УДК 651:681.3.06

ЯЗЫК ОПИСАНИЯ ПРОЦЕССОВ ДЕЛОПРОИЗВОДСТВА И ДОКУМЕНТООБОРОТА

Т.Г. Белова
(представил д.т.н., проф. Г.Г. Асеев)

Рассматриваются элементы языка описания деловых процессов, основанного на исчислении предикатов первого порядка. Приводится пример описания деловых процессов регистрации и согласования документов.

В рамках технологии workflow деловой процесс определяется как логически завершенный набор операций (деловых процедур), поддерживающих структуру предприятия (организации) и реализующих его политику, направленную на достижение поставленной цели [1]. Для органов государственной власти и управления основными деловыми процессами являются процессы делопроизводства и документооборота, направленные на документальное отражение и обеспечение управленческих процессов. В силу этого они приобретают доминирующий характер и нуждаются в собственной системе управления, использующей принципы технологии workflow.

Рассмотрим одну из задач, решаемых при создании workflow-системы – разработку описания деловых процессов. Процессы делопроизводства и документооборота в органах государственной власти и управления имеют сложный и динамичный характер, слабо формализуются, изменяются во времени [2]. Возникает потребность в разработке универсального языка описания деловых процессов, который даст возможность учесть приведенные выше особенности и создать базу данных (БД), содержащую правила выполнения деловых процессов.

Введем формальный язык описания деловых процессов (ЯОДП), основанный на многосортном исчислении предикатов первого порядка. Выбор в качестве ядра ЯОДП исчисления предикатов обусловлен его ясной формальной семантикой и операционной поддержкой в том смысле, что для него разработаны механизмы вывода [3]. В основе ЯОДП лежит формальная система, задаваемая как

$$\mathbf{M} = <\mathbf{T}, \mathbf{P}, \mathbf{Q}, \mathbf{R}>, \qquad (1)$$

где **T** – множество базовых элементов, для которых существует процедура **П(T)**, позволяющая за конечное число шагов определить, является ли элемент **x** элементом множества **T**; **P** – множество синтаксических правил, с помощью которых из элементов множества **T** образуются синтаксически правильные совокупности; **Q** – множество аксиом, $\mathbf{Q} \subseteq \mathbf{P}$; процедура **П(Q)** определяет, принадлежит ли синтаксически правильная совокуп-

© Т.Г. Белова, 2000

ность **X** множеству **Q**; **R** – множество правил вывода, причем процедура **П(R)** определяет, является ли синтаксически правильная совокупность **X** выводимой.

Множество базовых понятий **T** в ЯОДП содержит основные понятия технологии workflow [4] и состоит из следующих сущностей (в терминологии логических моделей – сортов): **Pr** – Процесс; **Op** – Операция; **Rl** – Исполнитель; **Ob** – Объект; **Ev** – Событие; **Time** – Время; **tOp**, **tRl**, **tOb**, **tEv** – соответственно типы Операции, Исполнителя, Объекта и События.

Зададим сигнатуру **Σ** ЯОДП, состоящую из множества выражений вида $\mathbf{f{:}A_1{\times}A_2{\times}...{\times}A_n{\rightarrow}B}$, где $\mathbf{A_j}$, **B** – сорта; **f** – функция или предикат в **Σ**, если **B={0,1}**. Сорта сигнатуры интерпретируются как множества.

Рассмотрим фрагмент предметной области, содержащей описание процессов регистрации и согласования документов, наиболее характерных для органов государственной власти и управления. Сигнатуру фрагмента зададим следующими отношениями и предикатами:

$$\mathbf{f_1{:}Op{\rightarrow}tOp;\ f_2{:}Rl{\rightarrow}tRl;\ f_3{:}Ob{\rightarrow}tOb;\ f_4{:}Ev{\rightarrow}tEv;} \tag{2}$$

$$\mathbf{нач{:}Op{\rightarrow}Time;\ кон{:}Op{\rightarrow}Time;} \tag{3}$$

$$\mathbf{p_1{:}Pr{\times}Op{\rightarrow}\{0,1\};\ p_2{:}Ev{\times}Pr{\rightarrow}\{0,1\};\ p_3{:}Op{\times}Rl{\rightarrow}\{0,1\};\ p_4{:}Op{\times}Ob{\rightarrow}\{0,1\};} \tag{4}$$

$$\mathbf{согл\_док{:}\ Pr{\rightarrow}\{0,1\};\ регистр\_док{:}\ Pr{\rightarrow}\{0,1\};} \tag{5}$$

$$\mathbf{вх\_док{:}\ {\rightarrow}tOp;\ распор\_док{\rightarrow}tOp;\ нормат\_док{\rightarrow}tOp;} \tag{6}$$

$$\mathbf{нач\_отд{:}{\rightarrow}tRl;\ делопроизв{:}{\rightarrow}tRl;} \tag{7}$$

$$\mathbf{созд\_док{:}\ {\rightarrow}tEv;\ получ\_док{:}\ {\rightarrow}tEv;} \tag{8}$$

$$\mathbf{индекс{:}\ {\rightarrow}tOb;\ доработка{:}\ {\rightarrow}tOb;\ созд\_ркк{:}{\rightarrow}tOb;\ архив{:}{\rightarrow}tOb;} \tag{9}$$

$$\mathbf{изучить\_док{:}{\rightarrow}tOb;\ сравнить\_док{:}\ {\rightarrow}tOb;\ проверить\_ссылки{:}\ {\rightarrow}tOb;} \tag{10}$$

$$\mathbf{проверить\_нормативы{:}\ {\rightarrow}tOb;\ контроль\_исполн{:}\ {\rightarrow}tOb;} \tag{11}$$

$$\mathbf{принять\_согласов{:}\ {\rightarrow}tOb;\ отклонить\_согласов{:}\ {\rightarrow}tOb;} \tag{12}$$

$$\mathbf{0{:}{\rightarrow}Time;\ 1{:}{\rightarrow}Time;\ ...\ 1{:}{\rightarrow}Time;} \tag{13}$$

$$\mathbf{+{:}Time{\times}Time{\rightarrow}Time;\ {\le}{:}Time{\times}Time{\rightarrow}\{0,1\}.} \tag{14}$$

Выражения (2) задают функции, значениями которых являются соответствующие сорта. В (3) задаются функции начала и конца сорта Операция, значения функций принадлежат к сорту Время. В (4) задаются двухместные предикаты, имеющие следующую интерпретацию: $\mathbf{p_1}$ – «операция участвует в процессе»; $\mathbf{p_2}$ – «событие инициирует выполнение процесса»; $\mathbf{p_3}$ – «исполнитель участвует в операции»; $\mathbf{p_4}$ – «операция использует объект». В (5) заданы два одноместных предиката **согл_док** (согласование документа) и **регистр_док** (регистрация документа). **согл_док(Pr)=1** тогда и только тогда, когда **Pr** является процессом согласования документов (для **регистр_док** аналогично). Выражения (6) – (12) задают константы соответствующих сортов, (13) – (14) задают константы, функцию и предикат для сорта Время.

Расширенная сигнатура **Σ‘** включает (2) – (14) и все литерные константы, означивающие элементы множеств, которые интерпретируют в Σ сорта.

Сигнатура задает в ЯОДП структурные связи между понятиями предметной области, представленными предикатами и функциями. Логические связи задаются формулами, которые записываются в сигнатуре.

Обозначим через Z^{Σ} множество всех синтаксически правильных выражений, построенных на основе сигнатуры Σ. Зададим правила **P** построения синтаксически правильных совокупностей.

Пусть **Tm** – множество всех термов, $\mathbf{Tm}_{\tau}$ – множество термов сорта τ, причем $\mathbf{Tm}_{\tau} \subseteq \mathbf{Tm}$. Зададим **Tm** следующим образом.

П1: Если $\mathbf{c}_{\tau}$ – переменная, принимающая значения из сорта τ, то $\mathbf{c}_{\tau} \in \mathbf{Tm}_{\tau}$.

П2: Если сигнатура Σ содержит выражение $\mathbf{f{:}A_1{\times}A_2{\times}...{\times}A_n{\rightarrow}B}$, причем $\mathbf{B \neq \{0,1\}}$, и $\tau_1 \in \mathbf{Tm}_{A1} ... \tau_n \in \mathbf{Tm}_{An}$, то $\mathbf{f(\tau_1,\tau_2,...\tau_n) \in Tm_B}$.

Атомы Z^{Σ} составляются из термов, предикатных символов и знака равенства по следующим правилам:

П3: Если сигнатура Σ содержит выражение $\mathbf{p{:}A_1{\times}A_2{\times}...{\times}A_n{\rightarrow}\{0,1\}}$, и $\tau_1 \in \mathbf{Tm}_{A1} ... \tau_n \in \mathbf{Tm}_{An}$, то $\mathbf{p(\tau_1,\tau_2,...\tau_n)}$ – атом.

П4: Если $\nu \in \mathbf{Tm}_{\tau}$ и $\sigma \in \mathbf{Tm}_{\tau}$, то выражение $\nu = \sigma$ есть атом.

Произвольная формула Z^{Σ} составляется из атомарных формул (атомов) с использованием логических связок $\neg$, $\wedge$, $\vee$, $\rightarrow$ и кванторов по следующим правилам.

П5: Каждый атом есть формула. Всякая входящая в атом переменная является свободной в этом атоме.

П6: Если α и β уже построенные формулы, то выражения $\neg\alpha$, $(\alpha\wedge\beta)$, $(\alpha\vee\beta)$, $(\alpha\rightarrow\beta)$ также формулы. Всякая переменная, входящая свободно (связанно) в α или β, входит свободно (связанно) и в новые формулы.

П7: Если α уже построенная формула, в которую свободно входит переменная **x**, значения которой принадлежат к сорту **A**, то выражения $(\forall \mathbf{x} \in \mathbf{A})\alpha$ и $(\exists \mathbf{x} \in \mathbf{A})\alpha$ также являются формулами. Переменная **x** считается связанной в новых формулах.

Используя правила П1 – П7, построим для сигнатуры Σ (2) – (14), описывающей фрагмент предметной области, множество формул Ω, $\Omega \subseteq Z^{\Sigma}$, истинных в (2) – (14):

$$(\forall x \in Ob)(\exists y,z \in Pr).\{\neg(f_3(x)=\text{вх\_док}) \wedge \text{согл\_док}(y) \wedge \text{регистр\_док}(z) \wedge$$
$$\wedge(\forall u,v \in Op).p_1(y,u) \wedge p_1(z,v) \wedge p_4(u,x) \wedge p_4(v,x) \wedge (\text{кон}(u) \leq \text{нач}(v))\}; \quad (15)$$
$$(\forall z \in Pr)(\exists y \in Op).[p_1(z,y) \wedge (f_1(y)=\text{контроль\_исполн})]; \quad (16)$$
$$(\forall x \in Ob)(\exists z \in Pr).[\neg(f_3(x)=\text{вх\_док}) \wedge \text{согл\_док}(z) \wedge (\forall y,u,v,g,r \in Op).$$
$$.(f_1(y)=\text{изучить\_док}) \wedge p_1(z,y) \wedge p_4(y,x) \wedge (f_1(u)=\text{сравнить \_док}) \wedge p_1(z,u) \wedge$$
$$\wedge p_4(u,x) \wedge (f_1(v)=\text{проверить\_ссылки}) \wedge p_1(z,v) \wedge p_4(v,x) \wedge$$
$$\wedge(f_1(g)=\text{проверить\_нормативы}) \wedge p_1(z,g) \wedge p_4(g,x) \wedge p_1(z,r) \wedge p_4(r,x) \wedge$$
$$\wedge\{(f_1(r)=\text{принять\_согласов}) \vee (f_1(r)=\text{отклонить\_согласов})\} \wedge$$
$$\wedge(\text{кон}(y) \leq \text{нач}(v)) \wedge (\text{кон}(y) \leq \text{нач}(u)) \wedge (\text{кон}(y) \leq \text{нач}(g)) \wedge (\text{кон}(v) \leq \text{нач}(r)) \wedge$$

$$\wedge(\text{кон}(u) \leq \text{нач}(r)) \wedge (\text{кон}(g) \leq \text{нач}(r))]; \quad (17)$$

$$(\forall x \in Ob)\ (\exists y \in Rl).[(f_3(x)=\text{вх\_док}) \wedge (f_2(y)=\text{делопроизв}) \wedge (\forall s \in Ev).$$
$$.(f_4(s)=\text{получ\_док}) \wedge (\exists v \in Op).(f_1(v)=\text{созд\_ркк}) \wedge p_3(v,y) \wedge p_4(v,x) \wedge$$
$$\wedge(\forall z \in Pr).\text{регистр\_док}(z) \wedge p_2(s,z) \wedge p_1(z,v)]; \quad (18)$$

$$(\forall x \in Ob)(\exists y,v \in Pr).p_4(y,x) \wedge p_4(v,x) \wedge \{[(f_1(y)=\text{отклонить\_согласов}) \rightarrow$$
$$\rightarrow (f_1(v)=\text{доработка})] \vee [(f_1(y)=\text{принять\_согласов}) \rightarrow ((\forall z \in Pr).p_1(z,v) \wedge$$
$$\wedge \text{регистр\_док}(z))]\}; \quad (19)$$

$$(f_3(\text{док}_1)=\text{вх\_док}); \quad (20)$$

$$p_3(\text{опер}_1,\text{док}_1) \wedge (f_1(\text{опер}_1)=\text{созд\_ркк}) \wedge (\text{нач}(\text{опер}_1)=10) \wedge (\text{кон}(\text{опер}_1)=50). \quad (21)$$

Формула (15) задает правило, по которому созданный документ сначала согласовывается, а затем регистрируется. Формула (16) означает, что в любой деловой процесс должна быть включена операция контроля. Формула (17) задает последовательность операций выполнения делового процесса «согласование документа»: сначала выполняется операция «изучение документа», затем параллельно выполняются операции «сравнение с аналогом», «проверка ссылочных документов» и «проверка соответствия нормативным документам», после чего в зависимости от результата выполняется операция «принять согласование» или «отклонить согласование». Формула (18) определяет, что на каждый полученный документ делопроизводитель должен завести регистрационно-контрольную карточку. В (19) указывается, что если документ согласован, он передается на регистрацию, если нет – на доработку. Формула (21) задает, что **док**$_1$ (20) принимает участие в операции **опер**$_1$, являющейся операцией создания регистрационно-контрольной карточки.

Таким образом, с помощью ЯОДП можно представить сложные по своей структуре знания о выполнении деловых процессов в органах государственной власти и управления.

Пусть Ψ – структура, интерпретирующая сигнатуру Σ. Пусть $\alpha \in Z^{\Sigma'}$ – замкнутая формула (т.е. формула, не включающая свободные переменные), причем $\Psi \models \alpha$ если α истинна в Ψ. Множество формул $Z^{\Sigma'}$ является замкнутым относительно подстановок любых литерных констант из Σ' вместо свободных переменных.

Определим формально $\forall \alpha,\beta \in Z^{\Sigma'}$ в Ψ по следующим правилам **R**.

П8. $\Psi \models \neg\alpha$ тогда и только тогда, когда нет $\Psi \models \alpha$.

П9. $\Psi \models (\alpha \wedge \beta)$ тогда и только тогда, когда есть $\Psi \models \alpha$ и $\Psi \models \beta$.

П10. $\Psi \models (\alpha \vee \beta)$ тогда и только тогда, когда есть $\Psi \models \alpha$ или $\Psi \models \beta$.

П11. $\Psi \models (\alpha \rightarrow \beta)$ тогда и только тогда, когда нет $\Psi \models \alpha$ или есть $\Psi \models \beta$.

П12. Если α содержит одну свободную переменную **x**, т.е. $\alpha=\alpha[x]$ то $\Psi \models (\exists x \in A)\alpha$ тогда и только тогда, когда $\Psi \models \alpha[a]$ для хотя бы одной константы **a** из сорта **A** из Σ'.

Структура Ψ, интерпретирующая Σ, является модельной структурой для $\Omega \in Z^{\Sigma}$, если $\Psi \models \alpha$ для всех $\alpha \in \Omega$, т.е. сигнатура Σ={(2), (3),…(14)} является модельной структурой для совокупности формул Ω={(15), (16),…(21)}.

Логическое следствие в ЯОДП определяется следующим образом: $\boldsymbol{\beta}$ следует из $\boldsymbol{\Omega}$ ($\boldsymbol{\Omega} \models \boldsymbol{\beta}$) если $\boldsymbol{\beta}$ истинно во всякой модельной структуре для $\boldsymbol{\Omega}$.

Рассмотрим следующее утверждение: в **опер**$_1$ участвует входной документ, т.е.

$$(\exists \in \mathbf{Ob}).\mathbf{p_3(опер_1,x) \wedge (f_3(x)=вх\_док)}. \tag{22}$$

Утверждение (22) является логическим следствием (20) и (21), т.е.

$$\{(20), (21)\} \models (22). \tag{23}$$

Докажем справедливость логического следствия (23). По определению $\boldsymbol{\Psi} \models \mathbf{(20)}$ и $\boldsymbol{\Psi} \models \mathbf{(21)}$. Тогда по П9 имеем из (21):

$$\boldsymbol{\Psi} \models \mathbf{p_3(опер_1,док_1)}; \quad \boldsymbol{\Psi} \models \mathbf{(f_1(опер_1)=созд\_ркк)}; \tag{24}$$

$$\boldsymbol{\Psi} \models \mathbf{(нач(опер_1)=10)}; \ \boldsymbol{\Psi} \models \mathbf{(кон(опер_1)=50)}. \tag{25}$$

В частности, формула $\mathbf{p_3(опер_1,док_1)}$ истинна в $\boldsymbol{\Psi}$. Следовательно, по П9 и П12 истинна и также и (22). Итак, во всякой структуре $\boldsymbol{\Psi}$, в которой истинны (20) и (21), будет истинна и (22), т.е. $\{(20), (21)\} \models (22)$, что и требовалось доказать.

Логический вывод в ЯОДП основывается на аксиомах (для приведенного фрагмента предметной области аксиомами являются (15) – (21)) и правилах вывода П8 – П12. Сформулируем задачу логического вывода в следующем виде: $\boldsymbol{\Omega} \models \boldsymbol{\beta}$ (формула $\boldsymbol{\beta}$ выводима из $\boldsymbol{\Omega}$) с помощью правил П8 – П12, если существует такая последовательность формул $\boldsymbol{\beta_1, \beta_2, \ldots \beta_n}$, что:

1) $\boldsymbol{\beta}$ совпадает с $\boldsymbol{\beta_n}$;

2) для каждого $\mathbf{i \le n}$, либо $\boldsymbol{\beta_i \in \Omega}$, либо $\boldsymbol{\beta_i}$ является непосредственным следствием некоторых формул $\boldsymbol{\beta_1, \beta_2, \ldots \beta_{i-1}}$ в силу одного из правил вывода П8 – П12.

Для логического вывода в ЯОДП будем использовать систему дедукции резолюционного типа [1], в основе которой лежит правило резолюции

$$\mathbf{[(x \vee y) \wedge (\neg y \vee z)] \rightarrow (x \vee z)}. \tag{26}$$

Для использования (26) исходная логическая формула приводится к пренексной нормальной форме, имеющей вид $\boldsymbol{\kappa_1 x_1\ \kappa_2 x_2\ \ldots\ \kappa_n x_n}$ $\mathbf{K}$, $\boldsymbol{\kappa_i \in \{\forall, \exists\}}$, $\mathbf{(1 \le i \le n)}$. В свою очередь **К** представляется в конъюктивной нормальной форме $\mathbf{F_1 \wedge F_2 \wedge \ldots \wedge F_n}$, где каждая из $\mathbf{F_1, F_2, \ldots\ F_n}$ есть дизъюнкция литер.

Преимущество ЯОДП состоит в том, что $\boldsymbol{\Sigma \cup \Omega}$ можно рассматривать как схему БД, причем $\boldsymbol{\Sigma}$ описывает структуру данных, а $\boldsymbol{\Omega}$ – зависимости и ограничения целостности. Конечная модельная структура $\boldsymbol{\Psi}$ для $\boldsymbol{\Omega}$ есть экземпляр схемы $\boldsymbol{\Sigma \cup \Omega}$.

В таком случае запрос к БД можно представить в терминах логического следствия. Для этого выразим свойства объектов некоторым предложением $\boldsymbol{\beta}\mathbf{[x]}$ в ЯОДП, где **x** – свободная переменная. Тогда запрос представим следующим выражением

$$\{\mathbf{x} \mid \boldsymbol{\Psi} \models \boldsymbol{\beta}\mathbf{[x]}\}. \tag{27}$$

Ответ на такой запрос можно вычислить путем перебора всех констант **x** данного сорта и отбора тех из них, для которых формула $\boldsymbol{\beta}\mathbf{[x]}$ истинна в $\boldsymbol{\Psi}$

(для определения истинности применяются П8 – П12), или логическим выводом формулы

$$(\forall x \in X)\{\beta[x] \rightarrow ответ(x)\} \tag{28}$$

из Ω_Ψ, однозначно характеризующей Ψ. Совокупность Ω_Ψ состоит из константных формул вида $\Psi \models (f(x_1, x_2 \ldots x_n)=x)$ или $\Psi \models (p(x_1, x_2 \ldots x_n)$.

Приведем пример такого запроса: найти всех исполнителей, работающих с документом **x** в момент времени **t**. Сформулируем запрос как нахождение множества:

$$\{\forall r \in RL | (\exists y \in Op).p_4(y,x) \wedge p_3(y,r) \wedge (t \leq кон(y)) \wedge (нач(y) \leq t)\}. \tag{29}$$

Запрос (29) к БД вычислим с помощью логического вывода. Для этого БД D представим совокупностью Ω_D атомарных константных формул вида **f(c)=d**, где **f** – имя функции, соответствующей паре отношение – атрибут, **c** – имя строки таблицы, **d** – значение атрибута, стоящее в этой строке. Тогда запрос (29) вычисляется с помощью вывода из Ω_D формулы

$$(\forall r \in RL)\{\beta[r] \rightarrow ответ(r)\}, \tag{30}$$

где $\beta[r]$ есть входящая в (29) формула.

Таким образом, описание предметной области на ЯОДП можно преобразовать в схему БД. Хотя вычисление запросов к БД, основанное на логическом выводе, является громоздким, но оно является универсальным и применимо к неполным БД, которые наиболее характерны для рассматриваемой предметной области. Под неполной БД [1] понимается такая БД **E**, которая не удовлетворяет зависимостям и ограничениям из Ω, но может быть расширена до допустимой базы данных (т.е. БД, удовлетворяющей всем зависимостям из Ω). Запрос к неполной БД **E** может быть вычислен с помощью логического вывода запросной формулы из совокупности формул $\Omega_E \cup \Omega$, где Ω_E – совокупность атомарных константных формул вида **f(c)=d**.

Другим применением ЯОДП может стать анализ модели предметной области и определение избыточной информации в модели.

ЛИТЕРАТУРА

1. Громов А., Каменова М., Старыгин А. Управление бизнес - процессами на основе технологии workflow // Открытые системы. – 1997. – №1. – С. 35 - 41.

2. Поспелов Д.А. Ситуационное управление: теория и практика. – М.: Наука, 1986. – 288 с.

3. Асеев Г.Г. Электронный документооборот. – Харьков: ХГАК, 2000. – 470 с.

4. Білова Т.Г. Формалізація представлення ділових процесів у workflow-системах // Системи обробки інформації. – Харків : НАНУ, ПАНМ, ХВУ. – 2000. - Вип. 2(8). – С. 67 - 71.

Поступила в редколлегию 28.11.2000