

**УНИВЕРСАЛЬНАЯ БАЗА ДАННЫХ С ВЫДЕЛЕННОЙ ТЯЖЁЛОЙ  
ИНФОРМАЦИЕЙ В СПРАВОЧНЫЕ СУЩНОСТИ  
НА ОСНОВЕ СИНТЕЗА МЕТАИНФОРМАЦИИ И ДАННЫХ**

**И. А. Микляев**

*Институт судостроения и морской арктической техники  
филиала Северного арктического федерального университета*

**Поступила в редакцию 19.12.2012 г.**

**Аннотация.** Универсальная база данных может применяться в проектировании информационных систем с целью унификации инструментария для структурирования и обработки информации в различных предметных областях, создания приложений, легко перенастраиваемых при изменениях в бизнес среде без модификации структуры базы данных и программного обеспечения. В Универсальном приложении реализован универсальный инструментарий для работы со всеми матричными универсальными базами данных. Программа рассчитана на самый широкий спектр пользователей. В зависимости от уровня знаний пользователь может начать с самого простого: ввода и работы с табличной информацией в стандартном виде, не отвлекаясь на дополнительную информацию, а далее по мере развития углубляться до комплексного управления сразу множеством таблиц, характеристик и значений.

**Ключевые слова:** универсальная база данных, унификация инструментария для структурирования информации, информационные системы.

**Annotation.** Universal database is intended for the informational systems design based on the using structuring and processing data unified tools in the different application domains, for the engineering easy update software in the variable business environment without data base structure and software update. In universal application is marketed universal toolbox for working with all matrix universal databases. Program is calculated on the most broad spectrum of users. Depending on the level of knowledge's an user can begin with simplest: entering and work with tabular information in the standard type, not distracting to additional information, but hereinafter on the measure of development to deepen before complex control of immediately ensemble of tables, features and values. Besides, exhibit is presented and in the manner of the compiled module, which program product developer, can comprise of its under development project.

**Keywords:** designing the control systems of databases, unification a toolbox for structuring information, universal type of data, universal database.

## **ВВЕДЕНИЕ**

Привычно рассматривать окружающий нас мир как совокупность объектов. Дерево представляется нам именно деревом, а не массой атомов. Мы можем затем начать выделять в нем корни, ветви, листья и т.д., но и они существуют в нашем восприятии в виде объектов. Объектная идеология позволяет построить процесс разработки программ в соответствии с естественным человеческим восприятием.

С другой стороны идея о том, что мир пронизан синергией – энергией взаимодействия,

нашла удивительный отклик во всех сферах современного мира. Феноменом этого отклика стало представление о нашей жизни в целом в виде бесконечной совокупности взаимосвязанных и взаимозависимых открытых и закрытых систем [1].

И вся эта “реальность, данная нам в ощущениях” требует создания соответствующего информационного пространства – синергетического. Описываемые базами данных (БД) предметные области и бизнес-процессы, как и все живые системы, обладают свойствами жизнестойкости, функциональности и адаптивности. Их информационное поле многомерно, многогранно и постоянно изменяется.

То же самое происходит и со значениями характеристик объекта. Взаимодействие – синергия с внешним миром является тем моментом, который придаёт значения характеристикам объектов. Вокруг нас синергетическое пространство нелинейное, неоднозначное, и синергетические образы объектов в нём носят отпечаток незавершённости и диалога [2].

В идеале модель информационного поля объекта должна предельно приближаться к реальному его содержанию. Для адекватного описания предметной области необходимо провести всесторонний анализ информации. И при столкновении с синергетическими характеристиками, имеющими сразу несколько значений, возникают естественные сложности их отображения в традиционных платформах БД.

Стандартный реляционный подход к организации данных делает ставку на максимально эффективное манипулирование данными, для этого применяет дезинтеграцию данных до их полной однородности. А поскольку реальные бизнес-среды практически никогда не обладают однородностью, это приводит к чрезмерному абстрагированию от предметной области [3], и разрастанию таблиц и количества связей в геометрической прогрессии [4].

Напротив, стандартный объектно-ориентированный подход к организации данных предполагает каждый объект реального мира описывать индивидуально, всей совокупностью его параметров, что является более естественным для сложных систем. Безусловно, это позволяет адекватно отобразить структуру и состав предметной области, многообразие синергетических характеристик и облегчает проектирование,

однако чрезвычайно затрудняет навигацию по данным [3].

## МОДЕЛЬ УБД

В [9] представлена общая концепция формирования УБД, в результате чего получена модель УБД в виде представленном на рис. 1.

При реализации механизма обработки информации в УБД на передний план выходят две проблемы:

1. Разнотипность значений атрибута (поля) «Значение» сущности (таблицы) «Характеристика экземпляра сущности»;
2. Хаотичность размещения информации в этой таблице, соответствующей каждому экземпляру сущности.

В логической модели на рис. 1 первая проблема решается через универсальный тип. В автономной матричной объектно-реляционной универсальной базы данных (МУОРБД) разработан механизм универсального типа данных на основе понижения системы счисления символа [6]. В [10] предложена идея выделения информации во внешние справочные сущности (таблицы), что делает оставшуюся часть БД более компактной и максимально динамичной.

Учитывая, что вся информационная нагрузка ложится на сущность «Характеристика экземпляра сущности» (рис. 1), а точнее на поле «Значение», то последний путь решения проблемы разнотипности видится максимально эффективным.

## СИНТЕЗ СБД И ДАННЫХ ПОЛЬЗОВАТЕЛЯ

Ещё одно направление расширения возможностей УБД (рис. 1) состоит в том, что рассмат-

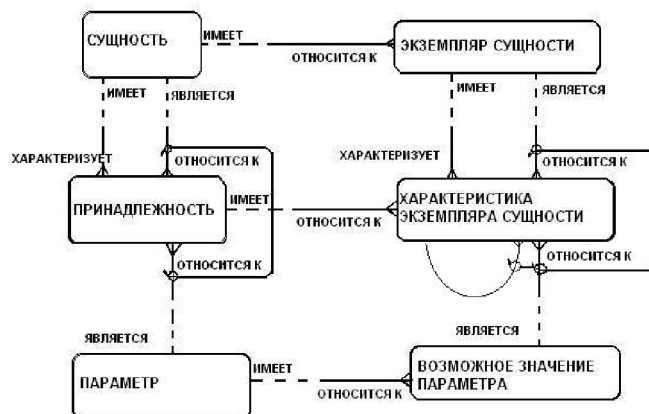


Рис. 1. Логическая модель простой УБД на основе универсального типа значений элемента кортежа

ривая сущности (таблицы) БД пользователя, описание которых располагается в сущности «Сущность», видим, что имеем дело с такими же объектами, как объекты предметной области пользователя, с таким же механизмом их описания. Поэтому первый шаг оптимизации УБД (рис. 1), состоит во внесении сущности «Сущность» в сущность «Характеристика экземпляра сущности», закрепив за ней первое значение первичного ключа, относящееся к первой сущности пользователя.

Далее естественным является и определение сущности «Параметр» как экземпляр сущности «Характеристика экземпляра сущности», для чего зарезервированы там второе и третье значение первичного ключа, относящиеся к соответствующим сущностям БД пользователя. Две сущности под описание параметров выделены из-за того, что перенесены сущности «Параметр», «Возможное значение параметра» и «Принадлежность» в сущность «Характеристика экземпляра сущности», туда попадают и атрибуты (поля) этих сущностей, которые являются служебными и определяют механизм работы с метаинформацией БД пользователя. Таким образом, при модернизации УБД очень полезно служебную информацию хранить отдельно от информации пользователя. что позволяет безболезненно, не затрагивая информацию пользователя расширять функционал СУБД УБД.

Сокращение сущности «Экземпляр сущности» не представляет большого труда, т.к. в этой сущности присутствует только поле первичного ключа, т.е. его кода, и кода внешнего ключа к сущности «Сущность». Поэтому при сохранении этих двух атрибутов в сущности «Характерис-

тика экземпляра сущности», сущность «Экземпляр сущности» теряет свою актуальность.

Таблица «Сущность» является ключевой и является не только составляющей схемы ядра УБД «сущность – экземпляр – характеристика», но и составляющей системы описания СБД. Поэтому Она может быть сокращена только после сущностей «Параметр», «Параметр сущности» и «Возможное значение».

Проведенные преобразования привели к изменению выражения единицы информации при описании экземпляра сущности в [5] и имеет вид:

$$\forall ue_{eu_e} \in Ue_e$$

$$\exists G_{eu_e} = \left\{ g_{eu_e 1}, g_{eu_e 2}, \dots, g_{eu_e g_{yu_e}}, \dots, g_{eu_e G_{yu_e}} \right\}, \quad (1)$$

а выражение элемента единицы информации [5] в

$$\forall g_{eu_e g_{yu_e}} \in G_{eu_e} \exists E_{eu_e g_{yu_e}} =$$

$$= \left\{ e_{eu_e g_{eu_e} 1}, e_{eu_e g_{eu_e} 2}, \dots, e_{eu_e g_{eu_e} i_{eu_e g_{yu_e}}}, \dots, e_{eu_e g_{eu_e} I_{eu_e g_{yu_e}}} \right\}, \quad (2)$$

где индексы:  $e$  – номер сущности (таблицы) БД пользователя,  $u$  – номер экземпляра сущности (строки таблицы),  $g$  – номер многострочной единицы информации кортежа,  $i$  – номер элемента многострочной единицы информации кортежа.

### ЛОГИЧЕСКАЯ МОДЕЛЬ

Таким образом, получаем следующую модель УБД с выделенной тяжёлой информацией в справочные сущности на основе синтеза метаинформации и данных представленную на рис. 2.

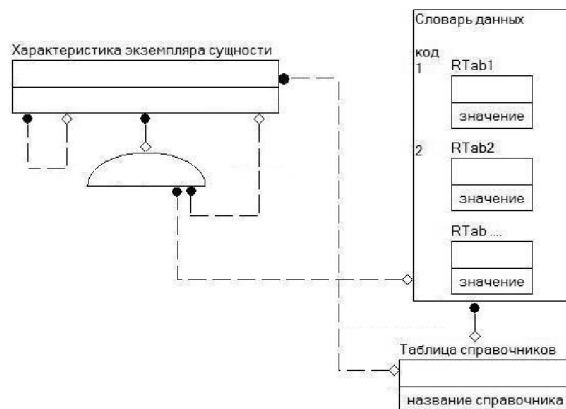


Рис. 2. Логическая модель УБД с выделенной тяжёлой информацией в справочные сущности на основе синтеза метаинформации и данных

Ядром МУОРБД на реляционной платформе является сущность «Характеристика экземпляра сущности», вся тяжёлая информация находится в сопутствующих справочниках. А связи между отдельными структурными элементами МУОРБД возложены на циклическую связь этой сущности самой на себя и различные вариации внешнего ключа.

Состав сущности «Характеристика экземпляра сущности» основан на структуре элемента единицы информации [5] и включает следующие атрибуты:

Атрибуты первичного ключа:

1. Код сущности
2. Код экземпляра сущности
3. Код единицы информации
4. Код элемента единицы информации

Атрибуты внешнего ключа атрибута сущности:

5. Код сущности атрибута
6. Код экземпляра сущности атрибута
7. Код единицы информации атрибута
8. Код элемента единицы информации атрибута

Атрибуты внешнего ключа значения:

9. Код сущности атрибута
10. Код экземпляра сущности атрибута
11. Код единицы информации атрибута
12. Код элемента единицы информации атрибута

Атрибуты внешнего ключа значения из справочника:

13. Код справочника
14. Код значения в справочнике

Атрибуты темпоральности:

15. Дата начала действия информации
16. Дата окончания действия информации

Атрибут поддержки древовидной структуры:

17. Код родительского элемента единицы информации

Атрибуты первичного ключа структурно понятны и соответствуют индексам элемента единицы информации (2), все они обязательны к заполнению и не могут принимать значение Null. Атрибуты внешнего ключа атрибута сущности так же обязательны к заполнению и не могут принимать значение Null. Атрибуты внешнего ключа значения не обязательны и могут принимать значение Null, т.к. значения вообще может и не быть, если информацией об атрибуте является его присутствие при описании эк-

земпляра сущности, либо значение внесено пользователем непосредственно и хранится в справочнике значений. Атрибуты внешнего ключа значения из справочника так же не обязательны, если информацией об атрибуте является его присутствие при описании экземпляра сущности, либо значением определяется через циклическую связь сущности «Характеристика экземпляра пользователя» самой к себе. На атрибуты внешнего ключа значения и атрибуты внешнего ключа значения из справочника наложено правило «исключающего или». Атрибуты темпоральности определяют временной диапазон действия информации. Атрибуты не обязательны и могут принимать значение Null. Атрибут поддержки древовидной структуры имеет целочисленное значение и используется, если необходимо установить подчинение между элементами внутри единицы информации.

## СВЯЗИ

Как уже говорилось, функции организации связей между отдельными структурными элементами МУОРБД в УБД возложены на различные модификации циклической связи сущности «Характеристика экземпляра сущности».

Связь к отдельному элементу единицы информации имеет вид:

$$e_{eu_e g_{u_e} i_{eu_e g_{u_e}}} \Rightarrow e_{e_R u_R g_R i_R} \quad (3)$$

Связь к единице информации в целом имеет вид:

$$e_{eu_e g_{u_e} i_{eu_e g_{u_e}}} \Rightarrow e_{e_R u_R g_R [\dots]} \quad (4)$$

Ссылка на экземпляр сущности определяется выражением:

$$e_{eu_e g_{u_e} i_{eu_e g_{u_e}}} \Rightarrow e_{e_R u_R [\dots][\dots]} \quad (5)$$

Т.е. внешний ключ имеет два первых значимых значения из четырёх, остальные пустые.

Так как описание СБД, которое выполнялось сущностями «Сущность», «Параметр» и «Параметр сущности», теперь ложится на сущность «Характеристика экземпляра сущности», то появляется необходимость и в четвёртой форме циклической связи:

$$e_{eu_e g_{u_e} i_{eu_e g_{u_e}}} \Rightarrow e_{e_R [\dots][\dots][\dots]} \quad (6)$$

в этом случае три последние составляющие имеют пустые значения. В этом случае опреде-

лена ссылка на сущность (таблицу) в целом, используется такого рода связь при установлении у одной сущности атрибута, имеющего значения – экземпляры другой сущности.

Универсальность физической модели данных позволяет автоматически развёртывать БД при первом запуске универсального приложения или программного продукта, в котором присутствует запрос автоматического развёртывания УБД.

Унификация и упрощение работы с данными БД пользователя, размещёнными в УБД, реализуется с помощью хранимых процедур и функций.

В УБД с выделенной тяжёлой информацией в справочные сущности на основе синтеза мета-информации и данных вторая сущность хранит служебные параметры, которыми описываются структурные элементы МУОРБД, и они обрабатываются комплексом хранимых процедур.

Таким образом, получаем, что после автоматического развёртывания УБД, необходимо автоматически заполнить первые две сущности. В первой загрузить описание трёх служебных сущностей. Во вторую сущность разместить служебные параметры. Т.к. описание служебных сущностей имеет и тяжёлую информацию, то частично заполняются и справочные таблицы.

## ХРАНИМЫЕ ПРОЦЕДУРЫ И ФУНКЦИИ

Работа с информацией УБД производится не непосредственно через поля таблиц УБД, а через функции, которые имеют универсальный функционал и хранятся в УБД.

В основной состав комплекса хранимых процедур сложной УБД на реляционной платформе входят:

1. *ObjectName* – функция возвращает строчное значение отображения экземпляра сущности по идентифицирующим атрибутам, определяемым наличием в единице информации, описывающей атрибут, элемента с параметром «Заголовок(в табл.форме)»;

2. *FieldVal\_un\_ess* – функция возвращает значение атрибута указанного экземпляра сущностей;

3. *Charact\_val* – функция возвращает значение указанной характеристики экземпляра сущности;

4. *ValueFromRefTab* – функция возвращает указанное значение из определённого справочника.

Кроме функций вывода информации БД пользователя, в надстройку СУБД входят и функции обработки информации словаря данных БД пользователя:

5. *NameEss* – функция возвращает наименование сущности по коду;

6. *ParametrName* – функция возвращает обозначение атрибута сущности;

7. *ParametrName\_exp* – функция возвращает подробное наименование атрибута сущности;

8. *ParametrName\_Full* – функция возвращает исходное наименование атрибута сущности;

9. *PN* – служебная функция возвращает значения для трёх предыдущих функций с указанием типа результата вывода;

10. *AdrVal* – функция возвращает адрес одной строкой.

Комплекс хранимых процедур УБД на реляционной платформе является универсальным, а соответственно его можно автоматически развёртывать с УБД при первом запуске универсального приложения или программного продукта, в котором присутствует запрос автоматического развёртывания УБД.

Рекурсивный вызов функций проводится в двух функциях *ObjectName* и *Charact\_val*.

Функция *ObjectName* собирает значения идентифицирующих атрибутов соответствующей сущности, определяемых наличием в единице информации, описывающей атрибут, элемента с параметром «Заголовок(в табл.форме)», в одно строчное значение отображения этого экземпляра сущности. Функция *Charact\_val* определяет значение по адресу характеристики экземпляра сущности, по которому могут находиться три вида информации: значение из справочников, значение по адресу другой характеристики или группы характеристик, определяющих единицу информации, и последнее – другой экземпляр сущности, отображением которого в свою очередь занимается функция *ObjectName*.

Глубина такой рекурсивной цепочки ничем не ограничена.

Зацикливание рекурсивной обработки автоматически обрабатывается и прекращается при первом появлении, при этом пользователю выводится уже сформированная часть результата с пометкой циклической ссылки.

Нужно заметить, что циклическая ссылка вполне может иметь место и в реальном информационном поле. Например, при описании

индивидов, атрибуты «Мать» и «Сын» звцикливают двух экземпляров сущности «Индивид».

### КОНЦЕПЦИЯ УНИВЕРСАЛЬНОГО ПРИЛОЖЕНИЯ СЛОЖНОЙ УБД

Для работы с информацией в сложной УБД разработано универсальное приложение, аналогичное универсальному приложению простой УБД.

В данной программе реализован универсальный инструментарий для работы со всеми таблицами пользователя в УБД, что не заставляет пользователя запоминать инструментарий каждой таблицы, вместе с тем для каждой таблицы может быть использован весь спектр инструментов этого приложения.

Универсальное приложение рассчитано на самый широкий диапазон пользователей от начинающего пользователя (домохозяйки) до разработчиков АИС на базе распределённых серверов.

В зависимости от уровня знаний пользователь может начать с самого простого: ввода и работы с табличной информацией в стандартном виде, не отвлекаясь на дополнительную информацию, а далее по мере развития углубляться до комплексного управления сразу множеством таблиц, характеристик и значений.

Кроме того, приложение представляется и в виде скомпилированных модулей, которые разработчик программных продуктов, может включить в свой разрабатываемый проект. После чего, он получает полный доступ к УБД и

инструментарий для работы с ней, которая развертывается автоматически при первом запуске приложения.

На данный момент интерфейс универсального приложения УБД разработан с целью исследования и максимальной демонстрацией функционирования системы управления УБД. Также он максимально адаптирован к обучению пользователей работы в УБД. Для достижения интерфейса уровня широкого пользования необходимо участие большого круга специалистов от дизайнеров до разработчиков крупномасштабных систем с широким спектром компьютерных технологий.

Программа состоит всего из двух основных форм пользователя, что не заставляет пользователя блуждать по формам в поисках нужной информации.

Среда универсального приложения УБД является Delphi. Что позволяет достаточно наглядно и просто продемонстрировать функционал УБД, а также на этой платформе просто создаются обучающие системы, как для самого обучения работы с УБД, так и по обучению созданию программных продуктов с включением универсального приложения.

Для подключения к серверу и БД используется технология ADO через udl-файл – посредник.

Таким образом, УБД может быть реализована на любой реляционной платформе, которая поддерживается ADO-технологией, а универсальное приложение подключено к ней после

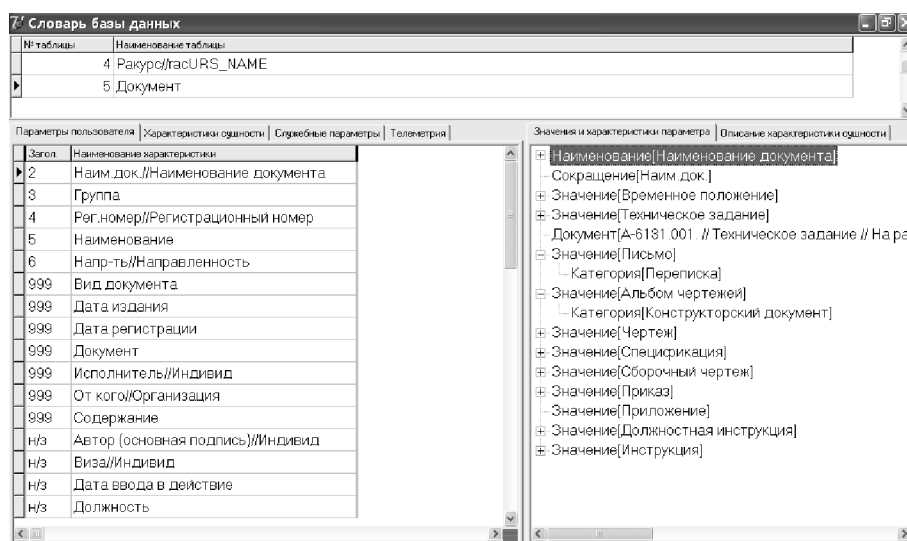


Рис. 3. Форма определения словаря пользователя в УБД с выделенной тяжёлой информацией в справочные сущности на основе синтеза метаинформации и данных

установления соответствующих параметров в udl-файле – посреднике.

### МОДУЛЬ ОПРЕДЕЛЕНИЯ СБД ПОЛЬЗОВАТЕЛЯ

Универсальное приложение состоит из двух форм: формы определения словаря УБД и главной универсальной формы отображения информации сущностей пользователя в УБД.

Форма определения СБД приведена на рис. 3.

Таблица отображения списка сущностей (таблиц) пользователя находится вверху формы. Наименования сущностей определяются функцией *NameEss*.

Запрос формирования сущностей пользователя строится на параметре «Наименования» при описании экземпляров первой сущности.

После выбора в верхней таблице нужной сущности заполняется левый нижний блок. Блок отображения характеристик сущности пользователя состоит из двух вкладок: параметры сущности, характеристики сущности.

В первом выводится список атрибутов сущности, во втором её описание. Таблица атрибутов сущности аналогична такой же в универсальном приложении простой УБД, но строится не на стационарной таблице, а на характеристиках первой сущности, со вторым параметром, соответствующим коду указанной сущности пользователя. Атрибут сущности определяется параметром «Допустимая характеристика».

На вкладке «Характеристики сущности» выведено всё описание сущности в древовидном виде.

Соответственно правый блок описания атрибута сущности пользователя на рис. 4 состоит из двух вкладок: значения и характеристики параметра, описание характеристики сущности.

На вкладке «Значения и характеристики параметра» отображаются в древовидном виде возможные значения и их дополнительное описание, а так же дополнительное описание самого параметра.

В случае если атрибут является сущностью, то отображается таблица, в которой выведены экземпляры этой сущности, как показано.

### РЕАЛИЗАЦИЯ СИНТЕЗА СБД И ДАННЫХ ПОЛЬЗОВАТЕЛЯ

В сложной УБД с выделенной тяжёлой информацией в справочные сущности на основе синтеза метаинформации и данных СБД хранится в одной структуре и представляет собой такие же данные пользователя, как и основная информация его БД. Таким образом, появляется возможность использовать сущности, их экземпляры, единицы информации и элементы единиц информации пользователя для описания СБД. Этот инструментарий имеет большие перспективы и заслуживает отдельного исследования.

Одним из механизмов управления СБД на основе данных пользователя в сложной УБД, появившимся в результате синтеза СБД пользователя и данных, является механизм управления системой обработки и отображения информации экземпляров сущности.

В традиционных СУБД СБД и данные разделены. И СБД диктует правила ввода информации основного содержания, но не наоборот.

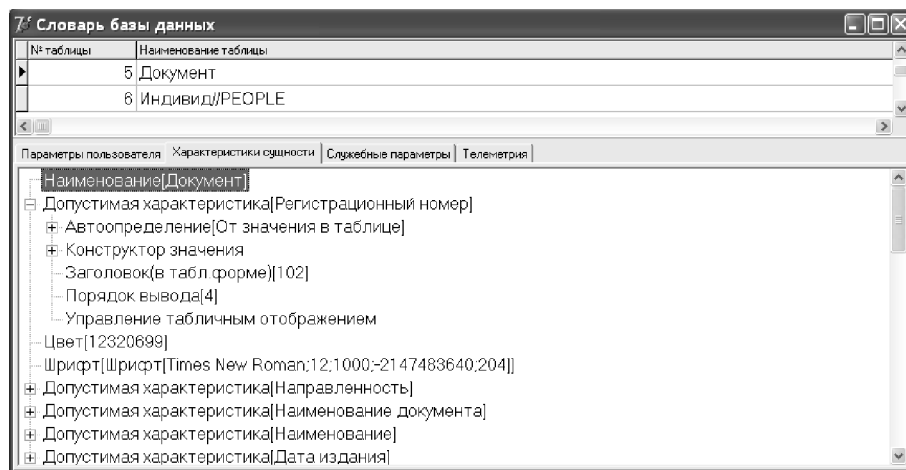


Рис. 4. Описание сущности пользователя

В сложной УБД появилась такая возможность. Это стало актуальным при необходимости разграничения правил обработки и отображения различных экземпляров одной сущности, учитывая их природные особенности, выраженные в их информации.

Например, если необходимо для документов с характеристикой «Тип документа» – «Письмо» и «Чертёж» установить разные возможные характеристики, то у характеристики «Тип документа» устанавливаем параметр «Управление табличным отображением», а у варьируемых характеристик (например, «Отправитель» и «Получатель») устанавливаем дополнительное описание «Тип документа» – «Письмо». В этом случае при включенном фильтре с параметром «Тип документа» и значением «Письмо» характеристики «Отправитель» и «Получатель» будут видны и доступны, а при значении «Чертёж» – нет.

Для запуска механизма атрибуты управления системой обработки и отображения информации экземпляров сущности в кортеже (-ах) атрибута, которому (-ым) устанавливаются в подчинение остальные атрибуты, добавляется элемент с параметром «Управление табличным отображением», т.е. устанавливается связь в подчиненном элементе кортежа первому элементу.

После этого система управления МУОРБД позволяет устанавливать в кортежах остальных атрибутов ссылку на экземпляр сущности, к которой установлена связь через этот атрибут, если такая существует, либо на возможное зна-

чение атрибута, приписанное к соответствующему экземпляру сущности «Параметр пользователя».

Результат такой системы приведён на рис. 5 для сущности «Документ» из АИС документооборота, где исследовались возможности работы с разнородными документами в одной сущности. Основной объём экземпляров в сущности «Документ» были конструкторские документы: чертежи, спецификации, сборочные чертежи, альбомы чертежей. Помимо них там содержались и письма, и приказы, и разного типа инструкции и многое другое. Документы очень разнотипные и поэтому естественным является установить разные правила обработки и отображения информации для разных категорий документов. Управляющими стали большая совокупность атрибутов, здесь и «Категория», и «Наименование документа», и «Регистрационный номер», и «Параметр чертежа» и многие другие.

Как видно из рис. 5 для категории со значением «Конструкторский документ» определён возможный параметр «Регистрационный номер», которому в свою очередь подчинён атрибут «Параметр чертежа», а ему в свою очередь атрибут «Значение». Два последних атрибута предназначены для описания групповых чертежей, в которых несколько регистрационных номеров, каждый из которых характеризуется своими обособленными значениями некоторых параметров чертежа.

Атрибут «Наименование документа» подчинён параметру «Категория», а также являет-

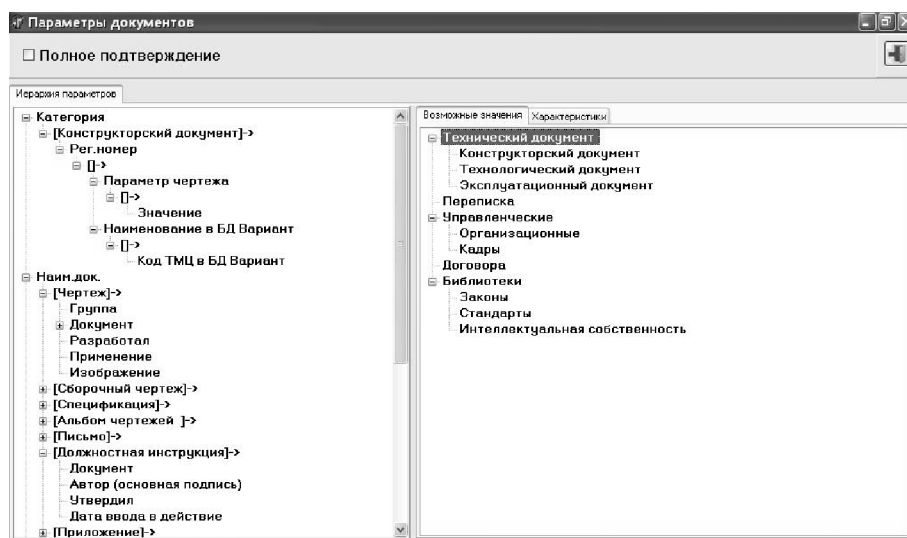


Рис. 5. Иерархия атрибутов и их возможных значений сущности «Документ»



ся сам управляющим атрибутом. В левом древовидном представлении показано, что экземпляры сущности «Документ» с разными значениями атрибута «Наименование документа» имеют разный список доступных атрибутов для их описания. Так для чертежей кроме общих атрибутов, доступны атрибуты: «Группа», «Документ», «Разработал», «Применение» и «Изображение», а для должностных инструкций: «Документ», «Автор(основная подпись)», «Утвердил» и «Дата ввода в действие».

Возможные значения атрибута «Наименование документа» также распределены в подчинении различных значений атрибута «Категория», причём они в свою очередь также имеют иерархическую зависимость между собой. Так, например, для значения атрибута «Категория» – «Технический документ» подчинены значения атрибута «Категория»: «Конструкторская документация» и «Технологическая документация». К техническим документам относится значение атрибута «Наименование документа» – «Техническое задание», а к конструкторским документам: «Альбом чертежей», «Чертёж», «Спецификация», «Сборочный чертёж», к технологическим документам отнесено значение «Инструкция».

Можно установить одно значение подчиненного атрибута и нескольким значениям управляющего атрибута.

Так, например, значение атрибута «Наименование документа» – «Приказ» может быть

установлено и для документов категории «Организационные» и для документов категории «Кадры».

Таким образом, новые возможности сложной УБД, вытекающие из синтеза СБД и основных данных позволяет заменить десятки сущностей реляционной БД, при этом сохранив все требования к структуризации информации предметной области.

### ГЛАВНАЯ УНИВЕРСАЛЬНАЯ ФОРМА ОТОБРАЖЕНИЯ ИНФОРМАЦИИ СУЩНОСТЕЙ ПОЛЬЗОВАТЕЛЯ

Главная универсальная форма отображения информации сущностей пользователя в сложной УБД позволяет в одном месте производить все манипуляции с информацией пользователя УБД. Её интерфейс представлен на рис. 6.

Главная универсальная форма отображения информации сущностей пользователя в простой УБД состоит из трёх блоков предназначенных для отображения сущностей пользователя (верхняя таблица), экземпляров сущностей (левая таблица) и описания экземпляров сущностей правая древовидная структура. А так же под правым блоком выведено уточняющее поле информации в её выделенной строке, для более подробного и наглядного отображения.

Верхняя таблица, так же как и в форме описания словаря УБД, отображает список сущностей (таблиц) пользователя. Единственное отличие в том, что оно не имеет контекстного

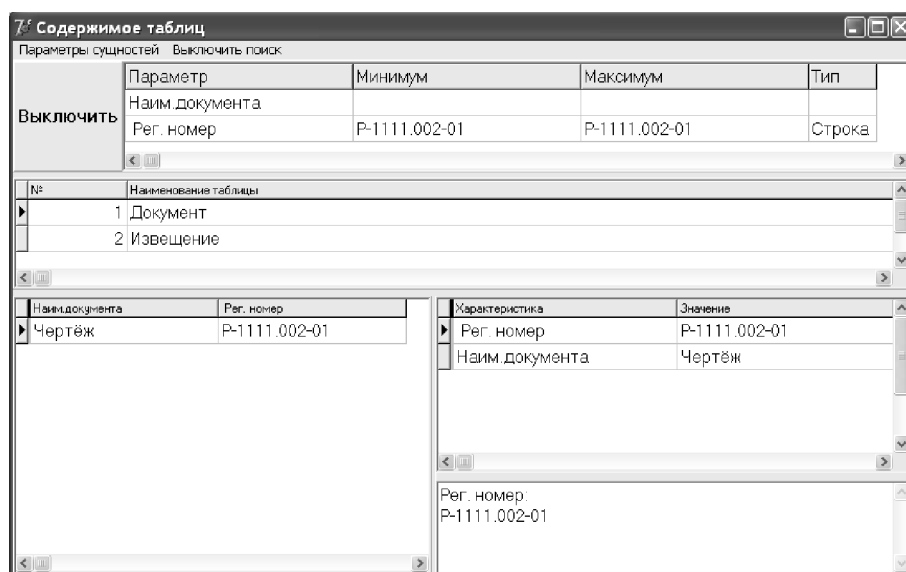


Рис. 6. Главная универсальная форма отображения информации сущностей пользователя в сложной УБД

меню, и вся работа с информацией этого списка проводится в форме описания словаря УБД.

Таблица параметров поиска на панели, вверху формы рис. 6, содержит четыре поля. В первом выведены обозначения атрибутов, во втором и третьем минимальные и максимальные значения для границ поиска по атрибутам. В четвёртом поле устанавливается тип данных. Оно заполняется автоматически и может быть исправлено пользователем. Возможные значения четвёртого поля: «Строка», «Число» и «Дата».

Система обработки информации может быть оптимизирована, расширена и доработана. Важным выводом является то, что все функции работы с информацией пользователя полностью доступны, просты и не зависят от структуры СБД пользователя, т.е. полностью унифицированы.

### ЗАКЛЮЧЕНИЕ

Переход от уже действующих баз данных на основу МУБД может быть осуществлён и автоматически. При этом первое и пятое измерение МУБД будут равны единице, а в шестом массиве будут использованы только нулевой и первый элементы, соответствующие наименованию атрибута и значению по этому атрибуту.

Верхнее измерение в массиве МУБД позволяет одновременную работу с несколькими базами данных и открывает возможности разработки распределённых баз данных.

Важным достоинством универсального приложения является лёгкость и простота, это может быть использовано и для бытового использования взамен программных продуктов Excel и Access при хранении структурированной информации.

### СПИСОК ЛИТЕРАТУРЫ

1. Хакен Г. Информация и самоорганизация: Макроскопический подход к сложным системам. М.: Мир, 1991.–240 с.

**Микляев И. А.** – доцент кафедры информационных технологий и систем. Институт судостроения и морской арктической техники филиала Северного арктического федерального университета. E-mail: ivanmia1@rambler.ru

2. Аршинов В. И. Синергетика как феномен постнеклассической науки. М.:ИФРАН 1999.

3. Конноли Т., Бегг К. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. 3-е издание.:Пер. с англ. / Т. Конноли, К. Бегг. М.: Издательский дом «Вильямс», 2003. С. 1440

4. Микляев И. А., Черткова О. В. Синергетическое информационное пространство МУОРБД // Материалы V Международной научно-практической конференции «ОБЪЕКТНЫЕ СИСТЕМЫ – 2011» (Зимняя сессия) С. 67–72.

5. Микляев И. А. Формализованное описание основной структуры представления данных матричной универсальной объектно-реляционной базы данных // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. 2011. № 1., Астр., С. 61–68.

6. Микляев И. А. Концепция разработки матричной универсальной базы данных с поддержкой древовидной структуры единицы информации и её универсального приложения // Вестник Воронежского государственного университета. Серия: Системный анализ и информационные технологии. 2010. № 2., Воронеж, С. 101–108.

7. Микляев И. А., Ундозерова А. Н., Кудяева М. В. Разработка информационных систем на основе модели универсальной базы данных // Автоматизация и современные технологии. 2011. № 05., М., С. 38–41.

8. Микляев И. А. Область применения матричного объектно-реляционного сервера баз данных // Материалы науч.-техн. конф. «Информационные технологии в исследовании Северных и Арктических территорий», Сев.(Арктич.) федер.ун-т. – Архангельск: ИПЦ САФУ, 2012, С. 41–51.

9. Микляев И. А., Ундозерова А. Н., Кудяева М. В. Универсальная логическая модель базы данных. // Вестник Поморского университета, серия «Естественные науки». – 2010. – № 1. Арх.: ПГУ – С. 93–98.

10. Банников Н. А. Объектно-ориентированные базы данных // <http://www.stikriz.narod.ru/art/oobd.htm>

**Miklyev I. A.** – Associate Professor, Department of “Information Technology and Systems” of the Institute of Shipbuilding and marine engineering of the Arctic branch Northern Arctic Federal University. E-mail: ivanmia1@rambler.ru