы стандартов решения изобретательских задач.....	2
II. Универсальная система стандартов на решение изобретательских задач.....	8
III. Текст и алгоритм применения универсальной системы стандартов на решение изобретательских задач (АИСТ-2010).....	13
1. Синтез элеполю.....	15
2. Развитие элеполюных структур.....	16
3. Синтез и повышение эффективности систем на измерение и обнаружение (систем со свойствами полей взаимодействия).	17
4. Линии развития систем.	18
ПРИЛОЖЕНИЯ к универсальной системе стандартов на решение изобретательских задач.	25
П1. Краткое введение в элеполюный анализ.....	25
П2. Алгоритм использования стандартов (АИСТ-2010).....	26
П3. Перечень типовых полей и веществ, применяемых в технических системах.....	28
П4. Алгоритм использования стандартов решение изобретательских задач и паттернов программирования (АИСТ-2010-П).	29
VI. Литература.....	32

Введение.

Имеются две основные причины необходимости перехода от веполю к элеполюм:
- отсутствие в нематериальных системах веществ
- некорректность принятых ранее правил построения веполюных моделей, в частности, в стандартах на решение изобретательских задач.

Исправление некорректности в веполюных моделях было предпринято в нескольких работах М.С.Рубина [9, 15]. Понятие элеполю было введено в работах М.С.Рубина и И.О.Одинцова при исследовании развития социально-культурных систем [13] и при использовании методов ТРИЗ в программировании и в информационных системах [14]. В работе О.И. Зиненко [11] была показана возможность применения универсальной системы стандартов для решения изобретательских задач в области программирования.

В данной работе сделана попытка изложить разрозненный по разным статьям материал об элеполюх и универсальной системе стандартов в относительно полном виде. Предлагаемый материал был успешно опробован на семинарах по ТРИЗ в 2009-2011 годах для программистов, менеджеров, специалистов в области маркетинга, информационных технологий.

I. Элеполи и применение универсальной системы стандартов решения изобретательских задач.

В ТРИЗ в качестве элементарной модели технической системы рассматривается веполь: объединение двух веществ при помощи полей взаимодействия. При этом в качестве поля взаимодействия рассматриваются обычные физические поля (магнитные, электрические, гравитационные) и взаимодействия, которые в физике не принято называть полями, например, запаховые, механические, акустические, химические, биохимические и т.д.

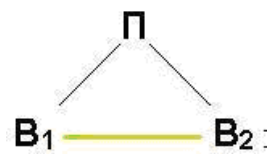


Рисунок 1. Веполь.

С переходом ТРИЗ к нетехническим областям возникла необходимость в формулировании элементарной модели взаимодействия для нематериальных систем. Были внесены необходимые изменения в формулировки и правила построения моделей, которые получили новое название: элеполь (элементы, связанные полями взаимодействия).

Элеполь имеет два вида связей между элементами. Непосредственная связь Э1–Э2 – это реализация той или иной необходимой функции или требования (что делается, какая функция или какое требование выполняется). Связь через поле взаимодействия Э1–П–Э2 – это то, с помощью чего удастся обеспечить необходимое действие или требование, как нужная функция или требование реализуются. Таким образом, элеполь должен отвечать на вопрос что делается и как делается.

В некоторых случаях (измерение, преобразование полей, развитие социокультурных система и т.д.) в основе элеполя находятся поля, а преобразование поля происходит элементом «Э». Такие структуры мы будем называть внешним элеполем.

Примерами внутренних элеполей (основанных на элементах со связями, направленными вовнутрь этой системы) могут быть сеть компьютеров или арифметический оператор над двумя и более переменными. Примером открытых элеполей (основанных на полях, направленных во вне системы) может быть дешифратор данных, преобразующий один поток информации в другой.

Одно и то же действие может быть реализовано при помощи разного набора полей. Например, поднять шарик из лунки в земле можно с помощью:

- воды и гравитации, если шарик пластмассовый
- магнита, если шарик металлический,
- обыкновенного пластилина, если шарик оказался из меди.

Модель реализации нужной функции останется во всех вариантах одной и той же: введение поля для достройки неэлеполевой структуры до полного элеполя.

<p>Описание моделей проблемных ситуаций: нет связей, есть недостаточная или вредная связь.</p>	<p>Элеполь: внутренний (с двумя элементами) и внешний (с двумя полями).</p>	<p>Повышение эффективности элеполей: комплексный, двойной, цепной элеполи.</p>

Рисунок 1. Элеполи: модели задач, модели решения и модели их развития.

При помощи элеполевых моделей могут быть обозначены проблемные ситуации (задачи) и модель их решения.

Можно привести примеры элепольных структур в программировании. Например, выделение в программе повторяющейся последовательности команд в цикл. Получаем следующую модель:

Э1 – часть программы с неповторяющейся последовательность команд

Э2 – часть программы с повторяющейся последовательность команд

П – поле взаимодействия, позволяющее объединить обе части программы в единую систему.

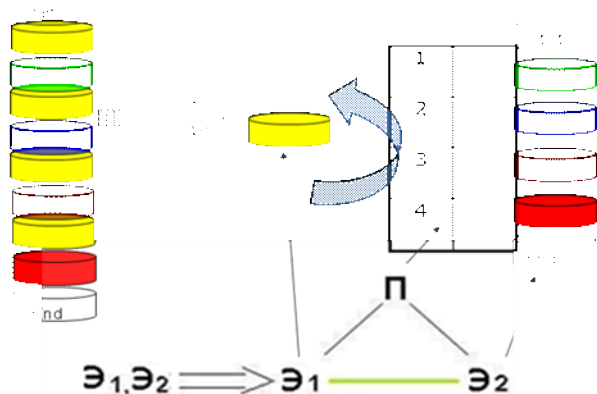


Рисунок 2. Цикл в программе как реализация элепольной структуры.

Придание программе вида с элепольной структурой повышает ее управляемость, снижает затраты на ее написание, отнимает меньше ресурсов, делает программу более надежной.

Примером элеполя может служить модель "сущность-связь", где элементами являются сущности анализируемой предметной области, а полем - связь между сущностями. Еще один пример - диаграмма классов языка UML, где элементами являются классы, а полями - взаимосвязи между ними.

Как и в классической ТРИЗ, так и в адаптации к области программирования существуют, как минимум, четыре сложности при моделировании с помощью элеполей:

- 1) не существуют системы, которые являются только полем или только веществом (элементом) – они всегда взаимосвязаны;
- 2) не всегда понятно, на каком уровне детализации (надсистемы-подсистемы) необходимо создавать веполю (элеполю), в каком аспекте рассматривать модель системы;
- 3) в конкретном взаимодействии бывает не одно, а сразу два и больше взаимодействий. Какое из взаимодействий нужно выбрать в каждом конкретном случае?

Рекомендуется каждый раз эти неопределенности решать, исходя из конкретных задач моделирования ситуации.

Примеры источников полей в программировании:

- механизм взаимодействия между элементами (обменом "сообщениями"). Например, запрос от клиента на выполнение действия объектом (вызов метода объекта);
- отношения между элементами. Например, наследование, ассоциация, агрегация классов.

Можно ввести некоторые общие правила для построения элепольных структур:

- элементы во внутреннем элеполе могут взаимодействовать только через поля взаимодействия;
- в элеполе поле действует на оба элемента, входящих во внутренний элеполю;
- во внешнем элеполе поле преобразуется в другое поле при взаимодействии с элементом (измерительные системы, социокультурные поля взаимодействия, преобразования потоков информации);
- элементно-полевые структуры, например, поток жидкости, информационный поток, художественное произведение и т.д. носят двойственный характер и рассматриваются либо в качестве вещества (элемента), либо в качестве поля взаимодействия в зависимости от рассматриваемой ситуации и решаемой задачи.

Приведем несколько примеров элепольных структур из различных областей деятельности человека.

Техника.

Судно на воздушной подушке (СВП). Элемент 1 – аппарат СВП, элемент 2 – опорная поверхность (водная, земная поверхность, снег, лед). Выполняемая функция – удерживать



над опорной поверхностью. Поле взаимодействия – воздушный поток. Если в этом элеполе в качестве поля взаимодействия использовать магнитное поле, то получится транспорт на магнитной подушке. Если поле взаимодействия – электростатика, то получим, например, машинку, способную передвигаться по вертикальным стенам (деревянными, кирпичными, бетонными, стеклянными, металлическим)¹.

Биология.

В биологии можно выделить два самостоятельных аспекта рассмотрения системы: материальный и нематериальный. Приведем по одному примеру на каждый из аспектов.

Материальный аспект. «Кузница» дятла². Элемент 1 – щель в дереве, элемент 2 – шишка еловая или сосновая. Выполняемая функция – удерживать шишку. Поле взаимодействия – механическое поле.

Нематериальный аспект.

Танец пчелы³. Элемент 1 – деталь танца медоносной пчелы-разведчицы, элемент 2 – пчелы в улье. Выполняемая функция – передача информации о месте расположения цветов-медоносов. Поле взаимодействия – визуальное и акустическое поля.



Искусство.

В искусстве тоже можно выделить материальный и нематериальный аспект.

Материальный аспект.

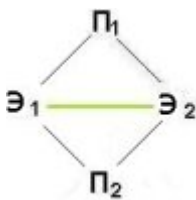
Подготовка холста для масляной живописи⁴. Элемент 1 – холст, элемент 2 – масляные краски. Выполняемая функция – создание поверхности, передающей и сохраняющей цветовую гамму масляных красок на холсте. Поле взаимодействия – механическое поле грунтовки на основе того или иного клея с добавкой белил и других компонент.

Нематериальный аспект.

А.П.Чехов «Жених». Без цитаты обойтись трудно: «- Прощай, моя прелесть! - говорил молодой человек, целуя девицу в белокурую головку. - Прощай! Я так несчастлив! Ты оставляешь меня на целую неделю! Для любящего сердца ведь это целая вечность! Про... щай... Утри свои слезки... Не плачь...»

Петя замахал платком. Но вдруг лицо его вытянулось... Он ударил себя по лбу и как сумасшедший вбежал в вагон.

- Варя! - сказал он, задыхаясь. - Я дал тебе для Мракова двадцать пять рублей... Голубчик... Расписочку дай! Скорей! Расписочку, милая! И как это я забыл?»



Элеполюную модель этого описания может представить в таком виде. Элемент 1 – жених 2 – Варя. Выполняемая функция – показать неискренность отношения жениха. Поле взаимодействия 1 – романтические взаимоотношения. Поле взаимодействия 2 – прагматичное отношение жениха к Варе.

Подобная же модель использована в рассказе А.П.Чехова «Злой мальчик»: «- Я рад, что мы наконец одни, - начал Лапкин, оглядываясь. - Я должен сказать вам многое, Анна Семеновна... Очень многое... Когда я увидел вас в первый раз... У вас клюет... Я понял

¹ <http://www.membrana.ru/particle/12571>

² <http://www.sivatherium.narod.ru/postcard/sled/sled.htm>

³ <http://ethology.ru/library/?id=360>

⁴ <http://zaplutus.ru>

тогда, для чего я живу, понял, где мой кумир, которому я должен посвятить свою честную, трудовую жизнь... Это, должно быть, большая клюет... Увидя вас, я полюбил впервые, полюбил страстно! Подождите дергать... пусть лучше клонет... Скажите мне, моя дорогая, заклинаю вас, могу ли я рассчитывать - не на взаимность, нет! - этого я не стою, я не смею даже помыслить об этом, - могу ли я рассчитывать на... Тащите!»

Психология личности и малых групп.

Личность человека. Элемент 1 – воспоминания личности о себе в прошлом (включая фотографии вещи и т.д.). Элемент 1' – представление личности о себе в настоящем. Функция – восприятие себя как цельной личности. Поля взаимодействия: сравнение ощущений, фактов, воспоминаний в различной форме.

Семья. Элемент 1 – мать, элемент 2 – отец, элемент 3 – дочь, элемент 4 – старший сын, элемент 5 – младший сын, элемент 6 – бабушка, элемент 7 – дедушка. Функции: воспроизводство жизни, жизнеобеспечение, ведение домашнего хозяйства, социализация. Поля взаимодействия: психологические, социальные, экономические, физические, физиологические.

Коллектив производственной компании. Элементы – члены коллектива, элементы производственной среды. Функции: производственная деятельность, управление. Поля взаимодействия: психологические, социальные, технические, физические.

Социокультурные системы.

Зарождение родовых отношений. Амулеты у эскимосов как символ отсутствующего (или умершего человека), защиты рода, плодовитости⁵. Элемент 1 – дочь. Элемент 2 – кукла, амулет, олицетворяющий мать. Функции – защита рода, обеспечение плодовитости. Поле взаимодействия – социально-психологическое поле, перешедшее от взаимоотношений дочь-мать к отношениям дочь-амулет, семья-амулет.

Цивилизация.

Формирование технической цивилизации⁶. Элемент 1 – естественная (природная) среда обитания. Элемент 2 – искусственная среда обитания. Выполняемая функция – формирование удобной, благоприятной для человечества среды обитания. Поля взаимодействия – культура в широком смысле этого слова (технологии, наука, искусство, политика, религии, язык общения и т.д.).

Программирование.

Материальный аспект.

Регулирование равномерности распределения температуры на плате многоядерного процессора компьютера за счет перераспределения нагрузки на разные ядра процессора⁷. Элемент 1 – ядра процессора. Элемент 2 – плата процессора. Функции – обеспечить равномерное распределение температуры на плате. Поля взаимодействия – программа регулирования загрузки ядер процессора, тепловое поле.

Нематериальный аспект.

Создание связанных таблиц в базах данных при помощи объединения ключевых полей этих таблиц. Например, в программе планирования встреч и собственных дел⁸ можно добавить информацию о погоде, расписании телепрограмм, афишу, дни рождения друзей и родственников и т.д. Элемент 1 – расписание собственных дел. Элемент 2 – данные о прогнозе погоды, расписании различных событий и т.д. Функция – отобразить сопутствующую информацию, полезную для планирования дел. Поле взаимодействия – установление связи с необходимой информацией через полу «дата».

Аналогично устанавливаются связи, например, в социальных сетях через поля «школа и дата ее окончания», «университет и дата его окончания», «друг» и т.д.

⁵ <http://www.temm.ru/ru/section.php?docId=4472>

⁶ <http://temm.ru/ru/section.php?docId=3470>

⁷ Пример из картотеки И.О.Одинцова

⁸ <http://calendar.yandex.ru/week>

Элепольное моделирование может использоваться для создания моделей задач и моделей их решения.

Задача 1⁹. Проблема разных интерфейсов связи приложения-монитора с Access Point: одни, например, допускают передачу бинарных данных, другие – только текстовых.

В программе-мониторе можно описывать все возможные варианты интерфейса, но это потребует много описания всех возможных вариантов интерфейса во всех подразделах программы.

Необходимо, чтобы приложение-монитор могло при минимальных средствах принимать и передавать разные типы данных. Необходимо, чтобы приложение могло работать по любому интерфейсу, и было устойчивым к нововведениям.

Модель этой ситуации можно описать следующим образом:

Э1 – Access Point,

Э2 – Приложение-монитор.

П – поле взаимодействия прием-передачи информации.

В нашем случае имеется одновременно и положительная и негативная связь между Э1 и Э2. Польза в том, что передаваемая информация нужна, а негатив в том, что типов данных и разных версий интерфейсов может быть неконтролируемо много. В такой модели конфликта в ТРИЗ существует модель стандартного решения.

Необходимо ввести какой-то 3-й элемент Э3 либо в первый, либо во второй элемент, который позволил бы ликвидировать нежелательное явление. Возможное решение: вводится промежуточный элемент (блок), который предварительно сохраняет данные в виде, независимом от интерфейса и только потом преобразуется к виду, соответствующему нужному интерфейсу. Нет необходимости повторять одни и те же команды несколько раз в разных подпрограммах и нет необходимости вводить изменения во все подпрограммы при появлении нового интерфейса – он вводится только один раз в программу-адаптер.

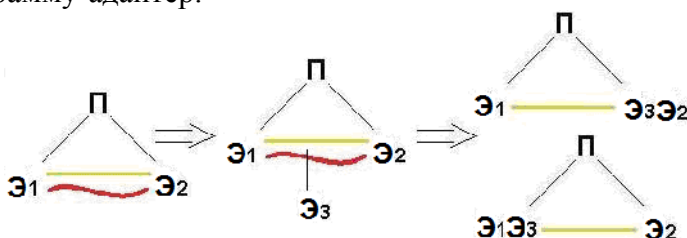


Рисунок 3. К задаче 1. Модель задачи и модель решения: комплексный элеполь.

В нематериальных системах, впрочем как и в материальных, далеко не всегда полевые взаимодействия можно отделить от вещественных (поле взаимодействия от элемента, связанного с этим взаимодействием). Например, не бывает магнитного поля без самого магнита, гравитационное поле действует только через массу тел, а звук и вовсе может распространяться только через то или иное вещество. То же происходит и в нематериальном взаимодействии. Например, социальные или психологические взаимодействия основаны на людях и предметах, которые участвуют в создании этих взаимодействий. Художественные системы всегда имеют материальный носитель и материальный «потребитель» – людей, зрителей.

Аналогичная ситуация и в информационных технологиях. При решении задачи 1, например, мы представили программу-адаптер в форме элемента Э3. При более детальном рассмотрении можно выяснить, что и сам Э3 (программа-адаптер) можно представить в форме другого элеполя со своими подсистемными элементами и полями взаимодействия.

⁹ Текст задачи предложил А.Кирдин в 2009 г..

Если попытаться формализовать записи элеполя, то можно дать следующие описания составляющих элеполя.

Элемент можно представить в форме набора параметров и связей между этими параметрами. Элемент всегда находится в определенном месте некоторого пространства.

Поля взаимодействия – это зависимости параметров одного элемента от параметров другого элемента, находящегося в другом месте того же пространства (не обязательно физического).

Приведем несколько примеров.



Рисунок 4. Элеполе: полировка зеркал.



Рисунок 5. Элеполе: сравнение площадей треугольника.



Рисунок 6. Элеполю: создание очереди на покраску забора.

Поля взаимодействия приводят к возникновению системного эффекта: возникают новые взаимодействия, которых не было до проявления поля взаимодействия между элементами и не сводятся к этому взаимодействию.

Необходимо выделить два важных понятия для элеполей – это пространство и время. Два или несколько элементов не могут находиться в одно и то же время в одном и том же месте пространства. Иначе эти элементы были бы просто не различимы. Для нематериальных систем пространство может быть нематериальное. Одна из характеристик поля взаимодействия – способность преодолевать то пространство, которое разделяет элементы.

II. Универсальная система стандартов на решение изобретательских задач.

Идея стандартов на решение изобретательских задач разрабатывалась в ТРИЗ с 1975 по 1985 годы. От отдельных стандартов был сделан переход к наиболее распространенной сейчас системе из 76 стандартов (Стандарты-76), объединенных в 5 классов:

Класс 1 Построение и разрушение веполей

- Построение или разрушение вепольной модели, если она неполная или выполняет вредную функцию.
- Имеет 2 подкласса, содержащие 13 Стандартов.

Класс 2 Развитие вепольных систем

- Внесение изменений в рамках существующей системы.
- Имеет 4 подкласса, содержащие 23 Стандарта.

Класс 3 Переход к надсистеме и на микроуровень

- Разработка решений на уровне надсистемы или подсистемы.
- Имеет 2 подкласса, содержащих 6 Стандартов.

Класс 4 Стандарты на обнаружение и измерение систем

- Решение проблем обнаружения и измерения в пределах существующей технической системы.
- Имеет 5 подклассов с 17 Стандартами.

Класс 5 Стандарты на применение стандартов

- Разработка решений, удовлетворяющих требованиям к идеальной системе.
- Имеет 5 подклассов, содержащих 17 Стандартов.

Система стандартов – 76 предназначена для решения изобретательских задач в технике. Ю.Мурашковским были разработаны стандарты для решения изобретательских задач в искусстве. Известна также система стандартов для решения изобретательских

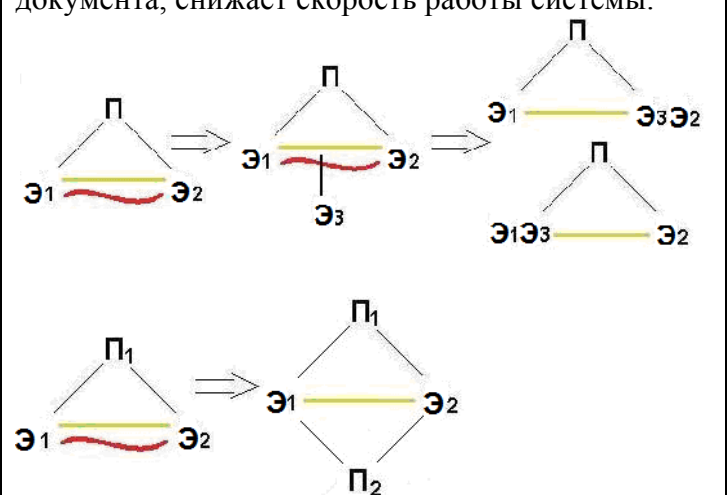
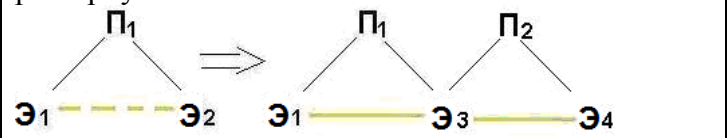
задач при развитии коллективов (Модели для творца Б. Л. Злотин, А. В. Зусман). На основе этих и других исследований была разработана универсальная система стандартов на решение изобретательских задач в различных областях деятельности человека, в том числе в области развития программного обеспечения. Текст универсальной системы стандартов и алгоритм использования стандартов (АИСТ-2010) приведены в приложении к тексту универсальной системы стандартов.

Для совместного использования универсальной системы стандартов и паттернов программирования был разработан алгоритм АИСТ-2010-П, также приведенный в приложении П5. Упрощенный вариант алгоритма АИСТ-2010-П приведен ниже на рисунке. В соответствии с этим алгоритмом от модели задачи необходимо перейти к модели решения при помощи рекомендуемого стандарта. Для повышения эффективности предлагаемых идей решения рекомендуется воспользоваться линиями развития систем (в том числе линиями и тенденции развития программного обеспечения). Для уточнения идеи решения можно перейти к рекомендациям по использованию соответствующих паттернов программирования.

В упрощенном виде алгоритм АИСТ-2010 приведен на сайте <http://temm.ru/ru/section.php?docId=4423>. С описанием паттернов программирования можно познакомиться на сайте <http://www.temm.ru/ru/section.php?docId=4515> или <http://codelab.ru/cat/patterns/>

Задача 2. Рассмотрим редактор документов, который допускает встраивание в документ графических объектов. Затраты на создание некоторых таких объектов, например больших растровых изображений, могут быть весьма значительны. Но документ должен открываться быстро, поэтому следует избегать создания всех «тяжелых» объектов. Как это сделать? Используйте систему стандартов на решение изобретательских задач.

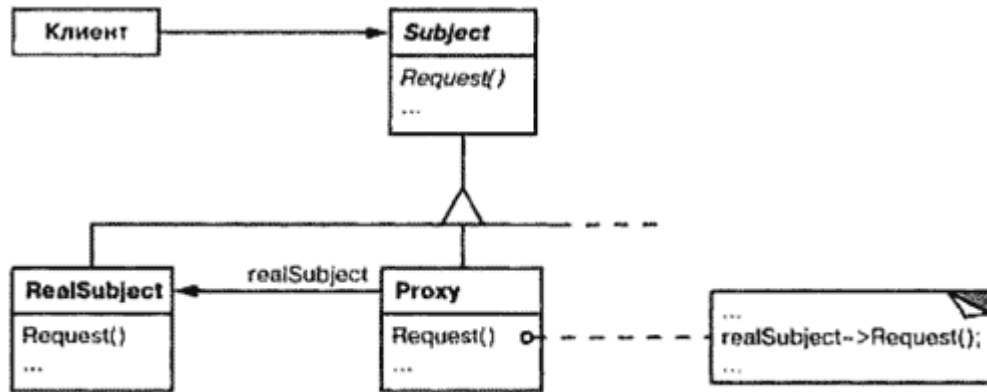
<p>1. Формулировка задачи. Выделение конфликтующих элементов, ключевых параметров. Описание модели задачи в элеполевой форме.</p>	<p>Редактор, документ, графический объект. Параметризация объектов. Для каждого объекта выделяются параметры в зависимости от аспекта рассмотрения ситуации: Редактор: размер кода. Документ: количество встроенных графических объектов. Графический объект: ширина, высота изображения, объем памяти, занимаемый изображением. Создание элеполевой структуры. Э1 - редактор, Э2 - графический объект Поле П - запросы к графическому объекту.</p>
<p>2. Если в задаче на изменение элеполь не полный, то перейти к стандарту 1.1 на создание элеполевых структур.</p>	<p>Элеполь полный.</p>
<p>3. Если в задаче на изменение имеются вредные связи, то использовать стандарты группы 1.2 на устранение вредных связей.</p>	<p>Между элементами возникает одновременно полезное и вредное взаимодействие, переход к стандартам на устранение вредных связей 1.2. Полезное взаимодействие обеспечивает возможность отображения и работы с</p>

	<p>изображением, вредное - создание изображения, пока оно не появляется в видимой части документа, снижает скорость работы системы.</p>  <p>Устранение вредных связей дополнением элементов</p> <p>Э2 и Э3 образуют независимый элеполь, с которым у Э1 полезное взаимодействие сохраняется, а вредное нет:</p> <p>Вводится новый объект Э3, который ведет себя так же и отображает настоящее изображение, на которое хранит ссылку, только в необходимых случаях. Э3 хранит размер изображения, может отвечать на запросы о своем размере, не создавая его.</p>
<p>4. Если в задаче на изменение имеются не эффективные связи, то рекомендуется группа стандартов 2 на развитие элепольных структур.</p>	<p style="text-align: center;">—</p>
<p>5. К стандартам части 2 рекомендуется обратиться в любом случае после рекомендаций стандартов 1-й части.</p>	<p>По стандарту 2.2. один из элементов может быть развернут в самостоятельный элеполь:</p> 
<p>6. Если задача на измерение или обнаружение, то перейти к разделу 3 текста стандартов.</p>	<p style="text-align: center;">—</p>
<p>7. Рекомендуется в любом случае рассмотреть предложения, описанные в линиях развития систем.</p>	
<p>8. Если задача сформулирована в связи с развитием программного продукта, то следует уточнить</p>	<p>Из перечисленных в 8.1 разделе АИСТ-2010-П (нужно устранить вредную связь) подходит к нашей задаче дополнительное условие: Нужно создавать элемент (получать к нему доступ)</p>

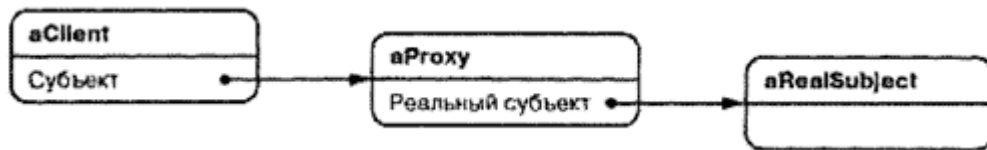
<p>предлагаемое решение в форме рекомендуемых паттернов программирования.</p> <p><i>8.1. Если нужно устранить вредную связь: ... (см. описание АИСТ-2010-П)</i></p>	<p>только в конкретных случаях. Соответственно получаем рекомендацию использовать паттерн «Заместитель».</p>
<p><i>8.2. Если нужно повысить эффективность: ... (см. описание АИСТ-2010-П)</i></p>	<p>—</p>
<p><i>8.3. Если удалось выбрать подходящий вариант, переход к описанию подходящего паттерна.</i></p>	<p>Нам удалось выбрать подходящий вариант паттерна, необходимо перейти к описанию этого паттерна, например на сайте http://www.temm.ru/ru/section.php?docId=4515 : «Заместитель» - паттерн, контролирующий доступ к элементам, предоставляя более оптимальное их взаимодействие.</p> <p>Разумно управлять доступом к элементу, поскольку тогда можно отложить расходы на его создание до момента, когда элемент действительно понадобится. Таким образом, выявляются элементы, функционирование которых проходит не совсем оптимально, и вводятся объекты-заместители, которые, дублируя внешний вид и поведение «проблемных» элементов, переадресуют им запросы лишь тогда, когда это действительно необходимо, либо после некоторых оптимизационных действий.</p> <p>Признаки применения:</p> <ul style="list-style-type: none"> · Когда требуется удаленный функционал Удаленный заместитель предоставляет локального представителя вместо целевого объекта, находящегося в другом (адресном) пространстве. · Когда нужен виртуальный заместитель Виртуальный заместитель создает «тяжелые» элементы по требованию. · Когда нужно контролировать доступ к исходному элементу Защищающий заместитель контролирует доступ к исходному элементу. <p>Более детальное описание этого паттерна и примеры его применения можно найти по адресу http://codelab.ru/pattern/proxy/</p>
<p><i>8.4. Если паттерн по описанию все же не соответствует желаемому результату, переход к группе родственных паттернов.</i></p> <p><i>8.5. Если в группе не удалось подобрать подходящий вариант,</i></p>	<p>Родственные паттерны (по ссылке http://codelab.ru/pattern/proxy/): Паттерн адаптер предоставляет другой интерфейс к адаптируемому объекту. Напротив, заместитель в точности повторяет интерфейс своего субъекта. Однако, если заместитель используется для ограничения доступа, он может отказаться</p>

<p><i>возможно задача не должна решаться с помощью паттернов.</i></p> <p><i>8.6. Для подходящего паттерна предоставляются рекомендации по использованию совместно с другими паттернами.</i></p>	<p>выполнять операцию, которую субъект выполнил бы, поэтому на самом деле интерфейс заместителя может быть и подмножеством интерфейса субъекта.</p>
<p>9. Если решение не найдено, то рекомендуется перейти к АРИЗ-Универсал-2010.</p>	<p>—</p>

Описание решения по паттерну «Заместитель»:



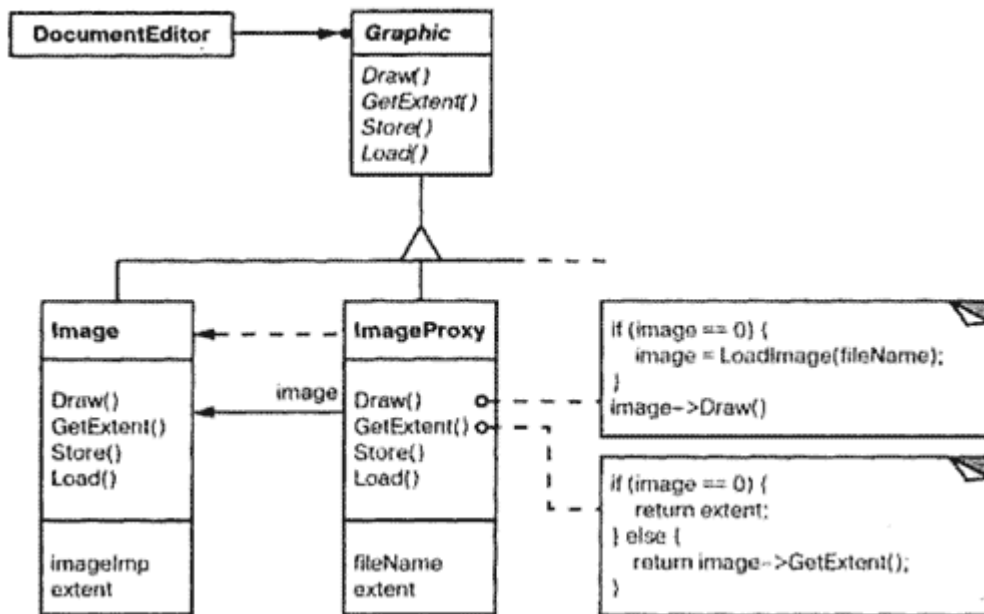
Вот как может выглядеть диаграмма объектов для структуры с заместителем во время выполнения:



«Заместитель» создает настоящее изображение, только если редактор документа вызовет операцию Draw у этой картинки. Все последующие запросы «заместитель» переадресует непосредственно изображению. Поэтому после создания изображения он должен сохранить ссылку на него.

Предположим, что изображения хранятся в отдельных файлах. В таком случае мы можем использовать имя файла как ссылку на реальный объект. Заместитель хранит также размер изображения, то есть длину и ширину. «Зная» ее, «заместитель» может отвечать на запросы формatera о своем размере, не инстанцируя изображение.

На следующей диаграмме классов этот пример показан более подробно (codelab.ru):



Алгоритм АИСТ-2010 позволяет искать стандартные решения изобретательских задач не только для программирования, но и для задач другого типа: технических, из области бизнеса, маркетинга и т.д.

III. Текст и алгоритм применения универсальной системы стандартов на решение изобретательских задач (АИСТ-2010).

Универсальная система стандартов на решение изобретательских задач.

©М.С. Рубин,
Вариант от 21.12.2010

В теории решения изобретательских задач (ТРИЗ) создана система моделей, подсказывающая идею решения изобретательских задач. Наибольшее распространение получила система стандартов на решение изобретательских задач, созданная Г.С.Альтшуллером в 1985 году (Стандарты-76) [1]. Она предназначена для решения только технических задач.

Предлагаемая универсальная система стандартов предназначена для применения не только в технике, но и в других областях знаний, в том числе, при решении изобретательских задач при создании программного обеспечения. При создании универсальной системы стандартов использовались в первую очередь работы [1, 2, 3, 9, 11, 12].

В основе языка описания стандартов на решение изобретательских задач лежит понятие элеполю: структура, состоящая из элементов и полей их взаимодействия (см. приложение 1). Для использования системы стандартов исходную задачу необходимо представить в виде модели и попытаться свести ее к одной из моделей задач в предлагаемой системе стандартов.

Система стандартов может применяться для решения изобретательских задач и для прогнозирования. Для решения изобретательских задач необходимо задачу (конфликтную ситуацию) представить в виде элеполюной модели. Лучше использовать

для этого АРИЗ-Универсал-2010. Зная модель задачи можно найти соответствующие рекомендации в системе стандартов или воспользоваться алгоритмом использования стандартов (АИСТ-2010 в приложении 2).

Для прогнозирования можно использовать два подхода. Первый – прогнозируемая система рассматривается как некая структура из элементов и полей взаимодействия без учета выполняемых функций. Стандарты подсказывают возможные изменения в этой структуре. Для новой структуры придумываются новые функции и возможные противоречия.

Второй подход прогнозирования на основе системы стандартов основан на построении модели принципа действия системы, выявлении внутренних противоречий системы и основных направлений развития. В этом случае технология прогнозирования в основном опирается на технологию выявления и решения изобретательских задач, но на уровне филогенеза.

Содержание системы стандартов.

1. Синтез элеполей

1.1. Создание элеполевой структуры (новой системы)

1.2. Устранение вредных связей в элеполе

1.2.1 Устранение вредных связей дополнением элементов

1.2.2 Устранение вредных связей дополнением полей

2. Развитие элеполевых структур

2.1. Переход к комплексному элеполю.

2.2. Построение цепного элеполя

2.3. Построение двойного элеполя

3. Синтез и повышение эффективности измерительных систем.

3.1. Обходные пути.

3.2. Синтез систем на измерение и обнаружение

4. Линии развития.

1. Переход в надсистему и к подсистемам (на микроуровень)

2. Линии коллективно-индивидуального использования систем

3. Линия введения элементов (веществ)

4. Линия введения и развития полей взаимодействия

5. Линия дробления и динамизации

6. Линии согласования-рассогласования и структуризации

7. Линия развития систем в соответствии с S-образными кривыми

8. Линии и тенденции развития программного обеспечения.

Приложения

П1. Краткое введение в элеполевый анализ.

П2. Алгоритм применения стандартов (АИСТ-2010)

П3. Перечень типовых полей и веществ применяемых в технических системах

П4. Алгоритм использования стандартов решение изобретательских задач и паттернов программирования (АИСТ-2010-П).

1. Синтез элеполей

1.1. Создание элепольной структуры (новой системы)

Если дан объект, плохо поддающийся нужным изменениям, и условия не содержат ограничений на введение элементов и полей, задачу решают синтезом элеполя, вводя недостающие элементы.



Рекомендации по развитию системы:

Рекомендуется применить линии введения элементов и полей.

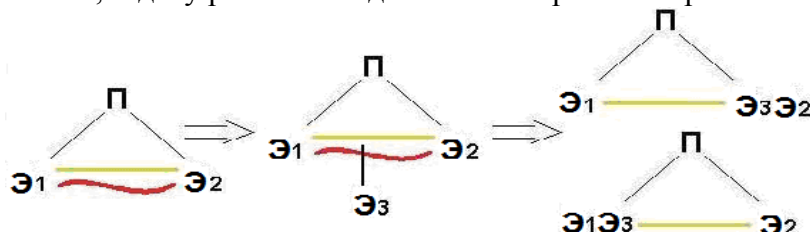
Примеры:

- 1) Для выбора прослушиваемой музыки предлагается ее ритм делать соответствующим ритму физиологических параметров слушателя (например, пульс). US Patent Application US 20060107822A1: Music synchronization arrangement.
- 2) Предложен метод построения композитных иконок, чувствительных к контексту. Используется базовая иконка и дополнительные элементы, отражающие контекст ее применения. US Patent 7 231 611 Apparatus and method for dynamically building a context sensitive composite icon.
- 3) Чтобы достать из узкой щели железный предмет можно использовать магнит, если предмет деревянный – можно использовать пластилин или пылесос.

1.2. Устранение вредных связей в элеполе

1.2.1 Устранение вредных связей дополнением элементов

Если между двумя элементами в элеполе возникают сопряженные - полезное и вредное - действия, задачу решают введением постороннего третьего элемента



Элемент Э3 либо нейтрализует, либо оттягивает на себя плохое взаимодействие.

Элемент Э3 в элеполь можно вводить различными способами:

- в виде добавки к Э1 или Э2;
- использовать в качестве Э3 видоизменения Э1 и/или Э2;

Рекомендации по развитию системы:

Использовать линию введения элементов.

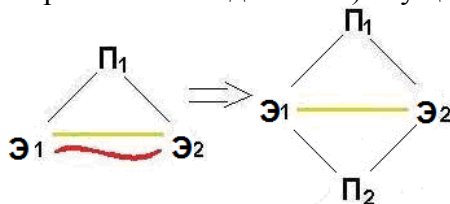
Примеры:

- 1) Художники-импрессионисты первыми перешли к чистым тонам, накладывая мазки один возле другого. Однако при такой технике мазки разных цветов контрастировали друг с другом. Чтобы этого не было, между этими мазками накладывались полутона [2].
- 2) Способ гибки ошипованной трубы намоткой ее в холодном состоянии на гибочный шаблон отличается тем, что с целью повышения качества при гибке трубы на радиус менее трех наружных диаметров трубы при намотке трубы ее шипы погружают в слой эластичного материала, например полиуретана. Авторское свидетельство № 724242.[1]

1.2.2 Устранение вредных связей дополнением полей

Если между двумя элементами в элеполе возникают сопряженные - полезное и вредное - действия, задачу решают переходом к двойному элеполю, в котором полезное

действие остается за полем П₁, а нейтрализацию вредного действия (или превращение вредного действия во второе полезное действие) осуществляет П₂.



Рекомендации по развитию системы:

Рекомендуется применять линию введения полей.

Примеры:

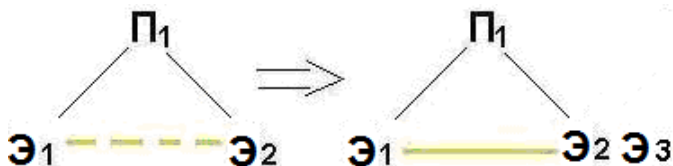
1) В рассказе Джека Лондона «На Сороковой Миле» два золотоискателя по ничтожному поводу собирались драться на дуэли. Друзья не в состоянии помешать им (кодекс чести). Принимается решение — победившего повесят. Естественно, дуэль не состоялась [2].

2) Для опыления цветов обдувают воздухом. Но цветок от ветра закрывается. Предложено раскрывать цветок воздействием электрического заряда. Авторское свидетельство № 755247.[1]

2. Развитие элепольных структур

2.1. Переход к комплексному элеполю

Если дан элеполю, плохо поддающийся нужным изменениям, и условия задачи не содержат ограничений на введение добавок в имеющиеся элементы, задачу решают переходом (постоянным или временным) к внутреннему комплексному элеполю, вводя в Э₁ или Э₂ добавки, увеличивающие управляемость или придающие элеполю нужные свойства:



Рекомендации по развитию системы:

Использовать линию введения элементов.

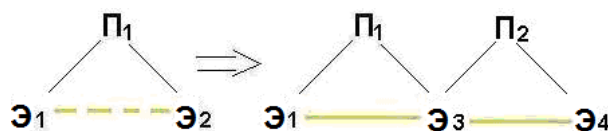
Примеры:

1) Во времена сухого закона в США бутлегеры ввозили спиртное по морю. Часто при появлении таможни ящики с виски приходилось выбрасывать в море. К этим ящикам добавляли мешки с солью. Когда представители таможни уезжали, то соль к этому времени в воде растворялась и ящики всплывали на поверхность.

2) При обнаружении неплотностей в агрегате холодильника имеется ограничение: люминофор нельзя вводить в жидкость. В этом случае вещество-обнаружитель может быть расположено на наружной поверхности агрегата (Авторское свидетельство № 311109). Возникает внешний комплексный элеполю.[1]

2.2. Построение цепного элеполя

Если нужно повысить эффективность элепольной системы, задачу решают превращением одной из частей элеполя в независимо управляемый элеполю и образованием цепного элеполя:



(Э3 или Э4 в свою очередь может быть развернут в самостоятельный элеполю).

Рекомендации по развитию системы:

Использовать линию введения элементов.

Рекомендуется применить линию введения полей

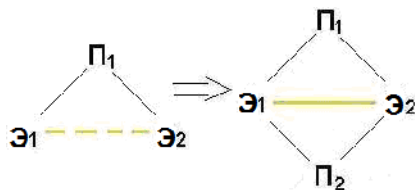
Примеры:

1) Герой произведения связывается с явлением или частью природы. Чтобы показать состояние героя показывается состояние природы.

2) Организация меню для редактора Word. Есть основное меню, которое имеет много разделов, занимает много места на экране и имеет возможность «прятаться», чтобы не мешать пользователю. Для повышения эффективности работы с меню пользователь имеет возможность самостоятельно дублировать основные иконки меню в дополнительную строчку меню, которая занимает мало места и содержит самые необходимые для работы команды меню.

2.3. Построение двойного элеполя

Если дан плохо управляемый элеполю и нужно повысить его эффективность, причем замена элементов этого элеполя недопустима или нецелесообразна, задача решается постройкой двойного элеполя путем введения второго поля, хорошо поддающегося управлению:



Рекомендации по развитию системы:

- использовать линию введения и развития полей,
- использовать линию введения элементов,
- использовать линии дробления и динамизации
- использовать линии согласования и структуризации

Примеры:

1) Светофор со звуком. Компьютеры и телефоны со звуковым дублированием интерфейса.

2) Способ регулируемого расхода жидкого металла из разливочного ковша, отличающийся тем, что с целью безаварийной разливки гидростатический напор регулируют высотой металла над отверстием разливочного стакана, вращая металл в ковше электромагнитным полем. Авторское свидетельство № 275331.[1]

3. Синтез и повышение эффективности систем на измерение и обнаружение (систем со свойствами полей взаимодействия).

3.1. Обходные пути.

Если дана задача на обнаружение или измерение, целесообразно так изменить систему, чтобы вообще отпала необходимость в решении этой задачи.

Если это не удастся, то целесообразно заменить непосредственные операции над объектом операциями над его копией (снимком).

Если это не удастся, то целесообразно перевести ее в задачу на последовательное обнаружение изменений.

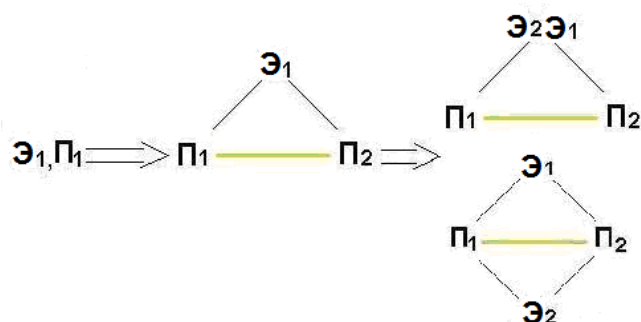
Примеры:

1) если нужно обнаружить изменения в состоянии некоторого объекта, объект наделяется способностью самостоятельно отсылать уведомление, если меняется его состояние;

2) если по какой-то причине это невозможно, то состояние объекта запрашивается после каждого обращения к нему.

3.2. Синтез и повышение эффективности систем на измерение и обнаружение.

Если неэлеполюная система плохо выполняет функции обнаружения или измерения, задачу решают, достраивая неэлеполюную структуру простого, комплексного (Э1-Э2) или двойного элеполя с полем на выходе:



Рекомендации по развитию системы:

- повысить эффективность созданной измерительной системы переходом от измерения функции к измерению первой производной функции и измерению второй производной функции.

- использовать линию введения элементов,
- использовать линии дробления и динамизации
- использовать линии согласования и структуризации
- использовать линии перехода в надсистему и к подсистемам;

Примеры:

1) Способ обнаружения момента начала кипения жидкости (то есть появления в жидкости пузырьков Э2). Через жидкость пропускают ток - при появлении пузырьков резко возрастает электрическое сопротивление. Авторское свидетельство № 269558.[1]

2) Способ определения фактической площади контакта поверхностей, отличающийся тем, что для окрашивания поверхностей применяют люминесцентные краски. Авторское свидетельство № 110314.[1]

3) Способ определения степени затвердевания (размягчения) полимерных составов, отличающийся тем, что с целью неразрушающего контроля в состав вводят магнитный порошок и измеряют изменение магнитной проницаемости состава в процессе его затвердевания (размягчения). Авторское свидетельство № 239633.[1]

4. Линии развития систем.

1. Переход в надсистему и к подсистемам (на микроуровень)

• На любом этапе внутреннего развития система может быть объединена с другими системами в надсистему с новыми качествами:

- образование бисистем или полисистем;
- развитием связей внутри бисистем и полисистем;
- увеличения различий между элементами системы: разные характеристики, разные элементы, противоположные элементы;
- свертывание би- и полисистем в моносистему с возможным повторением цикла образования полисистем;
- часть системы наделяется одним свойством, а другая часть или система в целом наделяется противоположным свойством.

- На любом этапе внутреннего развития эффективность системы может быть повышена переходом к развитию подсистемы (на микроуровень), в частности, заменой системы элементом.

Примеры:

- 1) однотипные данные объединяются в массив, упрощается их применение (доступ к нужным данным осуществляется по индексу);
- 2) множество браузеров, открытых для отображения множества страниц, свертываются в один, позволяющий хранить страницы во вкладках;
- 3) в целях уменьшения зависимостей между подсистемами сложной системы, подсистема заменяется элементом, который предоставляет унифицированный интерфейс ко всем возможностям классов подсистемы.

2. Линии коллективно-индивидуального использования систем

Это линия - частный случай развития бисистем, в которых одной из систем рассматриваемый объект, а другой – потребитель, пользователь этого объекта (человек, группа лиц, коллектив).

- Если имеется система индивидуального пользования, то происходит постепенное увеличение степени коллективного применения системы (часть системы общая, вся система общая, пользователи небольшая группа, пользователи - большая группа или неограниченное количество людей).
- Если имеется система коллективного пользования, то происходит постепенное увеличение степени индивидуальности применения системы (часть системы становится индивидуальной, вся система становится индивидуальной, количество пользователей постепенно снижается до одного индивидуума, для части жизни и деятельности индивидуума).
- Система индивидуального или коллективного пользования с развитием становится системой индивидуально-коллективного пользования, совмещаю преимущества той и другой системы.

Примеры:

1) Первые компьютеры были коллективного пользования, так были очень дорогими. Затем появились персональные компьютеры. После этого появились персональные компьютеры, объединенные в сеть (часть возможностей осталась индивидуальной, а часть – коллективной).

2) Интернет является системой коллективного пользования. Постепенно можно наблюдать процессы индивидуализации интернет: в поисковых программах учитывается в каком городе Вы находитесь, какие запросы Вы до этого уже делали, какие разделы информационных потоков Вас больше интересуют. Уровень индивидуализации интернет будет повышаться. Появился, например, военный интернет, могут возникнуть и другие варианты специализации интернет.

3) Редакторы текста появились как программа индивидуального пользования. Сейчас начали появляться редакторы текста общего пользования. Компания Google планирует выпустить оффлайновую версию своего текстового редактора Google Docs. Вместе с тем, онлайнный вариант программного обеспечения также сохранится и будет развиваться в дальнейшем. Google предложит систему WebOffice, которая объединит в себе как онлайнные, так и настольные офисные приложения. "Мы сможем заполнить пробел, образовавшийся сегодня между онлайнными и оффлайновыми программами", - говорят в Google¹⁰.

3. Линия введения элементов (веществ)

- Вместо элемента использовать "пустоту", вместо действия - бездействие;

¹⁰ <http://www.klerk.ru/soft/news/99162/>

- Если нужно ввести большое количество элемента, а это запрещено условиями задачи или недопустимо по условиям работы системы, в качестве элемента используют большое количество "пустоты";
 - Использовать для введения уже имеющиеся элементы или их модификации;
 - Вместо элемента использовать поле;
 - Вместо внутренней добавки использовать наружную добавку;
 - Вводить особо активную добавку в очень маленьких дозах;
 - Вводить добавку на время;
 - Вместо объекта используют его копию (модель), в которую допустимо введение добавки;
 - Добавку получают из внешней среды изменением ее в целом или по частям;
 - Введенный в систему элемент - после того, как он сработал, - должен исчезнуть или стать неотличимым от элемента, ранее бывшего в системе или во внешней среде.

Примеры:

1) для облегчения понимания кода программы используют комментарии, которые воспринимаются компилятором как "пустота";

2) при пересылке данных в виде блоков, состоящих из байтов, принимающая сторона должна удостовериться, что никакая часть из них не пропала. Для этого вместе с блоками должна передаваться проверочная информация, например, контрольная сумма. Она добавляется в качестве наружной добавки;

3) на этапе тестирования и отладки бывает полезно выводить текущее состояние программы с помощью расположенных в критических точках программы операторов вывода. Затем эти добавки удаляют, т.к. они не должны присутствовать на следующих этапах жизненного цикла.

4) Картина «Память» воинам-афганцам. Стилизована под фотографию. Живые – фигурами, погибшие – пустотами [2].

5) Для отслеживания скорости потока жидкости вводят метка в виде пузырьков воздуха.

4. Линия введения и развития полей взаимодействия

- Если в элепольную систему нужно ввести поле, то следует, прежде всего, использовать уже имеющиеся поля, носителями которых являются входящие в систему элементы;
 - При ограничениях на использование полей использовать поля, имеющиеся во внешней среде;
 - Если имеются ограничения на введение в систему поля, то следует использовать поля, носителями или источниками которых могут "по совместительству" стать элементы, имеющиеся в системе или во внешней среде.

Пример:

1) В разьеме сетевого адаптера MacBook Pro установлен магнит для упрощения подключения адаптера к компьютеру.



2) В системе есть абстрактный класс и его подкласс, вводится новый класс. Нужно ввести поле, чтобы обеспечить идентичность интерфейсов подкласса и нового класса. Используем поле, носителем которого является абстрактный класс: наследуем от него этот новый класс.[11]

3) Способ отделения пузырьков газа от жидкости в потоке жидкого кислорода. В системе два вещества. Оба являются носителями механического поля. Для решения задачи достаточно преобразовать движение этих веществ, "закрутив" поток. Центробежная сила отожмет жидкость к стенкам, а газ - к оси трубопровода.[1]

5. Линия дробления и динамизации

1. Выделить отдельный элемент, который рассматривается как целое.

2. Разделить элемент на две части (би-элемент) и соединить их между собой полем взаимодействия.
3. Сделать это поле взаимодействия более гибким, динамичным, управляемым, адаптирующимся к ситуации.
4. Разделить элемент не на две, а больше частей (поли-элемент) и соединить их между собой полями взаимодействия.
5. Сделать эти поля взаимодействия более гибкими, динамичными, управляемыми, адаптирующимися к ситуации.
6. Раздробить поли-элемент с динамичными полями взаимодействия до степени возникновения принципиально нового элемента.
7. Новое образование рассмотреть как самостоятельный элемент и изменить его по с пункта 1.

Пример:

- 1) Эволюция телефонов и компьютеров: жесткие, составные, с шарнирными соединениями, гибкие, с элементами поля вместо вещества (вместо клавиатуры – световое изображение клавиатуры).
- 2) Введение модульной структуры программных продуктов и динамизация связей между этими модулями – тенденция, которую прослеживается в развитии программного обеспечения.
- 3) Вместо зубчатой передачи – использование цепной передачи. Отдельные звенья цепи соединены подвижно и жестко одновременно.

6. Линии согласования-рассогласования и структуризации

- На любом этапе развития эффективность функционирования системы может быть повышена за счет согласования входящих в систему элементов и связей между ними.
- Рассогласование - это обратная сторона согласования. Если необходимо, например, защитить информацию, сделать недоступным изменения, то необходимо максимально рассогласовать возможные потоки информации. При передаче данных в том или ином виде происходит структуризация информация на разных уровнях.
- Этапы согласования: принудительное, буферное (специальным элементом или подсистемой), самосогласование (функции согласования-рассогласования есть, а специальной подсистемы для этого нет).
- Структуризация поля.
 - Если дана элеполевая система, ее эффективность может быть повышена переходом от полей однородных или имеющих неупорядоченную структуру к полям неоднородным или имеющим определенную структуру (постоянную или переменную)
 - Если элементу, входящему в элеполе (или могущему войти), должна быть придана определенная структура, то процесс следует вести в поле, которое имеет структуру, соответствующую требуемой структуре элемента.
- Структуризация элементов.
 - Если дана элеполевая система, ее эффективность может быть повышена переходом от элементов (веществ) однородных или имеющих неупорядоченную структуру к элементам (веществам) неоднородным или имеющим определенную структуру (постоянную или переменную)
 - Если нужно получить интенсивное воздействие в определенных местах системы (точках, линиях), в эти места следует заранее ввести активные, локально действующие добавки.

Пример: согласование скорости работы программы или передачи (обновления) данных со скоростью ее восприятия принимающей стороной (например, человеком).

Согласование состояния пользовательского интерфейса с выполняемыми пользователем действиями.

7. Линия развития систем в соответствии с S-образными кривыми

В результате взаимодействия системы с окружающей средой и надсистемой ее развитие может происходить в соответствии с S-образной кривой. Если удастся убедиться в том, что рассматриваемая система развивается в соответствии с S-образной кривой, а также выявить на каком этапе развития находится система, то можно выделить главные направления развития системы.

- Если система находится на 1-м этапе развития (начало развития), то
 - необходимо максимально использовать уже существующие инфраструктурные ресурсы и потребности
 - рекомендуется объединить систему с лидирующими в данный момент системами
 - рекомендуется развивать систему в конкретной области, где ее достоинства значительно превосходят ее недостатки
- Если система находится на 2-м этапе развития (бурное развитие), то
 - рекомендуется адаптировать систему к новым видам применения
 - адаптировать имеющиеся инфраструктурные ресурсы к нуждам развивающейся системы
- Если система находится на 3-м этапе развития (стабилизация, прекращение роста), то
 - На ближнюю и среднюю перспективы следует решать задачи по снижению затрат и развитию сервисных функций
 - На дальнюю перспективу следует предусмотреть смену принципа действия ТС или ее компонентов, разрешающую тормозящие развитие противоречия.
 - Очень эффективны глубокое свертывание, объединение альтернативных систем и другие способы перехода в надсистему
- Если система находится на 4-м этапе развития (спад), то
 - На ближнюю перспективу следует решать задачи по снижению затрат и развитию сервисных функций
 - На среднюю и дальнюю перспективы следует предусмотреть смену принципа действия ТС, разрешающую тормозящие развитие противоречия
 - Следует искать локальные области, в которых система все еще будет конкурентоспособной

(см. подробнее А.Любомирский, С.Литвин, Законы развития технических систем, <http://www.gen3.ru/3605/5454/>)..

8. Линии и тенденции развития программного обеспечения

1. SaaS (Software as Service), или, в более общем случае, EaaS (Everything as a Service). Компьютерные ресурсы и мощности предоставляются пользователю как Интернет-сервис. Перенос всего или части ПП на сторонний сервер и предоставление пользователю доступ к ПО через Интернет-браузер.

2. SOA (Service Oriented Architecture). Сервисно-ориентированная архитектура — это парадигма организации и использования распределенных информационных ресурсов таких как: приложения и данные, находящихся в сфере ответственности разных владельцев, для достижения желаемых результатов потребителем, которым может быть: конечный пользователь или другое приложение (OASIS). Многократное

использования функциональных элементов ИТ, ликвидация дублирования функциональности в ПО, унификации типовых операционных процессов.

3. Web 2.0 (Web 3.0). Методика проектирования систем, которые путём учета сетевых взаимодействий, становятся тем лучше (полнее), чем больше людей ими пользуются. (Тим О'Рейли). Использование большого количества пользователей для улучшения ПО.

4. RIA (Rich Internet Application). Приложение, доступное через Интернет, богатое функциональностью традиционных настольных приложений, не поддерживаемой браузерами непосредственно.

5. Облачные вычисления (Cloud Computing). Использование облачных вычислений ведет к их снижению затрат на ИТ-решения. Предприятия, использующие облачные сервисы, будут все чаще выступать в качестве облачных поставщиков приложений клиентам и партнерам. Более того, сейчас ведутся экспериментальные исследования по развертыванию кластеров виртуальных машин в облаках.

6. Комплексная аналитика. Сегодня аналитические инструменты используются во многих областях, включая оптимизацию и симуляцию бизнес-процессов. Следующий шаг - предоставлять для аналитики не только информацию, но и сами процессы, чтобы точнее понимать, что может случиться в будущем.

7. "Зеленые ИТ". Использование информационных технологий в качестве универсального способа повышения эффективности (в том числе экологической) различных технологий (документооборота, транспорта, отопительных систем, принятия управленческих решений и т.д.).

8. Изменения дата-центров. Меняется практика строительства дата-центров. Если раньше компании рассчитывали потенциальный рост на 15-20 лет, то сегодня этот срок сократился до 5-7 лет [<http://www.securitylab.ru/news/386807.php>].

9. Интеграция в социальные структуры. Компаниям необходимо уделять внимание использованию социального ПО и социальных ресурсов, а также интеграции в свою работу подобных сообществ.

10. Безопасность: мониторинг активности пользователей. Специалисты в области информационной безопасности сталкиваются с необходимостью выявлять вредоносную деятельность в постоянном потоке дискретных событий. В то же время, департаменты, занимающиеся безопасностью, сталкиваются с растущим спросом на средства анализа безопасности и отчетности. Инструменты для анализа и мониторинга позволяют организациям эффективно находить и расследовать подозрительную деятельность в режиме реального времени.

11. Флеш-память для дисков. Эта технология (SSD-накопители) не является новой, но уже сейчас она получает новые варианты использования в корпоративном секторе. Если ранее для компаний такие решения были слишком дорогими (пусть и производительнее HDD-накопителей), то на фоне снижения цен флеш-память начнет использоваться в гораздо большем объеме.

12. Виртуализация. Технологии в этой области уже довольно сильно влияют на развитие индустрии, но, согласно прогнозам Gartner, в 2010 году начнут распространяться новые элементы. К примеру, "живая" миграция, включающая в себя "перемещение" работающей виртуальной машины, в то время как ОС и ПО продолжают работать так, как если бы они остались на физическом сервере [10]. «От виртуальных ЦОД к облачным вычислениям, затем - к использованию «виртуальных клиентов», и далее – к «виртуальным устройствам». (М. Льюис)

14. Мобильные приложения . Как утверждают в Gartner, к концу 2010 года 1,2 млрд. человек будут использовать широкие возможности мобильных приложений. Однако для дальнейшего развития, по мнению экспертов, подобным технологиям нужна унификация платформы с полноценными персональными компьютерами. Будет

наблюдаться бум прикладных программ для аппаратных платформ iPhone и Intel Atom, а также операционных платформ Google Android и MeeGo .

15. Повышение уровня автоматизации (автоматизация автоматизации).

Любые системы автоматизации начинаются с уровня данных (получение, хранение и предоставление информации). Следующий уровень – автоматизация управления процессами, затем – автоматизации знаний и опыта, выдвижения идей.

ПРИЛОЖЕНИЯ к универсальной системе стандартов на решение изобретательских задач.

П1. Краткое введение в элепольный анализ.

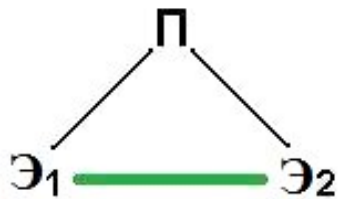


Рисунок 7. Элементарная система состоит из элементов и поля их взаимодействия (внутренний элеполь)

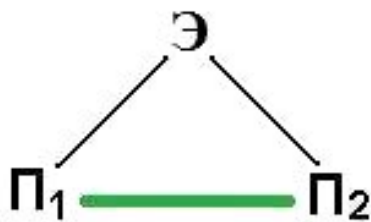


Рисунок 8. Элеполь на основе полей (внешний элеполь).

Минимальная модель системы (элеполь).

Элементарную модель системы можно представить в виде двух элементов, связанных между собой полем взаимодействия (элеполь¹¹). Элеполь имеет два вида связей между элементами. Непосредственная связь $\text{Э}_1\text{--}\text{Э}_2$ – это реализация той или иной необходимой функции или требования. Связь через поле взаимодействия $\text{Э}_1\text{--}\text{П}\text{--}\text{Э}_2$ – это то, с помощью чего удастся обеспечить необходимое действие или требование.

В некоторых случаях (измерение, преобразование полей, развитие теорий или искусства) в основе элеполя находятся поля, а преобразование поля происходит элементом «Э». Такие структуры мы будем называть внешним элеполем.

Если у элемента нет никаких свойств («валентности»), необходимых для образования связей (полей) и он не способен ни к захвату, ни к тому, чтобы он был захвачен другим элементом, то система просто не формируется.

<p>Описание моделей проблемных ситуаций: нет связей, есть недостаточная или вредная связь.</p>	<p>Элеполь: внутренний (с двумя элементами) и внешний (с двумя полями).</p>	<p>Повышение эффективности элеполей: комплексный, двойной, цепной элеполи.</p>

¹¹ Аналогом элеполя в технических системах является веполю (система, состоящая из веществ, объединенных тем или иным полем). Мы в данной работе будем различать закрытый, внутренний элеполь (носящий свойства элемента, связи которого направлены вовнутрь) и открытый, внешний элеполь (носящий свойства поля взаимодействия, связи которого направлены вовне).

П2. Алгоритм использования стандартов (АИСТ-2010)

Данный алгоритм применения стандартов основан на Универсальной системе стандартов на решение изобретательских задач 2010 года (АИСТ-2010). При решении технических задач даются рекомендации по использованию Системы стандартов – 76 (Стандарты-76).

Формулировку задачи и анализ проблемной ситуации рекомендуется проводить по АРИЗ-Универсал-2010.

1. Формулировка задачи.

Выделение конфликтующих элементов, ключевых параметров. Описание модели задачи в элепольной форме.

2. Если в задаче на изменение не полный элеполь, то перейти к стандарту 1.1 на создание элепольных структур.

Если рассматривается техническая система, то рекомендуется рассмотреть Стандарты-76, раздел 1.1.

3. Если в задаче на изменение имеются вредные связи, то использовать стандарты группы 1.2 на устранение вредных связей.

Если рассматривается техническая система, то рекомендуется рассмотреть Стандарты-76, раздел 1.2

4. Если в задаче на изменение имеются не эффективные связи, то рекомендуется группа стандартов 2 на развитие элепольных структур.

Если рассматривается техническая система, то рекомендуется рассмотреть Стандарты-76, класс 2

5. К стандартам части 2 рекомендуется обратиться в любом случае после рекомендаций стандартов 1-й части.

Если рассматривается техническая система, то рекомендуется рассмотреть Стандарты-76, класс 2

6. Если задача на измерение или обнаружение, то перейти к разделу 3 текста стандартов.

Если рассматривается техническая система, то рекомендуется рассмотреть Стандарты-76, класс 4.

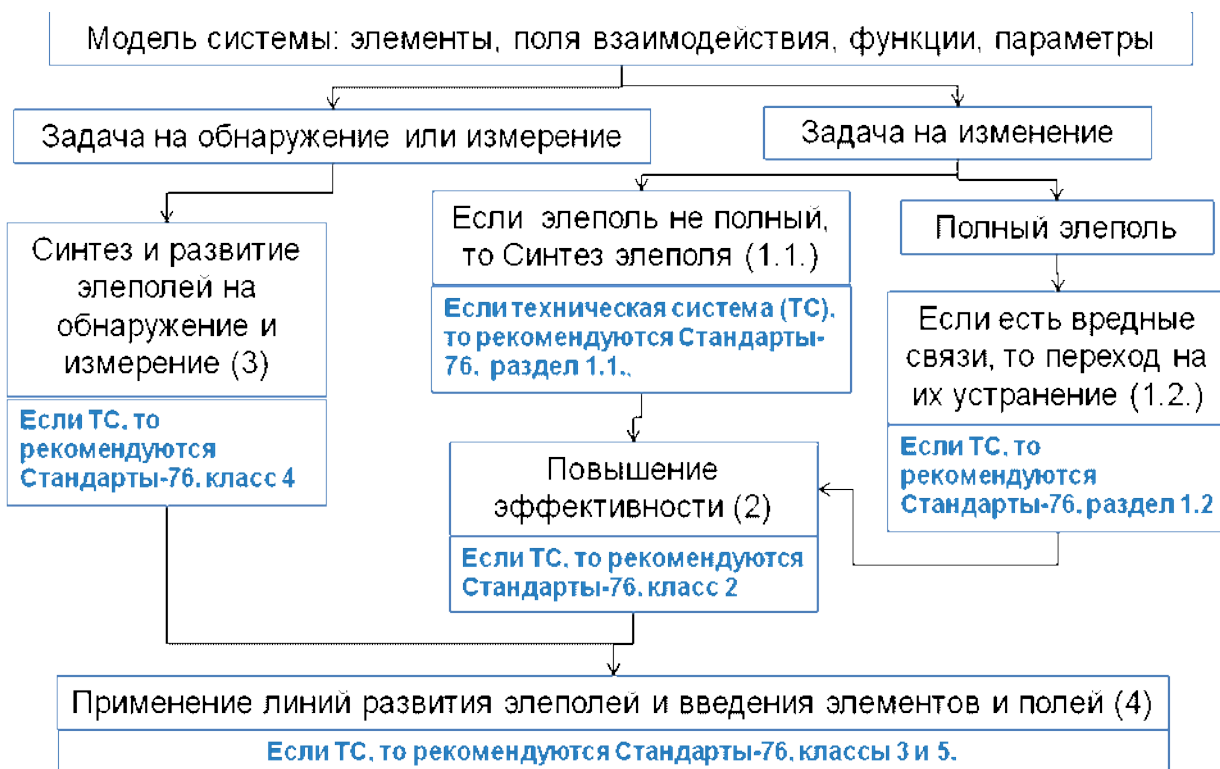
7. Рекомендуется в любом случае рассмотреть предложения, описанные в линиях развития систем.

Если рассматривается техническая система, то рекомендуется рассмотреть Стандарты-76, классы 3 и 5.

8. Если решение не найдено, то рекомендуется перейти к АРИЗ-Универсал-2010.

9. Если задача сформулирована в связи с развитием программного продукта, то следует использовать рекомендации алгоритма АИСТ-2010-П.

Упрощенное графическое представление АИСТ-2010:



В упрощенном виде алгоритм АИСТ-2010 приведен на сайте <http://temm.ru/ru/section.php?docId=4423>

П3. Перечень типовых полей и веществ, применяемых в технических системах

Типовые поля, применяемые в технических системах:

Механические – гравитационное, инерционное, центробежное, центорстремительное, перемещение объектов, давление, механические напряжения, силы трения силы поверхностного напряжения, сила адгезии, силы гидродинамические, силы аэродинамические, удары, силы упругости, сила кариолиса, диффузия, гиростатическое давление, давление струи жидкости или газа, осмос

Акустические – звук, инфразвук, ультразвук, вибрации

Тепловое – нагрев, охлаждение, стабилизация тепла

Химические – синтез молекул, разрушение молекул, особо активные вещества, катализаторы, ингибиторы, инертные вещества, запаховое, вкусовое, зрительное, осязательное

Электрические – электростатика, электролизация, коронный разряд, электрошок, электролиз, электрофорез, электродинамика, токи Фуко, электроразряды, сверхпроводимость

Магнитные – электромагнитное, радиоволны, свет, лазер, ультрафиолетовое, инфракрасное, рентгеновское.

Биологические – вирусы, микробы, биохимические взаимодействия.

Типовые вещества, применяемые в технических системах:

Биметаллы

Вещества с выраженным запахом

Вещества с магнитными свойствами

Вещества с эффектом Кюри

Вещества с эффектом памяти формы

Вещества, меняющие свой объём от внешних сил

Вещества, меняющие свой цвет от внешних сил

Вещества, меняющие сопротивление

Взрывчатые вещества (ВВ)

Вспенивающиеся вещества (пены)

Вязкие вещества (гели, пасты)

Легковывжигаемые вещества

Легкокипящие, газотворные вещества

Легкоплавкие вещества

Легкорастворимые вещества

Липучие вещества

Полимеризующиеся вещества

Теплоаккумулирующие вещества

Экзотермические вещества

Эндотермические вещества

П4. Алгоритм использования стандартов решение изобретательских задач и паттернов программирования (АИСТ-2010-П).

Данный алгоритм АИСТ-2010-П основан на алгоритме применения стандартов на решение изобретательских задач АИСТ-2010 и включает в себя расширение в область программирования для выбора подходящего паттерна проектирования.

Формулировку задачи и анализ проблемной ситуации рекомендуется проводить по АРИЗ-Универсал-2010.

1. Формулировка задачи.

Выделение конфликтующих элементов, ключевых параметров. Описание модели задачи в элепольной форме.

2. Если в задаче на изменение элеполь не полный, то перейти к стандарту 1.1 на создание элепольных структур.

3. Если в задаче на изменение имеются вредные связи, то использовать стандарты группы 1.2 на устранение вредных связей.

4. Если в задаче на изменение имеются не эффективные связи, то рекомендуется группа стандартов 2 на развитие элепольных структур.

5. К стандартам части 2 рекомендуется обратиться в любом случае после рекомендаций стандартов 1-й части.

6. Если задача на измерение или обнаружение, то перейти к разделу 3 текста стандартов.

7. Рекомендуется в любом случае рассмотреть предложения, описанные в линиях развития систем.

8. Если задача сформулирована в связи с развитием программного продукта, то следует уточнить предлагаемое решение в форме рекомендуемых паттернов программирования.

8.1. Если нужно устранить вредную связь:

• Если один элемент (создатель) содержит алгоритм создания второго элемента и при этом:

- Нужно заменять порождаемый элемент без изменения создателя.

Фабричный метод

- Нужно заменять семейство порождаемых элементов без изменения создателя. **Абстрактная фабрика**

- Алгоритм создания элемента не должен зависеть от того, из каких частей и в какой их комбинации состоит порождаемый элемент. **Строитель**

• Интерфейс одного элемента не должен зависеть от интерфейса второго. **Адаптер**

• Нужно создавать элемент/получать к нему доступ только в конкретных случаях.

Заместитель

• Нужно одинаково обращаться к элементам, не зависимо, являются они простыми и составными. **Компоновщик**

• Нужно отделить абстракцию элемента от реализации. **Мост**

• Нужно снизить зависимость между подсистемами. **Фасад**

• Нужно обеспечить слабую связанность между элементами системы. **Посредник**

• Нужно избежать связи между отправителем запроса и получателем. **Цепочка**

обязанностей

8.2. Если нужно повысить эффективность:

• Если один элемент (создатель) содержит алгоритм создания второго элемента и при этом:

▪ Нужно переопределить алгоритмы создания элементов в элементах-потомках. **Фабричный метод**

▪ Нужно переопределить алгоритмы создания семейств элементов в элементах-потомках. **Абстрактная фабрика**

▪ Нужно гарантировать, что элемент создан в единственном экземпляре.

Одиночка

▪ Новые элементы создаются путем копирования элемента-прототипа.

Прототип

• Нужно обеспечить совместное использование элементов с различными интерфейсами. **Адаптер**

• Нужно динамически расширить функциональность элемента, добавить ему новые обязанности на время. **Декоратор**

• Нужно заменить элемент до момента, когда он действительно понадобится.

Заместитель

• Нужно одинаково обращаться с простыми и составными элементами.

Компоновщик

• Нужно ограничить набор экземпляров элементов. **Приспособленец**

• Нужно предоставить доступ к элементам подсистемы с помощью одного элемента.

Фасад

• Нужно переопределить шаги алгоритма в элементах-потомках. **Шаблонный метод**

• Нужно предоставить последовательный доступ ко всем подэлементам составного элемента. **Итератор**

• Нужно представить запрос в виде элемента, ставить запросы в очередь, поддерживать отмену операций. **Команда**

• Элементы должны изменять свое состояние в зависимости от состояния других элементов. **Наблюдатель**

• Нужно объединить все связи между элементами в одном элементе. **Посредник**

• Нужно обойти элементы структуры, выполнив над каждым из них некоторую операцию. **Посетитель**

• Нужно изменять поведение элемента в зависимости от его состояния. **Состояние**

• Нужно определить семейство взаимозаменяемых алгоритмов в виде элементов.

Стратегия

• Нужно сохранить состояние элемента, восстановить элемент в нужном состоянии.

Хранитель

8.3. Если удалось выбрать подходящий вариант, переход к описанию подходящего паттерна.

8.4. Если паттерн по описанию все же не соответствует желаемому результату, переход к группе родственных паттернов.

8.5. Если в группе не удалось подобрать подходящий вариант, возможно задача не должна решаться с помощью паттернов.

8.6. Для подходящего паттерна предоставляются рекомендации по использованию совместно с другими паттернами.

9. Если решение не найдено, то рекомендуется перейти к АРИЗ-Универсал-2010.

С описанием выбранных паттернов программирования можно познакомиться на сайте <http://www.temm.ru/ru/section.php?docId=4515> или <http://codelab.ru/cat/patterns/>.

Упрощенное графическое представление АИСТ-2010-II:



VI. Литература

1. Альтшуллер Г.С. В сб. "Нить в лабиринте". - Петрозаводск: Карелия, 1988. - С. 165-230. Маленькие необъятные миры: стандарты на решение изобретательских задач. Стандартные решения изобретательских задач (76 стандартов). <http://www.altshuller.ru/triz/standards.asp>
2. Мурашковский Ю.С. Биография искусств. Ч.1. Петрозаводск: Скандинавия, 2007. 234 с. : ил. ; Ч.2. Петрозаводск : Скандинавия, 2007. 316 с. :
3. Б. Л. Злотин, А. В. Зусман. Модели для творца. Теория развития коллективов. <http://triz-summit.ru/ru/section.php?docId=3946>
4. Петров В. История развития стандартов, Тель-Авив, 2003 г. <http://www.triz-summit.ru/ru/section.php?docId=3811>
5. Поиск новых идей: от озарения к технологии (Теория и практика решения изобретательских задач)/ Г.С.Альтшуллер, Б.Л.Злотин, А.В.Зусман, В.И.Филатов. - Кишинев: Картя Молдовеняскэ, 1989.- 381 с
6. Алгоритм использования стандартов на решение изобретательских задач (АИСТ-77), Ленинград, 1988 г.
7. Любомирский А., Литвин С., Законы развития технических систем. 2003, <http://www.gen3.ru/3605/5454/>
8. Одинцов И.О., Рубин М.С. Повышение эффективности разработки программных продуктов на основе методов ТРИЗ, ТРИЗ-Фест, 2009, Санкт-Петербург, <http://www.temm.ru/ru/section.php?docId=4539>
9. Рубин М.С. Об универсальной системе стандартов на решение изобретательских задач, 2009 г., ТРИЗ-Фест, Санкт-Петербург, <http://www.triz-summit.ru/file.php/id/f4440/name/MRubin-Стандарты.doc>
10. Рубин М.С., Одинцов И.О., Пономарева А.В., Зиненко О.И., Прогнозирование развития программного обеспечения на основе ТРИЗ, ТРИЗ-Саммит, 2010, <http://www.triz-summit.ru/file.php/id/f4770/name/ПМС-прогнозирование-ПО-1-1.pdf>
11. Зиненко О.И. Систематизация и анализ паттернов проектирования на основе стандартов теории решения изобретательских задач. СПбГУ, Математико-механический факультет, Кафедра информатики. Дипломная работа. Руководители Одинцов И.О., Рубин М.С., СПб, 2010 г., <http://www.temm.ru/ru/section.php?docId=4538>
12. Пономарева А. В., Прогнозирование развития программных продуктов методами ТРИЗ, СПбГУ, Математико-механический факультет, Кафедра информатики, дипломная работа, руководители Одинцов И.О., Рубин М.С., 2010, <http://www.temm.ru/ru/section.php?docId=4538>
13. Рубин М.С. Филогенез социокультурных систем. Секреты развития цивилизаций. СПб 2010, <http://www.temm.ru/ru/section.php?docId=4472>
14. Одинцов И.О., Рубин М.С. Опыт применения методов ТРИЗ для повышение эффективности разработки ПО, Москва, 2009, <http://temm.ru/ru/section.php?docId=4419>
15. Рубин М.С. О новой системе стандартов на решение изобретательских задач, 2009 г., <http://www.temm.ru/ru/section.php?docId=4201>