

Дрождин В.В.

Системный подход к построению модели данных эволюционных баз данных с сохранением информации

Предложена модель данных для создания эволюционных баз данных, обеспечивающих хранение информации во времени и способных фиксировать факты, нарушающие закономерности.

Автоматизированная информационная система (АИС) предназначена для удовлетворения информационных потребностей пользователей путем решения задач сбора, хранения, обработки и выдачи информации. АИС получили очень широкое распространение как самостоятельные системы, а также как основа для создания других автоматизированных систем.

Наибольшую часть АИС составляет информация, оптимальность организации и эффективность обработки которой существенно влияют на эффективность всей системы.

Традиционный способ построения баз данных (БД), систем управления базами данных (СУБД) и АИС базируется на проектировании и создании хорошей системы, способной осуществлять автоматизацию информационных процессов с требуемым уровнем качества. Гибкость системы обеспечивается реорганизацией БД, заменой СУБД на более новую версию и изменением программного обеспечения АИС. Такой подход вполне приемлем для автоматизации отдельных несложных информационных задач, требующих создания небольших локальных АИС. Однако это положение принципиально меняется, например, при комплексной автоматизации информационных процессов предприятия, требующей создания большой корпоративной АИС с распределенной БД.

Использование CASE средств на этапе проектирования, стандартизации интерфейса между подсистемами, многоплатформенной реализации, вычислительных средств большой мощности и компьютерных сетей, а также поэтапной разработки и внедрения систем позволяют создавать более сложные АИС, но не позволяют принципиально решить проблемы создания и функционирования больших АИС с распределенными БД.

Более полное решение проблем создания больших АИС возможно на основе эволюционного принципа, предполагающего длительное существование и постепенное развитие системы в процессе функционирования. При этом первоначально созданная АИС может быть достаточно простой, но она должна качественно отличаться от традиционных систем. Развивающаяся АИС должна быть принципиально ориентирована на свое постоянное изменение без потери качества функционирования (качества обслуживания пользователей), а при функционировании в постоянных условиях должна повышать качество функционирования. Единственным методом, обеспечивающим такое поведение АИС, является адаптация.

Адаптация заключается в приспособлении системы к изменениям внешней среды и внутренней организации на основе механизмов обучения и самообучения,

т.е. путем сбора информации, выявления новых зависимостей (моделирования) и использования их в дальнейшем функционировании. Наличие самообучения имеет два последствия. Первое заключается в том, что человек (администратор системы), выступающий в качестве внешней контролирующей и управляющей системы, не может управлять АИС в реальном времени, так как компьютер способен быстро порождать и использовать большое количество зависимостей. С другой стороны, именно самообучение делает систему существенно независимой от внешней среды и обеспечивает необратимый процесс ее эволюции.

В данной работе предлагается формальная теоретико-системная модель данных, предназначенная для создания эволюционных баз данных (ЭБД) с сохранением информации.

Основные принципы организации ЭБД:

- открытость структур данных вниз до самых минимальных элементов (до байтов и битов) и вверх до построения единой структуры, включающей всю базу данных;

- очень высокие гибкость и устойчивость структур данных, т.е. структуры должны разрешать добавление и модификацию фактов для которых подтверждена правильность даже с нарушением поддерживаемых отношений (закономерностей).

Моделью данных называется совокупность допустимых структур данных и операций над ними.

Структура данных - это взаимосвязанная совокупность элементов данных.

Построение модели данных будем осуществлять на основе общей теории систем, разработанной Ю.А.Урманцевым [1, 2].

Объект-система (OS) – это композиция или единство элементов $m \subseteq M_{OS}$ построенная по отношениям (взаимодействиям) $r \subseteq R_{OS}$ и ограничивающим их условиям или законам композиции $z \subseteq Z_{OS}$. Поэтому в общем виде OS представляется тройкой:

$$S = \langle m, r, z \rangle.$$

Система объектов одного и того же i рода (R-система или RS) - это закономерное множество OS одного и того же качества (рода i), означающая, что каждая OS построена:

- из всех или части элементов $m \subseteq M^i$, выделенных из универсума U ($M^i \subseteq U$) по основаниям $a \subseteq A^i$;

- в соответствии со всеми или частью отношений $r \subseteq R^i$;

- в соответствии со всеми или частью законами композиции $z \subseteq Z^i$.

Поэтому RS представляется четверкой вида:

$$R = \langle U, A^i, R^i, Z^i \rangle.$$

Любой объект есть OS и каждая OS принадлежит хотя бы одной RS [1].

Далее будем считать, что каждая OS разрабатываемой модели строится на основе всех отношений R^i , удовлетворяет всем законам композиции Z^i и принадлежит строго одной RS. Поэтому модель OS будет иметь вид:

$$S = \langle m, R^i, Z^i \rangle,$$

а в случае известной RS:

$$S = \langle m \rangle.$$

Введем обозначения, более привычные в программировании и теории баз данных:

$$T = U, \text{ val} = m, F^i = R^i, P^i = Z^i, \text{VAL}^i = \langle T, A^i \rangle = M^i,$$

где T – множество допустимых типов данных;

F – зависимости между данными (ассоциации, отображения, классификации, многозначные, функциональные и др.), причем наиболее сильными (логическими) являются функциональные и многозначные зависимости, а наиболее слабыми – ассоциации, позволяющие ассоциировать между собой произвольные данные;

P – предикат, задающий ограничения целостности на данные;

VAL^i – множество элементов, составляющих RS .

Тогда модели OS и RS будут иметь вид:

$$S^i = \langle \text{val}^i, F^i, P^i \rangle \text{ или } S^i = \langle \text{val}^i \rangle,$$

$$R^i = \langle \text{VAL}^i, F^i, P^i \rangle.$$

Будем считать **R-системой 0-го рода** любой абстрактный тип данных [3] с атомарными значениями данных представленный в виде:

$$T = \{ \tau_g \},$$

$$\tau_g = (D_g, \Theta_g),$$

где g – имя типа;

D_g – множество атомарных элементов данных типа g ;

Θ_g – множество операций, выполняемых над элементами данных типа g .

Абстрактный тип данных задает самый низкий (синтаксический) уровень представления и обработки данных.

Тогда OS и RS 0-го рода, реализующие тип данных g , будут иметь вид:

$$S^0 = \langle \text{val}^0 \rangle,$$

$$R^0 = \langle \text{VAL}^0 \rangle.$$

где $A^0 = g$;

$$\text{VAL}^0 = D_g;$$

$$\text{val}^0 \in \text{VAL}^0;$$

$$F^0 = \emptyset;$$

$$P^0 = \emptyset.$$

На 0 уровне решается задача эффективной реализации абстрактного типа данных g , определяющей способ представления значений данных и операций их обработки.

Для эффективного представления, обработки и организации взаимодействия OS введем абстрактный элемент id следующим образом:

$$\text{id} \equiv \text{val}.$$

Элемент id будем называть **идентификатором** и считать, что он обладает следующими свойствами:

- не зависит от типа и структуры val ;

- принадлежит некоторому универсальному числовому типу.

Тогда модели OS и RS примут вид:

$$S = \langle \text{id}, \text{val}^i, F^i, P^i \rangle \text{ или } S = \langle \text{id}, \text{val}^i \rangle,$$

$$R = \langle \text{ID}, \text{VAL}^i, F^i, P^i \rangle.$$

Будем называть **R-системой 1-го рода** упорядоченное множество атомарных элементов val и эквивалентных им идентификаторов id :

$$S^1 = \langle \text{id}, \text{val}^1 \rangle$$

$$R^1 = \langle \text{ID}, \text{VAL}^1, P^1 \rangle,$$

где $A^1 = g_R$ - имя системы объектов R^0 , объекты S^0 которой преобразуются в val^1 ;

$$\text{VAL}^1 \subseteq \text{VAL}^0;$$

$$\text{val}^1 \in \text{VAL}^1;$$

$$F^1 = \emptyset;$$

$$P^1 = \{ p_{\beta^1} \mid \beta^1 \in \langle \text{ограничения на атомарные данные} \rangle \} - \text{ограничения, определяющие допустимое разнообразие объектов } S^1 \text{ в системе объектов } R^1.$$

Примерами R-систем 1-го рода являются домены реляционной модели данных: Список фамилий, Список имен, Список отчеств, Список дат рождения.

На 1 уровне решаются задачи эффективного представления, хранения и доступа к объектам S^1 в системе объектов R^1 .

Следующие, более высокие уровни организации систем, будут содержать сложные элементы val , структура которых будет задаваться набором отношений (устойчивых связей, взаимосвязей) F между компонентами (частями) $\text{val}' \in \text{val}$ и ограничениями целостности P . При этом уровень системы будет определяться мощностью взаимосвязей (обязательностью участия в них компонентов val' , допустимой степенью нарушений – количеством исключений, частотой совместного использования и обработки компонентов val') и допустимой степенью разнообразия val . Уровень системы будет тем ниже, чем выше мощность связей между компонентами val' и ниже степень разнообразия val .

Будем называть **R-системой 2-го рода** упорядоченное множество объектов S^2 , существующих во времени T , в каждом из которых компоненты $\text{val}' \in \text{val}^2$ жестко взаимодействуют (сильно связаны) между собой:

$$S^2 = \langle \text{id}, \text{val}^2, t_h, t_k \rangle,$$

$$R^2 = \langle \text{ID}, \text{VAL}^2, F^2, P^2, T \rangle,$$

где $A^2 = [g_{R1}, g_{R2}, \dots, g_{Rn}]$ – последовательность имен систем объектов R^1 , из объектов S^1 которых формируется элемент val^2 ;

$$\text{VAL}^2 \subseteq \text{VAL}_{g_{R1}}^1 \times \text{VAL}_{g_{R2}}^1 \times \dots \times \text{VAL}_{g_{Rn}}^1;$$

$$\text{val}^2 \in \text{VAL}^2;$$

$$F^2 = \{ f_{\alpha^2} \mid \alpha^2 \in \langle \text{зависимости между элементами данных во всех } \text{val}^2 \rangle \} - \text{отношения (взаимосвязи) между данными типа функциональных и многозначных зависимостей, определяющие необходимое подобие всех объектов } S^2 \text{ в системе объектов } R^2;$$

отношения (взаимосвязи) между данными типа функциональных и многозначных зависимостей, определяющие необходимое подобие всех объектов S^2 в системе объектов R^2 ;

$$P^2 = \{ p_{\beta^2} \mid \beta^2 \in \langle \text{ограничения на сложные данные} \rangle \} - \text{ограничения, определяющие допустимое разнообразие объектов } S^2 \text{ в системе объектов } R^2;$$

определяющие допустимое разнообразие объектов S^2 в системе объектов R^2 ;

$$T = \{ t \} - \text{физическое время};$$

$t_n, t_k \in T$ – начальный и конечный моменты времени, задающие интервал времени существования объекта S^2 .

Примерами R-систем 2-го рода являются отношения реляционной базы данных: Человек (Фамилия, Имя, Отчество, Дата рождения, Номер паспорта, Адрес), Предприятие (Название, Адрес, Руководитель, Телефон руководителя).

На 2 уровне решаются задачи эффективного выявления и поддержки отношений между данными, представления и обработки исключений, поддержки схемы данных в оптимальной форме, а также ведение динамической информационной модели предметной области с предысторией. При этом для темпоральных данных целесообразно создавать подсистемы $R^{2'}$ вида:

$$S^{2'} = \langle id, val^{2'}, t \rangle,$$

$$R^{2'} = \langle ID, VAL^{2'}, T \rangle,$$

где $val^{2'} \in VAL^{2'}$ – значение компоненты в некоторый момент времени;
 $t \in T$ – начальный момент времени существования компонента $val^{2'}$ в S^2 .

Будем называть **R-системой 3-го рода** упорядоченное множество объектов S^3 , каждый из которых представляет собой совокупность объектов S^2 и $S^{3'}$, в подавляющем случае обрабатываемых и используемых совместно:

$$S^3 = \langle id, val^3, t_n, t_k \rangle,$$

$$R^3 = \langle ID, VAL^3, F^3, P^3, T \rangle,$$

где $A^3 = [g_{R1}, g_{R2}, \dots, g_{Rn}]$ – последовательность имен систем объектов R^2 и $R^{3'}$, из объектов S^2 и $S^{3'}$ которых формируется элемент val^3 ;

$$VAL^3 \subseteq VAL_{g_{R1}}^2 \times VAL_{g_{R2}}^2 \times \dots \times VAL_{g_{Rn}}^{3'};$$

$$val^3 \in VAL^3;$$

$S^{3'}$ – это объект уже существующей $R^{3'}$, который включается в val^3 в качестве компонента;

$F^3 = \{f_{\alpha^3} \mid \alpha^3 \in \langle \text{взаимосвязи компонентов } val^2 \text{ и } val^{3'} \text{ во всех } val^3 \rangle\}$ – отношения, определяющие совместную обработку (возможные конфигурации) объектов S^2 и $S^{3'}$ и задающие необходимое подобие всех объектов S^3 в системе объектов R^3 ;

$P^3 = \{p_{\beta^3} \mid \beta^3 \in \langle \text{ограничения на совместную обработку объектов} \rangle\}$ – ограничения на отсутствие и форму объектов S^2 и $S^{3'}$, определяющие допустимое разнообразие объектов S^3 в системе объектов R^3 ;

$$T = \{t\} \text{ – физическое время};$$

$t_n, t_k \in T$ – начальный и конечный моменты времени, задающие интервал времени существования объекта S^3 .

Примерами R-систем 3-го рода являются, например, ФИО и Адрес в отношении Человек или Сотрудник и Студент, являющиеся наследниками Человека.

Таким образом, R-система 3-го рода позволяет, с одной стороны, декомпозировать существующие системы объектов 2 и 3 рода на более мелкие системы, но с более сильно связанными элементами, а с другой – формировать из существующих систем 2 и 3 рода более крупные системы при усилении связей между элементами этих систем.

На 3 уровне решаются задачи эффективного выявления и поддержки отношений совместного использования данных и формирования оптимальной схемы данных с учетом совместно используемых данных.

Будем называть **R-системой 4-го рода** единственный объект S^4 , представляющий собой взаимосвязанную совокупность объектов S^2 и S^3 и существующий автономно:

$$S^4 = \langle \text{val}^4, t_n, t_k \rangle,$$

$$R^4 = \langle \text{VAL}^4, F^4, P^4, T \rangle,$$

где $A^4 = [g_{R1}, g_{R2}, \dots, g_{Rn}]$ – последовательность имен систем объектов R^2 и R^3 , из объектов S^2 и S^3 которых формируется элемент val^4 ;

$$\text{VAL}^4 \subseteq \text{VAL}_{g_{R1}}^2 \times \text{VAL}_{g_{R2}}^2 \times \dots \times \text{VAL}_{g_{Rn}}^3;$$

$$\text{val}^4 \in \text{VAL}^4;$$

$F^4 = \{f_{\alpha^4} \mid \alpha^4 \in \langle \text{взаимосвязи компонентов } \text{val}^2 \text{ и } \text{val}^3 \text{ в } \text{val}^4 \rangle\}$ – отношения, определяющие автономную совокупность объектов S^2 и S^3 и формирующие целостный объект S^4 , являющийся единственным представителем системы объектов R^4 ;

$P^4 = \{p_{\beta^4} \mid \beta^4 \in \langle \text{ограничения на автономность объекта} \rangle\}$ – ограничения на необходимость, объем и интенсивность обработки объектов S^2 и S^3 , определяющие существование объекта S^4 , представляющего систему объектов R^4 ;

$$T = \{t\} \text{ – физическое время};$$

$t_n, t_k \in T$ – начальный и конечный моменты времени, задающие интервал времени существования объекта S^4 .

Примерами R-систем 4-го рода являются автономные подсистемы больших систем или самостоятельные автоматизированные системы.

Таким образом, R-система 4-го рода задает автономную (самостоятельную, целостную и относительно независимую) систему организации данных, для которой может быть разработан эффективный метод ведения и обработки данных, позволяющий системе существовать длительное (потенциально бесконечное) время.

На 4 уровне решаются задачи эффективного выявления и поддержки отношений автономности объекта и предоставления эффективного интерфейса взаимодействия с внешней средой.

Предложенная модель данных позволяет создавать устойчивые многоуровневые БД, допускающие хранение фактов, нарушающих зависимости, что позволяет обоснованно проводить реорганизацию и оптимизацию БД в процессе функционирования АИС.

Литература

1. Система, симметрия, гармония. - М.: Мысль, 1988. - 315 с.
2. Урманцев Ю.А. Эволюционика. - Пущино: 1988. - 79 с.
3. Нагао М., Катаяма Т., Уэмура С. Структуры и базы данных. – М.: Мир, 1986. – 197 с.