



Session 1

Transport

ИСПОЛЬЗОВАНИЕ ФУНКЦИЙ С ГИБКОЙ СТРУКТУРОЙ ДЛЯ ПРОГНОЗИРОВАНИЯ ГРУЗОБОРОТА В ЛОГИСТИЧЕСКОЙ СИСТЕМЕ

Миндаугас Мазура, Ольга Фадина

*Вильнюсский технический университет им. Гедиминаса
ул. Плитинес. 27, ЛТ-2040, Вильнюс, Литва
Ел. почта: Minduagas.Mazura@ti.vtu.lt, Olga@lineka.lt*

Экономический прогноз позволяет установить возможные направления и различные варианты развития логистической системы, а также помогает в выборе конкретных целей её функционирования. Поэтому основное назначение прогноза в логистике состоит в раскрытии тенденции изменения микро- и макрологической среды и получения вероятностных количественных и качественных оценок динамики логистических активностей, необходимых персоналу менеджмента фирмы. В статье предложены методы прогнозирования показателей логистической системы с использованием функций с гибкой структурой, проведена оценка точности и повышения достоверности прогнозирования, а также проанализированы методы измерения погрешностей прогноза.

Ключевые слова: Прогнозирование, логистическая система, менеджмент, функции с гибкой структурой, погрешности прогноза

1. Введение

Часто для логистического менеджмента важным является получение прогноза агрегированного показателя (например, объёма продаж готовой продукции) в зависимости от одного или нескольких факторов, характеризующих, например мощность и производительность технологического оборудования, использование основных фондов и оборотного капитала, трудовых ресурсов и т.п. В этих случаях, как правило, применяют однофакторные и многофакторные корреляционно-регрессионные модели, общим ограничением которых является невозможность непосредственного использования для целей прогнозирования. Кроме того, характер связи между факторами может отличаться от линейного, что создаёт дополнительные трудности вычислительного характера. Для построения прогнозов логистических показателей на основе использования предположения о наличии и статистической связи между двумя показателями y и x может быть эффективно использован математический аппарат функций с гибкой структурой (ФГС). Форма ФГС может изменяться и автоматически определяться в процессе прогнозирования, учитывая не только статистическую зависимость $y = f(x)$, но и собственные временные тенденции изменения каждого фактора. С помощью ФГС можно решать и важные для логистики обратные задачи: например, для запланированного объёма продаж определить экстенсивные и интенсивные производственные факторы и параметры.

2. Использование функций с гибкой структурой

Функция с гибкой структурой имеет вид:

$$F(x) = A_0 + \sum_{j=1}^n A_j \frac{\delta_j(x)}{D}, \quad (1)$$

где n – фиксированное натуральное число;
 x_0 – начальное значение фактора-аргумента на рассматриваемом интервале;
 A_0, A_1, A_n – параметры;
 D – определитель Ван-дер-Монда n -го порядка.

$$D = \begin{vmatrix} 1 & 1 & 1 & \dots & 1 \\ r_1 & r_2 & r_3 & \dots & r_n \\ r_1^2 & r_2^2 & r_3^2 & \dots & r_n^2 \\ \dots & \dots & \dots & \dots & \dots \\ r_1^{n-1} & r_2^{n-1} & r_3^{n-1} & \dots & r_n^{n-1} \end{vmatrix}$$

$\overline{r_1, r_n}$ – действительные или комплексные попарно-сопряжённые числа;
 $\delta_j(x-x_0)$ – функции, получаемые из определителя D путём замены j -й строки на экспоненты вида

$$\frac{1}{r_j^v} \left(1^{r_j(x-x_0)} - 1 \right), \quad v = \overline{1, n}. \quad (2)$$

Идея метода ФГС заключается в представлении исходного процесса в виде

$$y(x) = F(x) + R(x), \quad (3)$$

где $R(x)$ – функция точности приближения (остаток);

$F(x)$ – ФГС.

Основное влияние на структуру ФГС оказывают значения r_j ($j = \overline{1, n}$). Если все они – действительные числа, различные и не равные нулю, то ФГС будет представлять собой линейную комбинацию экспонент. Если числа r_j – комплексные, то ФГС является суммой экспонент, умноженных на гармонические (\sin – \cos) составляющие. При равенстве $r_j = 0$ ($j = \overline{1, n}$) ФГС преобразуется в степенной многочлен.

Таким образом, основным становится вопрос о нахождении оптимальных значений r_j , ($j = \overline{1, n}$).

Если функция $y(x)$ дифференцируема, то в качестве параметров A_0, A_1, \dots, A_j можно взять соответствующие производные от $y(x)$, тогда (1) запишется в виде

$$F(x) = y(x_0) + \sum_{j=1}^n y^{(j)}(x_0) \frac{\delta_j(x-x_0)}{D}. \quad (4)$$

При этом числа r_1, \dots, r_n рассматриваются как корни базисного уравнения

$$r^n + a_{n-1}r^{n-1} + a_{n-2}r^{n-2} + \dots + a_1r + a_0 = 0. \quad (5)$$

Аналитическое выражение остатка в формуле (3) имеет следующий вид

$$R(x) = \int_{x_0}^x \int_{x_0}^t \eta(\tau) \frac{\Delta_n(t-\tau)}{D} d\tau dt, \quad (6)$$

где $\Delta_n(t-\tau)$ получается из определителя D заменой n -й строки (последней) на строку экспонент вида

$$1^{r_i(t-\tau)}, (i = \overline{1, n}). \quad (7)$$

Функция $\eta(\tau)$ уравнения (6) определяется из уравнения

$$\eta(\tau) = y^{(n+1)}(\tau) + a_{n-1}y^{(n)}(\tau) + \dots + a_1y^{(2)}(\tau) + a_0y^{(1)}(\tau) \quad (8)$$

где $y^{(j)}$ – j -я производная y .

Идея оптимальной аппроксимации при использовании ФГС сводится к минимизации остатка $R(x)$, установлению таких значений параметров a_0, a_1, \dots, a_{n-1} , чтобы значение остатка в каждой точке отрезка аппроксимации не превышало некоторой заданной величины погрешности аппроксимации – это и определяет вид ФГС.

Рассмотрим простейший случай, когда $n=1$. Тогда выражение (4) можно записать в виде

$$F(x) = y(x_0) + y'(x_0) \left[\frac{1^{r(x-x_0)}}{r} \right], \quad (9)$$

Базисное уравнение (5) для r примет вид:

$$r + a_0 = 0. \quad (10)$$

Функция $\eta(\tau) = \eta(x)$ для остатка $R(x)$ из уравнения (8) превратится в

$$\eta(\tau) = y''(x) + a_0y'(x). \quad (11)$$

Условие минимума остатка можно представить в виде $\eta(x) = 0$, что обеспечит нулевую погрешность аппроксимации. Тогда, для того чтобы избавиться от производных в уравнении (11), проинтегрируем его

$$y'(x) + a_0y(x) = c_1. \quad (12)$$

Подставляя в это выражение значение начальной точки для аргумента $x = x_0$, получим

$$y'(x_0) = c_1 - a_0y(x_0). \quad (13)$$

Ещё раз интегрируем уравнение (12)

$$y(x) = a_0 \int_{x_0}^x y(x) dx = c_1(x - x_0) = c_0. \quad (14)$$

При $x = x_0$ мы имеем $c_0 = y(x_0)$, тогда уравнение (14) примет вид

$$c_1(x - x_0) - a_0 \int_{x_0}^x y(x) dx = y(x) - y(x_0). \quad (15)$$

из уравнения (15) видно, что оно содержит две неизвестные постоянные a_0, c_1 и интеграл. Интеграл задан табличной функцией $y(x)$ и может быть вычислен графически.

Для вычисления c_1 и a_0 образуем на основе уравнения (15) систему уравнений с двумя неизвестными. Это можно сделать, если подставить в (15) значение ещё двух точек, взятых из временного ряда $y(x)$. Тогда

$$\begin{cases} c_1(x - x_0) - a_0 \int_{x_0}^{x_1} y(x) dx = y(x_1) - y(x_0), \\ c_1(x_2 - x_0) - a_0 \int_{x_0}^{x_2} y(x) dx. \end{cases} \quad (16)$$

После вычисления интегралов находим неизвестные коэффициенты c_1 и a_0 . Затем определяем значение первой производной $y'(x_0)$ из уравнения (13). Корень базисного уравнения $r = a_0$ со знаком минус. Вычисленные параметры подставляем в формулу для ФГС (9) и получим аналитическое выражение для $y(x)$.

Пример

В качестве примера рассмотрим определение зависимости между коэффициентом обновления основных фондов автотранспортной фирмы (ρ) и объёмом её грузооборота (Q). Исходные данные приведены в табл. 1.

В начале выбираем значение трёх опорных точек, одна из которых (начальное значение) должна лежать в середине ряда для лучшей аппроксимации, т. е. для того, чтобы полученная функция одинаково точно приближала данные как в конце, так и в начале ряда.

Выбираем: $x_0 = 34.3$ $y_0 = 0.618 = (y(x_0))$;

$x_1 = 31.2$ $y_1 = 0.597$;

$x_2 = 37.8$ $y_2 = 0.631$.

Таблица 1.

Показатель	1996	1997	1998	1999	2000	2001	2002
$\rho, (y)$	0,597	0,597	0,608	0,618	0,615	0,618	0,631
Q, млн. усл. ед, (x)	31,2	32,3	33,4	34,3	34,5	35,5	37,8

Находим коэффициенты уравнений (16):

$$x_1 - x_0 = -3.1; \quad x_2 - x_0 = 3.5;$$

$$y_1 - y_0 = -0.021; \quad y_2 - y_0 = 0.013.$$

Вычислим интегралы, входящие в систему (16).

Интеграл $\int_{x_0}^{x_1} y(x) dx$ – площадь, ограниченная графиком функции $y(x)$ (эмпирической) и значениями x , равными 34,3 и 31,2. Так как верхний предел

интеграла меньше нижнего, то значение интеграла отрицательное. Площадь, ограниченная значениями x , равными 34,3 и 31,2 будет складываться из площадей трёх трапеций:

$$\int_{34.3}^{31.2} y(x) dx = - \left[\begin{aligned} &0.597(32.3 - 32.1) + \frac{0.597 + 0.608}{2} \cdot \\ &\cdot (33.4 - 32.1) + \frac{0.608 + 0.618}{2} \cdot \\ &\cdot (34.3 - 33.4) \end{aligned} \right] =$$

$$= -1.871$$

Значение интеграла $\int_{34.8}^{37.8} y(x) dx$ будет равно:

$$\int_{34.3}^{37.8} y(x) dx = - \frac{0.618 + 0.615}{2} (34.5 - 34.3) + \frac{0.615 + 0.618}{2} \cdot$$

$$\cdot (35.5 - 34.5) + \frac{0.618 + 0.631}{2} (37.8 - 35.5) = 2.176.$$

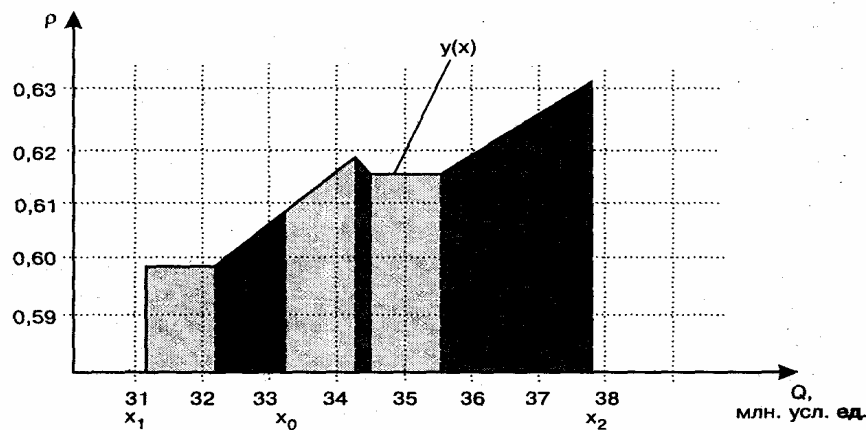


Рис. 1. Графическое определение интегралов (пример)

Графическое определение интегралов показано на рис. 1.

Получение значения коэффициентов и интегралов подставляем в уравнения системы (16). Получим:

$$\begin{cases} -3.1c_1 + 1.871a_0 = -0.021, \\ 3.5c_1 - 2.176a_0 = 0.013. \end{cases}$$

Решая эту систему, находим

$$c_1 = 0.108; \quad a_0 = 0.168.$$

Затем определяем значение первой производной в начальной точке, подставив в уравнение (13) полученные параметры:

$$y'(x_0) = c_1 - a_0 y(x_0) = 0.108 - 0.168 \cdot 0.618 = 0.00417.$$

Из базисного уравнения (10) находим

$$r = -a_0 = -0.168.$$

Теперь получив все необходимые параметры, найдём выражение для ФГС:

$$F(x) = 0.618 + 0.00417 \left[\frac{1^{-0.168(x-34.3)} - 1}{-0.168} \right].$$

Раскрыв скобки и сделав необходимые вычисления, окончательно получим

$$y = \rho = 0.643 - 0.0248e^{-0.168(Q-34.3)}.$$

Найдём прогноз на 2003 год. Учитывая, что запланированный объём грузооборота должен составить в 2003 г. 41 млн. усл. ед., получаем требуемый (прогнозируемый) коэффициент обновления основных фондов

$$\rho = 0.648 - 0.02481^{-0.168(41-34.3)} = 0.635.$$

При решении обратной задачи получим выражение для нахождения прогноза объёма грузооборота

$$Q_1 = 29.057 + 8.24 \exp \left[3t5(\rho_1 - 0.618) \right].$$

3. Погрешности прогноза

Точность прогноза определяется разницей между фактическими и прогнозными продажами. Повышение точности прогнозов требует измерения и анализа погрешностей. Для начала нужно определить подходящие способы измерения погрешностей. Затем следует выбрать надлежащий уровень выявления и оценки погрешностей. И, наконец, необходимо наладить механизм обратной связи.

4. Измерение погрешностей

Существует несколько методов измерения погрешностей прогнозирования в абсолютных или относительных величинах. В принципе понятно, что погрешность прогноза равна разнице между фактическим спросом и его прогнозным значением, но для расчётов и сопоставлений требуется более строгое определение.

Один из методов вычисления погрешности прогноза заключается в следующем. Сначала надо суммировать погрешности за все периоды. Вычислив сумму погрешностей за год, делим её на 12 и находим простую среднюю арифметическую погрешностей. Недостаток этого метода в том, что завышенные и заниженные прогнозные оценки взаимно погашаются, и это маскирует существенную неточность прогнозирования в отдельные месяцы. Избежать этого позволяет другой метод, который сводится к вычислению абсолютной погрешности. Этот метод часто применяют для измерения погрешностей, но проблема в том, что он придаёт равные веса и крупным и незначительным отклонениям. Ещё один путь – вычислить среднюю квадратическую погрешность. Преимущество средней квадратической погрешности состоит в том, что крупные погрешности получают в ней больший вес, чем малые.

Показатели средней, абсолютной средней и средней квадратической погрешности хороши для оценки точности прогнозов по отдельным единицам хранения и отдельным распределительным центрам, но именно поэтому они не слишком подходят для оценки

агрегированных прогнозов. *Например*, при отклонении прогноза от фактического объёма продаж на 50 единиц все эти показатели придадут одинаковое значение погрешности независимо от того, чему равен месячный спрос – 50 единицам или 5 тыс. единиц. Между тем в первом случае прогноз ошибочен на 100%, во втором только на 1%. Для сопоставления точности прогнозов по разным единицам хранения и рынкам с неодинаковым средним спросом принято исчислять относительную величину погрешности в процентном выражении; для этого среднюю погрешность прогноза надо разделить на среднюю величину спроса.

5. Уровень оценки

Вторая задача заключается в выборе уровня оценки или агрегирования. Имея все данные прогнозов по отдельным единицам хранения, погрешность прогноза можно вычислять для сочетаний «единица хранения – распределительный центр», для групп единиц хранения или распределительных центров либо для национального рынка в целом. Чем выше уровень агрегирования, тем меньше относительная величина погрешности прогноза.

При выборе уровня агрегирования прогноза следует учитывать два соображения. Во-первых, чем ниже этот уровень, тем легче выявлять проблемы и преодолевать ошибки, совершенствуя процесс прогнозирования. Во-вторых, чем менее агрегирован прогноз (и соответственно, чем более детальным может быть анализ погрешностей), тем больше требуется вычислений и исходных данных, поскольку у типичной фирмы число комбинаций «единица хранения – распределительный центр» обычно очень велико.

6. Обратная связь

Третий этап в обеспечении точности прогнозов предполагает создание механизма обратной связи, помогающего совершенствовать процесс прогнозирования. Для этого нужно, чтобы служащие были заинтересованы в выявлении проблем и в поиске путей совершенствования и чтобы они хорошо представляли себе, какое вознаграждение их за это ждёт. При наличии должной мотивации прогнозисты в состоянии выявить главные источники ошибок и разработать способы, позволяющие их избегать. В некоторых случаях развитие методов прогнозирования, примером служат методы фокусировки, даёт поразительное повышение точности прогнозов.

7. Оценка точности и повышения достоверности прогнозирования

При использовании в логистическом менеджменте различных методов и моделей прогнозирования необходимо большое внимание уделять их точности и оценке достоверности полученных результатов прогноза.

Оценка точности прогноза и его достоверности, или как часто говорят, верификация прогноза, является достаточно сложной задачей.

Верификация прогнозных моделей может быть осуществлена несколькими способами:

- *прямая верификация* заключается в получении тех же результатов;
- *косвенная верификация* – подтверждение прогноза ссылкой на приведённый в литературе прогноз того же объекта;
- *консеквентная верификация* – получение значения верификационного прогноза путём логического или математического выведения следствий из уже известных прогнозов;

- *инверсная верификация* – т. е. экстраполяция назад. При этом сопоставляются значения прогноза, полученного инверсной экстраполяцией с фактическими значениями ретроспективного ряда;
- верификация путём минимизации систематических ошибок.

Полную ошибку решения задачи прогноза можно определить суммой:

$$\varepsilon_n = \varepsilon_u + \varepsilon_m + \varepsilon_g + \varepsilon_n, \quad (17)$$

где ε_u – ошибки в исходной информации;

ε_m – ошибки метода прогнозирования;

ε_g – погрешность вычислений;

ε_n – нерегулярная ошибка, вызванная появлениями в будущем непредсказуемых событий, влияющих на прогнозируемую величину.

В случае инверсной верификации основным критерием правильности подобранной формы аппроксимирующей зависимости является среднеквадратическое отклонение теоретических значений функции от эмпирических данных ретроспективного ряда:

$$\sigma_y = \sqrt{\frac{\sum_{i=1}^n (y_i - \bar{y}_i)^2}{n-1}}, \quad (18)$$

где n – число наблюдений (членов ряда);

y_i – значение эмпирического ряда;

\bar{y}_i – расчётные (теоретические) значения ряда.

Кроме критерия (18) используются следующие дополнительные показатели, оценивающие точность подобранной аппроксимирующей зависимости:

– *коэффициент вариации*

$$v = \frac{\sigma_y}{y} 100\%, \quad (19)$$

где \bar{y} – оценка математического ожидания ретроспективного ряда;

– *индекс корреляции*

$$R^2 = 1 - \frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{\sum_{i=1}^n (\bar{y} - y_i)^2}, \quad (20)$$

– *критерий Фишера*

$$F = \frac{R^2 (n - m)}{(1 - R^2)(m - 1)}, \quad (21)$$

где m – число неизвестных параметров аппроксимирующей функции.

Лучшая аппроксимирующая функция выбирается из условий:

$$\sigma, v \rightarrow \min; \quad R^2, F \rightarrow \max \quad (22)$$

Так как прогнозирование носит вероятностный характер, то прогнозируемый показатель должен определяться с некоторыми доверительными границами (интервальная оценка прогноза).

При нормальном распределении отклонений $(y_i - y_i)$ и заданной σ доверительный интервал для прогноза y^* имеет вид

$$(y^* - \delta, y^* + \delta),$$

где δ – точность оценки, определяемая из выражения

$$\delta = \frac{t\sigma}{\sqrt{n}}; \tag{23}$$

Если задана доверительная вероятность γ , то параметр t определяется из соотношения

$$2\Phi(t) = \gamma, \tag{24}$$

где Φ – функция Лапласа.

При $\gamma = 0.95$ $\Phi(t) = 0.475$. По таблицам функции Лапласа находим, что $t = 1.96$.

Тогда

$$\delta = \frac{t\sigma}{\sqrt{n}} = \frac{(1.96\sigma)}{\sqrt{n}}.$$

Т. е. доверительный интервал имеет следующие границы:

$$\begin{cases} y_n^* = y^* - \frac{1.96\sigma}{\sqrt{n}}; \\ y_e^* = y^* + \frac{1.96\sigma}{\sqrt{n}}. \end{cases}$$

Для повышения точности и достоверности прогнозирования в ЛС можно использовать комбинированную схему (или комплексную оценку) прогноза, позволяющую объединить два и более метода. Комбинированная схема (синтез прогнозов) позволяют компенсировать недостатки одних методов достоинствами других, особенно в случае, если они построены на различной информационной базе.

Пример построения комбинированного прогноза.

Предположим, что прогнозирование спроса на грузооборот, производимый некоторой фирмой, осуществлялось тремя методами, результаты которых сведены в табл. 2.

Необходимо применить схему комбинированного прогноза для повышения достоверности.

Оценим граничные значения для 1-го и 2-го методов, путём нахождения доверительных интервалов.

Для прогноза по методу наименьших квадратов используем формулу для дисперсии в случае линейной зависимости $(y = a_0 + a_1t)$:

$$\begin{aligned} \sigma_y^2(t) &= \sigma_{a_0}^2 + \sigma_{a_1}^2 t^2 + 2tKa_0a_1, \\ \sigma_{a_0}^2 &= \sigma_y^2 B_0; \quad \sigma_{a_1}^2 = \sigma_y^2 B_1; \quad K_{a_0a_1} = \sigma_y^2 B_2. \end{aligned}$$

$$\left. \begin{aligned} B_0 &= \frac{\sum t_i^2}{\Delta} = \frac{1139}{1075} = 1.06; \\ B_1 &= \frac{N}{\Delta} = \frac{4}{1075} = 0.0037; \\ B_2 &= \frac{\sum t_i}{\Delta} = -\frac{59}{1075} = -0.055. \end{aligned} \right\}$$

Коэффициенты B_0, B_1, B_2 определяются из уравнений метода наименьших квадратов при $N = 4$; $t = 30$.

$$\sigma_y^2 = (0.082)^2 = 0.0067.$$

Определим среднее квадратическое отклонение

$$\begin{aligned} \sigma_y(t) &= \sqrt{\sigma_y^2 (B_0 + B_1 t^2 + 2t B_2)} = \\ &= \sqrt{0.0071 + 0.000025 t^2 + 0.00074 t}. \end{aligned}$$

При $t = 30$ получим $\sigma_y(t = 30) = 0.086$ тыс. усл. ед.

Определим доверительные интервалы при уровне доверительной вероятности $\gamma = 0.9$ по таблицам распределения Стьюдента при числе степеней свободы:

$$(k = N - m - 1 = 4 - 1 - 1) = 2: \quad t_{\gamma=0.9, k=2} = 2.92.$$

Интервальный прогноз спроса будет равен

$$Y_{МНК}^{(1)} = y_1 \pm t_r \cdot \sigma_{y(t)} = 3.22 \pm 2.92 \cdot 0.086 = 3.22 \pm 0.25 \text{ тыс. ед.}$$

Верхняя граница $y_B^{(1)} = 3.22 + 0.25 = 3.47$ тыс. ед.

Нижняя граница $y_B^{(1)} = 3.22 - 0.25 = 2.97$ тыс. ед.

Таблица 2.

Метод прогнозирования. Параметры прогноза	1. Интервальный прогноз МНК (период наблюдения $t=30$ месяцев)	2. Интервальный прогноз экспоненциальным сглаживанием (модель Брауна)	3. Точечный прогноз линейной интерполяцией
Среднее значение y , тыс. ед.	3,220	3,400	3,900
Среднее квадратическое отклонение σ_y , тыс. ед.	0,082	0,100	-

Рассчитаем доверительные границы для второго прогноза, полученного методом экспоненциального сглаживания для линейной модели Брауна. Они будут соответственно равны:

$$Y_{B,p}^2 = \bar{y}_2 \pm t_j \cdot \sigma_y = 3.40 + 0.29 = 3.69 \text{ тыс. ед.}$$

Верхняя граница $y_B^{(2)} = 3.40 + 0.29 = 3.69$ тыс. ед.

Нижняя граница $y_B^{(2)} = 3.40 - 0.29 = 3.11$ тыс. ед.

Проверка непротиворечивости полученных значений интегральных оценок прогнозов заключается в построении плотностей распределения $f_i(y)$ прогноза (рис. 2).

Из рис. 2 видно, что первые два метода имеют область пересечения и могут считаться непротиворечивыми. Третий метод (линейная интерполяция) исключаем как противоречивый.

Рассчитаем весовые коэффициенты для определения комбинированного прогноза:

$$\omega_1 = \sigma_2^2 / (\sigma_1^2 + \sigma_2^2) = (0.1)^2 / ((0.086)^2 + (0.1)^2) = 0.57;$$

$$\omega_2 = 1 - 0.57 = 0.43.$$

Определим параметры комбинированного прогноза:

$$y^* = \omega_1 y_1 + \omega_2 y_2 = 0.57 \times 3.22 + 0.43 \times 3.4 = 3.3 \text{ тыс. ед.}$$

$$\begin{aligned} \sigma_{y^*} &= \sqrt{\omega_1^2 \sigma_1^2 + \omega_2^2 \sigma_2^2} = \\ &= \sqrt{0.57^2 \times 0.086^2 + 0.43^2 \times 0.1^2} = 0.065 \text{ тыс. ед.} \end{aligned}$$

Интервальный комбинированный прогноз:

$$y_{н.н.}^* = 3,3 \pm 2,92 \times 0,065 = 3,3 \pm 0,2 \text{ тыс. ед.}$$

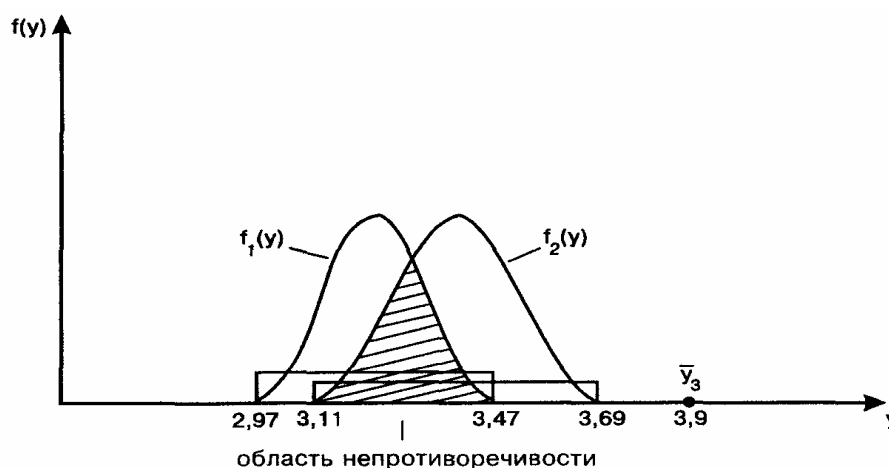


Рис. 2. Определение непротиворечивости прогнозов

8. Выводы

1. Для построения прогнозов логистических показателей на основе использования предположения о наличии и статистической связи между двумя показателями y и x можно эффективно использовать математический аппарат функций с гибкой структурой (ФГС).

2. Основное влияние на структуру ФГС оказывают действительные или комплексные попарно-сопряжённые числа, что позволяет находить оптимальные значения.

3. Оценка точности прогноза и его достоверности осуществляется при помощи прямой, косвенной, консеквентной и инверсной верификаций, которые позволяют оценить достоверность получаемых результатов прогноза.

4. Для повышения точности и достоверности прогнозирования в логистической системе используется комбинированная схема, которая компенсирует недостатки одних методов, достоинствами других.

5. С помощью прогнозов компании устанавливают для себя общие количественные цели, служащие рабочими ориентирами для всей логистической системы. Важная задача состоит в том, чтобы собирать как можно больше информации, анализировать её и своевременно строить на её основе прогнозы с желательной степенью точности.

6. Совершенствование процесса прогнозирования – в результате более полного обмена информацией, повышение качества анализа или накопление опыта – ведёт к значительному сокращению уровня запасов.

Литература

1. M. Mazūra. Predicting Major Trends of Transportation Development. *Transport Engineering* 2002, XXI t., Nr. 2.
2. Lithuania Academy of Science Book 22. A. Baublys, D. Griškevičienė, A. Žvaliauskas. Prognostication of Transport Activity Vilnius, Lithuania science, 1999, 9-61 p.
3. W.-H. Yang, K. Mathur, and R. H. Ballou, “Stochastic Vehicle Routing Problem with Restocking”, *Transportation Science* 34, 99-112, 2000.
4. F. Fumero, and C. Vercellis, “Synchronized Development of Production, Inventory and Distribution Schedules”, *Transportation Science* 33, 330-340, 1999.
5. M. I. Reiman, R. Rubio, and L. M. Wein, “Heavy Traffic Analysis of the Dynamic Stochastic Inventory-Routing Problem”, *Transportation Science* 33, 361-380, 1999.
6. E. D. Miller-Hooks, and H. S. Mahmassani, “Least Expected Time Paths in Stochastic Time - Varying Transportation Networks”, *Transportation Science* 34, 198-215, 2000.
7. H. Yang, “System Optimum, Stochastic User Equilibrium, and Optimal Link Tolls”, *Transportation Science* 33, 354-360, 1999.
8. Д. Дж. Бауерсокс, Д. Дж. Клосс. “Логистика. Интегрированная цепь поставок”, ЗАО “ОЛИМП-БИЗНЕС”, Москва, 2001, 640 с.

ВИРТУАЛЬНЫЕ МОДЕЛИ В СИСТЕМАХ ПРИНЯТИЯ РЕШЕНИЙ НА ЛАТВИЙСКОЙ ЖЕЛЕЗНОЙ ДОРОГЕ

В. Демидов

*Институт транспорта и связи,
Отдел систем управления базами данных ИВЦ ГАО "Latvijas Dzelzceļš"
Тургенева 21, Рига, LV1050, Латвия.
Тел.: (+371)-5834067. Факс: (+371)-5834366. E-mail: dem@ldz.lv*

В настоящей статье рассматриваются вопросы применения виртуальных моделей в системах принятия решений на Латвийской железной дороге на примере статистической отчетности по пассажирским перевозкам.

Ключевые слова: виртуальные модели, формализация, системы принятия решений, хранилища данных

1. Введение

Деятельность транспортного предприятия связана с функционированием множества информационных систем (ИС), отвечающих за те или иные сферы его бизнеса. Данные, генерируемые этими системами, позволяют оценивать эффективность деятельности предприятия и осуществлять перспективное планирование, используя технологии, применяемые в системах принятия решений (Decision Support Systems, DSS). Однако комплексное планирование деятельности всего предприятия затрудняется неоднородностью используемого информационного обеспечения. Например, информационная среда Латвийской железной дороги состоит из 5 крупных ИС, таких как система грузовых перевозок, система пассажирских перевозок, финансовая система, информационные системы инфраструктуры и подвижного состава. Каждая ИС включает в себя множество разнородных подсистем. Зачастую, эти подсистемы между собой совершенно не связаны. Это вызвано тем, что подсистемы развивались обособленно и для их разработки использовались разные технологии и средства. В такой ситуации сложно получить полную картину даже в рамках одной ИС. Так, например, в ИС пассажирских перевозок входят четыре подсистемы, разработанные в разное время на разной программной и аппаратной базе [1]. Эти системы генерируют более 12 миллионов транзакций в год (см. табл.1).

Таблица 1. Системы учета и регистрации пассажирских перевозок на LDz

<i>OLTP системы</i>	<i>Кол-во транзакций в год</i>
Международная система Экспресс-2	2 153 473
Билетно-кассовые аппараты (тип 1)	9 741 701
Билетно-кассовые аппараты (тип 2)	437 770
Ручная продажа	37 638
<i>Итого:</i>	<i>12 370 582</i>

Транзакция – это зарегистрированный факт продажи билета или группы билетов и не является эквивалентом поездки. Билеты могут быть разовыми или многоразовыми, в одном направлении или туда и обратно. Данные, полученные от этих систем, имеют различный формат и степень агрегации. Например, данные от ручной продажи

поступают в систему уже агрегированными по виду билета, станциям продажи, отправления и назначения. Чтобы получить информацию о количестве перевезённых пассажиров, эти данные необходимо подвергнуть дополнительной обработке.

2. Концептуальная схема системы

Для анализа динамики изменения ситуации и выявления закономерностей необходимо длительное время хранить все собранные данные, с возможностью удобного доступа к ним. Для обеспечения эффективного руководства предприятием в целом необходимо наличие DSS системы, использующей объединённые данные всех ИС предприятия.

Современные DSS системы целесообразнее строить на базе хранилищ данных (Data Warehouse, DW), где накапливаются большие объёмы данных из разнородных источников [2]. DW по сравнению с базами данных (Database, DB) оперативных систем (On-Line Transaction Processing, OLTP) наиболее полно описывает функционирование системы в целом, поскольку содержит исторические взаимосвязанные данные жизнедеятельности нескольких OLTP систем за весь период их существования, а также данные из внешних источников.

Транспортное предприятие можно определить как сложную систему, представленную совокупностью всех его ИС. Для анализа функционирования такой системы возникает необходимость создания её реальной модели, которая строится на основе данных, получаемых от различных источников, как это показано на рис. 1.

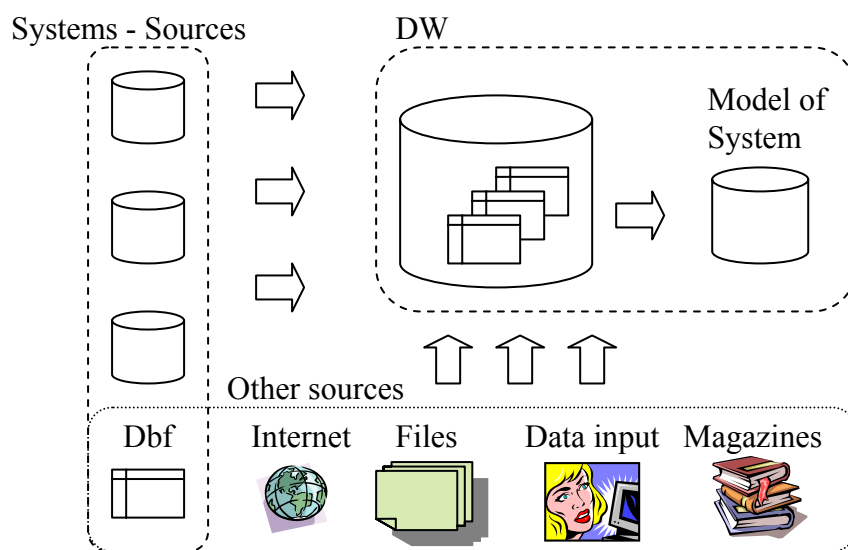


Рис. 1. Концептуальная схема построения аналитической системы

На основе этой модели, применяя методы интеллектуальной обработки данных (Data Mining), можно увидеть, какие воздействия на систему оказывались за наблюдаемый период и как она на них реагировала.

3. Математическая модель

3.1. Пассажиры-километры

Для планирования пассажирских железнодорожных перевозок необходимо проводить детальный анализ пассажиропотоков по отдельным участкам железной дороги на всей территории Латвии. Количество перевезённых пассажиров является важным показателем, характеризующим деятельность перевозчика, однако не даёт

реальной картины «населённости» поездов и нагрузки на те или иные участки железной дороги. Рассмотрим пример, приведённый на рис. 2.

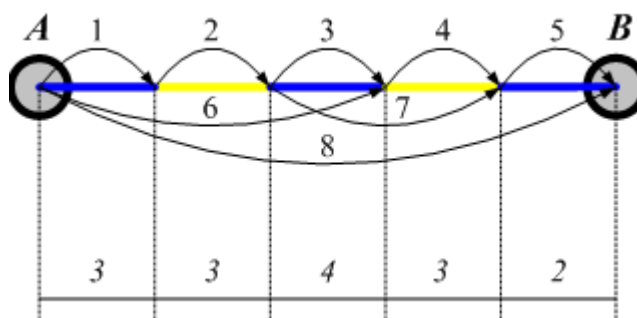


Рис. 2. Схема поездок пассажиров по маршруту

На этом рисунке показана упрощённая схема поездок пассажиров по маршруту от станции *A* до станции *B*. Дугами обозначены поездки, цифры над дугами – это порядковый номер поездки. Цифры внизу рисунка обозначают количество пассажиров, проехавших по соответствующему участку. Как видно из примера, между станциями *A* и *B* было перевезено 8 пассажиров, однако в поезде одновременно находилось не более 4 человек, а на последнем участке всего 2. Однако участки включают в себя несколько станций, и не всегда поездка начинается и заканчивается на границе участка, как показано на рис.3. Поэтому для каждой поездки необходимо определить какое расстояние проехал пассажир по тому или иному участку.

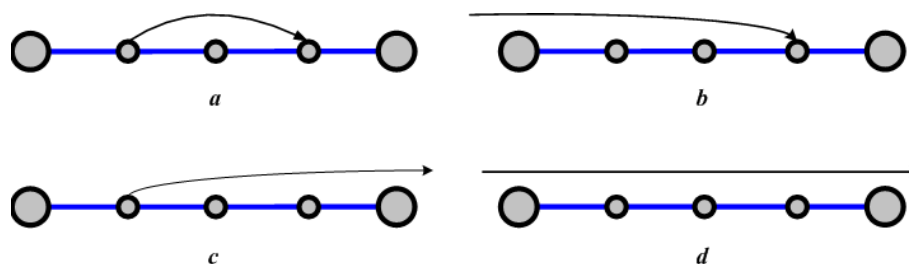


Рис. 3. Типы поездок по участкам

На рис. 3 представлены четыре варианта проезда по участку:

- поездка внутри участка – схема *a*;
- въезд на участок с соседнего участка – схема *b*;
- выезд с участка на соседний участок – схема *c*;
- транзитная поездка через участок – схема *d*.

Так как поездки могут осуществляться и в обратном направлении, то число возможных вариантов удваивается. Таким образом, учитывая сложный механизм расчёта пассажиро-километров и зависимость расчёта от схемы железной дороги, необходима дополнительная обработка данных с наложением каждой поездки на множество участков железной дороги, по которым она происходила. Математическое описание модели сети железной дороги представлено ниже.

3.2. Модели железнодорожных линий и участков

Латвийская железная дорога представляет собой сложную схему с множеством альтернативных путей проезда между двумя станциями. Для анализа пассажиропотоков разработана модель разбивки железной дороги на линии в соответствии с требованиями статистического отчёта СО-25 (см. рис. 4). Как видно из рисунка, некоторые линии частично накладываются друг на друга.

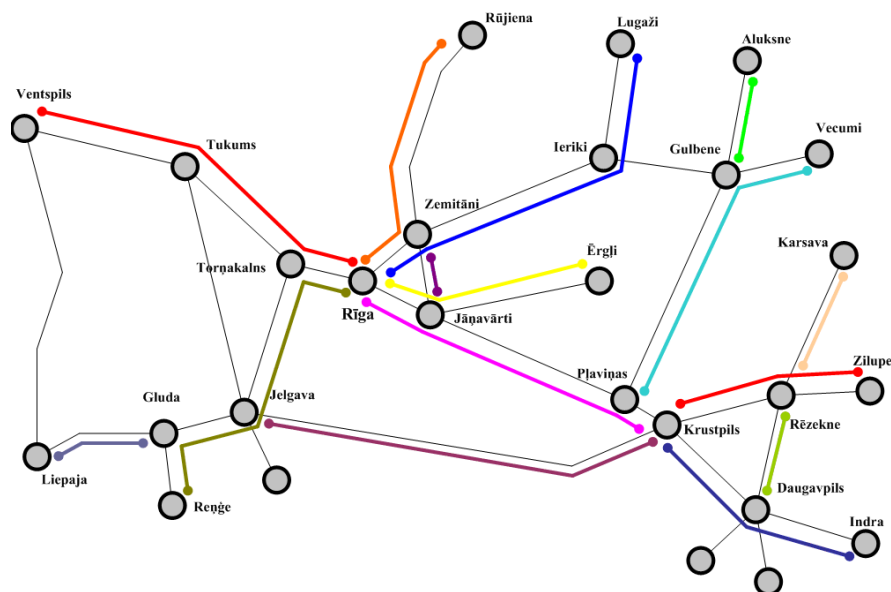


Рис. 4. Модель линий CO-25

Для более углублённого анализа каждая линия разделяется на участки (см. рис. 5) по следующим правилам:

- каждый участок принадлежит только одной линии;
- участки одной линии не перекрываются;
- два смежных участка, принадлежащих одной линии, имеют одну общую станцию;
- объединение всех участков одной линии даёт саму линию.

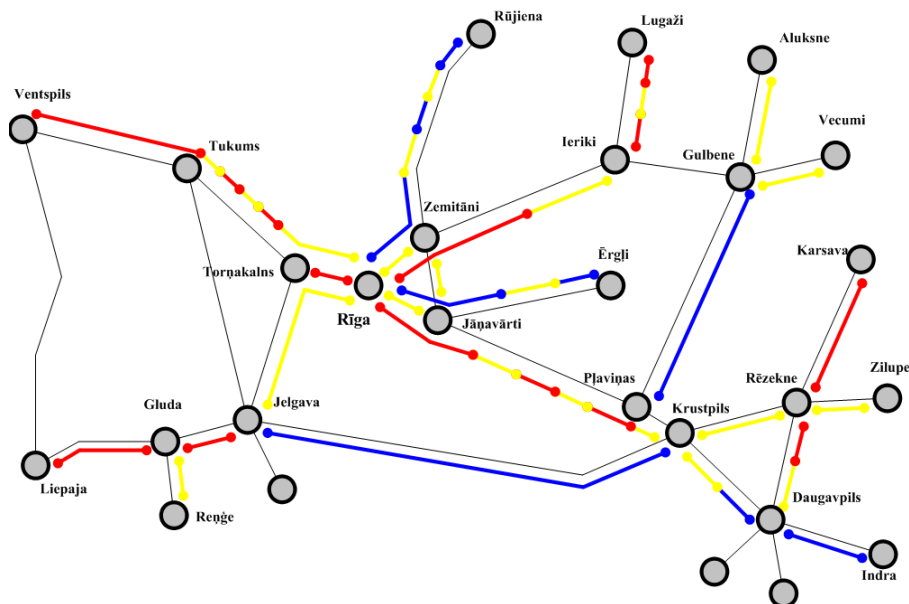


Рис. 5. Модель участков CO-25

Эти модели линий и участков не являются постоянными и со временем изменяются. Некоторые смежные участки могут объединяться в один, или один участок может разделяться на несколько. Могут появляться совершенно новые модели линий и участков. Одновременно допустимо существование сразу несколько моделей. Например, наравне с моделью CO-25 существует модель, разработанная на основе принципа принадлежности участков к территориям самоуправлений. Таким образом, система должна поддерживать несколько моделей сети железной дороги и позволять трансформировать эти модели в процессе анализа.

3.3. Неопределённости

При расчёте пассажиропотоков необходимо определить точный маршрут каждой поездки и наложить её на соответствующие линии и участки. Однако, в исходных данных, поступающих из разных систем, необходимой информации бывает недостаточно, а иногда эта информация бывает не достоверна. Вследствие чего возникает вероятность погрешности в формируемых отчётах. Задача усложняется ещё и тем, что модель СО-25 имеет наложение линий и участков друг на друга и допускает ряд неопределённостей, которые система должна обрабатывать. Рассмотрим фрагмент схемы железной дороги, изображённый на рис.6.

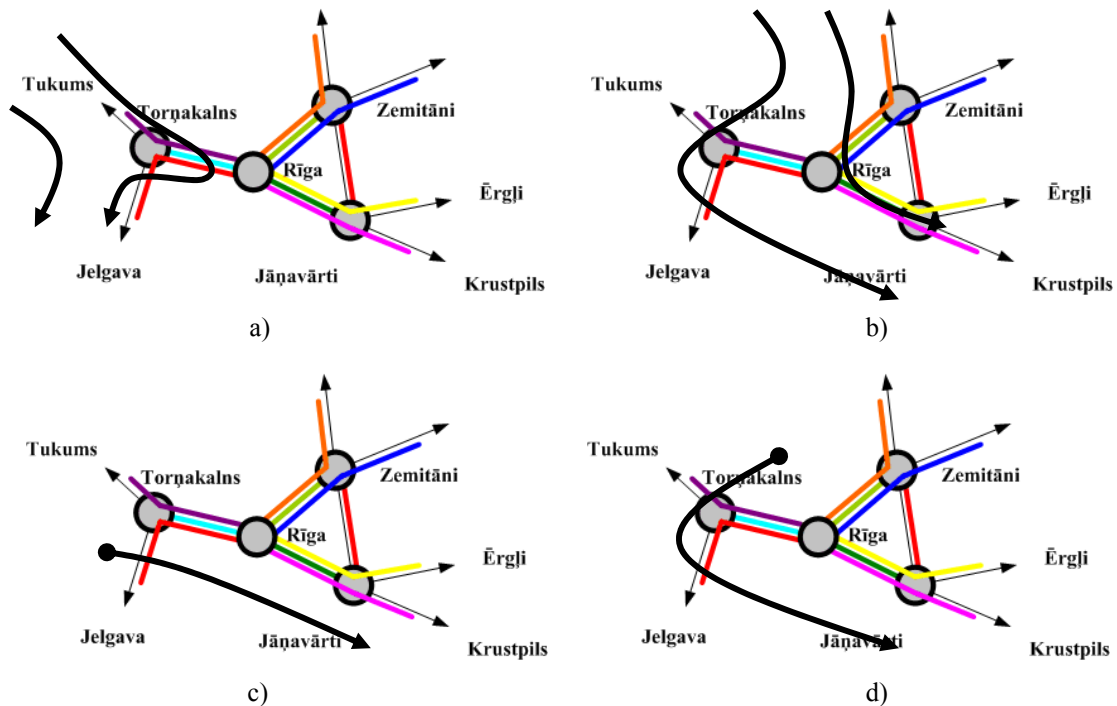


Рис. 6. Неопределённости в исходных данных

На рис. 6.a и 6.b показаны примеры, иллюстрирующие неоднозначность определения маршрута. Если данные о заезде на станцию Rīga отсутствуют, то при наличии нескольких альтернативных маршрутов появляется проблема определения актуального пути, а следовательно, и правильного расчёта пассажиро-километров:

- для маршрутов *Tukums – Torņakalns – Jelgava* или *Tukums – Torņakalns – Rīga – Torņakalns – Jelgava* (рис.6.a) разница в протяжённости составляет 6 километров;
- для маршрутов *Jāņavārti – Zemitāni* и *Jāņavārti – Rīga – Zemitāni* (рис. 6.b) разница в протяжённости составляет 4 километра.

На рис. 6.c и 6.d проиллюстрированы проблемы привязки части поездки к конкретному маршруту в случаях, когда поездка начиналась с таких станций, как *Torņakalns* или *Zemitāni* и далее осуществлялась через станцию *Rīga*. Это происходит потому, что между парами станций *Rīga – Torņakalns*, *Rīga – Zemitāni*, *Rīga – Jāņavārti* расположено по три участка, и определить к какому участку следует отнести поездку невозможно, даже имея все исходные данные из OLTP систем. Для устранения таких неопределённостей в системе должен быть описан ряд правил и ограничений, которые учитываются при расчётах. Учитывая непостоянный характер этих правил и ограничений, они должны быть отделены от модели сети железной дороги и от математической модели расчёта пассажиро-километров и интерпретироваться, как управляющие воздействия на систему [3].

Таким образом, для разработки DSS системы пассажирских перевозок и обеспечения гарантированной достоверности генерируемых отчётов необходимо формализовать следующие модели:

- модель сети железной дороги;
- модель исходных данных OLTP систем;
- модель управляющих воздействий (правила и ограничения);
- математическая модель системы;
- модель взаимодействия этих моделей между собой.

3.4. Формализация постановки задачи

Для формализации постановки задачи и описания моделей введем группу множеств, характеризующих железнодорожные объекты и саму систему в целом.

- $S = \{s_1, s_2, \dots, s_k\}$ – множество всех станций Латвийской железной дороги, где k – мощность множества.
- $P = \{p\}$ – множество пар станций, между которыми может быть осуществлена поездка, где $\forall p = \{s_i, s_j\}$, $s_i \in S$ и $s_j \in S$, $i \neq j$ и $i, j \in \{1, \dots, k\}$.
- $R = \{r_m\}$ – множество всех допустимых маршрутов проезда между двумя парами станций, где $r_m = \{s_1^{(m)}, s_2^{(m)}, \dots, s_\gamma^{(m)}\}$, $s_i^{(m)} \in S$, $i = \overline{1, \gamma}$, $\gamma \leq k$, $m = \overline{1, \beta}$ – идентификатор маршрута, β – мощность множества R .
- $L = \{l_q\}$ – множество линий, где $l_q = \{s_1^{(q)}, s_2^{(q)}, \dots, s_\lambda^{(q)}\}$, $s_i^{(q)} \in S$, $i = \overline{1, \lambda}$, $\lambda \leq k$, $q = \overline{1, \chi}$ – идентификатор линии, χ – мощность множества L .
- $U = \{u_z\}$ – множество участков, где $u_z = \{s_i^{(z)}, s_j^{(z)}\}$, $s_i^{(z)} \in S$ и $s_j^{(z)} \in S$, $i \neq j$, z – идентификатор участка, а $u_z \subseteq l_q$, что означает, что обе станции участка должны принадлежать одной линии $s_i^{(z)} \in l_q$ и $s_j^{(z)} \in l_q$, где $l_q \in L$.
- $Y = \{y\}$ – множество управляющих воздействий. Это правила и ограничения, накладываемые на модели линий и участков при расчёте пассажиро-километров и на сам процесс расчёта.

Опираясь на элементы описанных множеств, можно описать все модели, правила и ограничения, как это показано в следующих примерах:

- если выполняется следующее условие $r_i \cap l_j = \emptyset$, то линия и маршрут не пересекаются;
- если выполняется условие $r_i \cap l_j = l_j$, то линия полностью включена в маршрут;
- если выполняется условие $r_i \cap l_j = r_i$, то маршрут проходит в пределах одной линии;
- если $(r_i \cap l_j \neq \emptyset) \wedge (r_i \cap l_j \neq l_j) \wedge (r_i \cap l_j \neq r_i)$, то линия частично включена в маршрут;
- если $l_i \cap l_j = s_\delta$, где $s_\delta \in S$, то линии l_i и l_j пересекаются;
- если $l_i \cap l_j = \{s_\tau\}$, где $\tau = \overline{1, \varphi}$ и $\varphi > 1$, то линии l_i и l_j частично накладываются друг на друга.

4. Общая постановка задачи

Пусть известна некоторая предыстория системы, которая задается совокупностью значений вектора $X(t)$, измеренных в моменты прошлого t_1, t_2, \dots, t_n , т. е. даны вектора $X(t_1), X(t_2), \dots, X(t_n)$. Будем считать, что t_1 – начальный момент наблюдения системы, а t_n – текущий момент времени.

Для проведения более эффективного анализа работы системы необходимо изменить реальную модель системы с состоянием $X(t)$ на модель, удобную для анализа $X_f^*(t)$ в соответствии с вектором управляющих воздействий $Y_f(t)$, где f – номер модели. В этом случае исследуемая модель системы может быть записана в виде:

$$X^*(t) = F_f [X(t); Y_f(t), f = \overline{1, n}],$$

где n – количество анализируемых моделей, $F_f = \{F_f^{(net)}, F_f^{(math)}, F_f^{(t)}\}$ – некоторый функционал, являющийся по сути обобщённой моделью системы, состоящей из следующих моделей: модели сети железной дороги – $F_f^{(net)}$; математической модели – $F_f^{(math)}$; модели взаимодействия всех моделей между собой – $F_f^{(t)}$.

Данный подход заключается в том, что параллельно создаётся несколько виртуальных моделей, имеющих состояния $X_1^*(t), X_2^*(t), \dots, X_n^*(t)$, как это показано на рис. 7.

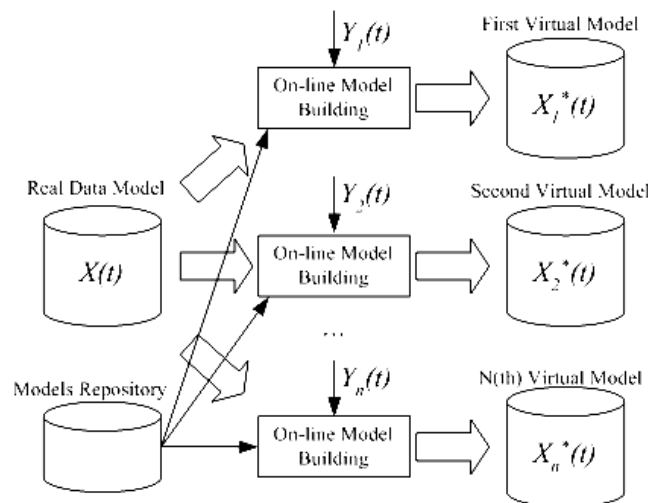


Рис. 7. Концептуальная схема создания виртуальных моделей

На рисунке изображена концептуальная схема создания виртуальных моделей $X_f^*(t)$. Данные о поездках пассажиров, необходимые для расчёта пассажиро-километров, берутся из реальной модели данных $X(t)$, созданной на основе загружаемых из OLTP систем данных. Модели сети железной дороги и математическая модель хранятся в репозитории моделей. Количество виртуальных моделей зависит от количества моделей сети. При построении виртуальной модели данных (On-Line Model Building) учитываются управляющие воздействия $Y_f(t)$, влияющие на процесс трансформации данных реальной модели в виртуальную.

5. Концептуальная модель

В соответствии с предложенной концептуальной схемой (см. рис. 7) была разработана общая модель системы, изображённая на рис. 8. Эта модель состоит из пяти фрагментов:

- Real Data Model – обработанные и описанные данные OLTP систем;
- Models Repository – модели, описывающие линии и участки железной дороги и процесс расчёта нагрузки на них;
- Repository of Control Action – правила и ограничения, управляющие процессом расчёта;
- Virtual Data Models – виртуальные модели данных;
- описание процесса взаимодействия используемых моделей на этапе преобразования данных из Real Data Model в Virtual Data Models.

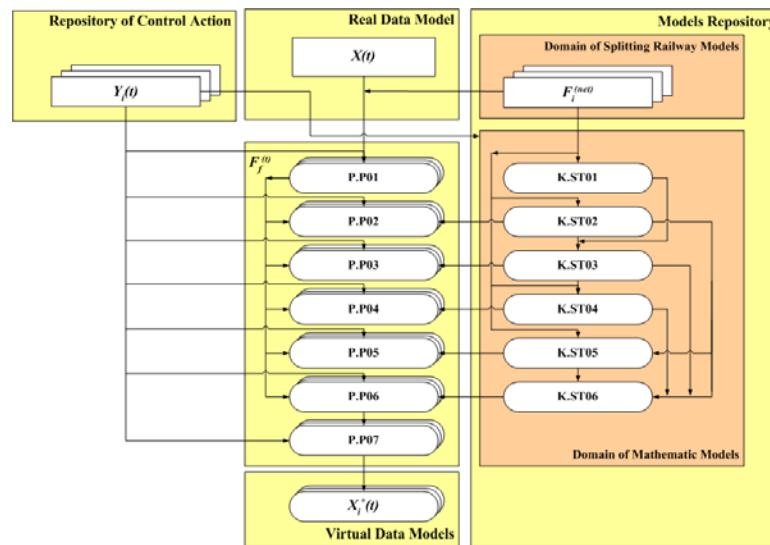


Рис. 8. Общая модель системы

Для обеспечения независимости модели сети железной дороги и модели расчётов друг от друга, эти модели хранятся в репозитории моделей в разных доменах (см. рис. 9). Поясним назначение этих доменов.

1. Domain of Splitting Railway Models – домен моделей разбиения сети железной дороги на линии и участки. В домене могут храниться различные варианты разбиения Латвийской железной дороги и даже других железных дорог.
2. Domain of Mathematic Models – домен абстрактной математической модели расчёта нагрузки на линии и участки, реализованной на базе иерархических представлений (Views) [4].

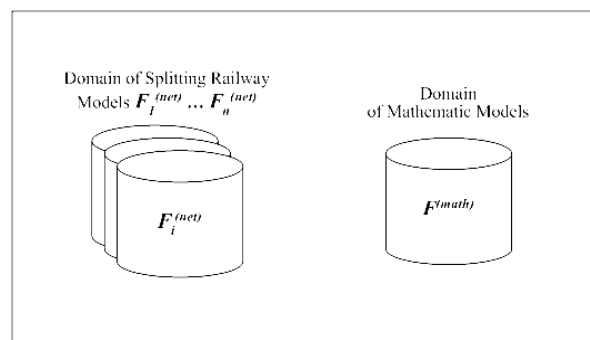


Рис. 9. Схема репозитория моделей данных

Таким образом, при разработке математической модели можно абстрагироваться от конкретной модели железной дороги и её разбивки на линии и участки и создать универсальную модель, независимую от правил распределения пассажиро-километров по спорным участкам и линиям и правил выбора альтернативного маршрута. Математическая модель определяет алгоритм расчёта пассажиро-километров между двумя станциями. Эта модель разделена на логические элементы, вычисляющие различные варианты поездок через абстрактные линии и участки:

- поездка по одной линии;
- поездка между двумя смежными линиями;
- поездка между двумя несмежными линиями;
- поездка по одному участку в пределах одной линии;
- поездка между двумя смежными участками в пределах одной линии;
- поездка между двумя несмежными участками в пределах одной линии.

Вместе эти элементы составляют полный набор вариантов расчёта для любого вида поездки.

Такое разделение моделей в свою очередь при необходимости (например, для получения разных типов отчётов) может иметь единую модель разбивки сети железной дороги для разных математических моделей.

6. Реализация математической модели

Далее рассмотрим примеры фрагментов реализации математической модели, реализованные в системе управления базами данных IBM DB2 Universal Database версии 7.2 на языке стандартного языка запросов к базам данных SQL.

Пример 1. Рассмотрим математическую модель для расчёта пассажиро-километров в поездке по участку, как показано на рис. 3. В этом примере происходит наложение абстрактной поездки из *k.st02* на абстрактный участок из *k.st01*. Привязка к реальным линиям и участкам производится на конечном этапе формирования отчёта в зависимости от выбранной модели разбивки.

Данное представление **VIEW** обрабатывает информацию о любом типе поездки по участку. В приведённом примере приведены запросы, обрабатывающие прямые поездки по участкам.

CREATE VIEW k.st03 AS

```
SELECT u.l, u.u, r.st1, r.st2, r.dist as dist, r.dist
      FROM k.st01 as a, k.st02 as b -- u – участок, r – поездка;
      WHERE u.l=r.l and r.km1>=u.km1 and r.km2<=u.km2 and r.km1<r.km2 OR r.km2>=u.km1
      and r.km1<=u.km2 and r.km1>r.km2
```

UNION

```
SELECT u.l, u.u, r.st1, r.st2, abs(r.km2-u.km1) as dist, r.dist
      FROM k.st01 as a, k.st02 as b
      WHERE u.l=r.l and r.km1<u.km1 and r.km1<r.km2 and r.km2>u.km1 and r.km2<=u.km2
```

UNION

```
SELECT u.l, u.u, r.st1, r.st2, abs(u.km2-r.km1) as dist, r.dist
      FROM k.st01 as a, k.st02 as b
      WHERE u.l=r.l and r.km2>u.km2 and r.km1<r.km2 and r.km1>=u.km1 and r.km1<u.km2
```

UNION

```
SELECT u.l, u.u, r.st1, r.st2, u.dist as dist, r.dist
      FROM k.st01 as a, k.st02 as b
      WHERE u.l=r.l and r.km1<r.km2 and r.km1<u.km1 and r.km2>u.km2 OR r.km1>r.km2 and
      r.km1>u.km2 and r.km2<u.km1
```

Каждый запрос «реагирует» только на свой тип поездки, описанной в условии запроса **WHERE**, и соответственно вычисляет протяжённость поездки по данному участку:

- поездка внутри участка, $dist = r.dist$ – длина всей поездки;
- въезд на участок с соседнего участка, $dist = abs(u.km1-r.km2)$ – длина поездки от начальной станции участка до конца поездки;
- выезд с участка на соседний участок, $dist = abs(r.km1-u.km2)$ – длина поездки от начальной станции поездки до конца участка;
- транзитная поездка через участок, $dist = u.dist$ – длина участка.

Обработка обратных поездок производится аналогичным способом с небольшим изменением условия WHERE. Далее все запросы объединяются оператором UNION.

Пример 2. Рассмотрим математическую модель для расчёта пассажиро-километров в поездке между двумя смежными линиями, как показано на рис. 10.

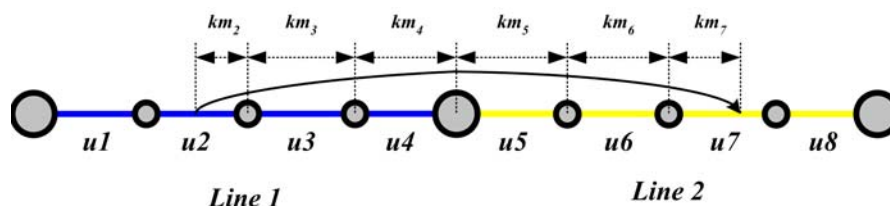


Рис. 10. Поездка по двум смежным линиям

Пассажир совершил поездку со станции участка $u2$ линии $L1$ до станции участка $u7$ линии $L2$, т.е. он частично проехал по участкам $u2$ и $u7$ и полностью по участкам $u3$, $u4$, $u5$ и $u6$. В верхней части рисунка над участками с $u2$ по $u7$ указано расстояние km_i , которое проехал пассажир по соответствующему i -му участку. На данном этапе определяется протяжённость поездки по каждому участку – km_i и по каждой линии: $km_{L1} = km_2 + km_3 + km_4$ и $km_{L2} = km_5 + km_6 + km_7$. Ниже показана реализация данной логики.

```
CREATE VIEW k.st04 (l, u, st1, st2, dist, km) AS
WITH CT (L1,L2,ST) as
(SELECT a.l, b.l, a.st FROM k.st_ch as a, k.st_ch as b WHERE a.st = b.st and a.l <> b.l)
SELECT a.l, a.u, a.st1, b.st2, a.dist, a.km+b.km
FROM k.st03 as a, k.st03 as b
WHERE a.st2 = b.st1 and a.st2 in (SELECT st FROM CT WHERE L1=a.l and L2=b.l)
UNION
SELECT b.l, b.u, a.st1, b.st2, b.dist, a.km+b.km
FROM k.st03 as a, k.st03 as b
WHERE a.st2 = b.st1 and a.st2 in (SELECT st FROM CT WHERE L1=a.l and L2 = b.l);
```

Вначале модель должна определить, что поездка осуществлялась между смежными линиями, т.е. $a.st = b.st$ and $a.l \neq b.l$. Так как линии могут частично перекрываться, то для определения станции, на которой происходит стыковка линий, используются данные таблицы $k.st_ch$, в которой описаны правила переезда с одной линии на другую. Далее происходит объединение множеств участков $U_{L1} = \{u2, u3, u4\}$ и $U_{L2} = \{u5, u6, u7\}$, по которым осуществлялась поездка, а для расчёта пассажиро-километров по входящим в линии $L1$ и $L2$ участкам используется алгоритм, заложенный в представлении $k.st03$.

Таким образом, представление $k.st04$ производит расчёт пассажиро-километров для абстрактной поездки по всем абстрактным участкам двух смежных абстрактных линий. Наложив эту модель на конкретную модель сети железной дороги и на исходные данные по фактам поездок за определённых промежутков времени, мы получим расчёт пассажиро-километров по всем поездкам между смежными линиями.

Совокупность всех представлений математической модели с $k.st01$ по $k.st06$ обрабатывает любой вариант поездок по любой описанной модели сети железной дороги.

Выводы

Предлагаемая концептуальная схема создания виртуальных моделей, использующая единую и независимую от схемы сети железной дороги математическую модель, даёт возможность параллельно иметь несколько анализируемых моделей данных и быстро создавать новые. Такой подход расширяет границы применения системы и на других железных дорогах. Для этого нет необходимости перестраивать всю систему, надо только описать схему сети соответствующей железной дороги и управляющие воздействия и загрузить новые транзакционные данные в реальную модель данных в DW.

Применение виртуальных моделей в совокупности с чёткой формализацией системы позволяет сократить время разработки новых отчётов и обеспечить их гарантированную достоверность.

Описываемая в этой статье система разработана и апробирована на Латвийской железной дороге и зарекомендовала себя гибкой системой, легко адаптируемой к постоянно меняющимся условиям бизнеса и требованиям потребителей информации.

Литература

1. E. Kopytov, N. Petoukhova, V. Demidov. Methodology of Huge Data Volume Processing System Development for Analysis of Latvian Railway Passengers Transportation. In: Proceedings of VI International Conference "TransBaltica 2001", June 7-8, 2001, Riga, Latvia. pp. 201-208.
2. Eric Sperley. Enterprise Data Warehouse. Planning, Building, and Implementation. Prentice Hall, Inc, 1999.
3. J. Kopitovs, V. Demidovs, N. Petoukhova. Virtual Models in Forecasting Systems of Railway Transportation. The international conference: Modelling and Simulation of Business Systems (MOSIBUS 2003). Vilnius, Lithuania, May 13-14, 2003. - pp. 265-268.
4. В.В. Демидов. Использование методов повышения достоверности данных в системах принятия решений на Латвийской железной дороге. В кн.: Transport and Telecommunication. Volume 4. Nr.1 – 2003, Riga, Latvia. - С. 16-22.

ПРИНЦИПЫ РАЗРАБОТКИ КОМПЛЕКСНЫХ СИСТЕМ БЕЗОПАСНОСТИ В РЕЛЯЦИОННЫХ БАЗАХ ДАННЫХ НА ПРИМЕРЕ ЛАТВИЙСКОЙ ЖЕЛЕЗНОЙ ДОРОГИ

Наталья Петухова

*Институт транспорта и связи
Ломоносова 1, Рига, LV 1019, Латвия
Тел.: (+371) 5834067. E-mail: natalia@ldz.lv*

В предлагаемой работе рассматривается подход к разработке специализированной системы комплексного контроля доступа в реляционных базах данных. Комплексность системы заключается в возможности реализации любых правил политики безопасности. В системе контроля доступа учитывается семантика защищаемых данных. Система построена на базе метода ограничения доступа к данным на уровне строк отношения. Приводится формальное описание задачи комплексного контроля доступа.

Ключевые слова: безопасность, реляционная база данных, контроль доступа, политика безопасности

1. Введение

Вопрос безопасности корпоративных данных всегда особенно остро стоял перед крупными предприятиями, такими как Латвийская железная дорога (LDz). Важность этого вопроса объясняется двумя основными причинами: ценностью накопленных информационных ресурсов и наличием развитой информационной инфраструктуры. Инфраструктура включает более тысячи пользователей вычислительной сети, из которых несколько сотен активно используют корпоративную базу данных (БД), управляя общими ресурсами предприятия [1].

Безопасное управление общими корпоративными ресурсами должно соответствовать политике безопасности (ПБ) предприятия, основанной на огромном количестве законов, норм, внутренних распоряжений. В качестве примера можно привести следующие задачи LDz: система учета объектов недвижимости, распределенных по всей территории Латвии, рабочее место товарного кассира, учет компьютерной периферии. Только в перечисленных задачах ПБ предполагает применение различных аспектов разграничения доступа, таких как территориальное и административное разделение, временные рамки, функциональные роли и конфиденциальность информации. Но, несмотря на то, что вопрос безопасности реляционных данных на сегодняшний день рассмотрен достаточно широко, внедрение в информационные системы (ИС) таких комплексных условий ПБ представляет серьезную проблему.

В настоящей работе автор предлагает решение этого вопроса, заключающееся в разработке систем комплексного контроля доступа (ККД) в реляционных базах данных. При этом использованы результаты предыдущей работы, а именно, применение метода ограничения доступа к данным реляционных систем на уровне кортежей отношения [2].

2. Проблемы известных подходов к защите данных

На сегодняшний день известны два подхода к управлению доступом к реляционным данным: дискреционный и мандатный.

Дискреционный контроль доступа позволяет разграничить доступ между поименованными субъектами и поименованными объектами [3, 4]. Он включен в состав большинства коммерческих реляционных продуктов и удовлетворяет

большинству информационных систем, но ему присущи ограничения в гибкости управления.

Примером элемента дискреционного подхода может служить SQL команда управления привилегиями GRANT. Синтаксис команды GRANT выглядит следующим образом:

<i>Оператор назначения полномочий доступа к объекту</i>	GRANT	
<i>Привилегии доступа</i>	(INSERT, DELETE, SELECT, UPDATE) (column-name)	<i>Доступ на чтение и модификацию можно ограничить несколькими столбцами</i>
<i>Тип объекта</i>	ON TABLE	
<i>Имя объекта</i>	table-name view-name	
	TO (USER, GROUP)	
<i>Имя пользователя или группы пользователей</i>	authorization-name	
<i>Привилегия передачи полномочий</i>	WITH GRANT OPTION	

Из синтаксиса команды видно, что привилегии доступа для пользователей (authorization-name) ассоциированы с поименованными информационными структурами (table-name, column-name, view-name), в которых содержатся данные, а связь с самими данными отсутствует. Системе безопасности ничего не известно об их семантике. Привилегии существуют отдельно от самих данных, которые, с точки зрения обеспечения безопасности, являются как бы обезличенными. Мандатный контроль доступа, с этой точки зрения, предпочтительней дискреционного.

Мандатный контроль доступа разграничивает доступ субъектов к объектам данных, основываясь на характеризуемой меткой конфиденциальности информации, которая содержится в объектах, и на официальном разрешении субъектов обращаться к информации такого уровня конфиденциальности [3, 4]. Для мандатного подхода примечательно жесткое следование рекомендациям «Оранжевой книги» Министерства обороны США и модели безопасности данных Bell La Padule [5]. Известные реализации выполнены в виде закрытых систем, что накладывает серьезные ограничения на переносимость. Этим объясняется ограниченное применение в реальных ИС Латвийской железной дороги, политика безопасности которой не укладывается в модели мандатного подхода.

Применение классических подходов, рассмотренных выше, не решает поставленной задачи, что требует поиска новых методов решения этой проблемы, в большей мере удовлетворяющих требованиям безопасности ИС LDz.

3. Разработка комплексных систем безопасности

Сформулируем требования к комплексной системе безопасности.

- Уровень доступа к данным определяется самими данными и зависит от их семантики, т. е. сами данные участвуют в процессе ограничения доступа к ним.
- Система контроля доступа, при необходимости, может работать как часть бизнес-системы и следовать реальным правилам поведения субъектов и объектов доступа ИС, т. е. возможно функционирование системы безопасности без прямого административного вмешательства.
- Система должна позволять реализацию любой политики безопасности и любых ее комплексных правил.
- В системе должна быть возможность организации процессов подотчетности (протоколирование действий пользователей).
- К одним и тем же наборам данных конкретный пользователь может иметь различный доступ на разные операции. Например, множества данных одного

и того же информационного объекта, доступные для чтения, удаления и модификации, могут различаться.

Доступ к данным должен зависеть от их семантики, а не от структурных элементов, в которых они располагаются. Следовательно, перед нами ставится задача разграничения доступа внутри поименованных информационных объектов, таких как реляционная таблица. Метод обеспечения доступа к данным на уровне кортежей отношения в реляционных системах был рассмотрен в работе автора [2].

Объектом защиты предложенного метода являются данные таблицы. Применяя этот метод можно ограничить доступ к данным таблицы таким образом, чтобы пользователь к одной записи таблицы смог получить доступ, а к другой – нет. Ограничение или разрешение доступа происходит на основе специальных меток безопасности строки, идентификатора пользователя и определенных правил безопасности.

Под *меткой безопасности* данных будем понимать некую метку, присутствующую в данных и являющуюся критерием доступа к ним. Другими словами, метка безопасности – это атрибут отношения, определяющий доступ к кортежу. Таким атрибутом может быть как специализированный атрибут безопасности, так и любой другой атрибут или его скалярная производная.

Технически разграничение доступа к данным отношения обеспечивается созданием одного представления (виртуальной таблицы), через которое происходит любое взаимодействие пользователей с исходной таблицей, и двух триггеров, ограничивающих доступ на модификацию и удаление данных.

Иллюстрация метода ограничения доступа на уровне кортежей отношения приведена на рис. 1.

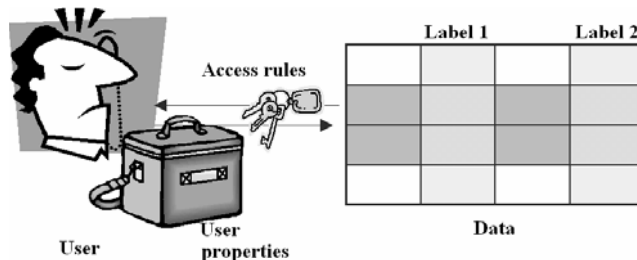


Рис. 1. Доступ пользователя к реляционной таблице

Как видно из рисунка, пользователь User, запрашивая данные у таблицы Data, проходит через слой системы безопасности, определенный правилами доступа Access rules. Согласно этим правилам, он получает доступ только к избранным записям, которые выделены на рисунке темным цветом. Доступные для пользователя записи определяются по значениям меток Label 1 и Label 2, которые соотносятся с полномочиями пользователя User properties на работу с данными, помеченными такими метками.

3.1. Формализация задачи ограничения доступа к кортежу отношения

Для формализации задачи определим группы множеств, участвующие в процессе ККД. Опишем последовательно кортежи, к которым следует ограничивать доступ, специализированные объекты системы СКК и элементы запроса пользователя.

Кортеж, к которому следует ограничивать доступ.

Структура кортежа t отношения T имеет вид:

$$t = \{A, L\},$$

где $A = \{a\}$ – множество атрибутов кортежа, не влияющих на доступ к нему;

$L = \{l\}$ – множество меток безопасности кортежа (множество атрибутов кортежа, определяющих доступ к нему) мощностью m .

Специализированные объекты системы контроля доступа.

Введем следующие множества:

$U = \{u\}$ – множество зарегистрированных пользователей системы;

$O = \{o\}$ – множество возможных операций доступа к кортежу;

$Q = \{Q_1, Q_2, \dots, Q_m\}$ – множество доменов значений меток безопасности, определенных для каждой метки $l_i \in L$, $i = \overline{1, m}$.

где $Q_i = \{q^{(i)}\}$ – домен значений меток безопасности, определенный для метки $l_i \in L$;

$q^{(i)}$ – возможное значение i -й метки;

$S = \{S_1, S_2, \dots, S_m\}$ – множество доменов пользовательских свойств, влияющих на доступ к меткам $l_i \in L$, где $i = \overline{1, m}$;

$S_i = \{s^{(i)}\}$ – домен значений пользовательских свойств, влияющих на доступ к метке $l_i \in L$;

$s^{(i)}$ – возможное значение пользовательского свойства.

$R = \{R_1, R_2, \dots, R_m\}$ – множество наборов правил доступа к меткам из L ,

где $R_i = \{r_1^{(i)}, r_2^{(i)}, \dots, r_{k_i}^{(i)}\}$ – набор правил доступа к метке $l_i \in L$;

k_i – количество правил доступа к метке $l_i \in L$, $i = \overline{1, m}$.

Правило $r_j^{(i)} \in R_i$ регламентирует доступ пользователей со свойством $p \in S_i$ к метке $l_i \in L$ со значением $q \in Q_i$, где $i = \overline{1, m}$, j – номер правила.

Структура правила $r_j^{(i)} \in R_i$ приведена ниже:

$$r_j^{(i)} = \langle p, q, o \rangle, \text{ где } p \in S_i, q \in Q_i, o \in O, j \text{ – номер правила.}$$

Элементы запроса пользователя к данным кортежа:

$u \in U$ – пользователь, пославший запрос;

$o^{(u)} \in O$ – операция запроса пользователя u ;

$P^{(u)} = \{p^{(u)} \in S\}$ – множество свойств пользователя $u \in U$, пославшего запрос;

$P_i^{(u)} \subseteq P^{(u)}$ – множество свойств пользователя, влияющих на его доступ к метке $l_i \in L$, $i = \overline{1, m}$, т. е. имеем $P_i^{(u)} \subseteq S_i$;

$L(u, o) \subseteq Q$ – множество наборов значений меток, доступных пользователю для выполнения операции $o^{(u)} \in O$ над кортежем t , определяемое как $L(u, o) = \{l_1(u, o), l_2(u, o), \dots, l_m(u, o)\}$,

$l_i(u, o) \subseteq Q_i$ – набор значений меток, доступных пользователю $u \in U$ для выполнения операции $o^{(u)} \in O$ над меткой $l_i \in L$, $i = \overline{1, m}$, который вычисляется следующим образом: $l_i(u, o) = P_i^{(u)} \times_{F_i, o^{(u)}} R_i$,

где $l_i(u, o) \in L(u, o)$, $i = \overline{1, m}$, \times – оператор декартова произведения,

F_i – набор функций ПБ, определенных для метки $l_i \in L$.

Вычисление доступности кортежа.

Множество фактов доступности меток кортежа – это булев вектор X , который определяется проверкой вхождения множества меток кортежа L в соответствующие множества доступных для пользователя меток:

$$X = L \subseteq L(u, o), \text{ то есть } x_i = \begin{cases} \text{true, если } l_i \in l_i(u, o), \\ \text{false, если } l_i \notin l_i(u, o), \end{cases} i = \overline{1, m}$$

Тогда факт доступности кортежа y определяется следующим образом:

$$y = B(X),$$

где B – булева функция. Иными словами, множество значений y есть $\{\text{true}, \text{false}\}$.

В случае истинности y доступ к кортежу $t = \{A, L\}$ разрешается и ограничивается в противном случае.

3.2. Задача протоколирования

Важным средством обеспечения безопасности является механизм протоколирования. Доверенная система должна фиксировать все события, касающиеся безопасности. Ведение протоколов должно дополняться аудитом, то есть анализом регистрационной информации [4].

Если фиксировать все события, объем регистрационной информации, скорее всего, будет расти слишком быстро, а ее эффективный анализ станет невозможным. Система ККД предусматривает наличие средств выборочного протоколирования, как в отношении пользователей (обращать внимание только на подозрительных), их свойств и признаков, так и в отношении операций и времени обращения.

Для математического описания задачи протоколирования введем следующие обозначения:

$O_z \subseteq O$ – множество протоколируемых операций;

$A_z \subseteq A$ – множество протоколируемых атрибутов;

$U_z \subseteq U$ – множество протоколируемых пользователей;

$P_z \subseteq S$ – множество протоколируемых свойств пользователей;

$T_z = \{\tau^{(z)}\}$ – множество протоколируемых периодов времени.

Процесс протоколирования запускается в случае соблюдения следующего условия:

$$(o^{(u)} \in O_z) \vee (A^{(u)} \subseteq A_z) \vee (u \in U_z) \vee (P^{(u)} \subseteq P_z) \vee (\tau^{(u)} \in T_z),$$

где $o^{(u)}, A^{(u)}, u, P^{(u)}, \tau^{(u)}$ – соответствующие множества окружения запроса пользователя.

При протоколировании события рекомендуется записывать следующую информацию:

- дата и время события;
- уникальный идентификатор пользователя – инициатора действия;
- тип события;
- имена затронутых объектов;
- старые и новые значения объектов.

Приведенный перечень протоколируемых параметров могут быть расширены исходя из возможностей той или иной СУБД.

4. Пример реализации комплексного контроля доступа

Учет аппаратного обеспечения LDz является показательным примером задачи комплексного контроля доступа.

Политика безопасности задачи учета формулируется следующим образом:

- доступ к данным аппаратных устройств определяется территориальными и административными сферами полномочий пользователей;
- при определении полномочий учитывается иерархия администраций;
- доступ возможен, если территориальные и административные атрибуты устройства находятся в сферах полномочий пользователя.

Пример показателен также тем, что в нем контроль доступа не является абстрагированным модулем политики безопасности, он находится в сильной зависимости от реального положения дел на предприятии, а именно, от меняющейся иерархии структурных подразделений и от принадлежности пользователя к той или иной административной структуре. Эти данные меняются в процессе жизнедеятельности предприятия. Система ККД мгновенно учитывает эти изменения, корректируя процессы доступа к данным.

Инструментами реализации системы ККД являются стандартные средства СУБД: SQL, триггеры и представления. Система выполнена на платформе СУБД IBM DB2 UDB v.7.2

Структура базы данных (БД) примера представлена на Рис. 2.

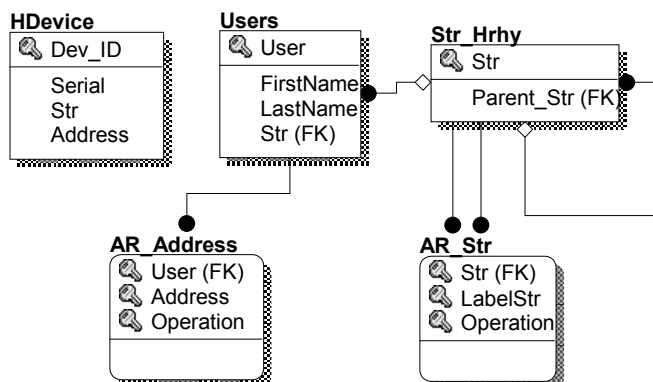


Рис. 2. ER модель БД

БД содержит следующие таблицы:

HDevice – таблица устройств вычислительной техники, к которой необходимо контролировать доступ;

Users – перечень всех пользователей системы с указанием их административной принадлежности (атрибут **Str**);

Str_Hrhy – иерархия структурных подразделений (администраций).

Перечисленные таблицы содержат только бизнес данные задачи учета аппаратного обеспечения, и в них не присутствуют специализированные элементы системы безопасности. Однако система безопасности использует некоторые из этих данных.

Специализированными таблицами системы безопасности являются:

AR_Address – территориальные полномочия доступа пользователей;

AR_Str– правила административного разделения доступа.

Рассмотрим формализацию данной задачи.

Формализация задачи примера

Информация об устройствах содержится в отношении **Hdevice**. Каждый кортеж этого отношения соответствует отдельному устройству вычислительной техники. Формально описать задачу контроля доступа к данным устройства, базируясь на формализации, описанной в п.3.1, можно следующим образом.

Структуру кортежа t отношения HDevice представим в виде: $t = \{A, L\}$, где $A = \{Dev_ID, Serial\}$ – множество бизнес атрибутов кортежа, не влияющих на ограничение доступа к нему;

L – множество меток безопасности кортежа, $L = \{l_{Str}, l_{Address}\}$.

Опишем специализированные объекты системы контроля доступа.

$Q = \{Q_{Str}, Q_{Address}\}$ – множество доменов значений меток безопасности, определенных для каждой метки из L .

$S = \{S_{Str}, S_{Address}\}$ – множество доменов свойств пользователя, состоящее из двух элементов: S_{Str} – домена структурных подразделений, определяющих административные сферы полномочий пользователей и $S_{Address}$ – домена адресов, которые уполномочен обслуживать пользователь. В данном примере $Q \subseteq S$.

$R = \{R_{Str}, R_{Address}\}$ – множество, состоящее из двух наборов правил доступа к меткам из L : $R_{Str} = \{r_{Str}\}$ и $R_{Address} = \{r_{Address}\}$.

$r_{Str} = \langle p_{Str}^{(r)}, l_{Str}^{(r)}, o^{(r)} \rangle$ – множество правил, ограничивающих доступ пользователей по административной сфере их полномочий, где $p_{Str}^{(r)} \in S_{Str}$, $l_{Str}^{(r)} \in Q_{Str}$, $o^{(r)} \in O$.

$r_{Address} = \langle p_{Address}^{(r)}, l_{Address}^{(r)}, o^{(r)} \rangle$ – множество правил, ограничивающих доступ пользователей по территориальной сфере их полномочий,

где $p_{Address}^{(r)} \in S_{Address}$ – значение адреса из домена адресов;

$l_{Address}^{(r)} \in Q_{Address}$ – значение метки из домена меток;

$o^{(r)} \in O$ – операция из домена операций.

Далее опишем элементы запроса пользователя к данным кортежа.

$o^{(u)} \in O$ – операция запроса.

$P^{(u)} = \{P_{Str}^{(u)}, P_{Address}^{(u)}\}$ – множества свойств пользователя $u \in U$, пославшего запрос, состоящее из $P_{Str}^{(u)} = \{p_{Str}^{(u)} \in S_{Str}\}$ – множества свойств пользователя, определяющее его административную сферу полномочий и $P_{Address}^{(u)} = \{p_{Address}^{(u)} \in S_{Address}\}$ – множества свойств пользователя, определяющее его территориальную сферу полномочий. Множество $P_{Str}^{(u)}$ строится на базе отношений Users, Str_Hrhy и AR_Str. Множество $P_{Address}^{(u)}$ строится на базе отношения AR_Address.

В момент запроса определяются множества меток $L_{Str}(u, o) = \{l_{Str}(u, o) \in Q_{Str}\}$, и $L_{Address}(u, o) = \{l_{Address}(u, o) \in Q_{Address}\}$, доступных пользователю $u \in U$ для выполнения операции $o^{(u)} \in O$. Множества вычисляются следующим образом:

$$l_{Str}(u, o) = p_{Str}^{(u)} \underset{F_{Str}}{\times} r_{Str},$$

где F_{Str} – функция ПБ, определенная для метки Str , вида:

$$(o^{(u)} = r_{Str} \cdot o) \vee (p_{Str}^{(u)} \cdot Str = r_{Str} \cdot Str);$$

$$l_{Address}(u, o) = p_{Address}^{(u)} \underset{F_{Address}}{\times} r_{Address},$$

где $F_{Address}$ – функция ПБ, определенная для метки $Address$, вида:

$$(o^{(u)} = r_{Address} \cdot o) \vee (p_{Address}^{(u)} \cdot Address = r_{Address} \cdot Address).$$

$X = L \subseteq L(u, o)$ – множество фактов доступности меток устройства, иначе

$$X = \{x_{str}, x_{Address}\} = \{l_{Str} \in L_{Str}(u, o), l_{Address} \in L_{Address}(u, o)\}$$

Тогда факт доступности записи об устройстве y определяется следующим образом:

$$y = B(X) = x_{str} \vee x_{Address}, \text{ где } B - \text{булева функция.}$$

В случае истинности y доступ к устройству разрешается, в противном случае – ограничивается.

Функция доступности записи об устройстве $y = B(X)$ определяется формулой реляционной алгебры:

$$y = \sigma_{x_{Str} \vee x_{Address}}(T),$$

где $x_{Str} = Str \subseteq \pi_{Str} \sigma_f (AR_Str \times Str_Hrhy \times Users)$;

$$x_{Address} = Address \subseteq \pi_{Address} \sigma_f (AR_Address);$$

$$f = (Operation = OPERATION) \vee (User = USER);$$

OPERATION – операция доступа;

USER – системная переменная – идентификатор пользователя, осуществляющего доступ.

Полученное выражение $y = B(X)$ достаточно несложно представить **на языке запросов SQL**:

```

y =      Hdevice.Str IN (
                SELECT a.Str from AR_Str AS a INNER JOIN Str_Hrhy AS s
                ON a.Str=s.Str INNER JOIN Users AS u ON s. Parent_Str=u.Str
                WHERE u.User=USER AND a.Operation=OPERATION
            )
AND
Hdevice.Address IN
            (
                SELECT a.Address from AR_ Address AS a
                WHERE a.User=USER AND a.Operation=OPERATION
            )
    
```

где *OPERATION* – операция доступа,

USER – системная переменная – идентификатор пользователя, осуществляющего доступ.

Представление, ограничивающее чтение данных таблицы¹ имеет вид:

```

create view net.v_ar_hdevice as
select * from HDevice
where Hdevice.Str IN (
                SELECT a.Str from AR_Str AS a INNER JOIN Str_Hrhy AS s
                ON a.Str=s.Str INNER JOIN Users AS u ON s. Parent_Str=u.Str
                WHERE u.User=USER AND a.Operation='Select'
            )
AND
Hdevice.Address IN
            (
                SELECT a.Address from AR_ Address AS a
                WHERE a.User=USER AND a.Operation='Select'
            )
    
```

Триггер, ограничивающий обновление данных об устройстве записывается в виде:

¹ Все триггеры и представления данных разработаны для IBM DB2 UDB v.7.2


```
create trigger net.tu_ar_hdevice no cascade before update on hdevice
referencing old as o new as n for each row mode db2sql
when ( not
      o.Str IN (
        SELECT a.Str from AR_Str AS a INNER JOIN Str_Hrhy AS s
        ON a.Str=s.Str INNER JOIN Users AS u ON s. Parent_Str=u.Str
        WHERE u.User=USER AND a.Operation='Update'
      )
AND
      o.Address IN
      (
        SELECT a.Address from AR_Address AS a
        WHERE a.User=USER AND a.Operation='Update'
      )
) signal sqlstate 'you have not privileges to update this record'
```

В теле представления и триггера в явном виде представлена формула вычисления доступности записи об устройстве $y = B(X)$. При изменении политики безопасности и, следовательно, доработки функции вычисления $y = B(X)$, она легко заменяется новой без изменения общей архитектуры системы безопасности.

5. Выводы

Предложенная система контроля доступа формирует для пользователя свой взгляд на общую базу данных, не позволяя ему выходить за рамки отведенной для него подсистемы БД. Система реализуется стандартными средствами СУБД: триггерами и представлениями. Математический аппарат системы – реляционная алгебра. Задача ККД относительно легко поддается формализации.

Комплексный контроль доступа нашел свое применение в информационных системах LDz, решил большинство проблем, связанных с безопасностью данных и показал свою надежность, гибкость применения и использования.

Предложенный метод контроля доступа может быть востребован и в других ИС, так как в его основе лежит условие адаптации к любым правилам корпорации. При этом использование только стандартных средств в реализации ККД делает возможным создание системы ККД на платформе практически любой современной реляционной СУБД, т.е. система ККД является переносимой.

Литература

1. Е. Копытов, N. Petoukhova, V. Demidov. Methodology of Huge Data Volume Processing System Development for Analysis of Latvian Railway Passengers Transportation. In: *Proceedings of VI International Conference "TransBaltica 2001"*, June 7-8, 2001, Riga, Latvia. pp. 201-208.
2. Петухова Н. «Детализированный доступ к данным, как средство повышения безопасности информационных систем». In: *Proceedings of the International Conference "Reliability and Statistics in Transportation and Communication (RelStat'02)"*, October 17-18, 2002, Riga, Latvia. - p. 73-74.
3. Козленко Л., «Информационная безопасность в современных системах управления базами данных», *КомпьютерПресс* 3, 2002
4. Галатенко В. А., Курс лекций «Основы информационной безопасности», http://publish.abitu.ru/courseslibrary/ims/c_yfksu/secbasics/lectures/lecture+5.esp
5. Алан Р. Саймон. Стратегические технологии баз данных: Менеджмент на 2000 год. Пер. с англ./Под ред. и с предисл. М.Р. Коголовского. – М.: Финансы и статистика, 1999. 402-405с.