

АКАДЕМИЯ НАУК УССР
ВЫЧИСЛИТЕЛЬНЫЙ ЦЕНТР
ОДЕССКОГО ОТДЕЛЕНИЯ ИНСТИТУТА ЭКОНОМИКИ

"УТВЕРЖДАЮ"

Директор Вычислительного центра
Одесского отделения Института
экономики Академии Наук УССР,
член-корреспондент АН УССР

М.Т.МЕЛЕШКИН

"УТВЕРЖДАЮ"

Директор Центрального научно-
исследовательского и проектно-
экспериментального института
автоматизированных систем в
строительстве, к.э.н.

А.А.ГУСАКОВ

УДХ 421.044:65.011.56

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

на программное обеспечение блока ДОКУМЕНТИРОВАНИЕ
метода автоматизированного проектирования систем
организационного управления

№ государственной
регистрации _____

Инв. № _____

Руководитель темы

Зам. директора ВЦ,
к.ф.н.

Портнов Г.Я.Портнов

Ответственные исполнители:

Зав.произв.группой, к.ф-м.н.

А.В.Айзенштат

Ст.инженер

Б.А.Закс

"Согласовано"

Зав.отделом ИО АСУС

С.П.Никаноров С.П.Никаноров

Ст.научный сотрудник,
к.ф-м.н.

Д.Б.Персиц Д.Б.Персиц

СПИСОК ИСПОЛНИТЕЛЕЙ

1. Портнов Г.Я. - руководитель темы, зам.директора ВЦ,
к.ф.н.
2. Айзенштат А.В. - ответственный исполнитель, зав.про-
изводственной группой, к.ф.-м.н.
3. Закс Б.А. - ответственный исполнитель, ст.инже-
нер
4. Самовалов А.Д. - ст.инженер
5. Язловицкая Г.Ф. - инженер
6. Бутина Г.А. - ст.техник

АННОТАЦИЯ

Настоящий документ содержит математические постановки основных задач, решаемых в блоке ДОКУМЕНТИРОВАНИЕ метода автоматизированного проектирования систем организационного управления и является заданием для разработки Технического проекта на блок ДОКУМЕНТИРОВАНИЕ.

Блок ДОКУМЕНТИРОВАНИЕ является составной частью системы пакетов прикладных программ (СППП) системы автоматизированного проектирования систем организационного управления (САП СОУ).

Блок ДОКУМЕНТИРОВАНИЕ предназначен для автоматизации процесса представления недокументированного решения, определяющего проектируемую СОУ, в форме документа-проекта.

В документе описана общая схема работы блока ДОКУМЕНТИРОВАНИЕ и основные операции блока.

**Основание для проведения работ по разработке
Технического задания.**

Настоящее Техническое задание разрабатывается в соответствии с договором ОI6-76 от 20 июля 1976 г. с Центральным научно-исследовательским институтом автоматизированных систем в строительстве (ЦНИИАСС) Госстроя СССР на тему "Разработка системы автоматизированного проектирования систем организационного управления. Технический проект на блок ДОКУМЕНТИРОВАНИЕ и блок ВЫБОР МЕТОДОВ" и является отчетом по первому этапу этого договора.

ОГЛАВЛЕНИЕ

	стр.
I. ВВЕДЕНИЕ	1
2. ОБЩАЯ СХЕМА БЛОКА ДОКУМЕНТИРОВАНИЕ	2
2.1. Требования к представлению результатов R-интерпретации..	2
2.2. Основные операции	5
2.3. Схема функционирования блока ДОКУМЕНТИРОВАНИЕ	6
3. АСПЕКТИРОВАНИЕ	9
3.1. Задачи, решаемые операцией	9
3.2. Вход в операцию	9
3.3. Описание операции АСПЕКТИРОВАНИЕ	10
3.4. Выход операции	13
3.5. Требования к внесению изменений	15
4. СОКРАЩЕНИЕ	16
4.1. Задачи, решаемые операцией	16
4.2. Вход в операцию	16
4.3. Содержание операции	16
4.4. Выход операции	23
4.5. Требования к внесению изменений	23
5. ТЕКСТИРОВАНИЕ	24
5.1. Задачи, решаемые операцией	24
5.2. Вход в операцию	24
5.3. Концептуальная схема операции	24
5.3.1. Определения	24
5.3.2. Построение модели текста	27
5.4. Выход операции	29
5.5. Требования к внесению изменений	29

6.	РАЗМЕЩЕНИЕ	31
6.1.	Задачи, решаемые операцией	31
6.2.	Вход в операцию	31
6.3.	Содержание операции	32
6.4.	Выход операции	37
6.5.	Требования к внесению изменений	37
7.	ВЫВОД	39
7.1.	Задачи, решаемые операцией	39
7.2.	Вход в операцию	39
7.3.	Концептуальная схема	39
7.4.	Выход операции	43
7.5.	Требования к внесению изменений	43

I. ВВЕДЕНИЕ

Настоящее Техническое задание содержит основные математические модели задач, решаемых в блоке ДОКУМЕНТИРОВАНИЕ.

Блок ДОКУМЕНТИРОВАНИЕ является составной частью метода автоматизированного проектирования систем организационного управления, разрабатываемого в ЦНИИАСС.

Для понимания документа необходимо знакомство с отчетом "Технический проект экспериментальной системы пакетов прикладных программ автоматизированного проектирования систем организационного управления (логико-интерпретационный блок)".

Блок ДОКУМЕНТИРОВАНИЕ предназначен для автоматизации процессов представления недокументированного решения, получаемого в логико-интерпретационном блоке в виде результатов R-интерпретации, в форме документа-проекта. Превращение недокументированного решения в проект осуществляется с помощью следующих операций блока ДОКУМЕНТИРОВАНИЕ:

- АСПЕКТИРОВАНИЕ;
- СОКРАЩЕНИЕ;
- ТЕКСТИРОВАНИЕ;
- РАЗМЕЩЕНИЕ;
- ВЫВОД;
- ВНЕСЕНИЕ ИЗМЕНЕНИЙ.

В п 2 представлена общая схема блока ДОКУМЕНТИРОВАНИЕ, в п.п. 3-7 описаны математические модели и схемы выполнения всех операций, кроме операции ВНЕСЕНИЕ ИЗМЕНЕНИЙ и сформулированы требования к операции ВНЕСЕНИЕ ИЗМЕНЕНИЙ от остальных операций.

2. ОБЩАЯ СХЕМА БЛОКА ДОКУМЕНТИРОВАНИЯ

2.I. Требования к представлению результатов R-интерпретации.

2.I.I. Вспомогательные определения.

Дадим рекурсивное определение квазиэлемента термина Π (родовой структуры).

Определение 2.I.I.I.

1. Π - квазиэлемент Π .

2. Если множество $\{t_i\}$ - квазиэлемент Π , то для любого i
 t_i - квазиэлемент Π .

3. Если вектор $\langle t_1, \dots, t_n \rangle$ - квазиэлемент Π , то для $1 \leq i \leq n$
 t_i - квазиэлемент Π .

Используя знак \in для указания принадлежности в качестве квазиэлемента, определение 2.I.I.I. можно записать формально:

1) $\Pi \in \Pi$,

2) $[(t \in \Pi) \wedge (t_1 \in t)] \Rightarrow (t_1 \in \Pi)$;

3) $[(t \in \Pi) \wedge (t = \langle t_1, \dots, t_n \rangle)] \Rightarrow [(\forall 1 \leq i \leq n) (t_i \in \Pi)]$.

Определение 2.I.I.2.

Каноническим представлением R-интерпретации ГРС назовем следующую семантическую сеть $\mathcal{M} = \langle G, \nu, \lambda, \mu, \alpha, \beta, \gamma \rangle$, где G - ориентированный граф, вершины которого соответствуют квазиэлементам родовой структуры (а также терминам ГРС, не вошедшим в родовую структуру), а дуги направлены от элементов и компонент, соответственно, к множествам и векторам, содержащим их;

$\nu : V/V \rightarrow MV$ - отображение множества вершин графа G ,

отличных от начальных, во множество

$MV = \{vect, set\}$;

λ - упорядочение дуг, входящих в вершины множества

$$Vect = \{ \sigma \in V : \nu(\sigma) = vect \}$$

$$\lambda : D_{Vect}^{-1} \rightarrow Z^+, \quad \lambda_v : D_v^{-1} \rightarrow Z^+_{|D_v^{-1}|} \quad \text{-биекция;}$$

$\mu : V \rightarrow STN$ - отображение множества вершин графа G во множество стандартных имен (если некоторый квазиэлемент не имеет стандартного имени, то ему должно быть приписано некоторое имя).

$\alpha : V \rightarrow SK$ - отображение множества вершин графа G во множество семантических кодов, необходимых для автоматизации получения модели аспекта^{*)}.

$\beta : V \rightarrow ET$ - отображение множества вершин графа G во множество единиц текста, отражающих реальный смысл соответствующих квазиэлементов;

$\gamma : V \rightarrow KИИ$ - отображение множества вершин графа G во множество кодов, идентифицирующих представление соответствующих квазиэлементов.

В последующем конструкция вида m будем называть каноническим представлением обобщенного термина, или кратко-обобщенным термином. (Обобщенным - потому, что в графе G могут быть несколько конечных вершин).

2.1.2. Требования к представлению результатов R-интерпретации ГРС.

1. R-интерпретация ГРС должна быть представлена канонически.
2. Для возможности использования в блоке ДОКУМЕНТИРОВАНИЕ модулей пакета R-интерпретации граф G должен быть задан в формативном или расслоенном представлении.

^{*)} Отображение α может быть не задано, однако в этом случае модели аспектов должны будут задаваться проектировщиком.

3. Семантический код каждого квазиэлемента должен соответствовать системе семантической идентификации ССИ, разработанной для данного класса систем организационного управления. (Принципы создания ССИ будут сформулированы в настоящем ТЭ, а затем должны быть конкретизированы в ТИ).

4. Код, идентифицирующий представление, каждого квазиэлемента, должен однозначно отражать представление данного квазиэлемента с учетом классификации видов представлений и соответствующих им программ вывода. (Начальный вариант классификации видов представлений будет дан в техническом проекте).

2.2. Основные операции.

Получение проекта в блоке ДОКУМЕНТИРОВАНИЕ реализуется с помощью следующих укрупненных операций:

- АСПЕКТИРОВАНИЕ, предназначенной для представления решения в виде функционально ориентированных частей;
- СОКРАЩЕНИЕ, предназначенной для удаления не интересующих данного пользователя частей, либо замены их представления другим путем введения в процесс проектирования известных пользователю фактов и процедур и устранения повторяющихся частей;
- ТЕКСТИРОВАНИЕ, предназначенной для формирования модели текста, представляющей собой набор связанных между собой текстовых форм, переменным полям которых поставлены в соответствие единицы текста, входящие в аспекты;
- РАЗМЕЩЕНИЕ, предназначенной для формирования модели размещения текста проекта путем "локализации" решений, относительно текстовых форм и размещения в них информации;
- ВЫВОД, предназначенной для реализации решений, относительно размещения, принятых в операции РАЗМЕЩЕНИЕ.

2.3. Схема функционирования блока ДОКУМЕНТИРОВАНИЕ.

Входом в блок ДОКУМЕНТИРОВАНИЕ является результат R-интерпретации, т.е. полученное после R-интерпретации решение относительно проектируемой ССУ. Операции блока ДОКУМЕНТИРОВАНИЕ выполняются в следующей последовательности: АСПЕКТИРОВАНИЕ, ТЕКСТИРОВАНИЕ, РАЗМЕЩЕНИЕ, ВЫВОД; операция СОКРАЩЕНИЕ может выполняться после каждой из операций: АСПЕКТИРОВАНИЕ, ТЕКСТИРОВАНИЕ или РАЗМЕЩЕНИЕ. Решение о включении СОКРАЩЕНИЯ принимает проектировщик.

Вход блока ДОКУМЕНТИРОВАНИЕ поступает в операцию АСПЕКТИРОВАНИЕ вместе с моделью аспекта, задаваемой явно или с помощью семантического кода аспекта. Выходом операции АСПЕКТИРОВАНИЕ является набор аспектов. Если после АСПЕКТИРОВАНИЯ применяется операция СОКРАЩЕНИЕ, то вместе с набором аспектов на вход операции поступают модель пользователя для СОКРАЩЕНИЯ и правила сцепления набора аспектов и модели пользователя. Выходом операции СОКРАЩЕНИЕ в этом случае является набор сокращенных аспектов, который поступает в операцию ТЕКСТИРОВАНИЕ. Одновременно проектировщиком могут указываться подкласс текстовых форм, используемый в операции, а также нестандартные процедуры выбора форм и их заполнения.

Выходом операции ТЕКСТИРОВАНИЕ является модель текста, т.е. набор взаимосвязанных заполненных текстовых форм. Затем может выполняться операция СОКРАЩЕНИЕ, аналогично описанному выше. Входом операции РАЗМЕЩЕНИЕ является модель текста, задание на размещение и правила размещения, а выходом операции является модель размещения текста на носителе. Затем может применяться операция СОКРАЩЕНИЕ, после чего применяется операция ВЫВОД, получающая решение относительно размещения в виде модели размещения

7

и реализующая вывод текста на носитель.

Схема функционирования блока ДОКУМЕНТИРОВАНИЕ представлена на Рис. 2.3.

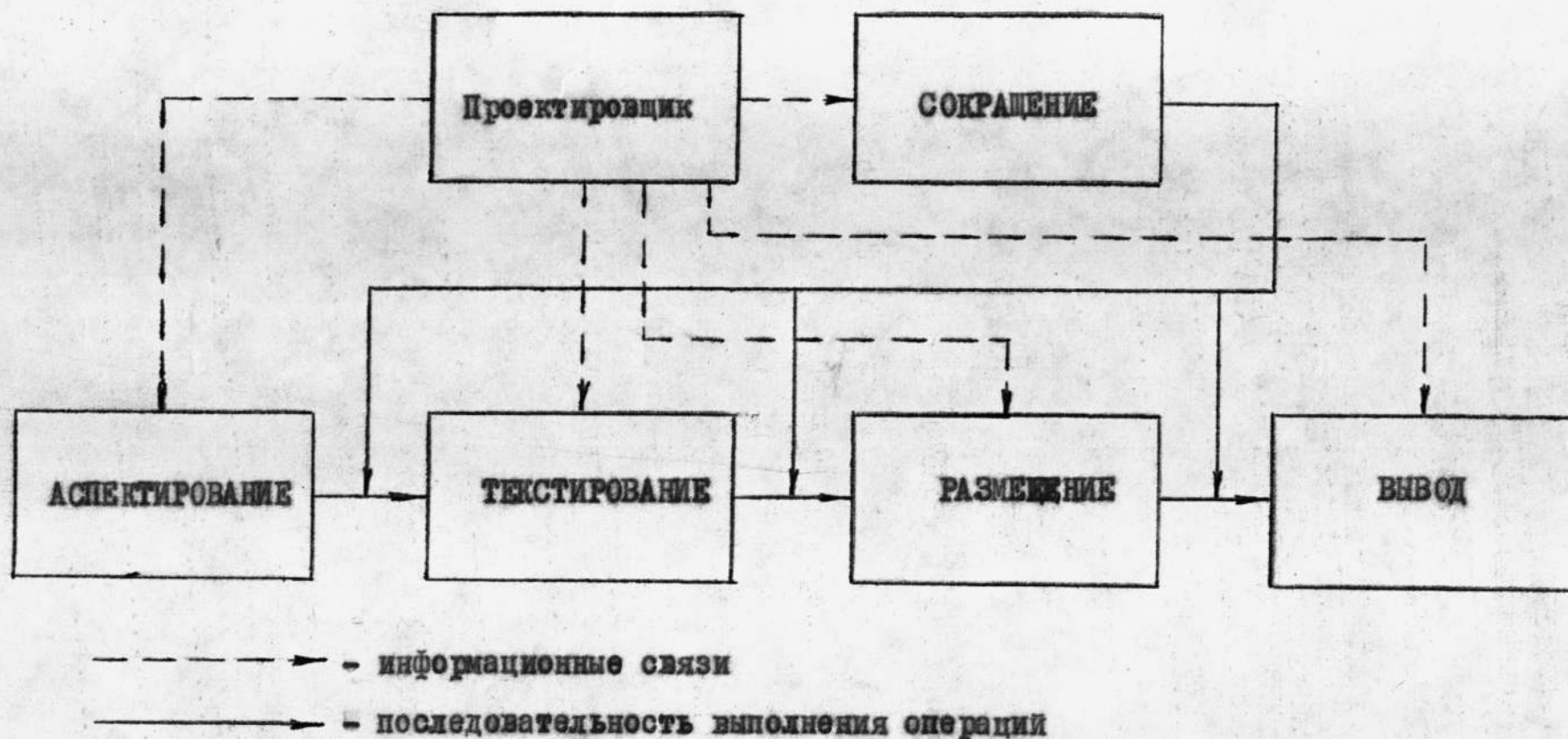


Рис. 2.3. Схема функционирования блока ДОКУМЕНТИРОВАНИЕ

3. АСПЕКТИРОВАНИЕ

3.1. Задачи, решаемые операцией.

АСПЕКТИРОВАНИЕ есть операция получения описания аспекта системы организационного управления по заданному описанию всей системы (результату R-интерпретации ГРС).

Необходимость операции **АСПЕКТИРОВАНИЕ** обуславливается одним из основных требований, предъявляемых к проекту системы организационного управления: разделы проекта должны быть строго функционально ориентированы, т.е. ориентированы на людей, выполняющих те или иные функции реализации проекта, функционирования спроектированной системы или внесения изменений в эту систему.

3.2. Вход в операцию.

Следует различать 2 режима выполнения операции **АСПЕКТИРОВАНИЕ**:

- А) **АСПЕКТИРОВАНИЕ** с автоматизированным получением модели аспекта;
- Б) **АСПЕКТИРОВАНИЕ** без автоматизированного получения модели аспекта.

В первом случае на вход операции поступают:

- 1) каноническая R-интерпретация родовой структуры (а также термов, не являющихся ее квазиэлементами);
- 2) семантический код аспекта.

Во втором случае на вход операции поступают:

- 1) каноническая R-интерпретация родовой структуры (а также термов, не являющихся ее квазиэлементами);
- 2) модель аспекта.

3.3. Описание операции АСПЕКТИРОВАНИЕ.

АСПЕКТИРОВАНИЕ является сложной операцией и включает в себя более простые операции:

1. ФОРМИРОВАНИЕ МОДЕЛИ АСПЕКТА;
2. ВЫДЕЛЕНИЕ ОБОБЩЕННОГО ТЕРМА;
3. ОПРЕДЕЛЕНИЕ МНОЖЕСТВА $V'(\sigma)$;
4. ФАКТОРИЗАЦИЯ ОБОБЩЕННОГО ТЕРМА;
5. ПЕРЕСТАНОВКА В ОБОБЩЕННОМ ТЕРМЕ;
6. СОКРАЩЕНИЕ ОБОБЩЕННОГО ТЕРМА.

Опишем каждую операцию отдельно.

3.3.1. ФОРМИРОВАНИЕ МОДЕЛИ АСПЕКТА.

Автоматизированное формирование модели аспекта может производиться, если для класса систем организационного управления, к которому относится проектируемая система, разработана система семантической идентификации ССИ, и отображение α в каноническом представлении результатов R-интерпретации задано.

Определение 3.3.1.1.

Системой семантической идентификации назовем следующую восьмерку: $\langle SK, SK_A, ET, ET_A, \sigma, \sigma_A, \sigma_1, \sigma_2 \rangle$, где

SK - множество семантических кодов квазиэлементов;

SK_A - множество семантических кодов аспектов;

ET - множество единиц текста, отражающих реальный смысл квазиэлементов;

ET_A - множество единиц текста, отражающих реальный смысл аспекта.

$\sigma : SK \rightarrow ET$ - кодификатор квазиэлементов,

$\sigma_A : SK_A \rightarrow ET_A$ - кодификатор аспектов,

$\sigma_i : SK_A \rightarrow B(SK)$ - отображение множества семантических кодов

аспектов в булеан множества семантических кодов квазиэлементов ($i = 1, 2$);

При этом для любого аспекта $a \in A$ и для любой системы организационного управления данного класса должно выполняться свойство:

$$G(V_2) \subset G(V_1),$$

где $V_i = \alpha^{-1}[G_i(SK_a) \cap \gamma_m \alpha]$ - множество вершин графа G , SK которых соответствует данному аспекту по отображению G_i , $i = 1, 2$, а $G(V_i)$ - нижнее замыкание множества V_i .

Определение 3.3.1.2.

Моделью аспекта назовем пару множеств $\langle V_1, V_2 \rangle$.

С содержательной точки зрения множество V_1 определяет обобщенный терм $\mathcal{M} = \langle G(V_1), \nu, \lambda, \mu, \alpha, \beta, \gamma \rangle$, внутри которого производится аспектирование, а V_2 - множество квазиэлементов этого обобщенного термина, относительно которых производится аспектирование.

3.3.2. ВЫДЕЛЕНИЕ ОБОБЩЕННОГО ТЕРМА.

Вход операции: 1) обобщенный терм \mathcal{M} ;

2) множество $V' \subset V$ - подмножество вершин графа G

Определение 3.3.2.1.

Выделить из $\mathcal{M} = \langle G, \nu, \lambda, \mu, \alpha, \beta, \gamma \rangle$ обобщенный терм, соответствующий множеству вершин V' , это значит получить обобщенный терм вида $\mathcal{M}'(V') = \langle G(V'), \nu, \lambda, \mu, \alpha, \beta, \gamma \rangle$, где $G(V')$ - нижнее замыкание множества V' в графе G .

3.3.3. ОПРЕДЕЛЕНИЕ МНОЖЕСТВА $V^{\downarrow}(v)$.

Вход операции: 1) граф G ;

2) вершина v графа G .

Определение 3.3.3.1.

Множеством $V'(v)$ называется множество вершин графа G , к которым направлены дуги от вершины v .

3.3.4. ФАКТОРИЗАЦИЯ ОБОБЩЕННОГО ТЕРМА.

Вход операции: 1) обобщенный терм $m = \langle G, \gamma, \lambda, \mu, \alpha, \beta, \delta \rangle$; 2) модель факторизации.

Определение 3.3.4.1.

Модель факторизации называется множество $V' \subset V$ вершин графа G , соответствующих факторизационным квазиэлементам (квазиэлементам, структура которых нас не интересует).

Определение 3.3.4.2.

Факторизацией m называется обобщенный терм $m' = \langle G', \gamma, \lambda, \mu, \alpha, \beta, \delta \rangle$, в котором граф G' получается из G удалением всех дуг, ведущих к вершинам множества V' , с последующим удалением всех частей графа G , не связанных с его конечными вершинами.

3.3.5. ПЕРЕСТАНОВКА В ОБОБЩЕННОМ ТЕРМЕ.

Вход операции: 1) обобщенный терм $m = \langle G, \gamma, \lambda, \mu, \alpha, \beta, \delta \rangle$; 2) модель перестановки.

Определение 3.3.5.1.

Модель перестановки обобщенного терма m называется множество перестановок

$$\{ P_v : Z^+ |D_v^-| \rightarrow Z^+ |D_v^+| \} v \in VEST$$

Определение 3.3.5.2.

Перестановкой обобщенного терма $m = \langle G, \gamma, \lambda, \mu, \alpha, \beta, \delta \rangle$, соответствующей модели $\{ P_v \} v \in VEST$, называется обобщенный

терм $m' = \langle G, \nu, \lambda', \mu, \alpha, \beta, \gamma \rangle$, где $\lambda'_v = P_v \cdot \lambda_v$, $v \in VECT$.

3.3.6. СОКРАЩЕНИЕ ОБОБЩЕННОГО ТЕРМА.

Вход операции: 1) обобщенный терм $m = \langle G, \nu, \lambda, \mu, \alpha, \beta, \gamma \rangle$;
2) модель сокращения.

Определение 3.3.6.1.

Моделью сокращения обобщенного терма m называется множество пар $\langle v, D_v^c \rangle$, где v - вершина графа G , отличная от начальной, а $D_v^c \subset D_v^{-1}$ ($D_v^c \neq D_v^{-1}$) - некоторое множество дуг, входящих в эту вершину.

Определение 3.3.6.2.

Сокращением m называется обобщенный терм

$$m' = \langle G', \nu, \lambda, \mu, \alpha, \beta, \gamma \rangle,$$

в котором граф G' получается из G удалением всех дуг, заданных в модели сокращения, с последующим удалением всех частей графа G , не связанных с его конечными вершинами.

3.3.7. Общая схема выполнения операции.

Общая схема выполнения операции представлена на Рис.3.3.7.1.

3.4. Выход операции.

Выходом операции АСПЕКТИРОВАНИЕ является совокупность обобщенных термов

$$\{ m_{\varphi, \eta} [V'(v)] \}_{v \in V_2}$$

- полный аспект,

$$\{ m_{\varphi, \eta, c} [V'(v)] \}_{v \in V_2}$$

- усеченный аспект

соответствующих аспектным вершинам $v \in V_2$.

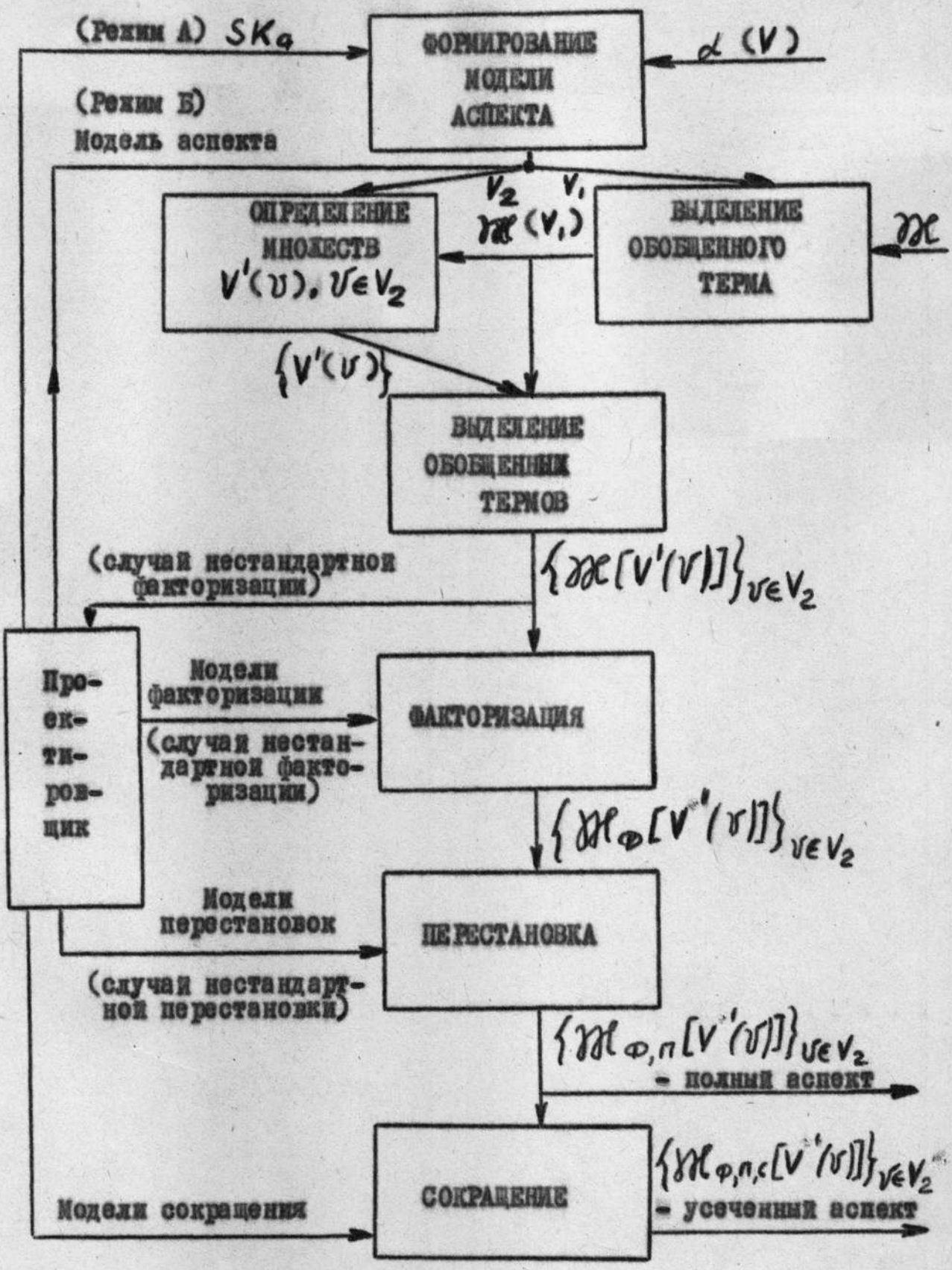


Рис. 3.3.7.1. Общая схема выполнения операции.

3.5. Требования к внесению изменений.

Операция ВНЕСЕНИЕ ИЗМЕНЕНИЙ должна обеспечить следующие возможности:

1. Пополнение и уменьшение множества V_1 в модели аспекта.
2. Пополнение и уменьшение множества V_2 в модели аспекта.
3. Пополнение и уменьшение множества V' в модели факторизации.
4. Замену некоторых перестановок P_{2r} из модели перестановки.
5. Пополнение и уменьшение множества дуг $\{D_2^c\}$ в модели сокращения.

4. СОКРАЩЕНИЕ

4.1. Задачи, решаемые операцией.

Операция СОКРАЩЕНИЕ предназначена для сокращения СТ-информации^{ж)} (СТИ) путем замены некоторых частей СТИ другими, имеющими для данного пользователя то же содержание, но занимающими меньше места, в частности, удаления некоторых частей текста, замены некоторых единиц текста ссылками, приведения к другому представлению.

4.2. Вход в операцию.

Входом в операцию СОКРАЩЕНИЕ является:

- модель СТИ;
- модель пользователя;
- правила "сцепления" модели СТИ и модели пользователя, т.е. получения на модели СТ-информации разметки, фиксирующей операции пользователя над соответствующими единицами СТ-информации.

4.3. Содержание операции.

В 4.3.1. приведена концептуальная схема, в 4.3.2. - основные операции, в 4.3.3. - общая схема выполнения операции.

4.3.1. Концептуальная схема.

Моделью СТ-информации (МСТ) $\alpha = \langle \Gamma, \varphi_1, x_1, \varphi_2, x_2, \dots, \varphi_k, x_k, \tilde{\varphi}_{k+1}, x_{k+1}, \dots, \tilde{\varphi}_n, x_n, \psi_1, z_1, \dots, \psi_e, z_e \rangle$ будем называть ориентированный граф Γ , на котором задана некото-

ж)

т.е. символьной или текстовой информации, определение основных понятий дано в 4.3.1.

рая "разметка", т.е. ряд отображений φ_i , возможно частичных, в соответствующие этим отображениям множества имен или значений, при этом среди отображений φ_i заданы "основные" $\varphi_1, \dots, \varphi_k$ и сервисные $\tilde{\varphi}_1, \dots, \tilde{\varphi}_n$, и сервисных отображений φ_i графа Γ в некоторые другие объекты из Z_i .

Интерпретация.

Поскольку операция СОКРАЩЕНИЕ применяется после различных операций блока ДОКУМЕНТИРОВАНИЕ, то для описания МСТ используется общая конструкция, задающая символьную или текстовую информацию.

В качестве "основных" отображений могут выступать:

- λ - задающее упорядочение ребер, входящих в данную вершину графа Γ ;
- β - задающее символьно-текстовые единицы, соответствующие вершинам графа Γ , при этом если вершина v предшествует вершине w , то $\beta(v)$ входит в $\beta(w)$;
- φ_j - задающие аспектирующую, текстовую идентификацию и идентификацию размещения на носителе;
- μ - задающее стандартные имена СТЕ, соответствующей данной вершине графа;
- ξ - задающее упорядоченную цепочку имен трансформаторов, соответствующих ребру графа Γ , после применения которых к СТЕ, соответствующей нижней вершине ребра, получается СТЕ, входящая в СТЕ, соответствующую верхней вершине ребра.
- τ - задающее отображение подграфов в соответствующие им текстовые формы (определяется после операции ТЕКСТИРОВАНИЕ), т.е. единицы текста.

Единицей СТИ (СТЕ) будем называть как элементарные тек-

ты, соответствующие некоторой вершине графа Г, так и тексты, соответствующие некоторым подграфам графа, получаемые из ранее определенных СТЕ в операции ТЕКСТИРОВАНИЕ.

Наличие отображения ξ может усложнить использование операции СОКРАЩЕНИЕ; в этом случае с помощью сервисных средств следует предварительно выполнить соответствующие ребра графа трансформаторы.

Основные отображения предназначены для задания логической структуры физического представления СТИ.

В качестве сервисных отображений φ_i могут выступать:

- γ - отображение, задающее коды представления СТЕ;
- α - отображение, задающее семантические коды соответствующих СТЕ;
- ξ - отображение, задающее индивидуальные обозначения СТЕ и другие.

В качестве сервисных отображений могут выступать:

- ρ - отображение графа Г в граф γ_2 ссылок.

Графом ссылок, соответствующим МСТ, называется ориентированный граф γ_2 , в котором наличие ориентированного ребра $\{v_1, v_2\}$ означает наличие в МСТ ссылки в СТЕ, соответствующей v_2 на СТЕ, соответствующую v_1 .

Сервисные отображения предназначены для упрощения работы пользователя.

Моделью пользователя M_n будем называть следующие множества трансформаций:

- 1) множество троек $\langle x_i, y_i, \kappa_i \rangle$, где κ_i - заменяемая единица СТ-информации;

y_i - заменяющая единица СТ-информации, K_i - КИП y_i ;

2) множество пар $\langle T_j, K_j \rangle$, где T_j - трансформаторы на множестве некоторых единиц СТ-информации, K_j - КИП, соответствующий результату применения T_j .

При этом для операции СОКРАЩЕНИЕ на модель пользователя накладываются следующие ограничения:

1) для троек $\langle x_i, y_i, K_i \rangle$ имеем а) СТЕ y_i имеет тот же смысл для пользователя, что и СТЕ x_i ; б) $\mu(x_i) \geq \mu(y_i) \forall i$

2) для пар $\langle T_j, K_j \rangle$ на T_j накладывается требование а) сжатия, т.е. если T_j применим к некоторой единице СТ-информации x , то $\mu(T_j x) \leq \mu(x)$, где μ - некоторая мера во множестве СТЕ, например, количество элементарных символов (для символьной информации) или величина занима^{мо}го места на носителе; б) сохранение смысла для пользователя, т.е. $T_j x$ имеет тот же смысл, что и x .

Интерпретация.

Трансформации первого типа задают известные пользователю "внешние" факты (в частности, позволяют удалить ненужную информацию или заменить ее другой, более короткой) или определяют более удобный способ представления информации внутри текста (например, замена повторяющихся частей ссылками), а трансформации второго типа определяют известные пользователю процедуры сокращения.

Пусть $\Pi \in \text{МП}$. Элементарным сцеплением МСТ с Π назовем частичное отображение $X: \mathcal{D} \rightarrow \Pi$, где \mathcal{D} - множество дуг графа Γ , такое, что если $\alpha = (v_1, v_2) \in \mathcal{D}$, то Π применимо к СТЕ, соответствующей нижней вершине v_1 .

Сцеплением МСТ с МП назовем суперпозицию отображений

$X = X_k \circ X_{k-1} \circ \dots \circ X_1$, соответствующих упорядоченному набору $\Pi_1, \dots, \Pi_k, \Pi_i \in \text{МП}$ ($i=1, \dots, k$).

Интерпретация элементарного сцепления.

Элементарное сцепление может порождаться некоторым условием P применимости Π , например:

- предикатом на разметке, в частности на семантических кодах или индивидуальных обозначениях; тогда Π применяется во всех случаях, если выполнено условие P и Π применимо к СТЕ, соответствующей нижней вершине ребра; P , вообще говоря, задается на ребрах, если же P задано на вершине, то в случае "сцепляемости" Π применяется ко всем ребрам, выходящим из данной вершины;

- явным указанием списка вершин или ребер;

- выделением особых случаев, когда не следует применять

Π и т.д.

Отсутствие условия P трактуется как условие применимости Π во всех случаях, когда СТЕ входит в область определения Π .

4.3.2. Основные операции.

Операция ВВЕДЕНИЕ ССЫЛКИ.

Вход и источники входов:

- МСТ; поступает из предшествующей операции;

- КЭМП , являющиеся ссылкой; $\Pi = \langle x, y, k \rangle$ - где x - заменяемая СТЕ, y - локализует ссылку в тексте или на носителе, k - КИП ссылки; задается проектировщиком;

- P - условие применимости Π^{\otimes} задается проектировщиком.

Содержание операции и выход:

Производится анализ МСТ на применимость к ребрам, исходящим из данной вершины, ссылки Π .

В случае, если часть ребер, но не все ребра, исходящие "вверх"

\otimes Описание возможных P приведено в 4.3.1.

из данной вершины, заменяются на данную ссылку, производится "расцепление" данной вершины на две-одна вершина, "старая", соответствует ребрам, к которым не применяется данная ссылка, а другая - "новая", - ребрам, к которым применяется данная ссылка.

Производится замена СТЕ, соответствующей "новой" вершине на ссылку и разметка КИП, на которую произведена ссылка, как ссылки; в случае, если заменяемая СТЕ является ссылкой, производится замена всех ссылок на данную СТЕ ссылками на "новую" ссылаемую СТЕ и соответствующая модификация графа ссылок; если же заменяемая СТЕ не является ссылкой, то производится пополнение графа ссылок.

Операция УДАЛЕНИЕ СТЕ.

Вход и источники входов:

- МСТ; поступает из предшествующей операции;
- ПЕМП- задание на удаление, $P = \langle x \rangle$ - где x -удаляемая СТЕ; задается проектировщиком.
- Р- условие применимости П; задается проектировщиком.

Содержание операции и выход.

Удаляются ребра, для которых произведено сцепление с трансформацией П и в случае, если это все ребра, исходящие вверх из заданной вершины, то сервисной программой удаляется из графа МСТ сильное нижнее замыкание этой вершины, если среди удаляемых вершин нет ссылаемых, в противном случае трансформация П отменяется и проектировщику, по желанию, могут быть выданы ссылаемые вершины из сильного нижнего замыкания, с помощью сервисных средств.

Операция ЗАМЕНА СТЕ.

Вход и источники входов:

- МСП; поступает из предыдущей операции;

- ПЄМП; являющееся заменой, $\Pi = \langle x, y, k \rangle$, где x -заменяемая СТЕ, y - заменяющая СТЕ, k - КИП y ; задается проектировщиком;

- Р-условие применимости Π , задается проектировщиком.

Содержание операции и выход:

Производится анализ МСП на применимость к ребрам, исходящим из данной вершины, замены Π . В случае, если часть ребер, но не все ребра, исходящие из данной вершины, сцеплены с Π , производится "расщепление" данной вершины на две, одна вершина - "старая", соответствует ребрам, с которыми Π не сцеплена, а другая, "новая", - ребрам, с которыми Π сцеплена. К новой вершине повторно приписывается нижнее замыкание старой вершины, и СТЕ, соответствующей новой вершине, приписывается КИП k .

Операция ПРЕОБРАЗОВАНИЕ СТЕ.

Аналогичная операции ЗАМЕНА СТЕ, $\Pi = \langle T, k \rangle$, ~~замена~~ осуществляется ^{замена} СТЕ x , с которой сцеплено Π , на Tx .

4.3.3. Общая схема выполнения операции.

Операция СОКРАЩЕНИЕ применяется после операции АСПЕКТИРОВАНИЕ, после операции ТЕСТИРОВАНИЕ, после операции РАЗМЕЩЕНИЕ. На вход операции поступает МСТ, МП и правила сцепления, в результате их взаимодействия получается сцепление МСТ с МП, т.е. цепочка элементарных сцеплений. Производится анализ очередного элементарного сцепления, которое может задаться с помощью некоторого условия применимости, задаваемого проектировщиком, например, в виде некоторого предиката на сервисной разметке. Если трансформация Π -ссылка, то выполняется операция ВВЕДЕНИЕ ССЫЛКИ, если Π -замена (но не удаление), то выполняется операция ЗАМЕНА СТЕ, если Π -удаление, то производится операция УДАЛЕНИЕ СТЕ, если Π -трансформатор, то применяется операция ПРЕ-

ОБРАЗОВАНИЕ СТЕ.

4.4. Выход операции.

Выходом операции СОКРАЩЕНИЕ является сокращение ИСТ.

4.5. Требование к внесению изменений.

Операция ВНЕСЕНИЕ ИЗМЕНЕНИЙ блока ДОКУМЕНТИРОВАНИЕ должна реализовать внесение изменений в следующие входы операции СОКРАЩЕНИЕ:

- модель пользователя;**
- правила сценария ИСТ и ИП;**
- ИСТ.**

5. ТЕКСТИРОВАНИЕ.

5.1. Задачи, решаемые операцией.

Операция ТЕКСТИРОВАНИЕ предназначена для формирования модели текста, соответствующего проаспектированному результату R -интерпретации.

(Определение модели текста см. в 5.3.1.)

Эта задача решается с помощью следующих подзадач:

- 1. выбор текстовой формы (см. 5.3.2);
- 2. заполнение текстовой формы.

5.2. Вход в операцию.

Для выполнения операции должны быть заданы:

- обобщенные термины, соответствующие аспектам;
- нестандартные процедуры выбора текстовых форм и их заполнения для некоторых вершин обобщенных термов.

5.3. Концептуальная схема операции.

5.3.1. Определения.

Определение 5.3.1.1.

Текстовой формой называется четверка $\langle P_1, P_2, \mathcal{F}, \theta_1 \rangle$,

где $P_1 \cup P_2$ - множество полей текстовой формы;

\mathcal{F} - множество отношений;

θ_1 - отображение множества P_1 во множество стандартных единиц текста.

Определение 5.3.1.2.

Пусть m - обобщенный терм, аспекта.

Заполнением текстовой формы Φ называется обобщенный терм \tilde{m} , удовлетворяющий условиям:

- 1. граф $g(\tilde{m})$ является подграфом $G(m)$;
- 2. $\varphi: g \rightarrow P_2(\varphi)$ -сюръекция;
- 3. на \tilde{m} выполняются отношения \mathcal{F} .

Определение 5.3.1.3.

Местность текстовой формы называется подмножество \mathcal{F} , обозначаемое \mathcal{F}_1 , определяющее число вершин графа, обобщенного термина, который может быть заполнением этой текстовой формы.

Определение 5.3.1.4.

Ограничениями на вход текстовой формы будем называть:

1. собственное подмножество \mathcal{F} , состоящее из:

1. \mathcal{F}_1 ;

2. \mathcal{F}_2 - множество отношений, определенных на КИП, задающее требования к КИП вершин графа обобщенного термина;

3. \mathcal{F}_3 - множество отношений, определенных на семантических кодах.

Определение 5.3.1.5.

Ограничениями размера текстовой формы называется подмножество \mathcal{F} , задающее ограничения на размеры формы через ограничения на размеры полей.

Определение 5.3.1.6.

Будем говорить, что текстовая форма Φ_2 является ^{под}формой Φ_1 , если ^{зак}конечные вершины графа заполнения Φ_2 являются начальными вершинами графа заполнения Φ_1 .

2. Ограничения размера текстовой формы Φ_2 удовлетворяют ограничениям размера полей Φ_1 , соответствующих Φ_2 .

Если для текстовой формы Φ_2 и Φ_1 выполняется условие 1, но не выполняется условие 2, будем говорить, что текстовая форма Φ_2 отделена от текстовой формы Φ_1 .

Определение 5.3.1.7.

Деревом заполненных текстовых форм называется тройка

$$D\Phi = \langle \Gamma, \varphi, \varepsilon \rangle$$
 , где

Γ - упорядоченное, ориентированное дерево, причем множество его дуг разбито на два подмножества - Q_1 и Q_2 . Если дуга $\langle v_1, v_2 \rangle \in Q_1$, то $\mathcal{X}(v_2)$ подформа $\mathcal{X}(v_1)$. Если дуга $\langle v_1, v_2 \rangle \in Q_2$, то текстовая форма $\mathcal{X}(v_2)$ отделена от текстовой формы $\mathcal{X}(v_1)$;

$\mathcal{X} : \Gamma \rightarrow M_{\mathcal{X}}$, $M_{\mathcal{X}}$ - множество текстовых форм; \mathcal{X} - биекция;

$\varepsilon : \Gamma \rightarrow M_{\varepsilon}$, M_{ε} - множество заполнений текстовых форм; ε - биекция;

Причем $\varepsilon(g)$ заполнение текстовой формы $\mathcal{X}(g)$, где $g \in \Gamma$.

Определение 5.3.1.8.

Пусть X - упорядоченное ориентированное дерево. Будем говорить, что n - расстояние от вершины v_1 до вершины v_2 , если путь от v_1 до v_2 (или путь от v_2 до v_1) содержит n дуг.

Определение 5.3.1.9.

Пусть \mathcal{D} - дерево заполненных текстовых форм. Определим для каждой вершины, в которую входит дуга из множества Q_2 , порядковый код:

1. порядковый код вершины, находящейся на минимальном расстоянии от вершины дерева - пара

$$\langle N, \cdot \rangle, \text{ где}$$

N - номер вершины;

2. порядковый код вершины \bar{v} , (не находящейся на минимальном расстоянии от вершины дерева) - тройка

$$\langle K, N_1, \cdot \rangle, \text{ где}$$

K - порядковый код вершины, находящейся на пути от вершины дерева до \bar{v} , и такой, что расстояние от нее до вершины \bar{v} - минимально;

$N_1 = \max_{\mathcal{D}} \{ N_1 \} + 1$, где \mathcal{D} - подмножество вершин дерева такое, что $\forall d \in \mathcal{D}$ $K_d = K$ (порядковый код d имеет вид $\langle K_d, N_1, \cdot \rangle$).

Определение 5.3.1.10.

Пусть $A_i, i=1, 2, \dots, n$ - аспекты. Назовем модель

текста проекта дерево заполненных текстовых форм, такое, что:
находящейся на расстоянии 1 от вершины дерева \mathcal{D} .

- 1. для вершины дерева $v_i, \mathcal{E}(v_i)$ -обобщенный терм, соответствующий аспекту A_i ;
- 2. каждой вершине, в которую входит дуга множества Q_2 , сопоставлен порядковый код.

Определение 5.3.1.11.

Начальным заполнением текстовой формы для вершины $v \in G$ назовем обобщенный терм, граф которого есть вершина v со всеми входящими из нее дугами, *и вершинами, соединенными этими дугами с v .*

Определение 5.3.1.12.

Пусть g подграф G .

Расширением g назовем такой подграф g' , что:

- 1. g - подграф g' ,
- 2. g' - подграф G .
- 3. Начальными вершинами g' являются все вершины g кроме одной вершины t и вершина t - является конечной вершиной подграфа G .

5.3.2. Построение модели текста.

Согласно данным выше определениям, построение модели текста сводится к построению дерева заполненных текстовых форм по обобщенному терму \mathcal{M} . Построение дерева заполненных текстовых форм производится сверху вниз, т.е. заданием вершины t и затем заданием вершин t_i , соединенных с t входящими дугами. Дерево, получаемое на каждом шаге построения модели текста, будем называть промежуточным деревом.

5.3.2.1. Построение заполнения текстовой формы для верши-

ны промежуточного дерева v_1 (шаг построения модели текста).

Пусть v_2 непосредственно предшествующая v_1 вершина промежуточного дерева и d - заполнение текстовой формы, соответствующее вершине v_2 , а v - нижняя вершина d , соответствующая вершине v_1 .

Если для вершины $v \in G(\overline{m})$ ^{не} указана проектировщиком процедура выбора заполнения текстовой формы, то применяется следующая стандартная процедура перебора с возвратом.

Строится начальное заполнение текстовой формы для вершины v и осуществляется поиск формы, ограничениям на вход которой удовлетворяет начальное заполнение. Причем всегда поиск осуществляется либо в некотором подмножестве форм (если это подмножество определено выбором предыдущих заполнений тестовых форм), либо во всем множестве форм. Если поиск оказался безрезультатным, то строится расширение начального заполнения и поиск повторяется.

Если поиск безрезультатен для всех расширений начального заполнения, то необходимо произвести возврат. Для этого надо продолжить поиск для заполнения текстовой формы, построенной раньше, т.е. заполнения, для которого v - конечная вершина графа заполнения. Этот возврат может вызвать новый возврат и т.д. Если возвращаться уже нельзя и поиск оказался безуспешным, то построение модели текста стандартным способом не может быть произведено.

После того, как выбрано заполнение текстовой формы для вершины v_1 , промежуточное дерево расширяется:

I. Рассматриваемой вершине дерева v_1 сопоставляется текстовая форма Φ_1 и ее заполнение;

2. строятся e новых вершин (E), где e — число полей текстовой формы Ф, которым сопоставляются подформы и отдельные текстовые формы;

3. новые вершины (E) соединяются дугами с вершиной σ , и для каждой дуги указывается элемент/этого множества, Q_1 или Q_2 , она является;

4. для каждой вершины, в которую входит дуга из Q_2 , строится порядковый код;

5. каждая новая вершина помечается как незаполненная.

5.3.2.2. Выбор вершины промежуточного дерева для заполнения.

Этот выбор производится среди незаполненных вершин промежуточного дерева.

1. Пусть X — шаг, после которого производится выбор вершины. Если среди новых вершин рассматриваемого шага есть вершина, в которую входит дуга из множества Q_1 , то эти вершины ~~искомы~~. В противном случае эта проверка выполняется для шага предыдущего X и т.д.

2. Если в пункте 1 не выбрана вершина, то берется вершина промежуточного дерева, имеющая максимальный порядковый код.

3. Если в пункте 2 не выбрана вершина, это означает, что построение модели текста закончено.

5.4. Выход операции.

Выходом операции является модель текста (см. 5.3.1.11), которая является входом операции РАЗМЕЩЕНИЕ.

5.5. Требования к внесению изменений.

Обеспечить построение измененной части модели текста, когда:

1. меняются аспекты,

2. меняются заложённые в систему текстовые формы,
3. меняется начальное заполнение текстовой формы.

6. РАЗМЕЩЕНИЕ

6.1. Задачи, решаемые операцией.

Операция РАЗМЕЩЕНИЕ предназначена для выработки решения относительно размещения текста на носителе. При разработке операции решаются две основные задачи:

- определение структуры размещения и задания на размещение;
- "локализация" решений, принятых в операции ТЕКСТИРОВАНИЕ относительно размещения единиц текста в формах и формирования управляющей информации для операции ВЫВОД.

В качестве ограничения при разработке операции является ее ориентация на вывод на АЦПУ.

6.2. Вход в операцию.

Входом в операцию РАЗМЕЩЕНИЕ являются:

- модель текста, получаемая после операции текстирования и, возможно, последующего сокращения текста операцией СОКРАЩЕНИЕ;
- задание на размещение;
- правила размещения.

Требования к входу:

I. Единицы размещаемого должны быть частично размечены с помощью кодов, идентифицирующих представление (КИП), однозначно отражающих представление единицы текста с учетом классификации видов представлений и соответствующих им программ вывода.

Разметка текста производится на нескольких этапах работы СПП:

- 1) результаты операции АСПЕКТИРОВАНИЕ, в силу принятого в операции ограничения, представляются в каноническом виде;
- 2) при трансформации СТИ в операции СОКРАЩЕНИЕ трансформато-

ры идентифицируют КИП результата;

3) при выполнении операции ТЕКСТИРОВАНИЕ определяются КИП единиц текста, как на уровне элементарных единиц текста, так и на уровне неэлементарных, путем задания соответствующих выходных форм и их заполнения.

И. Формы, представленные в качестве выхода операции ТЕКСТИРОВАНИЕ, должны учитывать ограничения на возможность размещения их на носителе, т.е. допускать разбиение на логически связанные части, помещающиеся в полосу, соответствующую АЦПУ.

6.3. Содержание операции.

6.3.1. Концептуальная схема.

Модель текста М - приведена в 5.3. при описании операции ТЕКСТИРОВАНИЕ.

Носителем S будем называть двойку $S = \langle M; \Phi \rangle$, где M - некоторое множество, $\Phi = \langle \Phi_1, \dots, \Phi_k \rangle$ - отношения на M .

Интерпретация.

Приведенное определение носителя является весьма общим: в него, в частности, можно включить "обобщенный" носитель, состоящий из различных типов носителей, части носителя, формы для представления и т.д.

В качестве отношений Φ_1, \dots, Φ_k могут выступать отношения порядка, метрические отношения и т.д.

В случае диска имеем три уровня: тома, цилиндры, дорожки; т.е. три класса "мест" и отношения между ними - вложенности и упорядоченности; в случае магнитной ленты или перфоленты имеем набор линейно упорядоченных множеств; в случае АЦПУ имеем линейно упорядоченное множество линейно упорядоченных множеств, сос-

тоящих не более чем из 128 элементов.

Размещением текста на носителе будем называть тройку

$R = \langle M, S, \theta \rangle$, где M - модель текста, S - модель носителя,
 $\theta : M \rightarrow S$.

Приведенное определение является неконструктивным, поэтому далее приведена конструкция построения размещения.

Структурой размещения на носителе назовем тройку $C = \langle \mathcal{G}_S, \nu, \mathcal{N} \rangle$, где \mathcal{G}_S - дерево с единственной конечной вершиной, $\nu : V \rightarrow \mathcal{N}$, где \mathcal{N} - множество имен, причем ν таково, что для любых двух путей, исходящих из конечной вершины, соответствующие им цепочки имен различны.

Заданием на размещение назовем тройку $\langle C, \mu, M \rangle$, где M - модель текста, C - структура размещения, $\mu : \mathcal{G}_S \rightarrow \Gamma$ - гомоморфизм графа \mathcal{G}_S структуры размещения в граф Γ модели текста.

Пусть \tilde{M} - множество подграфов (и соответствующих им форм) графа Γ , являющихся нижним замыканием прообразов вершин \mathcal{G}_S при отображении μ , назовем \tilde{M} - исходной моделью текста для размещения.

Интерпретация.

Структура размещения на носителе определяет иерархию единиц носителя (с их именами), на которых будет представлен текст.

Задание на размещение определяет содержание этих рубрик на носителе. Особо можно выделить случай задания содержания логически, например, книги - это аспекты, разделы, подаспекты "I-ого уровня" и т.д.

Моделью пользователя при размещении будем называть перечень кодов идентификации представления (КИП) текстовой информации, т.е. множество $K = \langle K_1, K_2, \dots, K_n \rangle$.

Интерпретация.

В качестве КИП могут, например, выступать:

- код представления информации списком;

- код табличной информации;
- код некоторой формы;
- код ссылки и т.д.

Перечень КИП определяется, исходя из проведенной классификации видов представления информации (в частности, выходных форм) и обеспечиваемых операцией Вывод программы вывода.

Сцеплением модели текста с моделью пользователя назовем $S = \langle \tilde{M}, K, \xi, L, \delta, T, \hat{M} \rangle$, где \tilde{M} - исходная модель, K - модель пользователя, $\xi: D \rightarrow K = K \times L$, где D - множество дуг графа $\tilde{\Gamma}$ исходной модели текста для размещения, L - некоторое множество, $\delta: D \rightarrow T$, где T - трансформатор единицы текста из одного представления в другое, \hat{M} - модель трансформированного текста.

Интерпретация.

Множество L служит для уточнения КИП, в частности, в случае определения числовых параметров $L = \bigcup_{k=1}^{\infty} R^k$, где R - множество действительных чисел, R^k - k -я декартова степень R .

Кроме того, при сцеплении некоторые КИП определяются с помощью правил размещения. В качестве таких правил могут выступать принятые в системе способы размещения, например, способы выделения на носителе рубрик текста, оформление титульных листов, печать стандартного текста, правила переноса, представление "больших" единиц текста более "мелкими" и т.д. В ряде случаев включается трансформация единицы текста к "каноническому" виду, т.е. обеспеченному программами вывода; например, может производиться детализация единицы текста, соответствующей некоторой форме, т.е. переход от этой формы к формам, представляющим поля этой формы и соответствующим единицам текста; эта процедура производится, если "абстрактная" форма, выдаваемая операцией ТЕКСТИРОВАНИЕ, не помещается в полосу, соответствующую АЦПУ.

Элементарным размещением текста на носителе назовем $\varepsilon_k = \langle \hat{k}, t_k, \mathcal{O}_k, S \rangle$, где $\hat{k} \in \hat{K}$ - КИП, $t_k: \mathcal{O}_k \rightarrow S$ - отображение, соответствующее \hat{k} , \mathcal{O}_k - множество единиц текста, имеющих данный КИП, S - носитель.

Интерпретация.

Элементарное размещение представляет задание имени некоторой программы вывода, соответствующей КИП. Сопоставление единиц текста с ее КИП имени программы вывода осуществляется с помощью правил размещения, закладываемых в операцию РАЗМЕЩЕНИЕ.

Модель размещения текста на носителе назовем четверку $\langle \tilde{M}, C, \tau, R \rangle$, где \tilde{M} - исходная модель текста для размещения, C - сцепление модели текста с моделью пользователя, $\tau: \hat{K} \rightarrow R$, где R - множество элементарных размещений.

Интерпретация.

Размещение представляется как набор элементарных размещений единиц текста, реализуемых программами вывода.

6.3.2. Основные операции.

6.3.2.1. Операция РАЗМЕТКА (модели текста для размещения).

Вход и источники входов:

- модель текста с разметкой укрупненными КИП единиц текста; поступает из операции ТЕКСТИРОВАНИЕ или операции СОКРАЩЕНИЕ, выполненной после операции ТЕКСТИРОВАНИЕ;
- задание на размещение, определяющее состав укрупненных единиц носителя, задается проектировщиком;
- дополнительная информация для детализации единиц текста; поступает от проектировщика;
- дополнительная информация о КИП и их параметрах; поступает по мере необходимости в процессе работы операции от проектиров-

щика.

Содержание операции и выход:

В соответствии с заданием на размещение анализируются единицы текста и их КИП, при этом:

- некоторые единицы текста "детализируются", т.е. представляются как собираемые из других, более "элементарных" единиц с помощью операции ДЕТАЛИЗАЦИЯ, и производится разметка КИП этих элементарных единиц;

- имеющиеся КИП уточняются для последующего вывода операцией ВЫВОД с помощью сервисной операции МЕТРИЧЕСКИЕ ЗНАЧЕНИЯ, служащей для определения параметров (всех или части), соответствующих этим КИП программ вывода;

- часть параметров уточняется с помощью запроса проектировщику.

После детализации текста и уточнения КИП единицы текста производится разметка этой единицы текста идентификатором соответствующего трансформатора (если необходимо) и программы вывода.

6.3.2.2. Операция МЕТРИЧЕСКИЕ ЗНАЧЕНИЯ.

Вход и источники входов:

- единица текста, поступает из операции РАЗМЕТКА;
- КИП единицы текста, поступает из операции РАЗМЕТКА.

Содержание операции и выход:

Операция определяет некоторые параметры единицы текста и распадается на ряд элементарных операций, соответствующих различным КИП.

6.3.2.3. Операция ДЕТАЛИЗАЦИЯ.

Вход и источники входов:

- единица текста, поступает из операции РАЗМЕТКА;
- выделение в форме, соответствующей единице текста, некото-

рых полей.

Содержание операции и выход:

Операция "расщепляет" единицу текста на более элементарные единицы текста, соответствующие выделенным полям.

6.3.3. Общая схема выполнения операции.

Множество заполненных форм, являющееся выходом операции ТЕКСТИРОВАНИЕ (или последующей операции СОКРАЩЕНИЕ), разбивается на классы, соответствующие укрупненным единицам носителя (книгам, разделам и т.п.); это разбиение производится по заданию на размещение, задаваемому проектировщиком, в результате получается исходная модель текста для размещения.

Исходная модель текста для размещения поступает для обработки операцией РАЗМЕТКА, которая детализирует единицы текста операцией ДЕТАЛИЗАЦИЯ и определяет значения параметров КИП единиц текста с помощью операции МЕТРИЧЕСКИЕ ЗНАЧЕНИЯ или запроса проектировщику, после чего размечает единицы текста для вывода операцией ВЫВОД.

6.4. Выход операции.

Выходом операции РАЗМЕЩЕНИЕ является размеченная именами трансформаторов и вывода программ, модель размещения текста на носителе, представляющая задание для операции ВЫВОД.

6.5. Требования к внесению изменений.

Операция ВНЕСЕНИЕ ИЗМЕНЕНИЙ должна обеспечить следующие возможности:

- изменять структуру размещения;
- изменять задание на размещение;

- изменять КИПы и уточняющие их параметры;
- изменять детализацию единиц текста;
- изменять модель текста.

7. ВЫВОД

7.1. Задачи, решаемые операцией.

Операция предназначена для реализации решения операции РАЗМЕЩЕНИЕ по выводу форм.

Эта задача распадается на ряд подзадач:

- 1. Распараллеливание вывода форм;
- 2. Выбор формы для вывода на носитель в определенный момент времени;
- 3. Формирование и вывод формы.

7.2. Вход в операцию.

Входом операции являются:

- 1. Модель размещения текста на носителе;
- 2. Задаваемые проектировщиком имена частей носителя (в соответствующей иерархии частей носителя), допускающих одновременный вывод на разные части носителя.

7.3. Концептуальная схема.

Определение 7.3.1.

Пусть A - дерево с вершиной v . Назовем поддеревьями вершины v деревья A_i , удовлетворяющие условиям:

- 1. A_i - поддерево A ;
- 2. Вершина A_i соединена дугой с v .

Обходом дерева A будем называть последовательность следующих действий:

- 1. Обход слева направо всех поддеревьев v ;
- 2. Выполнение действия B для вершины v .

Эти действия определяются для каждого обхода при употреблении этого понятия.

Определение 7.3.2.

Деревом вывода формы Φ (где Φ соответствует вершине σ трансформированной модели текста \hat{M}) назовем двойку $\langle DB, \alpha \rangle$, где DB - подграф \hat{M} такой, что:

- 1. DB - дерево с вершиной σ ;
- 2. Все вершины DB входят в нижнее замыкание σ ;
- 3. Вершины DB не имеют порядковых кодов;

$\alpha : DB \rightarrow \{P\Phi\}$, $P\Phi$ - поле формирования, α - биекция.

Определение 7.3.3.

Выводом формы Φ в часть носителя A называется обход дерева вывода формы Φ , где в качестве действия для вершины z принято формирование (программой вершины z , по КИП и параметрам z) $\alpha(z)$.

Дополнительным действием для вершины DB является вывод ее поля формирования в часть носителя A .

Определение 7.3.4.

Деревом форм части носителя A называется набор форм $\{\Phi\}$, удовлетворяющий следующим условиям:

- 1. Формы $\{\Phi\}$ сопоставлены части A носителя;
- 2. Формы $\{\Phi\}$ имеют порядковый код;
- 3. Формы $\{\Phi\}$ представлены в виде дерева (которое строится по порядковым кодам) и порядковые коды вершин одного уровня возрастают слева направо.

Определение 7.3.5.

Заполнением части носителя A называется обход дерева форм части носителя A . Сопоставляемым каждой вершине дерева действием является вывод форм этой вершины.

Определение 7.3.6.

Пусть A - дерево с вершиной v . Будем считать, что каждой вершине дерева A сопоставлены действия c и действия ϑ . Обходом дерева A с двойным проходом вершин назовем последовательность следующих действий: Выполнение действий c для v , обход с двойным проходом вершин слева направо всех поддеревьев; выполнение действий ϑ для вершины v . Будем считать, что для начальных вершин A определены стандартные действия ϑ не производить никаких действий.

Определение 7.3.7.

Заполнением носителя E называется обход дерева структуры E с двойным проходом вершин. Действия c - оформление начала структурной единицы с вершиной v , если v - не начальная вершина, и заполнение части носителя, соответствующей v , в противном случае.

Действия ϑ - оформление конца структурной единицы v .

Определение 7.3.8.

Будем говорить, что вершина v дерева \mathcal{K}_s допускает параллельный вывод, если $\mu (v_i)$ допускает вывод на разные носители; здесь v_i - поддеревья вершины v (определение μ см. 6.3. в определении задания на размещение).

Определение 7.3.9.

Иницированием обхода (обхода с двойным проходом вершин) дерева A будем называть выбор первого действия, (определенного для вершин дерева A), которое можно выполнить.

Завершением обхода дерева A будем называть завершение действия (последнего действия) для вершины A .

Определение 7.3.10.

Состоянием процесса вывода назовем подграф \tilde{F}_s дерева \mathcal{K}_s та-

кой, что:

- 1. $\tilde{\mathcal{F}}_s$ имеет единственную конечную вершину, совпадающую с вершиной \mathcal{F}_s ;
- 2. Все вершины $\tilde{\mathcal{F}}_s$ допускают параллельный вывод;
- 3. Все вершины $\tilde{\mathcal{F}}_s$ разбиты на 3 класса: В (выполнено), Н (невыполнено), П (промежуточный).

Определение 7.3.II.

Начальным состоянием процесса вывода называется состояние, при котором все вершины $\tilde{\mathcal{F}}_s$ принадлежат классу Н.

Конечным состоянием процесса вывода называется состояние, при котором вершина $\tilde{\mathcal{F}}_s$ принадлежит классу В.

Определение 7.3.I2.

Выводом называется процесс перехода от начального состояния в конечное. Процесс перехода задается обходом $\tilde{\mathcal{F}}_s$, при котором вершине v сопоставляются следующие действия:

1. Для всех вершин из класса П проверяется - не завершено ли заполнение соответствующего носителя. Все вершины, для которых заполнение завершено, считаются вершинами класса В, а соответствующие им носители - свободными;

2. Если существует свободный носитель, то инициируется заполнение этого носителя, вершина v переносится в класс П, и продолжается обход $\tilde{\mathcal{F}}_s$;

3. Если свободного носителя не существует, то обход прекращается до тех пор, пока не освободится один из носителей. После этого инициируется заполнение этого носителя, вершина v переносится в класс П, и продолжается обход $\tilde{\mathcal{F}}_s$.

7.4. Вывод операции.

Выходом операции является размещенный на носителях текст проекта, представленный в виде отдельных упорядоченных наборов форм.

7.5. Требования к внесению изменений.

Внесение изменений должно обеспечивать:

- 1. Изменения имен частей носителя, допускающих одновременный вывод на различные носители;
- 2. Изменения модели размещения текста на носителе.