

МЕТОДИКА ПЕРЕХОДА ОТ IDEF0 К МОДЕЛИ В ТЕРМИНАХ ТЕОРИИ СИСТЕМ МАССОВОГО ОБСЛУЖИВАНИЯ ПРИ ИССЛЕДОВАНИИ БИЗНЕС-ПРОЦЕССОВ ОРГАНИЗАЦИИ

Тихонов С. В.

(Южный федеральный университет, Ростов-на-Дону)

kfs001@mail.ru

В работе предлагается методика построения модели бизнес-процессов произвольной организации, сформулированной в терминах теории систем массового обслуживания (СМО) на основе их описания средствами функционального моделирования IDEF0. Методика основана на сужении интерпретаций исходных примитивов IDEF0 и однозначном сопоставлении им основных элементов СМО, что позволяет использовать ее безотносительно содержания рассматриваемых бизнес-процессов.

Ключевые слова: бизнес-процессы, IDEF0, дискретно-событийное моделирование, СМО.

Введение

С помощью моделирования различных областей деятельности можно достаточно эффективно анализировать «узкие места» в управлении и оптимизировать общую схему бизнеса. Понятие «моделирование бизнес-процессов» пришло в быт большинства аналитиков одновременно с появлением на рынке сложных программных продуктов, предназначенных для комплексной автоматизации управления предприятием. Для решения подобных задач моделирования сложных систем существуют хорошо обкатанные методологии и стандарты, к которым относятся, в первую очередь, методологии семейства IDEF. С их помощью можно эффективно отображать и анализировать модели дея-

тельности широкого спектра сложных систем в различных разрезах. Наибольшее распространение получила методология функционального моделирования IDEF0. Однако сам по себе IDEF0 недостаточен для построения модели, адекватной для предсказания поведения системы в ответ на управляющие воздействия и выработки предупреждающих и корректирующих мер, прежде всего, вследствие наличия логико-лингвистических противоречий в определении исходных примитивов (см., напр., [2]).

Для построения имитационной модели системы на основе ее описания с использованием методологии IDEF0 требуется, во-первых, адекватная целям моделирования интерпретация стандарта, во-вторых, набор правил, согласно которым исходным примитивам IDEF0 ставятся в соответствие основные понятия выбранного аппарата моделирования.

В данной работе предлагается методика перехода от описания системы в IDEF0 к дискретно-событийной модели с использованием аппарата систем массового обслуживания (СМО) для построения адекватной имитационной модели бизнес-процессов.

Как уже было сказано, для формального описания бизнес-процессов используется аппарат специального раздела теории вероятностей – теории систем массового обслуживания. Это класс математических моделей, используемых при анализе функционирования таких сложных систем, как автоматические телефонные станции, автоматизированные информационные системы, системы связи, диспетчерские службы, системы снабжения, поточные линии и т. д. Во всех этих случаях мы имеем дело с массовой обработкой некоторых объектов при учете влияния случайных факторов. Для повышения эффективности работы таких систем определяют их статистические характеристики, учитывающие наличие очередей, ожидание начала обслуживания, простой, среднее время нахождения в системе и др. Все эти величины описываются случайными величинами и случайными процессами.

Рассматриваемые бизнес-процессы допускают весьма удобную и наглядную интерпретацию с точки зрения теории систем массового обслуживания и тесно связанной с ней теории входящего потока.

Основные понятия стандарта IDEF0 рассмотрены в [1, 3]. Рассмотрим основные этапы перехода от описания бизнес-процессов, выполненного средствами IDEF0, к модели рассматриваемой системы, позволяющей трактовать ее в терминах теории СМО.

1. Сопоставление основных элементов модели в терминах СМО исходным примитивам IDEF0

Итак, необходимо выработать методику перехода от функционального описания системы, обеспечиваемого IDEF0, к событийному, лежащему в основе аппарата СМО, а также описать правила перехода от абстракций стандарта IDEF0 к основным понятиям, используемым при описании системы как СМО. Изначально событийной интерпретации стандарт IDEF0 не предусматривает, поскольку на это ориентирован стандарт IDEF3 (см. [4]).

Однако, как показано в [2], путем упомянутого выше сужения интерпретаций его исходных примитивов можно получить достаточно простые правила, позволяющие определять моменты начала (а иногда и окончания) работ, пользуясь только аппаратом IDEF0. Впрочем, эти правила плохо подходят для перехода к модели в терминах СМО, делая ее слишком абстрактной и неудобной для последующей формализации. В данной работе используется другая методика, позволяющая однозначно поставить в соответствие исходным примитивам IDEF0 основные элементы модели СМО и тем самым легко переходить к ним от IDEF0-диаграмм безотносительно содержания последних.

Входы процессов рассматриваются в качестве поступающих заявок. Очевидно, что IDEF0-диаграммы отображают входы как классы объектов, в то время как входами для СМО явля-

ются конкретные экземпляры этих классов (это утверждение справедливо и для остальных элементов модели).

Выходы процессов рассматриваются в качестве исходящих потоков заявок.

В соответствии с понятием декомпозиции рассматривается иерархия заявок, соответствующая различным уровням детализации процессов на диаграммах IDEF0. В частности, контекстной диаграмме (определение см. в [3]) ставятся в соответствие заявки 0-го уровня, дочерним блокам контекстной диаграммы – заявки 1-го уровня и т. д. Тогда бизнес-процесс s -го уровня логично понимать как процесс обслуживания заявки s -го уровня.

Обслуженная заявка данного класса покидает систему, вместо нее в систему поступают (в качестве входного потока), вообще говоря, несколько заявок других классов (очевидно, возможна и обратная ситуация), поскольку число входов и выходов процесса может не совпадать.

Более сложным для интерпретации в рамках аппарата СМО является понятие «управление». В данной работе управления процесса рассматриваются также в качестве входящего потока событий, т. е. входящий поток событий «управления» является потоком поступления представителей классов управляющих воздействий, поскольку каждое такое событие изменяет состояние системы. Это позволяет учесть два важных момента: во-первых, требование обязательного наличия управления при осуществлении какого-либо процесса, во-вторых, возможность корректировки процесса в процессе моделирования.

Проиллюстрируем сказанное на примере схемы абстрактного бизнес-процесса. Рассмотрим модель процесса с N входами, K выходами и L управлениями:

Обозначим:

$$(1) \quad \{z_{in}, i \geq 1\}_{n=1}^N$$

– входящие потоки заявок, где z_{in} – интервал времени между поступлением $(i - 1)$ -ой и i -ой заявки входящего потока с номером n ;



Рис. 1. Пример схемы абстрактного процесса

$$(2) \{u_{ik}, i \geq 1\}_{k=1}^K$$

– исходящие потоки заявок, где u_{ik} – интервал времени между моментами ухода $(i - 1)$ -ой и i -ой заявки исходящего потока с номером k ;

$$(3) \{w_{il}, i \geq 1\}_{l=1}^L$$

– потоки поступления управляющих воздействий, где w_{il} – интервал времени между моментами поступления $(i - 1)$ -го и i -го управляющего воздействия типа l .

Для сопоставления входящих/исходящих потоков и соответствующих им управлений требуется дополнительная информация о структуре системы, нежели диаграмма процесса рас-

смаатриваемого уровня. Ее источником, очевидно, являются диаграммы дочерних процессов (для удобства на рис. 2 диаграммы дочерних процессов наложены на диаграмму родительского процесса):

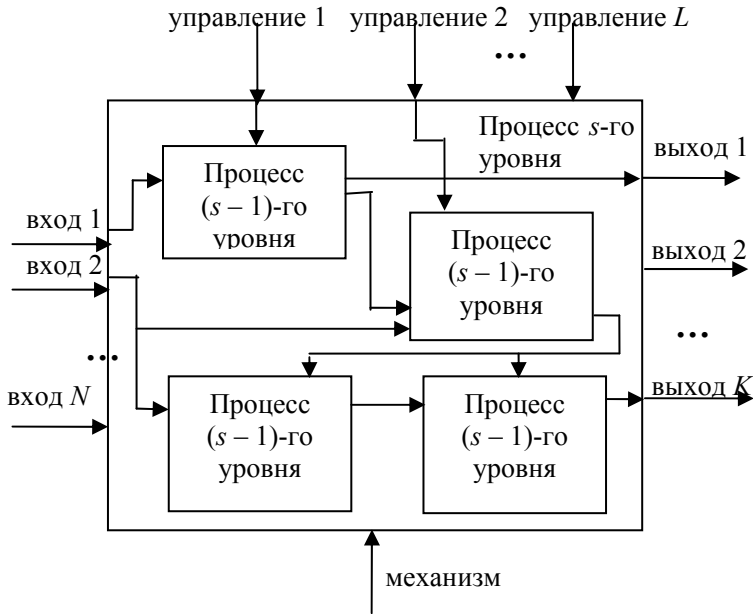


Рис. 2. Процесс с наложенными диаграммами дочерних процессов

В данном примере входящий поток, соответствующий входу 1, сопоставляется с исходящими потоками 1, 2 и K; входящий поток, соответствующий входу 2, сопоставляется с исходящими потоками 2 и K. Следует заметить, что сопоставление это может быть не прямым, т. е. через один дочерний процесс, а опосредованным, т. е. через несколько дочерних процессов, для которых выход одного из них будет входом или даже управляющим воздействием для другого.

Таким образом, каждому входящему потоку может соответствовать не один, а несколько исходящих потоков, как, впрочем, и наоборот.

Аналогичные рассуждения применимы и к потокам управляющих воздействий.

Указанное сопоставление удобно представить в виде матрицы M размера $N + L \times K$, элементы которой

$$(4) \quad m_{ij} = \begin{cases} 1, & \text{входу/ управлению } i \text{ соответствует выход } j \\ 0, & \text{в противном случае} \end{cases},$$

$i = 1, \dots, N + L, j = 1, \dots, K$.

Рассмотрим теперь важнейший элемент СМО: время обслуживания заявок. Пусть входящая заявка в процессе обслуживания проходит ряд этапов (т. е. связана с исходящей заявкой опосредованно, через ряд дочерних процессов), и на вход каждого из них подается несколько внешних потоков заявок (от родительского процесса) и несколько – от других дочерних процессов (рис. 3).

Эти этапы образуют цепочку переходов входящей заявки типа i родительского процесса в заявки его дочерних процессов до ее выхода в качестве исходящей заявки типа j родительского процесса. Если цепочка раздваивается на каком-либо из этапов (что соответствует, например, нескольким альтернативным процедурам принятия решения), то будем рассматривать две цепочки, каждая из которых соответствует разным путям перехода заявок.

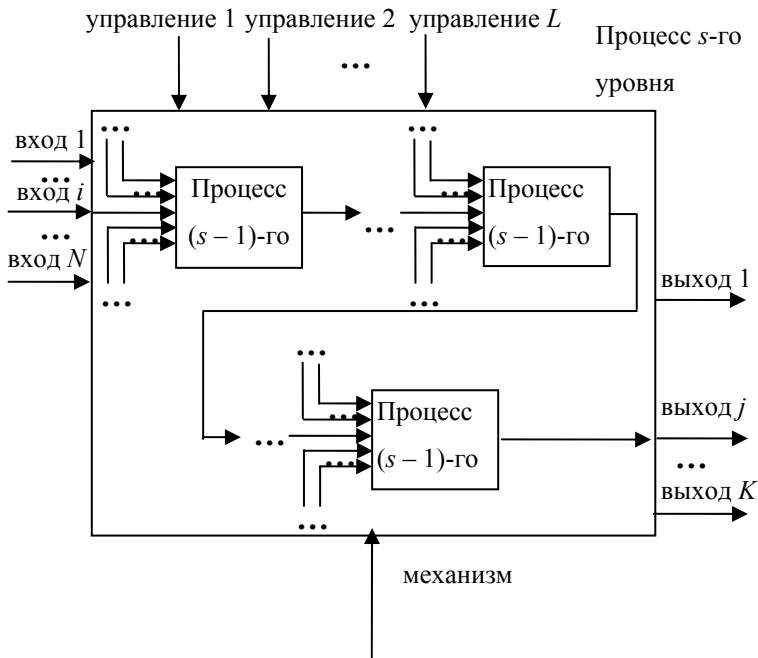


Рис. 3. Этапы обслуживания заявки

Введем обозначения:

t'_p – момент генерации исходящей заявки p -го дочернего процесса, входящего в цепочку, связывающую i -й вход родительского процесса и его j -й выход;

τ_p – время генерации выхода вышеуказанного дочернего процесса в ответ на поступившие входы;

t_s – момент поступления входящей заявки s -го потока;

$\hat{\tau}_s$ – время ожидания заявки s -го потока в очереди по причине занятости устройства

Тогда очевидным будет соотношение

$$(5) \quad t'_p = \max_{\substack{r \in R_p \\ s \in S_p}} \{t'_r + \hat{\tau}_r, t_s + \hat{\tau}_s\} + \tau_p,$$

где R_p – множество входящих потоков p -го дочернего процесса, являющихся исходящими потоками других дочерних процессов; S_p – множество входящих потоков p -го дочернего процесса, являющихся внешними по отношению к родительскому процессу.

Тогда момент t'_j генерации выхода j родительского процесса в ответ на его вход i будет определяться соотношением

$$(6) \quad t'_j = t'_p = \max_{\substack{r \in R_p \\ s \in S_p}} \{t'_r + \hat{t}_r, t_s + \hat{t}_s\} + \tau_p,$$

или

$$(7) \quad \tau_j = t'_j - t_i = \max_{\substack{r \in R_p \\ s \in S_p}} \{t'_r + \hat{t}_r, t_s + \hat{t}_s\} + \tau_p - t_i$$

для каждой из цепочек дочерних процессов, соответствующих переходу $i \rightarrow j$.

Далее, поскольку для каждого из процессов, стоящих в цепочке ранее, также выполняется соотношение

$$(8) \quad t'_p = \max_{\substack{r \in R_p \\ s \in S_p}} \{t'_r + \hat{t}_r, t_s + \hat{t}_s\} + \tau_p,$$

то

$$\begin{aligned} t'_j = t'_p &= \max_{\substack{r \in R_p \\ s \in S_p}} \{t'_r + \hat{t}_r, t_s + \hat{t}_s\} + \tau_p = \\ &= \max_{\substack{r \in R_p \\ s \in S_p}} \left\{ \max_{\substack{r_1 \in R_r \\ s_1 \in S_r}} \{t'_{r_1} + \hat{t}_{r_1}, t_{s_1} + \hat{t}_{s_1}\} + \tau_r + \hat{t}_r, t_s + \hat{t}_s \right\} + \tau_p = \dots = \\ (9) \quad &= \max_{\substack{r \in R_p \\ s \in S_p}} \left\{ \max_{\substack{r_1 \in R_r \\ s_1 \in S_r}} \left\{ \dots \max_{\substack{r_{p-1} \in R_{p-1} \\ s_{p-1} \in S_{p-1}}} \{t'_{r_{p-1}} + \hat{t}_{r_{p-1}}, t_{s_{p-1}} + \hat{t}_{s_{p-1}}\} + \tau_{r_{p-2}} + \right. \right. \\ &\quad \left. \left. + \hat{t}_{r_{p-2}}, \dots + \hat{t}_{r_1}, t_{s_1} + \hat{t}_{s_1}\right\} + \tau_r + \hat{t}_r, t_s + \hat{t}_s \right\} + \tau_p \end{aligned}$$

Здесь было сделано одно важное допущение, касающееся определения уровня детализации модели и учета моментов поступления входящих/исходящих заявок. А именно, генерация

выхода дочернего процесса начинается **только** после поступления **всех** входов, связанных с ним. Иначе говоря, ситуация ожидания на каком-либо из этапов внешней входящей заявки невозможна. Это позволяет не детализировать при рассмотрении процесса уровня s модель ниже уровня $s - 1$, рассматривая вместо этого величины τ_i дочерних процессов уровня $s - 1$. Таким образом, детализация модели определяется тем уровнем, на котором указанным фактом можно пренебречь.

Следует сделать замечание по поводу управлений процесса: несмотря на то, что управляющие воздействия, как и заявки, представлены входящими потоками событий, они имеют существенные отличия. Во-первых, время нахождения управляющего воздействия в системе равно нулю, поскольку оно оказывает влияние на переменные состояния системы и не требует никакой «обработки», поэтому, во-вторых, оно не участвует в вычислении операционных характеристик системы как СМО.

Механизм осуществления процесса логично представить в виде устройства обслуживания, на которое поступают входящие заявки и управления.

Поскольку зачастую в организационных системах бизнес-процессы одновременно обрабатывают несколько экземпляров входов (генерируют несколько экземпляров выходов), к примеру, отдел работает над несколькими проектами сразу, устройства обслуживания являются, вообще говоря, многоканальными.

Таким образом, получаем сеть СМО с неординарными (поскольку заявки могут поступать и группами), вообще говоря, произвольными входящими потоками. Обслуживание поступивших в блок требований осуществляется различными устройствами, допускающими очередь неограниченной длины. Поток обслуживаний в общем случае также является произвольным, поскольку величины t_s и τ_p случайны.

Полученная таким образом модель бизнес-процессов как СМО уже пригодна для проведения экспериментов и вычисления некоторых функциональных характеристик, связанных с наличием очередей, вынужденным ожиданием начала обслужи-

вания, простом прибором и т. п. С практической точки зрения она может быть полезна на этапе проектирования, позволяя оптимальным образом увязать во времени выполнение работ и процессов. В области управления бизнес-процессами чистая модель СМО полезна для оценки управленческих решений, изменяющих структуру системы.

Заключение

Предложенная методика построения дискретно-событийной модели бизнес-процессов с использованием аппарата СМО на основе IDEF0 является достаточно универсальной и может применяться для моделирования бизнес-процессов любой организации. В рамках работы в данной области описанная методика применяется для имитационного моделирования основных бизнес-процессов инвестиционно-строительной организации, целью которого является разработка инструмента, служащего основой для выработки и оценки управленческих решений.

Литература

1. ВЕРНИКОВ Г. *Обзор стандарта IDEF0*. Интернет-ресурс: www.idefinfo.ru.
2. РУБЦОВ С. *Опыт использования стандарта IDEF0 // Открытые системы*. 2003. №11.
3. *Integration Definition for Function Modeling (IDEF0)*. Draft Federal Information Processing Standards Publication 183. 1993, December 21.
4. *Information Integration for Concurrent Engineering (IICE) IDEF3 Process Description Capture*. Method Report. Knowledge Based Systems, Incorporated. 1995, September.

*Статья представлена к публикации
членом редакционной коллегии В.М. Вишневым*