

МЕТОД ПРЕОБРАЗОВАНИЯ УНАСЛЕДОВАННЫХ БАЗ ДАННЫХ В БАЗУ ДАННЫХ С УНИВЕРСАЛЬНЫМ БАЗИСОМ ОТНОШЕНИЙ

В. И. ЕСИН, В. В. ВИЛИГУРА

Предлагается метод преобразования унаследованных баз данных в базу данных с универсальным базисом отношений, позволяющий расширить функциональность унаследованных БД, улучшить определенные их характеристики качества, в том числе, способность к адаптации в условиях динамичных изменений предметных областей, способствующий снижению стоимости сопровождения баз данных.

Ключевые слова: база данных, унаследованная база данных, база данных с универсальным базисом отношений.

ПОСТАНОВКА ЗАДАЧИ

Устаревают платформы систем управления базами данных (СУБД), растет количество разрозненных источников информации, данные которых продолжают иметь значимую практическую ценность, но их изолированность, несогласованность, различное представление мешает связывать и анализировать содержащуюся в них информацию. Все это в целом на фоне возрастающих требований к поддерживаемой функциональности, качеству действующих баз данных (БД) информационных систем (ИС) приводит к необходимости адаптации ранее созданных и функционирующих БД и программ работы с ними к новым языкам, операционным средам и платформам. То есть ведет к росту востребованности проектов модернизации, интеграции, замены существующих систем – реинжинирингу действующих систем и их основного компонента – БД.

При этом условия, в которых приходится осуществлять реинжиниринг БД, характеризуются такими особенностями как: а) априорной неопределенностью в периодичности потребности данных различных типов, структур; б) жесткостью требований к значениям обеспечиваемых характеристик надежности, безопасности хранения и доступности использования актуальных, согласованных больших объемов структурированных и неструктурированных данных из существенно отличающихся ПрО; в) ограниченностью отводимых временных и финансовых ресурсов. Организации обычно расходуют (20-40) % своего ИТ-бюджета на эволюционирование данных путем миграции (изменение местоположения), очистки или преобразования (изменения формы или структуры данных) [1].

В сложившейся ситуации продолжает иметь место объективная потребность в исследовании и переосмыслении существующих подходов, методологий и технологий реинжиниринга ИС, актуализирующая задачи: разработки целостных методологий реинжиниринга баз данных ИС, механизмов адаптации мето-

дологий реинжиниринга БД в реальных проектах; инструментальных средств, обеспечивающих комплексное решение задач по реинжинирингу БД; оценки качества создаваемых в процессе реинжиниринга БД ИС.

Опираясь на результаты проведенного анализа: различных информационных технологий (ИТ) управления данными, интеграции данных, реинжиниринга информационных систем, а также тенденций их развития; классических методов проектирования баз данных и, в первую очередь, реляционных; различных моделей данных, используемых при моделировании предметных областей, учитывая требования, предъявляемые к корпоративным БД ИС, на основании созданных моделей [2–7] и схемы БД, инвариантной к предметным областям [8], была разработана информационная технология, обеспечивающая механизм адаптируемости БД к изменениям условий функционирования. Она, как совокупность методов, в основу которых были положены обоснованные и разработанные модель «объект-событие» [2–5], модель данных с универсальным базисом отношений [6, 7], инвариантная к предметным областям (ПрО) схема БД [8], а также специально созданного программного обеспечения, не привязана к конкретным аппаратным платформам, хотя некоторые ее реализации связаны с определенными программными системами.

Один из методов данной технологии – метод преобразования унаследованных баз данных, ведущий к замещению одной БД другой, построенной на основе инвариантной к ПрО схемы [8], предложен ниже.

МЕТОД ПРЕОБРАЗОВАНИЯ УНАСЛЕДОВАННЫХ БАЗ ДАННЫХ

Основные технологические операции данного метода, связываемые с соответствующими этапами преобразования унаследованных баз данных, представлены на рис. 1 в виде функциональных блоков методологии моделирования IDEFO.

Рассмотрим их для каждого этапа более подробно.

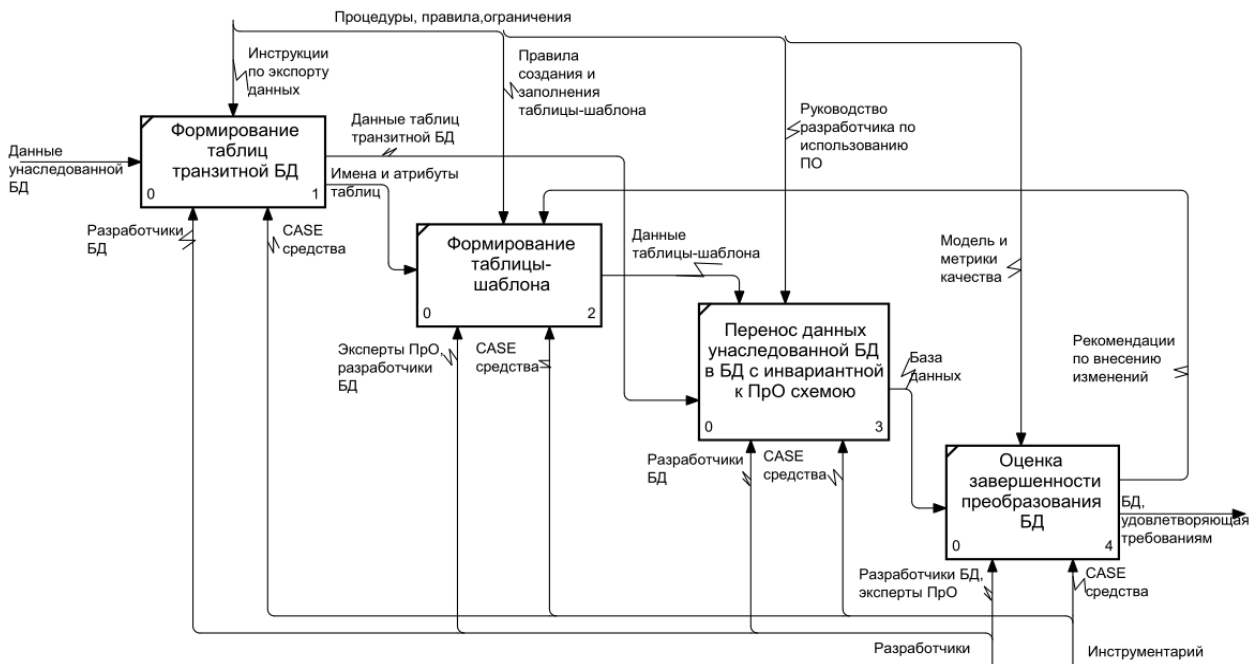


Рис. 1. Представление основных технологических операций метода

Этап 1. Формирование таблиц транзитной БД.

Данные унаследованной БД экспортируются во временные таблицы, так называемой, транзитной (временной) БД, реализованной на платформе реляционной СУБД, такой как: Oracle, PostgreSQL, Access или другой, совместимой с ODBC. Перенос данных между различными СУБД можно осуществить, например, как описано в работе [9], используя либо специальный драйвер (интерфейс), в частности ODBC, OLE DB (в этом случае две СУБД соединяются друг с другом и непосредственно передают данные), либо транзитные файлы (как правило, формата dbf, т. к. большинство СУБД, формат хранения данных которых отличается от формата dbf, снабжены утилитами или драйверами, которые позволяют перенести данные в этот формат), в которые копируются данные из унаследованной БД для последующего переноса в транзитную БД. При этом если унаследованная СУБД не является реляционной, то данные транзитных файлов должны быть приведены к табличному виду и первой нормальной форме.

Этап 2. Формирование таблицы-шаблона.

В транзитной БД также создается, так называемая, таблица-шаблон, заголовок которой формально можно представить следующим образом:

$$H_p = \langle A_{id}, A_{tt}, A_{op}, A_v \rangle \quad (1)$$

Основными компонентами кортежа (1) являются:

- A_{id} – первичный ключ временной таблицы;
- $A_{tt} = \{a_{tt}^1, a_{tt}^2, a_{tt}^3\}$ – множество имен атрибутов, значениями которых являются: имена временных таблиц транзитной БД (a_{tt}^1 = "преобразуемая таблица"),

в которые были перенесены данные из унаследованной БД; имена атрибутов (a_{tt}^2 = "атрибут") в соответствующих временных таблицах; определенные конструкции языка модели данных (ЯМД) [10] (a_{tt}^3 = "конструкция ЯМД"), заканчивающиеся знаком равно, которым может быть присвоено значение атрибута a_{tt}^2 соответствующей строки временной таблицы;

– $A_{op} = \{a_{op}^1, a_{op}^2, a_{op}^3, a_{op}^4, a_{op}^5\}$ – множество имен атрибутов, значениями которых являются конструкции ЯМД: для a_{op}^1 = "оператор РазделВлад": конструкция ЯМД – "<Раздел>=" или последовательность операторов ЯМД (в случае иерархии разделов в моделируемой ПрО) обязательно заканчивающаяся конструкцией – "<Раздел>="; для a_{op}^2 = "оператор ЭкзОВлад": конструкция ЯМД "<ЭкзО>=" или последовательность операторов ЯМД (в случае иерархии экземпляров объектов), обязательно заканчивающаяся конструкцией – "<ЭкзО>="; для a_{op}^3 = "оператор КлассОЭкзОВл": конструкция ЯМД – "<КлассО>=" или последовательность операторов ЯМД (в случае иерархии классов объектов), обязательно заканчивающаяся конструкцией – "<КлассО>="; для a_{op}^4 = "оператор ТипОЭкзОВл": конструкция ЯМД – "<ТипО>=" или последовательность операторов ЯМД, обязательно заканчивающаяся конструкцией – "<ТипО>="; конкретный (из числа допустимых) оператор ЯМД (без указания значения после знака равенства, в соответствии с синтаксисом ЯМД) или после-

довательность операторов ЯМД (для $a_{op}^5 = \emptyset \cup \{a_v^{51}, \dots\}$, если $a_{op}^5 \neq \emptyset$, то в зависимости от необходимости, каждому элементу множества a_{op}^5 будет присвоено имя следующего формата: «оператор n», где $n=1, \dots$, например, $a_{op}^{51} = \text{"оператор 1"}$ и т. д.);

– $A_v = \{a_v^1, a_v^2, a_v^3, a_v^4, a_v^5\}$ – множество имен атрибутов, значениями которых являются: для $a_v^1 = \text{"Раздел Влад"}$: имя раздела, присваиваемое значению атрибута a_{op}^1 соответствующей строки таблицы-шаблона, которое может задаваться константой или выбираться как из строк временной таблицы транзитной БД в соответствии с указанным атрибутом, так и собственно из формируемой БД (в которую переносятся данные); для $a_v^2 = \text{"ЭкзОВлад"}$: имя экземпляра объекта, присваиваемое значению атрибута a_{op}^2 соответствующей строки таблицы-шаблона, которое также может задаваться константой или выбираться как из строк временной таблицы транзитной БД в соответствии с указанным атрибутом, так и собственно из формируемой БД; для $a_v^3 = \text{"КлассОЭкзОВлад"}$: имя класса объекта-владельца, присваиваемое значению атрибута a_{op}^3 соответствующей строки таблицы-шаблона, которое задается или выбирается аналогично рассмотренным выше для a_v^1, a_v^2 именам; для $a_v^4 = \text{"ТипОЭкзОВл"}$: имя типа объекта-владельца, присваиваемое значению атрибута a_{op}^4 соответствующей строки таблицы-шаблона, которое также задается или выбирается аналогично рассмотренным выше для a_v^1, a_v^2, a_v^3 именам; имя, присваиваемое последнему оператору конструкции ЯМД a_{op}^5 соответствующей строки таблицы-шаблона (для $a_v^5 = \emptyset \cup \{a_v^{51}, \dots\}$, $(a_{op}^5 = \emptyset) \Rightarrow (a_v^5 = \emptyset)$, если $a_{op}^5 \neq \emptyset$, то в зависимости от количества элементов a_{op}^5 аналогичную мощность будет иметь множество a_v^5 ($|a_v^5| = |a_{op}^5|$), и в этом случае каждому элементу множества a_v^5 будет присвоено имя следующего формата: «Значение n», где $n=1, \dots$, например, $a_v^{51} = \text{"Значение 1"}$ и т. д.).

Заполнение таблицы-шаблона осуществляется в соответствии с синтаксисом ЯМД и следующими правилами.

При составлении конструкций, помещаемых в таблицу-шаблон, следует учитывать особенности их формирования, а именно те, которые связаны с добавлением определенных служебных символов в стан-

дартные конструкции операторов строк метаописаний ЯМД. Рассмотрим их более подробно.

Символ \$, указанный первым в строке атрибутов, начинающихся словом «оператор» таблицы-шаблона ($A_{op} = \{a_{op}^1, a_{op}^2, a_{op}^3, a_{op}^4, a_{op}^5\}$), свидетельствует о том, что информация, обрабатываемой строки, используется для формирования строки метаописания. При этом за символом \$ обязательно должен следовать необходимый оператор ЯМД (любые другие символы будут игнорироваться) и последним ключевым (служебным) словом в этой синтаксической конструкции должно быть одно из следующих: <КлассО>, <КлассС>, <ЭкзО>, <Папка>, <Документ>.

Символ \$, указанный первым в строке атрибутов, начинающихся словом «значение» таблицы-шаблона (a_v^5), свидетельствует о том, что информация о значении, обрабатываемой строки, будет выбираться из формируемой БД, построенной на основе инвариантной к ПрО схемы. При этом следует учитывать, что исполнение такой синтаксической конструкции в некоторых случаях может вызвать исключительную ситуацию – выбор нескольких значений. Поэтому использовать эту возможность необходимо в обоснованных случаях, в которых заранее известно, что они не вызовут такую ситуацию.

Символ #, указанный первым в строке атрибутов $A_v = \{a_v^1, a_v^2, a_v^3, a_v^4, a_v^5\}$, или указанный внутри строки метаописания, определенного символом \$ в строке атрибутов $A_{op} = \{a_{op}^1, a_{op}^2, a_{op}^3, a_{op}^4, a_{op}^5\}$ таблицы-шаблона, свидетельствует о том, что данные выбираются из соответствующего атрибута (имя атрибута определяется идентификатором, следующим за символом #) указанной временной таблицы транзитной БД.

Пример сформированной таблицы-шаблона приведен в таблице 1.

Этап 3. Перенос данных унаследованных БД в БД с инвариантной к ПрО схемой.

На основании содержания таблицы-шаблона (данных ее строк и столбцов) и временных таблиц транзитной БД, специально разработанное приложение в соответствии с алгоритмом преобразования, блок-схема которого представлена на рис. 2, формирует строки метаописания на ЯМД, которые затем и исполняет.

В результате таких действий данные, ранее хранящихся в исходной БД и перемещенные во временные таблицы транзитной БД, помещаются в БД, построенную на основе инвариантной к ПрО схемы, в виде определенных метаданных и данных моделируемой ПрО. При этом следует отметить, что возникающие при традиционном способе миграции унаследованных систем осложнения, вызванные различием логических структур данных, систем кодирования

Таблица 1

Пример таблицы-шаблона

ID	Преобразуемая таблица	Атрибут	Конструкция ЯМД	Оператор Раздел Влад	Раздел Влад	оператор ЭкзОВлад	ЭкзОВлад	Оператор КлассОЭкзОВл	КлассО ЭкзОВлад	Оператор ТипО ЭкзОВл	ТипО ЭкзОВл	Оператор 1	Значение 1
1	Vehicle classifier	Код типа техники	<ЗначХО>=	<Раздел>=	UGES	<ЭкзО>=	#Наименование_типа_техники	<КлассО>=	классификация техники	<ТипО>=	-	<ТипХО>[] (строковая, любая, фактическая)=	Код типа техники
2	Vehicle classifier	Код подтипа техники	<ЗначХО>=	<Раздел>=	UGES	<ЭкзО>=	#Наименование_типа_техники	<КлассО>=	классификация техники	<ТипО>=	-	<КлассО>=	Наименование подтипа техники
3	Vehicle classifier	Код наименования техники	<ЗначХО>=	<Раздел>=	UGES	<ЭкзО>=	#Наименование_подтипа_техники	\$<КлассО>= классификация техники; <ТипО>=; <ЭкзО>= # Наименование_типа_техники ; / <КлассО>=	Наименование подтипа техники	<ТипО>=	-		
4	Motor	Марка двигателя		<Раздел>=	UGES			<КлассО>=	Двигатель	<ТипО>=	#Марка двигателя		
5	Driver	Номер водительских прав	<ЗначХС>=	<Раздел>=	UGES	<ЭкзО>=	#ФИО водителя	<КлассО>=	Водитель	<ТипО>=	-	\$<ВремяНС>=# Дата получения водительских прав; <ТипХС>[] (строковая, любая, номер водительских прав, <КлассО>=	Получение водительских прав
6	Driver	Категория прав	<ЗначХС>=	<Раздел>=	UGES	<ЭкзО>=	#ФИО водителя	<КлассО>=	Водитель	<ТипО>=	-	\$<ВремяНС>=# Дата получения водительских прав; <ТипХС>[] (строковая, списочная, 1)= Категория водительских прав; <КлассО>=	Получение водительских прав
...

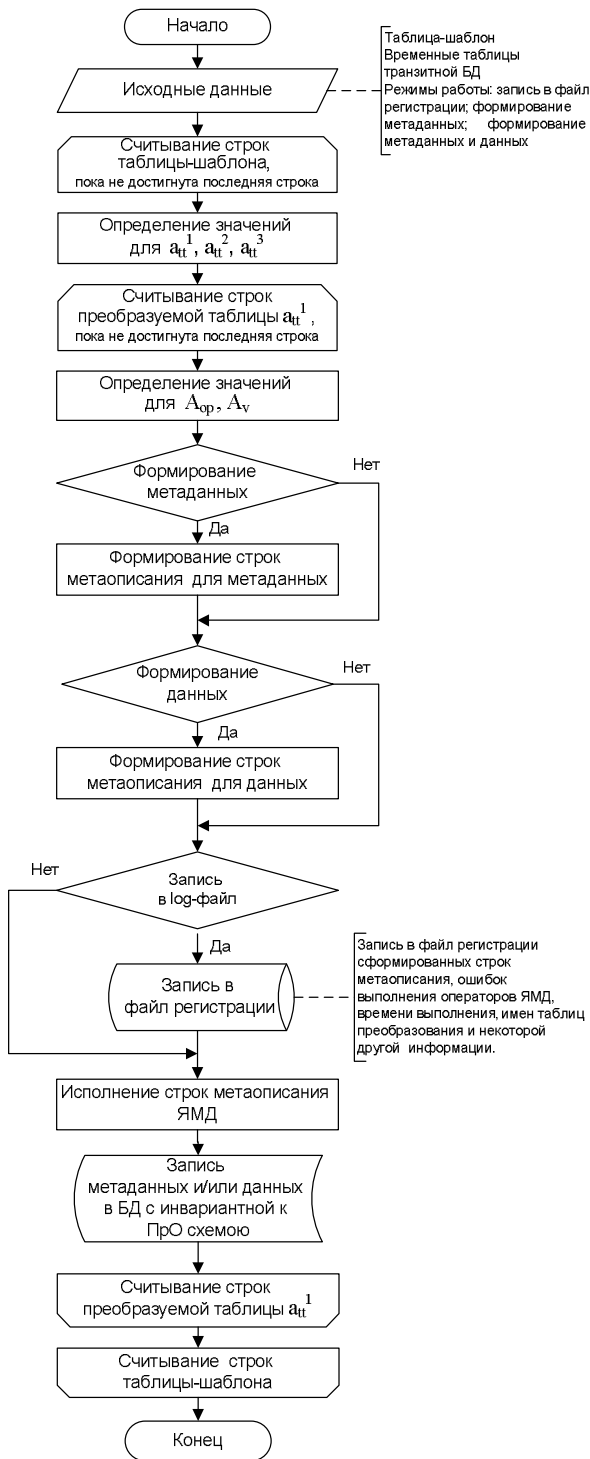


Рис. 2. Блок-схема алгоритма преобразования данных унаследованной БД в БД с инвариантной к ПРО схемой

информации, типов СУБД, сложностью автоматического выполнения переноса таких объектов и конструкций, как функции, триггеры, хранимые процедуры и т. д., в рассматриваемом методе отсутствуют благодаря использованию схемы БД с универсальным базисом отношений.

Этап 4. Оценка завершенности преобразования БД.

Решение о завершении процесса преобразования унаследованной БД, в БД, построенную на основе инвариантной к ПРО схемы, принимается на основе

сравнительного анализа значений атрибутов качества созданной БД, полученных с использованием метрик модели качества [11]:

$$Q_{DB} = \{H_i^{DB}, S_{ij}^{DB}, M_{jk}^{DB(i)}, At_{jl}^{DB(i)}\}, \quad (1)$$

где H_i^{DB} – i -я характеристика качества БД ($i=1, \dots, I$); S_{ij}^{DB} – j -я подхарактеристика ($j=1, \dots, J$) i -й характеристики качества; $M_{jk}^{DB(i)}$ – k -я метрика ($k=1, \dots, K$) j -й подхарактеристики i -й характеристики качества; $At_{jl}^{DB(i)}$ – l -й атрибут ($l=1, \dots, L$); j -й подхарактеристики i -й характеристики качества – переменная, которой присваивается значение в результате измерения (применения метрики), $At_{jl}^{DB(i)} \in Z$, и требований, предъявляемых к этим атрибутам со стороны потребителя информационного продукта. В случае неудовлетворительных значений атрибутов качества БД формулируется набор рекомендаций по внесению соответствующих изменений, и этапы 2–4 повторяются.

В результате, предлагаемый метод обеспечивает комплексное решение задачи преобразования унаследованных БД. Ценность такого решения заключается в том, что оно способствует расширению функциональности унаследованных БД, снижению стоимости их сопровождения, улучшению определенных характеристик качества, в том числе способности к адаптации в условиях динамичных изменений Про, как результат использования в качестве новой БД базы данных, построенной на основе схемы, инвариантной к предметным областям.

Данный метод был апробирован на нескольких БД для газовой промышленности, систем бухгалтерского учета.

ВЫВОДЫ

В данной научной статье предложен метод преобразования унаследованных баз данных в базу данных с универсальным базисом отношений, обеспечивающий комплексное решение задачи модернизации унаследованных БД. Разработанный метод позволяет расширить функциональность унаследованных БД, улучшить определенные их характеристики качества, в том числе, способность к адаптации в условиях динамичных изменений предметных областей, способствует снижению стоимости сопровождения БД ИС. Благодаря разработанному методу становится возможным использование имеющих значимую практическую ценность данных унаследованных БД в системе, основанной на новой технологии.

Литература

- [1] Measuring data management practice maturity: a community's self-assessment / [P. Aiken, M. D. Allen, B. Parker, A. Mattia] // IEEE Computer Society. – 2007. – V. 40. – №. 4. – С. 42–50.

- [2] *Есин В. И.* Семантическая модель данных «объект-событие» / В. И. Есин // Вісник Харківського національного університету імені В. Н. Каразіна. Сер.: Математичне моделювання. Інформаційні технології. Автоматизовані системи управління. – 2010. – № 925. – С. 65–73.
- [3] Security and noise immunity of telecommunication systems: new solutions to the codes and signals design problem. Collective monograph. – Minden, Nevada, USA : ASC Academic Publishing. – 2017. – 198 p. (Yesin V. I., Yesina M. V. Chapter 8, Means for conceptual modeling of information system databases, P. 160–196).
- [4] *Есин В. И.* Модель данных «объект-событие»: требования и синтез модели // Computer science and cyber security – International electronic scientific journal. – 2017. – Issue. 3 (7). – P. 33–44, <https://periodicals.karazin.ua/cscs/article/view/10003>, last accessed 2018/07/28.
- [5] *Есин В. И.* Выразительные средства модели данных «объект-событие» / В. И. Есин // Радиотехника: Всеукр. межвед. науч.-техн. сб. – 2017. Вып. 191. – С. 99–112.
- [6] *Есин В. И.* Универсальная модель данных и ее математические основы / В. И. Есин // Системи обробки інформації. – 2011. – № 2(92). – С. 21–24.
- [7] *Есин В. И.* Модель данных с универсальной фиксированной структурой / В. И. Есин // Теоретичні та прикладні аспекти побудови програмних систем : матеріали міжнародної наукової конференції, м. Київ, 15–17 грудня 2014 р. – Кіровоград : ФО-П Александрова М. В., 2014. – С. 112–116.
- [8] *Есин В. И.* Инвариантная к предметным областям схема базы данных и ее отличительные особенности / В. И. Есин // Радиотехника: науч.-техн. журнал. – 2018. Вып. 193. – С. 133–142.
- [9] *Лаврищева Е. М.* Методы и средства инженерии программного обеспечения // Е. М. Лаврищева, В. А. Петрухин. – М. : МОН РФ, 2007. – 415 с.
- [10] *Есин В. И.* Язык для универсальной модели данных / В. И. Есин, М. В. Есина // Системи обробки інформації. – 2011. – № 5(95). – С. 193–197.
- [11] *Yesin V. I.* A cybernetic approach to solving the problem of database reengineering // Telecommunications and Radio Engineering. – 2018. Volume 77, Issue 5. – P. 399–409. doi: 10.1615/TelecomRadEng.v77.i5.40.

Надійшла до редколегії 25.12.2018



Есин Виталий Иванович, доктор технических наук, доцент, профессор кафедры безопасности информационных систем и технологий ХНУ имени В. Н. Каразина. Область научных интересов – модели и методы разработки баз данных информационных систем и обеспечение их безопасности.



Вилигура Владислав Викторович, аспирант факультета компьютерных наук ХНУ имени В. Н. Каразина. Область научных интересов – базы данных и обеспечение их безопасности.

УДК 004.652 : 004.658.3

Єсін В. І. Метод перетворення успадкованих баз даних у базу даних з універсальним базисом відношень / В. І. Єсін, В. В. Вилигура // Прикладна радіоелектроніка: наук. – техн. журнал. – 2018. – Том 17, № 3, 4. – С. 176–181.

Пропонується метод перетворення успадкованих баз даних в базу даних з універсальним базисом відношень, що дозволяє розширити функціональність успадкованих БД, поліпшити певні їхні характеристики якості, в тому числі, здатність до адаптації в умовах динамічних змін предметних областей, що сприяє зниженню вартості супроводу баз даних.

Ключові слова: база даних, успадкована база даних, база даних з універсальним базисом відношень.

Табл.: 01. Іл.: 02. Бібліогр.: 11 найм.

UDC 004.652 : 004.658.3

Yesin V. I. **The method of converting legacy databases into a database with an universal basis of relations** / V.I. Yesin, V.V. Vilihura // Applied Radio Electronics: Sci. Journ. – 2018. – Vol. 17, № 3, 4. – P. 176–181.

The method of converting legacy databases into a database with an universal basis of relations is proposed, which enables to extend the functionality of legacy databases, improve their certain quality characteristics, including the ability to adapt in the conditions of dynamic changes in subject domains, which contributes to reducing the cost of maintaining databases.

Keywords: database, legacy database, the database with an universal basis of relations.

Tab. 01. Fig. 02. Ref.: 11 items.