

УДК 044.03

С.В. Штангей

Харьковский национальный университет радиоэлектроники, Харьков

МЕТОД АВТОМАТИЗИРОВАННОГО СИНТЕЗА СХЕМЫ ХРАНИЛИЩА ДАННЫХ ДЛЯ DESKTOP-OLAP ПРИЛОЖЕНИЙ

В работе сформулирована научно-прикладная задача создания специализированных методов и средств автоматизации синтеза схем хранилищ данных, которые могут быть положены в основу специализированных Desktop-OLAP пакетов, ориентированных на эксплуатацию в условиях Украины. В статье предлагается метод автоматизированного синтеза схемы хранилища данных для Desktop-OLAP приложений. Преимуществом предлагаемого метода является его общая направленность и доступность для реализации практически в любых Desktop-OLAP приложениях.

Ключевые слова: хранилище данных, автоматизированный синтез, DOLAP приложение.

Введение

Постановка проблемы и анализ последних исследований и публикаций. В настоящее время одной из основных технологий анализа данных, накопленных в оперативных базах данных (БД) информационных систем управления предприятиями и организациями, является технология OLAP (On-Line Analytical Processing). Эта технология оперативной аналитической обработки данных использует методы и средства сбора, хранения и анализа многомерных данных в целях поддержки процессов принятия управленческих решений. Основное назначение OLAP-систем – поддержка аналитической деятельности, произвольных запросов пользователей-аналитиков. Целью OLAP-анализа является проверка гипотез, возникающих у аналитиков и менеджеров организации в ходе поиска и формирования наиболее рационального управленческого решения. Конечной целью использования OLAP является анализ данных и представление результатов этого анализа в виде, удобном для восприятия и принятия решений [1].

В настоящее время используются такие способы хранения данных в OLAP [1 – 3]:

– MOLAP (Multidimensional OLAP) – архитектура OLAP-систем, основанная на представлении многомерных структур данных в виде многомерных массивов;

– ROLAP (Relational OLAP) – архитектура OLAP-систем, основанная на представлении многомерных структур данных в виде реляционных баз данных;

– HOLAP (Hybrid OLAP) – гибридная архитектура OLAP-систем, сочетающая подходы MOLAP и ROLAP;

– DOLAP (Desktop OLAP) – настольный OLAP. Является недорогой и простой в использовании OLAP-системой, предназначенной для локального анализа и представления данных, которые загружаются из реляционной или многомерной БД на машину клиента.

На данный момент основным источником данных OLAP-систем являются хранилища данных (ХД) и их разновидности – витрины или киоски данных (ВД). Создание метода автоматизированного формирования структур ХД и предварительной обработки данных для анализа является важной задачей, поскольку от скорости и корректности этого процесса напрямую зависит способность информационной системы поддерживать (сопровождать) процессы принятия решений. Однако на данный момент задача формализации процесса трансформации реляционной БД нормализованной и, зачастую, распределенной архитектуры в удобную с точки зрения анализа форму ХД практически не решена.

Попытки решения указанной выше проблемы осуществляются различными способами. В настоящее время основным способом автоматизации синтеза схем ХД являются специализированные пакеты прикладных программ. Среди таких пакетов следует особо выделить пакеты фирм Microsoft (архитектуры Microsoft Data Warehousing + Microsoft Analysis Services из пакета SQL Server 2000), Hyperion (Essbase), Oracle (Oracle 9i) и Informix (DataCudes) [1, 4]. Однако основным недостатком данных пакетов является их изначальная ориентация исключительно на БД и ХД, созданные в конкретных СУБД. В то же время на современных предприятиях Украины наблюдается частое использование одновременно нескольких СУБД в рамках эксплуатации нескольких информационных систем различного назначения.

Еще одним способом автоматизации синтеза схем ХД является применение специализированных студий для построения ХД. Такие студии представляют интегрированные наборы продуктов, содержащие инструменты для проектирования и преобразования ХД, БД, средства управления метаданными и администрирования ХД, предлагают набор шаблонов и заготовок для быстрого создания ХД. С помощью таких наборов у предприятий имеется

возможность сэкономить время и средства на реализацию бизнес-решений и одновременно увеличить гибкость и масштабируемость создаваемого ХД. Наиболее типичным примером студии можно считать набор продуктов Industry Warehouse Studio компании Sybase. Данный набор, первоначально содержавший четыре студии для таких областей бизнеса, как страхование от несчастных случаев, здравоохранение, розничное банковское обслуживание и телекоммуникации, был дополнен студиями для работы в сфере страхования жизни, кредитных карт и рынка ценных бумаг [5].

Возможность синтеза схем ХД реализована и в ряде универсальных CASE-инструментальных средств разработки БД, например, в CASE-средстве ERWin. В основе данной возможности CASE-средства ERWin положена методология проектирования ХД Dimensional. Однако проектирование ХД на основе модели Dimensional требует выбора так называемого "центрального вопроса". На практике таких вопросов может быть довольно много, следовательно, под каждый из них необходимо создавать отдельное ХД. Это приводит к неоднократному выполнению сложной квалифицированной работы по описанию структуры будущего ХД, маршрута извлечения данных из множества реляционных таблиц (при этом надо хорошо представлять структуру исходной БД) и запроса на заполнение ХД. Очевидно, что автоматизация этих процессов значительно разгрузит разработчика и сократит сроки внедрения [6].

Следует отметить, что большинство рассмотренных выше способов и средств решения проблемы синтеза схем ХД ориентированы, в первую очередь, на работы с БД крупных предприятий, которые могут позволить себе эксплуатацию весьма дорогостоящих OLAP-систем и коммерческих корпоративных СУБД. В то же время практически не решена проблема автоматизации синтеза схем ХД для DOLAP-систем, которые являются весьма актуальным средством анализа данных и добычи новых знаний для мелких и средних предприятий.

Чаще всего современные DOLAP-системы представляются самими компаниями-разработчиками БД, многомерных и реляционных. Это SAS Corporate Reporter, являющийся почти эталонным по удобству и красоте продуктом, Oracle Discovery, комплекс программ MS Pivot Services и Pivot Table и другие. Большая группа программ поставляется в рамках компании "OLAP в массы", которую проводит корпорация Microsoft. Эти программы предназначены для работы с MS OLAP Services. Как правило, они являются улучшенными вариантами Pivot Table и предназначены для использования в рамках MS Office или Web (Matrux, Knosys и т.п.) [7]. Однако большинство из этих продуктов неприменимы для мелких и средних предприятий Украины в силу

высокой стоимости и сложности эксплуатации для низкоквалифицированных кадров.

Формулировка цели статьи. Таким образом, возникает необходимость в специальных исследованиях, направленных на решение научно-прикладной задачи создания специализированных методов и средств автоматизации синтеза схем ХД, которые могут быть положены в основу специализированных DOLAP-пакетов, ориентированных на эксплуатацию в условиях Украины и стран СНГ. В частности, в данной статье предлагается метод автоматизированного синтеза схемы хранилища данных для Desktop-OLAP приложений.

Изложение основного материала исследования

Для разработки метода автоматизированного синтеза схем ХД прежде всего следует выбрать способ формализованного описания исходных схем БД, содержащих оперативные данные об управляемом объекте, и ожидаемых в результате синтеза схем ХД или ВД. В настоящее время имеется свыше тридцати моделей представления данных. Но на практике наиболее распространенным средством моделирования данных являются диаграммы "сущность-связь" (Entity-Relation Diagram, ERD). С их помощью определяются важные для предметной области объекты (сущности), их свойства (атрибуты) и отношения друг с другом (связи). ERD непосредственно используются для проектирования реляционных БД. Одно из наиболее полных формализованных описаний ERD, выполненное с использованием математического аппарата теории множеств, изложено в [8]. Разработанное на основе данного описания формализованное теоретико-множественное описание схем ХД и ВД изложено в [9].

В настоящее время наиболее распространенными схемами реализации многомерного представления данных являются схемы "звезда" и "снежинка". Чаще всего используется схема данных "звезда", которая состоит из двух типов таблиц: одной таблицы фактов (центр "звезды") и нескольких таблиц измерений (лучи "звезды"). Таблица фактов является основной таблицей ХД. Как правило, она содержит сведения об объектах или событиях, совокупность которых будет в дальнейшем анализироваться. Обычно говорят о четырех наиболее часто встречающихся типах фактов [1, 2, 10]:

а) факты, связанные с транзакциями (Transaction facts). Они основаны на отдельных событиях (типичными примерами является телефонный звонок или снятие денег с помощью банкомата);

б) факты, связанные с "моментальными снимками" (Snapshot facts). Они основаны на состоянии объекта (например, банковского счета) в определенные моменты времени (например, на конец дня или

месяца). Типичными примерами таких фактов является объем продаж за день или дневная выручка;

в) факты, связанные с элементами документа (Line-item facts). Они основаны на том или ином документе (например, счете за товар или услуги) и содержат подробную информацию об элементах этого документа (например, о количестве, цене, проценте скидки);

г) факты, связанные с событиями или состоянием объекта (Event or state facts). Они представляют возникновение события без подробностей о нем (например, просто факт продажи или факт отсутствия таковой без иных подробностей).

Для разработки DOLAP-приложений выбор соответствующих фактов, которые войдут в таблицу фактов ХД, ограничивается локальной моделью данных, отражающей специфику настольных приложений.

Таблица фактов, как правило, содержит уникальный составной ключ, объединяющий первичные ключи таблиц измерений. При этом как ключевые, так и некоторые неключевые поля должны соответствовать измерениям ХД. Помимо этого, таблица фактов содержит одно или несколько числовых полей, на основании которых в дальнейшем будут получены агрегатные данные. Для многомерного анализа пригодны таблицы фактов, содержащие как можно более подробные данные, т.е. соответствующие членам нижних уровней иерархии соответствующих измерений. В таблице фактов нет никаких сведений о том, как группировать записи при вычислении агрегатных данных. Например, в ней есть идентификаторы продуктов или клиентов, но отсутствует информация о том, к какой категории относится данный продукт или в каком городе находится данный клиент. Эти сведения, используемые в дальнейшем для построения иерархий в измерениях куба, содержатся в таблицах измерений.

Таблицы измерений содержат неизменяемые, либо редко изменяемые данные и расшифровывают ключи, на которые ссылается таблица фактов. В подавляющем большинстве случаев эти данные представляют собой по одной записи для каждого члена нижнего уровня иерархии в измерении. Таблицы измерений также содержат как минимум одно описательное поле (обычно с именем члена измерения) и, как правило, целочисленное ключевое поле (обычно это суррогатный ключ) для однозначной идентификации члена измерения. Если измерение, соответствующее таблице, содержит иерархию, то такая таблица также может содержать поля, указывающие на "родителя" данного члена в этой иерархии. Каждая таблица измерений должна находиться в отношении "один-ко-многим" с таблицей фактов.

Обычно данные в таблицах измерений денормализованы: ценой несколько неэффективного использования дискового пространства удается уменьшить число участвующих в операции соединения таблиц, что обычно приводит к сильному уменьшению вре-

мени выполнения запроса. Скорость роста таблиц измерений должна быть незначительной по сравнению со скоростью роста таблицы фактов. Например, новая запись в таблицу измерений, характеризующую товары, добавляется только при появлении нового товара, не продававшегося ранее.

В сложных задачах с многоуровневыми измерениями используется расширение схемы "звезда" – схема "снежинка". Это расширение может проявляться в случае большого числа сложных атрибутов в таблице измерений, при котором эти атрибуты детализируются в отдельные таблицы измерений. Иными словами отдельные измерения содержатся не в одной, а в нескольких связанных между собой таблицах. Это уменьшит степень дублирования информации, но снижает скорость выполнения запросов, поскольку увеличивает степень нормализации. Поэтому даже при наличии иерархических измерений с целью повышения скорости выполнения запросов к хранилищу данных нередко предпочтение отдается схеме "звезда" [1, 12]. Сказанное выше определяет основной подход к синтезу схемы ХД для DOLAP-приложений как выделение одной из БД, содержащих оперативные данные об управляемом объекте, или же фрагмента такой БД как основы для синтезируемой схемы ХД с последующим автоматическим преобразованием схемы выделенной БД или ее фрагмента в схему ХД типа "звезда". Метод, реализующий данный подход, будет иметь следующий вид.

Этап 1. Формирование кортежного описания выделенной БД или фрагмента БД, содержащего элементы БД, которые будут использоваться для формирования схемы ХД. Данное описание в общем случае имеет вид:

$$\langle E, \text{Attr}_E, R, \text{Pow}_E^R, \text{PK}_E, \text{FK}_E \rangle, \quad (1)$$

где $E = \{E_i\}$ – множество сущностей выделенной БД или фрагмента БД, $i=1,t$; Attr_E – множество атрибутов, используемых для формирования заголовков всего множества сущностей E ; $R = \{R_i\}$ – множество связей между сущностями ERD выделенной БД или фрагмента БД; $\text{Pow}_E^R = \{(\text{Pow}_{E_m}^{R_i}, \text{Pow}_{E_n}^{R_i})\}$ – множество мощностей связей сущностей выделенной БД или фрагмента БД; PK_E – множество первичных ключей выделенной БД или фрагмента БД; FK_E – множество внешних ключей выделенной БД или фрагмента БД.

В общем случае каждый элемент множества R имеет следующий вид [8, 9]:

$$R_i = \langle n_{R_i}, \text{Attr}_{E_m}^{R_i}, \text{Attr}_{E_n}^{R_i}, \text{Pow}_{E_m}^{R_i}, \text{Pow}_{E_n}^{R_i}, S_{E_m}^{R_i}, S_{E_n}^{R_i} \rangle, \quad (2)$$

где n_{R_i} – имя связи R_i ; $\text{Attr}_{E_m}^{R_i} \subseteq \text{Attr}_{E_m}$ – подмножество атрибутов сущности $E_m \in E$, участвующих в образовании связи R_i ; $\text{Attr}_{E_n}^{R_i} \subseteq \text{Attr}_{E_n}$ – подмно-

жество атрибутов сущности $E_n \in E$, участвующих в образовании связи R_i ; $Pow_{E_m}^{R_i}$ – мощность связи R_i для сущности $E_m \in E$; $Pow_{E_n}^{R_i}$ – мощность связи R_i для сущности $E_n \in E$; $S_{E_m}^{R_i}$ – степень участия сущности $E_m \in E$ в связи R_i ; $S_{E_n}^{R_i}$ – степень участия сущности $E_n \in E$ в связи R_i .

Каждый элемент PK_{E_i} множества PK_E в общем случае имеет следующий вид [8, 9]:

$$PK_{E_i} = Atr_{E_i}^{PK} \subseteq Atr_E, \quad (3)$$

где $Atr_{E_i}^{PK}$ – подмножество атрибутов, образующих первичный ключ для сущности E_i , выделяемое по условию [8, 9]:

$$\begin{cases} val_{E_i}^{kPK} = \{val_{E_i}^{kjPK}\} \subseteq \langle val_{E_i}^{kj} \rangle; \\ val_{E_i}^{kjPK} \neq \emptyset; r(Atr_{E_i}^{PK}) \rightarrow \min; \\ val_{E_i}^{kPK} \neq val_{E_i}^{lPK} \text{ если } k \neq l; \\ val_{E_i}^{kPK} = val_{E_i}^{lPK} \text{ если } k = l, \end{cases} \quad (4)$$

где $val_{E_i}^{kPK}$ – значение первичного ключа для экземпляра сущности $e_{E_i}^k$; $val_{E_i}^{kjPK}$ – значение j -го атрибута, который участвует в образовании первичного ключа, присутствующее в экземпляре сущности $e_{E_i}^k$; $r(Atr_{E_i}^{PK})$ – функция, определяющая мощность подмножества атрибутов $Atr_{E_i}^{PK}$, образующих первичный ключ сущности E_i .

Каждый элемент FK_{E_m} множества FK_E для некоей сущности $E_m \in E$ в общем случае имеет следующий вид [8, 9]:

$$FK_{E_m} = Atr_{E_m}^{FK} \subseteq Atr_{E_m}, \quad (5)$$

где $Atr_{E_m}^{FK}$ – подмножество атрибутов, образующих первичный ключ для сущности $E_m \in E$, выделяемое по условию [8, 9]:

$$\begin{cases} \exists E_n, \text{ причем для } e_{E_n}^k \in B_{E_n} \exists e_{E_m}^l \in B_{E_m}; \\ val_{E_n}^{kPK} = val_{E_m}^{lFK}; \\ val_{E_n}^{kPK} = \{val_{E_n}^{kjPK}\}; \\ val_{E_m}^{lFK} = \{val_{E_m}^{ljFK}\}; \\ val_{E_m}^{ljFK} \in D^j; val_{E_n}^{kjPK} \in D^j; D^j \in D, j = 1, t, \end{cases} \quad (6)$$

где $B_{E_m} = \{e_{E_m}^k\}, k = 1, 2, \dots$ – тело сущности $E_m \in E$; $val_{E_n}^{kPK}$ – значение первичного ключа для

экземпляра $e_{E_n}^k$ сущности $E_n \in E$; $val_{E_m}^{lFK}$ – значение вторичного ключа для экземпляра $e_{E_m}^l$ сущности $E_m \in E$; $val_{E_n}^{kjPK}$ – значение j -го атрибута, который участвует в образовании первичного ключа, присутствующее в экземпляре $e_{E_n}^k$; $val_{E_m}^{ljFK}$ – значение j -го атрибута, который участвует в образовании внешнего ключа, присутствующее в экземпляре $e_{E_m}^l$; D^j – домен атрибутов $atr_{E_n}^{jPK}$ и $atr_{E_m}^{jFK}$.

Этап 2. Подсчет количества таблиц k в множестве E описания (1) фрагмента оперативной БД, на основе которого строится схема ХД.

Этап 3. Сбор сведений об объектах или событиях, совокупность которых в дальнейшем будет анализироваться, определение требований к ХД и выделение из кортежного описания (1) подмножества атрибутов $\langle atr_{E_i}^j \rangle \subseteq Atr_E$, которые обязательно войдут в таблицу фактов ХД.

Этап 4. Формирование на основе множества атрибутов таблицы фактов $E_F: E_F = \langle atr_{E_i}^j \rangle$.

Этап 5. Выделение ранее не рассмотренной сущности $E_i \subset E$ из множества сущностей кортежного описания (1).

Этап 6. Поскольку таблицей измерения ХД может быть только родительская таблица БД, которая образует единичный конец связи (должно быть не более одного значения атрибута, образующего связь), производится проверка условия:

$$E_i \neq E_F; Pow_{E_i}^R = 1 \text{ или } FK_{E_i} = 0. \quad (7)$$

В случае невыполнения условия (7) осуществляется возврат к этапу 5. В случае выполнения условия (7) осуществляется переход к этапу 7.

Этап 7. Выделение ранее не рассмотренной сущности $E_j \subset E$ из кортежного описания (1).

Этап 8. Проверка условия на наличие связи между выделенными на этапе 5 и этапе 7 сущностями E_i и E_j :

$$E_j \neq E_F, \exists R = \langle Pow_{E_i}^R, Pow_{E_j}^R \rangle, \quad (8)$$

при дополнительном условии:

$$Pow_{E_i}^R = 1; Pow_{E_j}^R \geq 1. \quad (9)$$

В случае невыполнения условий (8) и (9) осуществляется возврат к этапу 7. В случае выполнения условий (8) и (9) осуществляется переход на этап 9.

Этап 9. Объединение сущностей E_i и E_j :

$$E_i \cup E_j = E_i, \quad (10)$$

Этап 10. Производится проверка условия:

$$Pow_{E_i}^R = 1 \text{ или } FK_{E_i} = 0. \quad (11)$$

В случае невыполнения условия (11) осуществляется возврат к этапу 7. В случае выполнения условия (11) осуществляется переход к этапу 11.

Этап 11. Формирование на основе сущности E_i таблицы измерений ХД E_D .

Этап 12. Определение первичных ключей таблиц измерений PK_{E_D} :

$$PK_{E_i} = PK_{E_D}. \quad (12)$$

После этого осуществляется возврат к этапу 5 до тех пор, пока не будут просмотрены все оставшиеся сущности множества E .

Этап 13. Определение составного ключа таблицы фактов SK_{E_F} :

$$\forall PK_{E_D} = SK_{E_F}, PK_{E_D} \subset SK_{E_F}. \quad (13)$$

Этап 14. Определение связей между таблицей фактов E_F и таблицами измерений E_D :

$$R = \langle SK_{E_F}^R, PK_{E_D}^R, Pow_{E_F}^R, Pow_{E_D}^R \rangle, \quad (14)$$

при условии

$$Pow_{E_F}^R \geq 1; Pow_{E_D}^R = 1. \quad (15)$$

Этап 15. Отображение синтезированной схемы данных ХД.

Исполнителем данного метода является администратор БД, на основе которой будет сформирована схема ХД.

Выводы

Разработанный метод является автоматизированным. Этапы 1, 2, 3 выполняет человек, поскольку компьютер не может определить, какие данные из исходной БД будут использованы для дальнейшего анализа и поэтому не сможет сформировать таблицу фактов будущего ХД. Это является основным недостатком данного метода.

Второй недостаток метода – ориентация исключительно на схему "звезда". Преимуществом же предлагаемого метода является его общая направленность и доступность для реализации практически в любых DOLAP приложениях.

На основе предлагаемого метода в дальнейшем возможно построение общего метода схем ХД и ВД, который будет предоставлять возможность создания как схемы "звезда", так и схемы "снежинка".

Список литературы

1. Методы и модели анализа данных: OLAP и Data Mining [Текст]: Учеб. пособие / А.А. Барсегян, М.С. Куприянов, В.В. Степаненко, И.И. Холод. – СПб.: БХВ-Петербург, 2004. – 336 с.
2. Создание системы поддержки принятия решений на основе хранилищ данных [Электронный ресурс]. – Режим доступа к ресурсу: <http://www.dvgu.ru/meteo/PC/sys.htm>. – Загл. с экрана.
3. Классификация аналитических систем [Электронный ресурс]. – Режим доступа к ресурсу: <http://www.tcont.ru/library/1/2>. – Загл. с экрана.
4. OLAP в финансовом управлении [Электронный ресурс]. – Режим доступа к ресурсу: <http://www.cfin.ru/management/practice/supremum2002/02.shtml>. – Загл. с экрана.
5. Industry Warehouse Studio [Электронный ресурс] // Сайт компании Sybase CIS. – Режим доступа к ресурсу: <http://www.sybase.ru/products/iws>. – Загл. с экрана.
6. Бакулева М.А. Модели и алгоритмы автоматизации проектирования структур хранилищ данных для аналитической обработки числовых показателей [Текст]: автореф. дис. ... канд. техн. наук: 27.09.07 / Бакулева М.А.; [Рязанский государственный радиотехнический университет]. – М., 2007. – 16 с.
7. Настольные OLAP-программы и OLAP-компоненты [Электронный ресурс]. – Режим доступа к ресурсу: <http://www.iso.ru/journal/articles/20.html>. – Загл. с экрана.
8. Левыкин В.М. Параллельное проектирование информационного и программного комплексов информационной системы [Текст] / В.М. Левыкин, В.С. Сугробов, М.В. Евланов // Радиотехника. – 2006. – Вып. 146. – С. 89-98.
9. Евланов М.В. Формализованное описание информационной модели бизнес-процесса [Текст] / М.В. Евланов, Е.В. Корнеева // Системи обробки інформації: зб. наук. пр. – X.: ХУ ПС, 2009. – Вып. 6(80). – С. 167-171.
10. Принципы построения систем, ориентированных на анализ данных [Электронный ресурс]. – Режим доступа к ресурсу: <http://masters.donntu.edu.ua/2005/kita/sroka/library/book10.htm>. – Загл. с экрана.

Поступила в редколлегию 8.08.2012

Рецензент: д-р техн. наук, проф. В.Е. Саваневич, Харьковский национальный университет радиоэлектроники, Харьков.

МЕТОД АВТОМАТИЗОВАНОГО СИНТЕЗУ СХЕМИ СХОВИЩА ДАНИХ ДЛЯ DESKTOP-OLAP ДОДАТКІВ

С.В. Штангей

В роботі сформульована науково-прикладна задача створення спеціалізованих методів і засобів автоматизації синтезу схем сховищ даних, які можуть бути покладені в основу спеціалізованих Desktop-OLAP пакетів, орієнтованих на експлуатацію в умовах України. У статті пропонується метод автоматизованого синтезу схеми сховища даних для Desktop-OLAP додатків. Перевагою пропонованого методу є його загальна спрямованість і доступність для реалізації практично в будь-яких Desktop-OLAP додатках.

Ключові слова: сховище даних, автоматизований синтез, DOLAP додаток.

METHOD OF THE AUTOMATED SYNTHESIS OF INFORMATION DEPOSITORY CHART FOR DESKTOP-OLAP APPENDIXES

S.V. Shtangej

The scientifically-applied task of creation of the specialized methods and facilities of synthesis automation of information depositories charts, which can be fixed in basis of the specialized Desktop-OLAP packages, oriented to exploitation in the conditions of Ukraine, is in article formulated. In the article the method of the automated synthesis of information depository chart for Desktop-OLAP appendixes is offered. Advantage of the offered method is his general orientation and availability for realization practically in any Desktop-OLAP appendixes.

Keywords: depository of information, automated synthesis, DOLAP appendix.