

Рис. 6. Изображение сплава позвоночного ствола с выявлением очага поражения (артефакта)

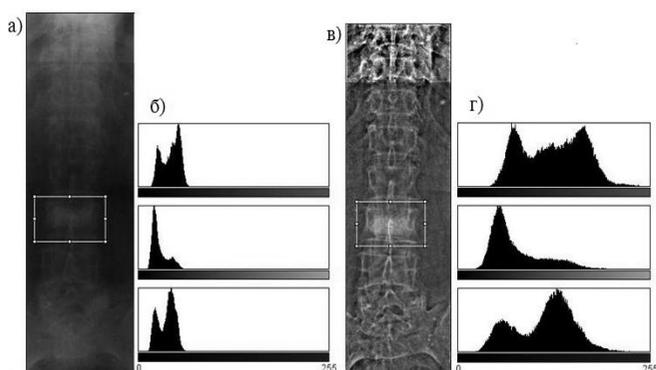


Рис. 7. Гистограммный анализ: а, б) изображение сплава без фильтрации и его гистограмма; в, г) изображение сплава с фильтрации и его гистограмма

Гистограммный анализ результирующих изображений сплава (см. рис. 7) показал, что полосовая Фурье – фильтрация расширяет динамический диапазон, тем самым улучшая визуальное распознавание артефактов.

В заключение отметим, предложенная процедура и модель продемонстрировала свою действенность в распознавании артефактов в области позвоночного ствола. Вместе с тем, применение рассмотренной модели к исследованию поверхности сердечно-сосудистой системы, как показала практика, эффективна.

Литература

1. Цифровая обработка изображений в информационных системах. Новосибирск: Изд. НГТУ, 2002. – 352 с.
2. Яне Б. Цифровая обработка изображений. Пер. с англ. М.: Техносфера, 2007. – 584 с.
3. Гонсалес Р., Вудс Р. Цифровая обработка изображений. Пер. с англ. М.: Техносфера, 2006. – 1072 с.

THE MODELS OF THE FUSION OF X-RAY AND SCINTIGRAPHY IMAGES FOR THE DETECTION OF THE ARTIFACTS

Khlestkin A.Yu., Starogilova O.V.

The article proposes a model of the fusion scintigraphy and X-ray images to detect artifacts. This model bases on projective change, polynomial interpolation and Fourier filtering. There are examples of alloys images of the developed model.

Keywords: fusion scintigraphy and X-ray images, detect artifacts, Fourier filtering X-ray images, projective change scintigraphy images, polynomial interpolation scintigraphy images.

Хлесткин Андрей Юрьевич, аспирант Поволжского государственного университета телекоммуникаций и информатики (ПГУТИ). Тел. 8-903-309-30-53. E-mail andreyx1@mail.ru

Старожилова Ольга Владимировна, к.т.н, доцент Кафедры «Высшая математика» ПГУТИ. Тел. (8-846) 228-00-23.

НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 658.512.22

ОБРАБОТКА И ХРАНЕНИЕ КЛАССА ПРОЕКТНЫХ РЕШЕНИЙ В ИНТЕГРИРОВАННОЙ ИНСТРУМЕНТАЛЬНОЙ СРЕДЕ

Похилько А.Ф.

В статье рассматриваются возможности выделения из проектной деятельности структур проектных решений, построения моделей классов объектов проектирования, а также хранения, отображения и дальнейшего использования информации в кон-

тексте интегрированной инструментальной среды (ИИС).

Ключевые слова: интегрированная инструментальная среда, проектная деятельность, модель, класс, объект проектирования, хранение, модификация.

Введение

Современная стратегия развития производственных систем определяется в значительной мере инфокоммуникационными технологиями [1], обеспечивающими создание так называемых «виртуальных предприятий», то есть производственных структур, взаимодействие между подразделениями которых осуществляется на базе современных средств телекоммуникаций с использованием стандартов информационного взаимодействия (CALS стандартами) между информационными системами (проектирование, производство, управление процессами) на всех стадиях жизненного цикла изделий и услуг. Существующий опыт и практика развития соответствующих инфокоммуникационных технологий, выводят на передний план необходимость исследования и развития методов представления проектной информации (проектных решений), обеспечивающих не столько обмен между различными информационными системами или сервисами, но в первую очередь методов и способов сохранения, модификации и обобщения информационных представлений проектных решений [2]. В [3] изложены теоретические предпосылки (математическая модель) описания и модификации проектных решений на основе формализации процесса их получения в форме многоуровневой системы протоколов их создания средствами современных САПР. В данной работе более детально рассматривается структура информационного представления проектных решений и указывается, что описание проектного решения при указываемых предпосылках создает основу для обобщения, проектных решений в форме создания описания класса проектных решений, как более высокоуровневой абстракции информационного представления [4].

Постановка задачи

Исходя из вышесказанного, задачу можно сформулировать следующим образом: необходимо создать ИИС, которая бы обеспечивала выполнение следующих требований:

- возможность интеграции разнородных приложений образом, удобным для пользователя;
- «развязку» разнородной информации для обработки в специализированных приложениях;
- работа с содержимым проектного решения, его идейным наполнением;
- «контекстность» предоставляемой пользователю информации;

- «отвязку» от конкретных форматов данных и их преобразований.

Для возможности создания интегрированной инструментальной среды необходимо разделить информационные потоки между модулями системы с тем, чтобы обеспечить работу специализированных приложений только с характерной для них информацией, также разработать и реализовать базу данных и модуль управления, которые бы обеспечивали:

- хранение проектных решений;
- динамическое наполнение множества проектных решений;
- хранение сопутствующей информации (описаний, вычислений и пр.);
- управление внешними функциональными модулями-компонентами среды, обмен данными между модулями и их сохранение;
- выдачу и отображение в удобном виде хранящейся информации;
- удобное управление проектами со стороны пользователя.

Модель и содержание решения

Как показывает практика, создание системы удовлетворяющей всем возникающим требованиям (используя единую информационную среду, обеспечивающую однородность данных и автоматизацию обмена данными на протяжении всего процесса проектирования, причем не только проектирования, но и всего жизненного цикла изделия) могут позволить себе лишь очень крупные концерны. Отсутствие возможности, на данный момент, создать систему, которая бы удовлетворяла всех, приводит к мысли о создании инструмента, позволяющего объединять существующие в настоящее время и используемые в процессе проектирования, и всего жизненного цикла изделия, приложения. Упрощенная схема такой системы приведена на рис. 1.

Далее встает вопрос о модели представления процесса получения проектных решений. Проводившиеся ранее исследования и работы показывают эффективность и доступность для пользователя и программиста метода представления процесса проектирования в виде И-ИЛИ-графа. Вершинами графа являются элементы технической системы (ТС) и их признаки, а дуги показывают иерархическую соподчиненность между элементами и их признаками, а также принадлежность признаков элементам. В вершинах множества И располагаются элементы, выполняющие различные функции, а также признаки элементов, относящихся к разным группам призна-

ков. Вершины ИЛИ объединяют альтернативные элементы, выполняющие одинаковые или очень близкие функции, и признаки, характеризующие индивидуальные особенности каждой ТС.

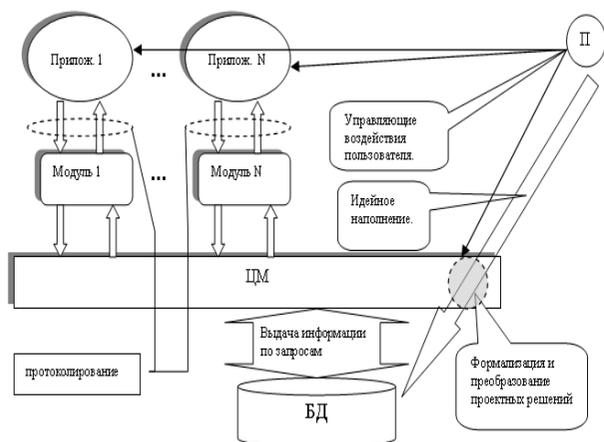


Рис. 1. Упрощенная схема ИИС

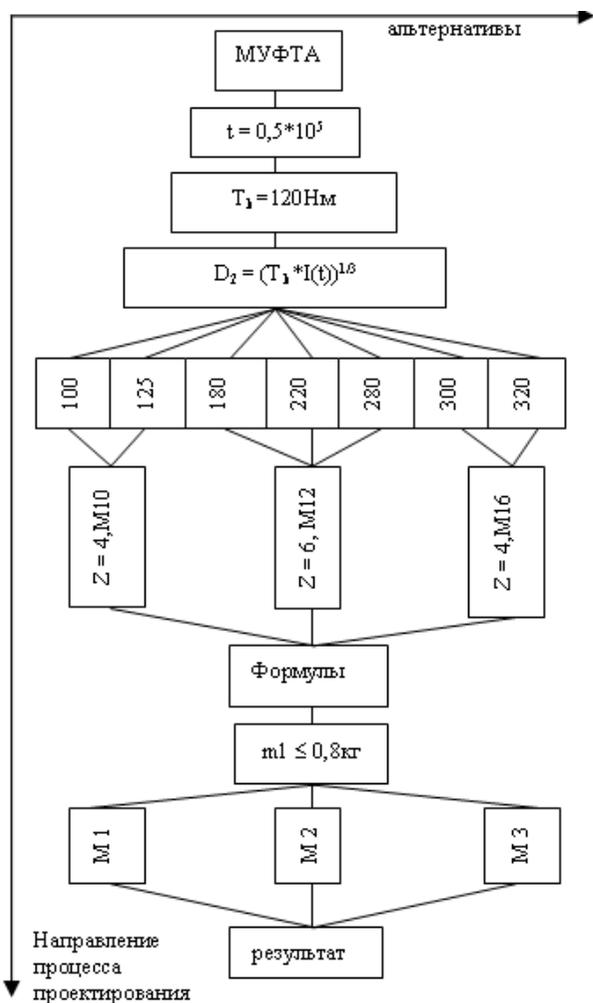


Рис. 2. Пример дерева решения

Множество путей достижения решения (прохождение по вершинам И-ИЛИ-графа) может быть представлено в виде логической

записи (предиката), где при выполнении всех условий принимается значение «истина», в противном случае – «ложь». Форма записи следующая:

$$P(A \& (B \cup C \cup D) \& F = [0;1], \quad (1)$$

где A, F – элементы, выполняющие различные функции; B, C, D – альтернативные элементы, выполняющие близкие функции. Решение системы уравнений, представленных в предикатной форме, это и есть искомое техническое решение. Пример такого дерева приведен на рис. 2.

Недостатки существующих реализаций этого метода:

- неполнота набора вариантов ТС;
- невозможность внесения изменений;
- невозможность добавления новых решений.

Другим подходом может стать динамическое наполнение множества проектных решений, то есть построение и наращивание графа во время самого процесса проектирования, что сразу же снимает вышеописанные недостатки, но накладывает более жесткие требования по обработке и хранению данных. В ходе анализа задачи были выделены несколько сущностей, реализация и использование которых дали бы возможность динамически строить граф проекта, использовать полученные решения ТС, удобно отображать сопутствующую проекту служебную и справочную информацию.

- проект (Пр) – наибольшая абстракция, которая включает в себя дерево решений и всю сопутствующую информацию.

- проектная операция (ПО) – некое действие, выполняемое пользователем и выделяемое им в отдельную смысловую единицу. Она разделяется в свою очередь на 3 элемента:

- операция тип1 (O1) – операция, являющаяся выполнением некоторых действий, которые лишь в совокупности обладают смыслом для пользователя. То есть применительно к разрабатываемой системе, это набор строчек макроса, выполнение которых внешним модулем или другой программой приводит к некоему результату, имеющему смысл для пользователя (скажем, отрисовка окружности с последующей вытяжкой (для SolidWorks));

- операция «назначения» (тип 2) (O2) – операция, являющаяся назначением параметру приложения, «участвующего» в системе, значения (например назначение размеру некоторого числа);

- условие (У) – операция проверки истинности некоторого условия, задаваемого в ходе

работы с проектом. Обеспечивает «ветвление дерева проекта». Накладывается на параметр (см. термин ниже), может быть нескольких типов (существование, равенство, различные неравенства и принадлежность диапазону);

- параметр (П) – некая абстракция, интерпретируемая в зависимости от контекста. По смыслу близка к переменной. В реализуемой системе может содержать в себе постоянную величину, расчет, SQL – запрос, интерактивный запрос. В перспективе может хранить любую информацию, интерпретация которой будет зависеть от контекста, в котором применяется параметр (например, рис. 2).

Основу модели составляют:

- множество проектов, $Pr\{Pri\langle key, Name, Desc\rangle\}$, где key – уникальное число-идентификатор проекта (уникальное числовое значение записи в БД, (далее будет опускаться)); $Name$ – имя проекта (краткое смысловое содержание); $Desc$ – описание содержания проекта;

- множество проектных операций $PO\{POi\langle ProjID, Type, Name, Desc, NOP1, NOP2\rangle\}$, где $Type$ – тип элемента множества (1 – O1, 2 – O2, 3 – Y); $Name$ – имя элемента, отражающее смысловое наполнение; $Desc$ – подробное описание; $NOP1$ – уникальное число, определяющее элемент множества, на который происходит переход после выполнения условия Y; $NOP2$ – уникальное число, определяющее элемент множества, на который происходит переход после невыполнения данного условия;

- множество операций тип 1 $O1\{O1i\langle OpID, Content\rangle\}$, где $OpID$ – уникальное число, определяющее элемент множества ПО, с которым однозначно связан данный элемент; $Content$ – содержимое элемента (текст макроса, скрипта и т.п.);

- множество операций тип 2 $O2\{O2i\langle OpID, ParamID\rangle\}$, где $OpID$ – уникальное число, определяющее элемент множества ПО, с которым однозначно связан данный элемент; $ParamID$ – уникальное число, определяющее элемент множества П, числовое значение которого будет использовано при выполнении данного элемента множества O2;

- множество условий $Y\{Yi\langle OpID, ParamID1, ParamID2, Condition\rangle\}$, где $OpID$ – уникальное число, определяющее элемент множества ПО, с которым однозначно связан данный элемент; $ParamID$ – уникальное число, определяющее элемент множества П, числовое значение на которое будет накладываться условие; $ParamID1$ – уникальное число, определяющее

элемент множества П, числовое значение которого будет использовано при выполнении данного элемента множества Y (левая граница диапазона); $ParamID2$ – уникальное число, определяющее элемент множества П, числовое значение которого будет использовано при выполнении данного элемента множества Y (правая граница диапазона); $Condition$ – собственно условие, налагаемое на элемент множества П;

- множество параметров $P\{Pi\langle ProjID, Name, Type, Content, Desc, Value\rangle\}$, где $ProjID$ – уникальное число, определяющее принадлежность данного элемента к проекту; $Name$ – имя элемента; $Type$ – тип(число), согласно которому интерпретируется содержимое элемента; $Content$ – содержимое элемента; $Desc$ – описание элемента; $Value$ – числовое значение, возможно получаемое при интерпретации содержимого элемента.

Процесс проектирования можно представить в виде совокупности элементов множества ПО. Каждый элемент множества PO_i представляет собой отдельную реализованную подзадачу проектирования. Таким образом, существует возможность описать проект Pr , используя реализованные проектные решения:

$$Pr = O1_i \& O2_j \& Y_k((O1_i \& O2_j \& \dots) \text{or} (Y_k(\dots))) \quad (2)$$

где i, j, k – индексы элементов множеств, адекватные проектным ситуациям.

Создание нового и пополнение проектного решения начинается с выделения некоторой области задач, в которую будет вписываться проект (создание или выбор элемента множества Pr). Далее следует пополнение и (или) изменение элементов всех остальных множеств в контексте данного проекта, что позволяет автоматически поддерживать связь между элементами множеств. Посредством этих связей в дальнейшем становится возможным восстановление дерева проекта.

Воспроизведение проектного решения заключается в выборе элемента Pr_i и дальнейшем прохождении по ветвям дерева с обработкой элементов множества ПО.

Для реализации вышерассмотренного подхода была смоделирована система, состоящая из трех компонентов: центрального модуля (он же модуль работы с БД) и интерфейсных модулей для моделилера SolidWorks, и мэтпроцессора MathCAD. Для хранения и обработки данных была создана база данных на основе Paradox7. Структура БД показана на рис. 3.

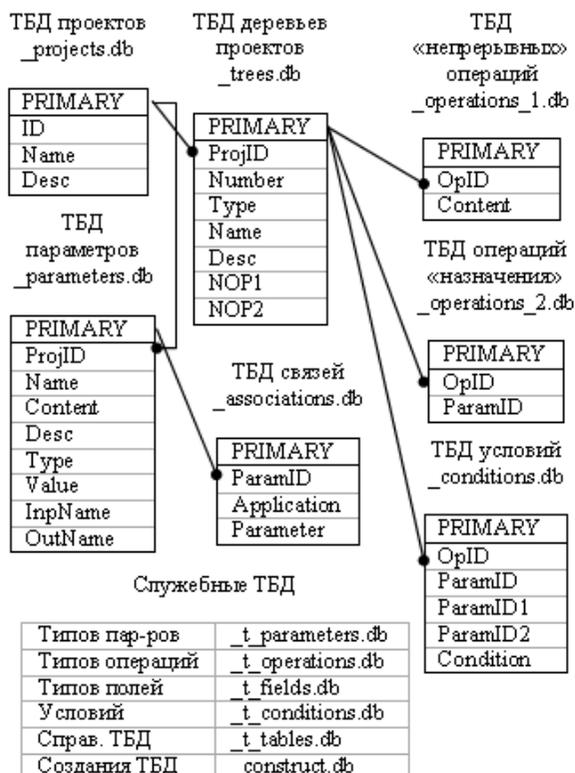


Рис.3. Структура БД

Апробация

Результатом реализации стал макет программного комплекса из трех компонентов Core, SWService, MCSservice и проведена ее апробация на множестве типовых задач проектирования машиностроительных деталей и узлов. В исходном виде процесс проектирования описывается как типовая методика, включающая описание как задачи в текстовом виде, так и графических эскизов и последовательности математического расчета. Подтверждена принципиальная работоспособность системы, ее возможность сохранять накопленные проектные решения. Функциональные возможности системы соответствуют исходным предположениям.

Заключение

К направлениям для дальнейшей работы можно отнести усовершенствование механизма проведения расчетов и наглядное отображение дерева проекта.

Произведенный программный эксперимент, несмотря на некоторые недоработки в ИИС в целом, в связи с согласованием работы отдельных компонентов, показал возможность создания подобных интегрированных сред и успешного хранения не только сопровождающей проектной информации, но и информации, касающейся «идейного» наполнения проекта, эффективно решать задачи построения модели процесса проектирования, моделирования классов объектов проектирования.

Перспективы развития:

- улучшение механизма расчетов в матпроцессоре;
- добавление возможностей параметризации не только размеров, но других величин (включая нечисленные);
- обеспечение работы не только с деталями, но и со сборками, то есть включение других проектов как подветвей;
- оптимизация взаимодействия с пользователем;
- расширение набора служебных модулей для взаимодействия с другими приложениями;
- переход к новой модели хранения проектных решений в виде потокового графа.

Литература

1. Судов Е.В. Интегрированная информационная поддержка жизненного цикла машиностроительной продукции. Принципы. Технологии. Методы. Модели. М.: ООО ИД «МВМ», 2003. – 264 с.
2. Sobolewski M. Foreword Next Generation Concurrent Engineering: Smart and Concurrent Integration of Product Data, Services and Control Strategies. ISPE, 2005. – 620 p.
3. Похилько А.Ф. Масляницын А.А. Скворцов А.В. Удовиченко А.В. Формальное представление процесса проектной деятельности в инструментальной инфокоммуникационной среде САПР // ИКТ. Т.6, № 1, 2008. – С. 80-84.
4. Гради Буч. Объектно-ориентированный анализ и проектирование с примерами приложений. М.: ISBN 978-5-8459-1401-9, 2008. – 720 с.

PROCESSING AND STORAGE OF THE CLASS OF DESIGN DECISIONS IN INTEGRATED INSTRUMENTALITY ENVIRONMENT

Pokhilko A.F.

In the article are considered possibilities of selection from design activity of structures of design solutions, constructions of models of classes of objects of designing, and also a storage, map and further use of the information in a context of the integrated tool medium.

Keywords: *integrated instrumentality environment, designed activity, model, class, object of designing, modification.*

Похилько Александр Федорович, к.т.н., доцент, профессор Кафедры «Системы автоматизированного проектирования» Ульяновского государственного технического университета. Тел. 8-902-356-84-16. E-mail: afp@ulstu.ru

УДК 681.518

СОВЕРШЕНСТВОВАНИЕ БИЗНЕС-ПРОЦЕССА ОБСЛУЖИВАНИЯ КОРПОРАТИВНЫХ КЛИЕНТОВ БАНКА С ИСПОЛЬЗОВАНИЕМ ЭЛЕКТРОННЫХ ТЕХНОЛОГИЙ

Димов Э.М., Луковкин С.В., Третьяков Р.В.

В статье рассмотрены альтернативные подходы к ведению документооборота между банком и клиентом, проведен их анализ в интересах повышения эффективности и качества управления, а также снижения издержек на их реализацию.

Ключевые слова: бизнес-процессы обслуживания, совершенствование бизнес-процесса, системы дистанционного банковского обслуживания, оптимальные решения в обслуживании.

Введение

Интенсивное развитие информационных технологий и прорыв в области телекоммуникаций, приведшие к повсеместному использованию компьютеров и глобальной сети интернет, оказали существенное воздействие на подход банков к обслуживанию клиентов. Важную роль в привлечение клиентов играет высокое качество обслуживания, что означает возможность банка предоставить широкий круг услуг там и тогда, где и когда пожелает клиент. При этом должна обеспечиваться высокая точность, оперативность, безопасность ведения банковских операций. Новые условия требуют не только активного совершенствования традиционных методов обслуживания клиентов, но и внедрения новых банковских решений, основанных на передовых достижениях науки и техники. Одним из современных инструментов, позволяющих ускорить и упростить взаимодействие банка с клиентом, является использование системы дистанционного банковского обслуживания (СДБО).

Постановка задачи

Объектом исследования является отделение коммерческого банка, а именно производственные бизнес-процессы обслуживания юридических лиц. Целью анализа является поиск оптимальных решений в обслуживании клиентов и снижение производственных издержек, связанных с обеспечением этого процесса. Перед авторами стоит

задача описать существующие бизнес-процессы, выявить их недостатки и найти пути их устранения. Необходимо, с одной стороны, обеспечить клиента актуальной информацией о состоянии его счетов и движений по ним, дать возможность оперативно распоряжаться денежными средствами (отправлять платежные поручения, покупать/продавать валюту), а, с другой стороны, минимизировать издержки банка на обслуживание этой задачи.

Анализ бизнес-процесса обслуживания счета клиента

Бизнес-процесс обслуживания счета можно разбить на два основных этапа, которые условно можно назвать «Выписка» и «Документы». Для того чтобы правильно спланировать свою расчетную деятельность на текущий день, клиенту необходимо знать остаток на его расчетных счетах и движение по ним за предыдущий период («Выписка»), а для осуществления текущей расчетной деятельности составлять документы и доставлять их в банк для исполнения («Документы»).

Обобщенная модель бизнес-процесса обслуживания банковского документа представлена на рис. 1. Разобьем исследуемый бизнес-процесс на условные подпроцессы, а систему представим в виде трех подсистем. В таблице 1 представлены альтернативные реализации исследуемого бизнес-процесса.

Рассмотрим классическую реализацию процесса обслуживания счета клиента в течение операционного дня банка. Классический метод получения выписки предполагает, что клиент (доверенное лицо клиента) приезжает в банк и у своего операционного работника, либо из персональной банковской ячейки получает бумажный носитель, содержащий информацию о движении по его расчетному счету, а также копии исполненных документов с отметкой банка. Если у клиента в течение дня возникает необходимость