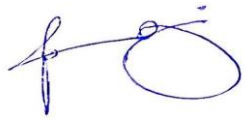
Министерство науки и высшего образования Российской Федерации
федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технологический университет «СТАНКИН»

На правах рукописи



Ньи Ньи Хтве

РАЗРАБОТКА МЕТОДА И СРЕДСТВ ПОДДЕРЖКИ ПРОЦЕССОВ ОБРАБОТКИ КОНЦЕПТУАЛЬНЫХ ПРЕДСТАВЛЕНИЙ ПРЕДМЕТНЫХ ЗАДАЧ

2.3.1 – Системный анализ, управление и обработка информации,статистика

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель:
доктор технических наук, профессор
Волкова Галина Дмитриевна

Москва – 2024

Оглавление

ВВЕДЕНИЕ……6

ГЛАВА 1. АНАЛИЗ СУЩЕСТВУЮЩИХ ПОДХОДОВ, МЕТОДОВ И ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ ПОДДЕРЖКИ МОДЕЛИРОВАНИЯ И ОБРАБОТКИ ЗНАНИЙ……11

1.1 Анализ применяемых методов и подходов для семантического моделирования предметных задач в процессе автоматизации 11

1.1.1 Общая характеристика подходов к автоматизации прикладных задач..... 11

1.1.2 Сравнительный анализ методов для семантического моделирования предметных задач..... 14

1.2 Анализ существующих инструментальных средств для семантического моделирования предметных задач, подлежащих автоматизации 36

1.2.1 Общая характеристика инструментальных средств, используемых при автоматизации информационных и интеллектуальных процессов 36

1.2.2 Сравнительный анализ инструментальных средств, используемых в рамках методологии автоматизации интеллектуального труда 38

1.3 Особенности концептуального моделирования в рамках методологии автоматизации интеллектуального труда 50

1.4 Выводы по главе 1..... 56

ГЛАВА 2. УТОЧНЕНИЕ ФОРМАЛЬНОГО ОПИСАНИЯ КОНЦЕПТУАЛЬНЫХ МОДЕЛЕЙ И РАЗРАБОТКА ФОРМАЛЬНОГО АППАРАТА ДЛЯ ПРОЦЕДУР ИХ ОБРАБОТКИ……57

2.1 Исследование особенностей процесса концептуального моделирования в рамках методологии автоматизации интеллектуального труда..... 57

2.2 Уточнение формального описания концептуальных представлений предметных задач..... 58

2.2.1 Общее описание 58

2.2.2 Уточнение формального описания концептуальной модели 1-го рода предметной задачи на объектном уровне 61

2.2.3 Уточнение формального описания концептуальной модели 1-го рода предметной задачи на конкретном уровне 66

2.2.4 Уточнение формального описания взаимосвязей концептуальных моделей 1-го рода предметной задачи объектного и конкретного уровней..... 74

2.3 Формальное описание типологии предметных ограничений 1-го рода..... 79

2.3.1 Общая характеристика типологии предметных ограничений 1-го рода..... 79

2.3.2 Формальное описание типологии предметных ограничений 1-го рода на объектном уровне 80

2.3.3 Формальное описание типологии предметных ограничений 1-го рода на конкретном уровне 85

2.4 Разработка формального описания статистических предметных ограничений 1-го рода 92

2.4.1 Общая характеристика статистических предметных ограничений 1-го рода 92

2.4.2 Формальное описание статистических предметных ограничений 1-го рода 93

2.5 Выводы по главе 2 95

ГЛАВА 3. РАЗРАБОТКА МЕТОДИЧЕСКИХ ОСНОВ ВЫДЕЛЕНИЯ, ОПИСАНИЯ И ОБРАБОТКИ СТАТИСТИЧЕСКИХ ПРЕДМЕТНЫХ ОГРАНИЧЕНИЙ 1-ГО РОДА ПРИ КОНЦЕПТУАЛЬНОМ МОДЕЛИРОВАНИИ……..97

3.1 Общая характеристика процесса концептуального моделирования 97

3.2 Разработка начального модельного представления задачи «Обработка статистических предметных ограничений 1-го рода при концептуальном моделировании прикладных задач» 103

3.2.1 Общая характеристика задачи в целом 103

3.2.2 Разработка начальной модели для процедуры A 104

3.2.2.1 Формирование диаграмм системы предметных действий для процедуры A 104

3.2.2.2 Формирование диаграммы системы предметных действий в целом для процедуры A 123

3.2.2.3 Формирование множества параметров для процедуры A 127

3.2.2.4 Формирование матричных диаграмм для процедуры A 129

3.2.3 Разработка начальной модели для процедуры B 141

3.2.3.1 Формирование диаграмм системы предметных действий для процедуры B 141

3.2.3.2 Формирование диаграммы системы предметных действий в целом для процедуры B 147

3.2.3.3 Формирование множества параметров для процедуры B 148

3.2.3.4 Формирование матричных диаграмм для всех конструкций процедуры B 148

3.2.4 Разработка начальной модели для процедуры C 153

3.2.4.1 Формирование диаграмм системы предметных действий для процедуры C 153

3.2.4.2 Формирование диаграммы системы предметных действий в целом для процедуры C..... 159

3.2.4.3 Формирование множества параметров для процедуры C..... 161

3.2.4.4 Формирование матричных диаграмм для процедуры C 161

3.2.5 Разработка начальной модели для процедуры D..... 167

3.2.5.1 Формирование диаграмм системы предметных действий для процедуры D 167

3.2.5.2 Формирование диаграммы системы предметных действий в целом для процедуры D 172

3.2.5.3 Формирование множества параметров для процедуры D 173

3.2.5.4 Формирование матричных диаграмм для процедуры D..... 174

3.3 Выводы по главе 3 177

ГЛАВА 4. РАЗРАБОТКА ПРОГРАММНЫХ СРЕДСТВ ПОДДЕРЖКИ ОБРАБОТКИ СТАТИСТИЧЕСКИХ ПРЕДМЕТНЫХ ОГРАНИЧЕНИЙ ПРИ КОНЦЕПТУАЛЬНОМ МОДЕЛИРОВАНИИ ПРИКЛАДНЫХ ЗАДАЧ……178

4.1 Общее описание программного комплекса..... 178

4.2 Структура информационной базы программного комплекса 180

4.2.1 Описание табличной структуры базы данных 180

4.2.2 Описание содержания таблиц базы данных..... 184

4.3 Состав и структура программных модулей..... 195

4.4 Инструкции по установке и по эксплуатации программного обеспечения.... 198

4.4.1 Функции администратора..... 198

4.4.2 Функции пользователя - аналитика..... 202

4.4.3 Функции пользователя - эксперта 212

4.5 Выводы по главе 4..... 217

ЗАКЛЮЧЕНИЕ……218

СПИСОК ЛИТЕРАТУРЫ……...220

ПРИЛОЖЕНИЕ А……229

ПРИЛОЖЕНИЕ Б……238

ПРИЛОЖЕНИЕ В. Справка об использовании результатов диссертационного исследования……246

СПИСОК ОБОЗНАЧЕНИЙ

ЭОД – электронная обработка данных
ИТ – информационные технологии
НИТ– новые информационные технологии
КП – когнитивный подход
МАИТ – методология автоматизации интеллектуального труда
АС –автоматизированная система
ПАС – прикладная автоматизированная система
ИС – инструментальные средства
УКП – универсальное концептуальное представление
КППЗ – концептуальное представление предметных задач
ПрЗ – предметная задача
КМ – концептуальная модель
ПЗ-1– предметная зависимость 1-го рода
СПЗ-1 – система предметных зависимостей 1-го рода
ЭПЗ-1– экземпляр предметной зависимости 1-го рода
ПК– предметная категория
КС – концептуальная структура на ПК
ЭПК– экземпляр предметной категории
$\boldsymbol{LX_i}$ – описывает подмножество бинарных связей вида “упорядочивание” («У») между элементами одного подмножества $\boldsymbol{X_i}$
$\boldsymbol{LX_iX_j}$ – описывает подмножество бинарных связей вида “состав” («С») между элементами разных подмножеств $\boldsymbol{X_i}$ и $\boldsymbol{X_j}$
$\boldsymbol{LX_j/X_i}$ – описывает подмножество бинарных связей вида “компоновка” («К») между элементами одного подмножества $\boldsymbol{X_j}$ при условии (/) их связи с одним элементом другого подмножества $\boldsymbol{X_i}$
Описание структуры выражений:
$\boldsymbol{X}$ = {$\boldsymbol{x_i}$ | < условие_1 для x_i **&** условие_2 для x_i **&**….**&** условие_n для x_i >},
где $\boldsymbol{x_i}$ -элемент множества
Характеристики алгоритмических конструкций:
ФП – функция последовательности, **ФА –** функция альтернативы, **АА –** аргумент альтернативы, **СЧ** – счетчик итераций, **АИ –** аргумент итерации, **ФИ** – функция итерации, **ФЦ** – функция цикла
МД – Матричная диаграмма
PHP (Hypertext Preprocessor) – интерпретируемый скриптовый язык программирования общего назначения
HTML (HyperText Markup Language) – стандартизированный язык гипертекстовой разметки документов для просмотра веб-страниц в браузере
CSS (Cascading Style Sheets) – формальный язык декодирования и описания внешнего вида документа (веб-страницы), написанного с использованием языка разметки (чаще всего HTML или XHTML)
JavaScript – мультипарадигменный язык программирования
MySQL – свободная реляционная система управления базами данных

ВВЕДЕНИЕ

Актуальность исследования. Интеллектуальный ресурс любой организации или промышленного предприятия включает как опыт и знания работающих в них специалистов и руководителей, так и разнообразные фонды, хранящиеся и в документной, и в электронной форме.

Фонды могут иметь различное назначение, например, организованы в виде архивов технической и организационной документации, библиотечных фондов, фондов регламентов, стандартов, патентов, методик, типовых решений и др. По сути, документные фонды предприятий представляют собой аккумулированные знания многих поколений специалистов.

В современных условиях, связанных с дефицитом квалифицированных инженерных кадров и ситуацией «разрыва поколений», на промышленных предприятиях остро стоит вопрос сохранения и дальнейшего развития интеллектуального потенциала. Специфика инженерных знаний заключается в их сложной организации, обусловленной необходимостью их увязывать в рамках жизненного цикла создаваемых машиностроительных изделий.

Одним из решений этого вопроса является перенос интеллектуального ресурса в компьютерную среду. Но простое сканирование и/или формирование документов непосредственно в компьютере не позволит решить эту проблему. Требуется перевести в компьютерную среду не только «форму», но и «содержание» документа. Это возможно при наличии формализованных методов семантического моделирования предметных областей, в которых знания зафиксированы в различных средах: бумажных носителях, в компьютере, в памяти специалистов.

Степень разработанности темы исследования. Метод концептуального моделирования предметных задач в рамках методология автоматизации интеллектуального труда (МАИТ), разработанный на кафедре «Информационные технологии и вычислительные системы» МГТУ «СТАНКИН», позволяет представить систему знаний предметной области в виде формализованных

моделей. Эти семантические модели обеспечивают единую интерпретацию последующих формально-языковых представлений предметных задач, подлежащих автоматизации.

МАИТ, обеспечивающая промышленный способ создания автоматизированных систем (АС) в рамках когнитивного подхода, теоретический обоснована доктором технических наук, профессором Волковой Галиной Дмитриевной.

Развитие формальных методов и методик данной методологии получили в работах Новоселовой О.В., Семячковой Е.Г., Курышева С. М., Щукина М.В., Сироты И.М., Бычковой Н.А., Протасовой С.В., Володиным Д.А.,Тюрбеевой Т.Б. и др.

Обозначенные проблемы, связанные с моделированием и поддержкой системы предметных знаний при проектировании и реализации АС, могут быть решены не только с помощью методов моделирования знаний, но требуются способы и средства поддержки обработки концептуальных представлений предметных задач. Это позволило сформулировать цель работы и поставить научную задачу.

Целью данного исследования является повышение эффективности процесса концептуального моделирования предметных задач за счет разработки метода и средств обработки компонентов семантического представления, разработанных в рамках методологии автоматизации интеллектуального труда.

Для достижения поставленной цели в работе была решена **научная задача,** включающая:

– изучение применяемых методов и подходов, а также инструментов для семантического моделирования предметных задач;

– разработку уточненного формального описания концептуального представления предметных задач;

– разработку метода обработки статистических предметных ограничений при концептуальном моделировании прикладных задач;

– разработку методических основ выделения, описания и обработки статистических предметных ограничений при концептуальном моделировании;

– разработку программных средств поддержки обработки статистических предметных ограничений при концептуальном моделировании прикладных задач.

Объектом исследования является семантические модели предметной области.

Предметом исследования является формальное описание концептуальной модели, представляющей документированные технические знания с учетом различной степени их формализации.

Научная новизна:

– установлены связи между характеристиками моделей статистических (таблично оформленных) знаний и их аналитической интерпретацией;

– разработано уточненное формальное описание концептуального представления предметных задач;

– разработан метод обработки статистических предметных ограничений 1-го рода при концептуальном моделировании прикладных задач.

Теоретическая значимость. Разработанный метод может быть использован для теоретического обоснования систематизации и классификсации концептуальных представлений таблично организованных знаний в различных предметных областях при автоматизации интеллектуального труда.

Практическая значимость. Разработано методическое обеспечение в виде начального модельного представления задачи «Обработка статистических предметных ограничений при концептуальном моделировании прикладных задач» как совокупности процедур:

– выделение всех статистических предметных зависимостей 1-го рода и их предварительная обработка, определение содержания всех статистических предметных зависимостей 1-го рода;

– формирование основной концептуальной структуры для фрагмента концептуальной модели под статистическую предметную зависимость 1-го рода;

– определение типологии статистических предметных зависимостей 1-го рода и документирование моделей для всех статистических предметных зависимостей 1-го рода;

– разработаны инструментальные средства для поддержки обработки концептуальных моделей.

Методы исследования. При разработке теоретических положений диссертационной работы были использованы аппараты теории множеств, математической логики, реляционной алгебры, теории баз данных, метод концептуального моделирования (в соответствии с МАИТ).

Положения, выносимые на защиту:

– уточненное формальное описание концептуального представления предметных задач как совокупности моделей 1-го и 2-го рода;

– метод обработки статистических ограничений 1-го рода при концептуальном моделировании предметных задач;

– методическое обеспечение в виде совокупности начальных моделей;

– программные модули для процедур обработки концептуальных моделей при создании прикладных АС.

Соответствие паспорту специальности. Научная работа соответствует формуле научной специальности 2.3.1- Системный анализ, управление и обработка информации, статистика, а именно: п.2- «Формализация и постановка задач системного анализа, оптимизации, управления, принятия решений и обработки информации»; п.4- «Разработка методов и алгоритмов решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации».

Степень достоверности и апробации результатов. Результаты, полученные в диссертационной работе, подтверждаются соответствием экспериментальных и теоретических исследований. Также результаты исследования рекомендовано применять в учебном процессе кафедры «Информационные технологии и вычислительные системы» МГТУ «СТАНКИН» при подготовке специалистов по направлению «Информатика и вычислительная техника».

Основные положения и результаты диссертационного исследования докладывались и обсуждались на российских и международных конференциях: III всероссийская научно-практическая конференция «Цифровая экономика: технологии, управление, человеческий капитал», Москва, 2020г.; XIII всероссийская конференция с международным участием «Машиностроение: традиции и инновации (МТИ – 2020)», Москва, 2020г.; XXVIII конференция «Математика. Компьютер. Образование – 2021» МКО-2021, 25-30 января 2021 г.; Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus) 2021 IEEE.

Публикации по теме работы. По теме диссертации опубликовано 7 научных работ, в том числе в изданиях, рекомендованных ВАК РФ - 2 научные работы, в изданиях, входящих в базы данных Scopus - одна научная работа.

Структура и объем диссертации. Диссертация состоит из введения, 4глав, заключения, списка литературы из 91 наименований. Работа содержит 246 страниц сквозной нумерации, включая 183 рисунков, 43 таблиц и 18 страниц приложений.

ГЛАВА 1. АНАЛИЗ СУЩЕСТВУЮЩИХ ПОДХОДОВ, МЕТОДОВ И ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ ПОДДЕРЖКИ МОДЕЛИРОВАНИЯ И ОБРАБОТКИ ЗНАНИЙ

1.1 Анализ применяемых методов и подходов для семантического моделирования предметных задач в процессе автоматизации

1.1.1 Общая характеристика подходов к автоматизации прикладных задач

Широкое использование компьютеров в различных областях промышленности и в управлении, научных исследованиях и бизнесе, сфере обслуживания и сфере образования привело к становлению новой индустрии - компьютерной. Прикладные автоматизированные системы внедряются и используются практически повсеместно. Понятие «эффективные технологии» много раз менялось на протяжении десятилетий использования компьютеров в различных сферах и это определяло существенное влияние на автоматизацию информационных и интеллектуальных процессов [1,2].

Появление и развитие новых технологий всегда связано с текущими проблемами и насущными вопросами. Их решение обуславливает становление новых методов и средств автоматизации интеллектуальной деятельности. Можно условно выделить несколько этапов развития и смены технологий в этой области: [2,79]:

- электронная обработка данных (ЭОД) или технологии традиционного программирования,
- информационные технологии (ИТ) или технологии структурного программирования,
- новые информационные технологии (НИТ) или технологии объектно-ориентированного программирования и методов искусственного интеллекта,

- когнитивный подход или когнитивные технологии.

Основным критерием, с помощью которой можно охарактеризовать все перечисленные этапы, является набор модельных представлений предметных задач, подлежащих автоматизации. Модельные представления лежат в основе всех методов и средств вышеуказаных технологий.

К таким модельным представлениям следует отнести:

- представления об окружающем мире,
- концептуальное или методологическое представление,
- представление, связанное с размещением информации в программно-технической среде.

Компьютерное представление определяет:

- организацию информации в вычислительной среде, т.е. хранение данных;
- организацию доступа к данным и обработки данных в вычислительной среде;
- организацию пользовательского интерфейса – представление информации в удобной форме для пользователя с различными возможностями внутри интерфейса.

Технические и вычислительные возможности определяли развитие технологии традиционного программирования. В качестве основы выделялось упрощенное представление реального мира - множество реальных объектов с присущими им свойствами [4,5,6]. Сначала вся информация представлялась как множество переменных, размещенных в оперативной памяти компьютер. Далее данные сохранялись не только в оперативной памяти, но также во внешней памяти. Это привело к необходимости введения операторов «ввода-вывода» и обмена данными между носителями в языках программирования [4]. Для обеспечения взаимодействия с пользователем появляется «оконный» интерфейс.

Формальной моделью информационного представления в компьютре выступает теория формальных грамматик [4]. С увеличением сложности решаемых задач возникали трудности в их автоматизации. Возникали различные проблемы: большое количество переменных усложняло процесс написание программ, их отладку и тестирование; часто происходило дублирование данных в больших

задачах; сложность извлечения данных из памяти компьютера с использованием специальных программ, реализующих запросы пользователей [2, 4, 5, 6].

Переход к этапу информационных технологий ознаменовался качественным скачком при разрешении количественных проблем – множество переменных преобразились в структуры данных. Реальный мир стал отражаться в вычислительной среде через объекты и их связи.

Моделирование данных в виде структур данных получило формальные воплощения в компьютерной среде. Наиболее полным представлением явилась реляционная модель Кодда. Методологическое обоснование новой парадигмы моделирования данных было сформулировано Питером Ченом в 70-х годах прошлого века и получило название «сущность-отношение-атрибут» или ERA – подход. Этот подход является доминирующим в моделировании данных и в настоящее время.

Переход к организацию информации в виде баз данных првело к появлению операторов доступа к данным в виде языков манипулирования данными (ЯМД) в системах управления базами данных (СУБД). Со временем это привело к качественному изменению в интерфейсе - от позиционирования символа к позиционированию целого блока. Этот этап стал первым шагом в развитии графического интерфейса пользователя [4].

Однако, разделение доступа к структурам данных в базах данных и манипулирования данными в программах обработки не могло продолжаться долго.

Промежуточным решением проблемы разделение доступа и манипулирования данными было появление концепции абстрактных типов данных для языков программирования. А также был сформирован структурный подход в программировании. Кроме того, необходим был единый подход на уровне одного языка программирования для представления как сложных данных, так и их категорий, доступа к ним и манипуляцией как данными, так и их категориями с организацией всех типов интерфейсов - символьного, визуального или звукового.

На этапе новых информационных технологий появились новые технологий: фреймовой подход (методе Мински), логическое и объектно-ориентированное

программирование. На данном этапе появилось множество новых непроцедурных формализмов - семантические сети, фреймы, логические системы, продукционные модели [2,4].

Особенностью первых трех этапов эволюции автоматизации иинформационных и интеллектуальноых процессов является то, что отображение окружающего мира в простую семантическую конструкцию «сущность-связь-атрибут» порождает огромную размерность информационного представления в компьютерной среде, что значительно усложняет обработку и перегружает интерфейс [4].

Когнитивный подход как этап развития автоматизации интеллектуального труда продолжает развиваться и в настоящее время. Суть этого этапа заключается в многоуровневом представлении реального мира, где каждый уровень содержит в себе три функциональных центра - контекстуальный, структурный и монадический, связанные между собой на основе закона цикличности научного познания [2,7,79]. Сводная таблица по этапам приведена на табл. 1.1.

Анализ развития автоматизации иинформационных и интеллектуальных процессов показал, что смена этапов отражает последовательное усложнение информационного представления задач в компьютерной среде, а первые три этапа можно сопоставить с соответствующими центрами когнитивного подхода [2,4,79].

При этом для всех этапов главной провлемой является переход от естественно-языкового представления информации и знаний предметных задач к их формально-языковому представлению в памяти компьютера [4].

1.1.2 Сравнительный анализ методов для семантического моделирования предметных задач

Анализ методов семантического моделирования выполнялся по следующим критериям: область деятельности, для которой создавался мотод; начало разработки; основания семантического моделирования/ базовые элементы; виды и

разнообразие статических отношений/структур; виды и разнообразие динамических отношений/структур; наличие закономерностей формирования структур/ конструкций; наличие формального аппарата для описания отношений/структур и для их интеграции; -наличие формального аппарата для обработки моделей/ структур; методическое обеспечение; программная поддержка [2,79]. В качестве основных методов для анализа были выбраны следующие представители:

в рамках этапа ЭОД: метод тезаурусов [8];

в рамках этапа ИТ: метод Чена (ERA- подход) [9,10,11,12,13,14,33];

в рамках этапа НИТ: декларативные методы искусственного интеллекта (семантические сети, фреймы) [22,23,24,25,26], процедуральные методы искусственного интеллекта (логический вывод, продукции) [23,27,28], метод онтологий [29,30,31,32,42], методы семантического моделирования в объектно-ориентированном подходе [34-41,43,44];

в рамках этапа когнитивного подхода: метод концептуального моделирования в методологии автоматизации интеллектуального труда [1,2,3].

Метод тезаурусов. Развитие технологий часто подразумевает создание определенных систем данных. Системы данных состоят из разных категорий информации, которые должны быть представлены в конкретной форме, подходящей для автоматизации процесса работы с данными. Основывается такой подход на тезаурусе предметной области. Тезаурус предметной области – это совокупность ключевых понятий определенной сферы, которые связаны друг с другом определенными смысловыми отношениями. Это понятие должно представлять терминологию для описания ресурсов данных, которые относятся к конкретной предметной области [8].

Первые тезаурусы появились с началом автоматизации библиотечной деятельности в 60-х годах XX века для обеспечения поиска документов по ключевым словам (понятиям предметной области). Базовые элементы тезауруса – это понятия-элементы и семантические (парадигматические) отношения на них, такие как родовидовые, синонимии, ассоциации.

Табл.1.1. Этапы развития автоматизации интеллектуального труда

N	Представления о реальном мире	Представления о концептуальном мире	Представления об информационном мире					
			Организация информации и знаний		Организация доступа	Организация обработки	Организация интерфейса	
			данные	категории			тип	реализация
1	Свойства объектов	Атрибуты	Переменные/ данные	-	ЯП	ЯП	символьный	ЯП
2	Объекты и их состояния, связи объектов, и их свойства	ERA-подход	Структуры на переменных/ структуры данных (ЯОД)	-	ЯМД доступ к данным	ЯП/ (ЯМД) управляющие конструкции	символьный, графический	ЯП (позиционирование курсора, меню)
3	Объекты, связи объектов, свойства объектов, события	ERA-подход	Структуры на переменных/ структуры данных (фрейм /объект)	Структуры на переменных/ структуры данных (фрейм /объект)	ЯЛП или ЯФП, ЯП, ООЯП	ЯЛП или ЯФП, ЯП, ООЯП (методы к объектам)	символьный, графический звуковой	ЯП, ООЯП (позиционирование курсора, блока, выпадающие меню, окна)
4	Многоуровневое представление ситуаций или явлений: *i*- уровень	Функциональные центры: *К*-контекстуальный, *С*-структурный, *М*- монадический. Закон цикличности: *M(i)=K(i+1)* Концептуальные представления и модели	Структуры на переменных/ структуры данных	Структуры на переменных/ структуры данных	Системы логических доступов	Системы логических обработок	символьный, графический звуковой	Системы интерфейсных элементов

Организация понятий выполняется по нескольким аспектам: по алфавитному перечню понятий; по иерархии понятий; по классам условной эквивалентности.

Тезаурусы, которые описывают какую-либо предметную область, часто создаются долго и планомерно, в основном здесь работает индивидуальный подход. Однако, этот процесс сопряжен со многими трудностями, так как вся терминология создается отдельно вручную.

Тезаурус может относиться к любой из следующих реализаций:

1. Тезаурус - это программный инструмент, входящий в состав некоторых текстовых процессоров, который предоставляет синонимы для выбранных слов по команде. Пользователи, использующие Microsoft Word, могут открыть тезаурус, выделив слово, которое они хотят найти, и нажав комбинацию клавиш Shift +F7.

2. Тезаурус – это собрание понятий, программное обеспечение или онлайн-сервис, который предоставляет альтернативные или похожие слова к данному понятию.

Тезаурус используется для группировки разных слов с одинаковым значением (синонимов) и похожих слов. С другой стороны, словарь объясняет значение слова.

Тезаурус как контрольный словарь может быть важным инструментом в обработке естественного языка (NLP). Однако создание тезауруса вручную экспертами может занять много времени. Кроме того, субъективность каждого эксперта может повлиять на структуру тезауруса. Уже реализовано множество методов для создания автоматического тезауруса на языках, которые классифицируются как богатые языковые ресурсы.

Обновление тезауруса происходит не сразу, так как связанной с этим работы достаточно много. Это «слабое место», так как для быстрого создания новых проектов трудно заново изобретать новые тезаурусы. Поэтому, как следствие, автоматизировать этот процесс стало необходимостью. Одним из ключевых препятствий, мешающих этому стало возникновение лишнего «шума». Для получения знаний большей точности используется последующий ручной процесс обработки полученного материала.

Формирование тезаурусов и их качество зависит от квалификации их разработчиков и определяется уровнем их кругозора. При этом отсутствуют формализмы для описания структур тезаурусов. Для формирования поисковых запросов к массивам документов используется аппарат математической логики.

Программная поддержка тезаурусов осуществляется в рамках автоматизированных информационно-поисковых систем как документального, так фактографического типа.

Методы структурного подхода. К данному подходу относят методы, основанные в том числе на SADT-методологии.

IDEF0 - IDEF0 (Integration Definition) был разработан компанией Integrated Computer-Aided Manufacturing (ICAM) ВВС США в 80-х годах XX века. Существует множество различных методов IDEF [15,16,17,18]. Полный список методов IDEF идет от IDEF0 до IDEF14. Каждый метод полезен для описания определенного аспекта предприятия. Метод моделирования функций IDEF0 предназначен для моделирования решений, действий и деятельности организации или системы. IDEF0 позволяет пользователю изобразить представление процесса, включая входы, выходы, элементы управления и механизмы (которые обычно называются метками ICOMs) [16,17]:

- входные данные: они обозначены стрелками в поле слева от представления процесса. Это ресурсы, которые процесс может использовать и преобразовать в результат (например, производственная информация, сырье);
- элементы управления: они представлены стрелками, которые входят в рабочее поле процесса сверху и определяют «почему» и «как» его выполнять. Они контролируют, ограничивают, инициируют и координируют действия (заказы клиентов, бизнес-стратегии, требования к дизайну и т.д.);
- выходы: они обозначены стрелкой в поле справа и являются результатом операции (например, график поставки, предлагаемая модель);

- механизмы: это стрелки, которые вводятся в поле снизу, они представляют ресурсы (машины, программное обеспечение, человеческие ресурсы и т.д.), необходимые для выполнения операции.

Абстрагирование от времени стало самым важным в методе IDEF0.

Как инструмент анализа, IDEF0 помогает разработчику модели правильно отражать, что происходит в текущей системе. В этом случае создание модели IDEF0 часто является одной из основных задач совершенствования системы.

В 1970-х годах ВВС США начали работу над программой интегрированного автоматизированного производства (ICAM). Так зародилось семейство языков моделирования IDEF. Аббревиатура «IDEF» первоначально обозначала «Определение ICAM», но теперь расшифровывается как «Определение интеграции», отражая его возможное использование для обмена информацией между различными языками моделирования. Вместо того, чтобы указывать один универсальный язык моделирования, проект ICAM определил следующие языки для различных задач:

- моделирование процессов IDEF0;
- концептуальное моделирование данных IDEF1;
- имитационное моделирование IDEF2.

Позже были добавлены другие языки, в том числе:

- моделирование данных IDEFIX («IDEF1 extended»). ;
- моделирование процессов IDEF3.

Несмотря на то, что IDEFIX основан на концептуальном языке IDEF1[12,13,15], он был изменен, чтобы сосредоточиться на логическом моделировании данных. Со временем IDEF3 включил в себя большую часть IDEF2 и IDEF5. В настоящее время IDEF0 и IDEF1X являются наиболее часто применяемыми на практике языками IDEF. Они поддерживаются в различных CASE-системах и широко используются в государственных секторах США, особенно в оборонной сфере. Язык IDEF3 также используется, хотя и в меньшей степени.

IDEF1X - это гибридный язык, объединяющий некоторые концептуальные понятия (например, сущность, отношения) с конструкциями реляционной базы данных (например, внешние ключи). Он был принят в качестве стандарта Национальным институтом стандартов и технологий (NIST) в 1993 году (NIST 1993). Предлагаемый преемник под названием $IDEF1X_{97}$ был одобрен в июне 1998 года Советом по стандартам IEEE-SA (IEEE 1999). Также известный как $IDEF_{object}$, этот расширенный $IDEF1X_{97}$ с объектно-ориентированными функциями облегчает реализацию в объектно-ориентированных базах данных и языках программирования, сохраняя при этом совместимость с IDEF1X для реализации реляционных баз данных. Эта обратная совместимость дает $IDEF1X_{97}$ одно преимущество перед UML для внедрения в государственных секторах США.

Тем не менее будущее $IDEF1X_{97}$ неопределенно, поскольку его объектно-ориентированные расширения менее всеобъемлющи, чем те, что используются в UML. Хотя UML еще не получил широкого распространения при проектировании баз данных, на сегодняшний день это наиболее широко используемый язык для разработки объектно-ориентированного кода, он поддерживается многими инструментами CASE и был принят в качестве стандарта Международной организацией по стандартизации (ISO). Немногие практикующие специалисты или поставщики инструментов выразили заинтересованность в внедрении $IDEF1X_{97}$. В отличие от этого, оригинальная (1993) версия языка IDEF1X продолжает очень широко использоваться для проектирования баз данных и поддерживается многими инструментами моделирования.

Язык моделирования данных IDEF1X (Integration Definition for Information Modeling) позволяет формировать семантические модели данных, которые обеспечивают управление информацией при интеграции ресурсов и информационных систем и создании компьютерных баз данных.

Основные понятия IDEF1X - это типы сущностей, отношения и обобщения.

DFD [19,20] - основными признаками DFD являются потоки данных, процессы, хранилища данных и внешние объекты (терминаторы). Расширения графа потока данных используются для декомпозиции данных и управления в

реальном времени. С этой целью вводятся процессы управления типом, хранение управления, подключение управления и узлы изменения.

Компоненты системы, указанные на диаграмме потока данных, описываются в словаре данных. Для визуализации может быть использована нотация Бэкуса-Наура. Процессы, которые не нуждаются в подробном объяснении с использованием DFD, даются в описании структурированных процессов на естественном языке.

Одна из ключевых особенностей DFD метода заключается в использовании методических правил, которые являются скорее рекомендациями нежели формальными правилами. Это усложняет процесс автоматизации сложных для задач, включая получение информации и утверждение результата будущей работы с заказчиком. Данный метод опирается на моделирование ситуаций. В то же время, к примеру, SADT предназначен для работы и анализа любой существующей системы.

IDEF3 (документирование технологических процессов) относится к так называемым языкам IDEF «следующего поколения», появившимся в ответ на новые потребности в области корпоративного моделирования [21]. Процессы IDEF3 описываются с использованием процессно-ориентированных и объектно-ориентированных стратегий получения знаний. Стратегия, ориентированная на процесс, организует знания о процессах, их времени, случайности и логических взаимосвязях на сцене. Вторая стратегия организует знания о процессах объектов, их состояниях и изменениях в поведении в текущей сцене или нескольких явлениях.

В основе IDEF3 лежит две группы элементов-символов для отображения диаграммы. В первую группу условных символов входят части диаграммы процессов. UOB (unit of behavior – от английского языка «единица поведения») – ключевой элемент в главном сценарии. Также существуют элементы-связки, обеспечивающие логическое соединение единиц процессов между собой во временном аспекте. Вторая группа элементов создана для того, чтобы создать

«диаграммы объектов» в описании предметной области. Части этой диаграммы объектов называются уровнями объектов.

Разработка описания процесса с использованием метода IDEF3 происходит в несколько этапов. На предварительных этапах определяется цель проекта и устанавливаются границы его предметной области. Более того, собирается команда профессионалов, проводится поиск информации и заполнение формуляров. Следующий шаг – это поиск и анализ полученной информации. На третьем этапе формируют полную объектную и технологическую схемы проекта. Затем эксперты постепенно дополняют и уточняют детали проекта, а после идет подведение итогов и проверка готовности проекта, правильности его функционирования.

Основное преимущество данного способа заключается в том, что он использует две связанные между собой диаграммы для проверки корректности проекта и модели в целом. Но есть недостатки - неутонченный характер связи между объектами и другие.

Метод Чена. В вычислительной среде моделирование структур данных наиболее полно реализовано с помощью реляционной модели Кодда. Методологическое обоснование применения реляционной модели было определено в начале 70-х прошлого столетия Питером Ченом и названо методологией «сущность-связь-атрибут» (или в последствии - подходом Чена) [9-14,23,33]. В практике моделирования данных он является доминирующим и в настоящее время.

Модель «сущность-связь» основывается на существенной информации о реальном мире и предназначена для описания на мета-уровне. С ее помощью осуществляется детализация структур данных проектируемой системы, включая идентификацию важных для предметной области объектов (сущностей -Entity), их свойств (атрибутов -Attribute) и отношений с другими объектами (связей-Relationship) [15].

Семантическое описание предметной области осуществляется перед созданием баз данных или информационных систем. В методологии Чена главный

упор делается на семантическое моделирование статической составляющей предметной области, при этом само моделирование осуществляется в визуальной форме – форме специальных диаграмм. Особенностью методологии Чена является отображение семантических элементов и конструкций в синтаксические – реляционные отношения Кодда. Так и сущности, и связи отображаются в реляционные отношения в виде множеств атрибутов-доменов. При этом в методологии отсутствует какой-либо формальный аппарат и формирование сложных семантических структур возлагается на специалистов-аналитиков, их опыт и знания.

Программная поддержка моделей «сущность-связь» осуществлялась в рамках программных комплексов ER Win, IDEF1X [12,13].

В рамках третьего этапа выделяют методы моделирования информации и знаний, связанные с искусственным интеллектом (декларативные, процедурные), онтологическим и объектно-ориентированным подходом [22-44].

Декларативные методы искусственного интеллекта. В данной группе методов различают модели семантических сетей и фреймов. В основе моделирования знаний в виде семантических сетей лежит идея описания необходимой информации в виде бинарных связей между понятиями. При этом отношения могут иметь различную природу [22], обусловленную предметной областью автоматизируемой задачи. В задачах ситуационного управления - временные, пространственные и каузальные (причинные) связи между объектами, в системах планирования- связи типа «цель-средство», «цель-подцель», в системах функционирования - связи «вход-выход», «аргумент-функция». Теория семантических сетей не завершена, ее развитие привело к таким расширениям, как растущие семантические сети, ассоциативные сети и др.

Семантические сети - это тип представления данных, включающий лингвистическую информацию, которая описывает понятия или объекты, дуги и отношения или зависимости между ними [23,24,25]. Дуга обозначается именем отношения, которое она представляет. Несколько дуг могут иметь одно и то же имя. Однако каждое понятие представлено только одним узлом. Дуги отличаются от

указателей доступа. Для каждого узла в сети предполагается, что сеть обеспечивает прямой доступ ко всем отношениям, в котором узел участвует, независимо от направления дуги.

Фреймовая методология [22,26]. Представление знаний в виде фреймов трансформировалось от предложенного Мински минимального описания, содержащего существенную информацию, до гибкой конструкции «имя фрейма - имя слота - значения слота» [22], позволяющей подставить в значение слота любые константы, выражения, ссылки и другие фреймы.

Основанная на «рамочной» конструкции модель представления знаний о домене определяет подвид семантических сетей, который должен отвечать ряду синтаксических ограничений. Концепция «рамки» в искусственном интеллекте претерпела некоторые изменения: первоначально под «рамкой» понималось минимальное описание объекта, содержащего всю необходимую информацию о нем, соответственно без этой части описания невозможен дальнейший анализ объекта, а также невозможно его отделить от ряда себе подобных объектов.

Спустя какое-то время специалисты отказались от подобного требования и в дальнейших исследованиях «рамки» являлись структурами формы. Любой слот представляться в виде формы: «имя слота» и «значение слота». Так же слот может быть фреймом. В таком случае пара слот-значение выступают в качестве не одного слота, а множества. Слоты могут заполнять рамки и переменные, и другие слоты. Можно сделать вывод, что понятие «рамка» может иметь множество применений для отображения информации.

Основные достоинства фреймовой методологии:

1. Гибкость структуры: есть возможность дополнять ее различными элементами (продукциями, вычислимыми слотами).
2. Модульность структуры: используется механизм наследования, то есть черты общих сущностей могут быть заимствованы элементарными сущностями.

К недостаткам фреймовой методологии можно отнести:

1. Сложность отображения структуры из-за большого количества составляющих.

2. Иерархичность связей при отсутствии других видов связей.
3. Примитивные механизмы манипулирования и вывода.
4. Отсутствие в методологии общей терминологии данного метода.

Процедурные методы искусственного интеллекта. К этой группе методов относят логическую модель вывода в виде логических исчислений предикатов и продукций, которые выражают простые взаимодействия, преобразования.

Системы логического вывода основаны на языке Prolog. Prolog - это язык логического программирования, который по сути выступает единицей компьютерной лингвистики и даже искусственного интеллекта [23,27]. От других языков программирования он отличается наличием формальной логики. Пролог основан изначально на логике первого порядка. Это декларативный язык, так как отношения между объектами выражаются логикой программы и выступают в виде терминов, которые обозначают различные факты и установки. Запрос по этим отношениям и есть вычисление.

Этот язык был разработан и внедрен в Марселе, Франция, в 1972 году Аленом Кольмерауэром совместно с Филиппом Русселем на основе процедурной интерпретации положений клаузул Хорна, предложенной Робертом Ковальски в Эдинбургском университете.

Prolog идеально подходит для конкретных задач, которые используют логические запросы, основанные на правилах, таких как поиск по базам данных, системы голосового контроля и шаблоны заполнения. Логическое программирование – это основная методология, по которой предполагалось разрабатывать программы для компьютеров пятого поколения.

Продукционная система (или система продукционных правил) [23,27,28] - это компьютерная программа, используемая для обеспечения определенного типа задач искусственного интеллекта, обычно состоящая из набора правил поведения, но также включает в себя механизм, необходимый для следования этим правилам, поскольку система реагирует на состояния окружения. Эти правила, называемые продукциями, являются базовым представлением, используемым в автоматизированном планировании, экспертных системах и др.

Исторически сложилось так, что основным подходом к описанию тематического пространства является использование продукционных правил - формализмов вида: ЕСЛИ <*условие*>, ТО <*действие*> [81], который устанавливает элементарные этапы преобразований и выводов. Условие определяется как ограничение, используемое для поиска в базе знаний; действие определяет процедуру, которую требуется выполнить при удовлетворении условия.

В продукционной модели система состоит из мысленного объекта. Он содержит в себе: фактологическую базу, ряд продукционных правил и интерпретатор. Началом работы считается постановка задачи, а затем интерпретатор устанавливает детали, факты и правила. Правила состоят из переменных, которые регулируются и дополняются при помощи фактов. Интерпретатор состоит в том, что основная система имеет полное управление над порядком выводов, выполнение которых может быть представлено как последовательность элементарных шагов.

Таким образом, продукционные системы – это метод организации информации «на основе фактов». Такие модели достаточно просты, естественны и, как следствие, доступны для понимания. Недостатки продукционной системы заключаются в следующем: во-первых, стратегия выбора влияет на этапы вывода, во-вторых, присутствует неполнота знаний о предметных областях, что влечет за собой сбои, в-третьих, через некоторое время с возникновением большого количества продукций производительность системы существенно падает, возникают сложности в системе выводов.

Метод онтологий. Онтология состоит из формального наименования [29,30,31,32,42], представления и определений свойств и категорий, связей между понятиями, терминами, сущностями, которые являются частью одной предметной области. Другими словами, онтология – это демонстрация свойств предметной области, при помощи определения связей между составляющими, то есть вычисление категорий и понятий, к которым они относятся.

В каждой из существующих дисциплин есть онтологии, которые ограничивают сложность восприятия информации. Благодаря этому решаются проблемы в разных областях.

В программировании онтология – это способ отображения данных о реальном мире. Онтологии применяют в семантической паутине, в искусственном интеллекте и даже при работе над бизнес-процессами. Языки онтологий могут различаться, однако принцип их построения один. В онтологии входят: атрибуты, экземпляры, отношения и понятия.

Онтология имеет свою специализацию. Например, существуют предметно-ориентированные онтологии, которые представляют реальный мир, либо его часть. В них входит набор определенных терминов, которые характерны для предметной области.

Таким образом, онтологии показывают сущность понятий во многообразии их значений. Поэтому эти онтологии имеют словарь терминов и тезаурус, который описывает значения понятий в каждой из сфер.

Со временем необходимо объединение онтологий, так как иногда системы, в которых применяются онтологии могут развиваться. В связи с этим нужно правильно отображать изменения. Это довольно трудоемкий процесс. Иногда близкие по сути онтологии несовместимы. Так как особенности восприятия в разных культурах разные, большое значение имеет местоположение, а также сложности вызывают разные языки общения и описания.

В некоторых случаях объединить онтологии может только специалист, а в каких-то это можно осуществить частично автоматически. Однако, это долго и дорого. Поэтому, как уже было сказано ранее, проще всего создать единую онтологию и единый глоссарий. Большинство работ, существующих в этой области, имею теоретический характер.

В онтологиях есть свои языки, который используют для кодирования, объединенные общими стандартами. Для создания рабочих процессов с использованием языка онтологий используется несколько методов.

Метод семантического моделирования в объектно-ориентированном подходе. Метод семантического моделирования можно охарактеризовать как решение задачи обработки данных с помощью интеллектуальных объектов, которые работают друг с другом. Также у них есть собственные функции и они запоминают значения друг друга.

В объектно-ориентированном подходе используются определения объектов и отношений между ними [33-44]. Система же описывается при помощи терминов обмена информацией и отношениями между объектами. Наибольшее развитие подобные подходы получили в 90-е годы, где у каждой такой системы была собственная система терминов. Наиболее известны три метода объектного моделирования, предложенные Бучем, Якобсоном и Рэмбо соответственно.

UML (Unified Modeling Language) – базовая нотация для визуализации процесса создания программных систем, которая берет свое начало в 1997 году [35,36]. UML выступает в качестве связующего звена между участниками проекта, так как для всех она понятна в равной степени. Визуально она выражается в диаграммах (графических схемах). Это необходимо, чтобы посмотреть на элементы с разных точек зрения. Всего существует восемь видов диаграмм и в их составе определены специальные элементы.

На сегодняшний день это самый популярный подход к моделированию.

В UML существуют несколько типов диаграмм для анализа:

- диаграмма прецедентов -Use case diagram;
- диаграмма активности -Activity diagram;
- диаграмма последовательностей действий -Sequence diagram;
- диаграммы отношений сущностей - Collaboration diagram.

Диаграмма является альтернативой обычным структурным обозначениям. Но при этом, чтобы описать и провести анализ бизнес-модели, недостаточно описания функций системы таких как прецеденты и акторы. Также к недостаткам можно отнести: -нехватку регламента выполнения функции; -трудности понимания

механизмов и особенностей управления процессами; - трудности в понимании логики действий.

Диаграмма активности – это набор схем регуляции потоков. Потоки могут проходить и параллельно и иметь другие альтернативные варианты. По сути это воплощение двух моделей: IDEF0 и IDEF3. Например, понятие «пересечение» будет означать «линейная синхронизация», либо «выбор», а «работа» обозначать «действие». Данный вариант плохо отражает сложную логику процессов, однако такой подход можно использовать, чтобы упростить представление заказчика.

События, которые происходят благодаря исполнителям, отражаются на диаграмме последовательностей. Это диаграмма в общем виде включает понятия предметной области, а сама система в ней представлена как «черный ящик». Эта удобно, если необходимо отобразить общую схему информационной модели. В таком случае есть входная информация и выходная. Входная информация выражена в терминах IDEF0.

На этапе проектирования используются следующие разновидности диаграмм: диаграмма состояний, диаграмма классов, диаграмма компонентов и другие.

Первостепенная диаграмма UML – диаграмма классов, которую используют для создания кода. При помощи данной диаграммы создается визуальное представление концепции программирования и ориентированного на объекты проектирования.

Достоинство этой системы в том, что она позволяет корректировать проектное решение, не меняя при этом логики решения, так как выгоднее вносить корректировки, не меняя всю структуру заново, как это делается при структурном подходе. То есть перемена классов и отношений не нарушит целостность первичной модели.

На данный момент можно говорить о широком применении онлайн-редакторов диаграмм, которые могут применятся на основе браузера. Таким образом, программное обеспечение начинает выполнять функцию сервиса. Удобство состоит в том, что оно не требует установки на компьютер и может

использоваться бесплатно. Однако, при этом трудно сохранить модель для последующей работы с ней. Таким образом, можно сказать, что для анализа система UML не очень подходит, однако для профессионалов это не будет препятствием.

Метод концептуального моделирования в МАИТ. Развитие автоматизации интеллектуального труда инициировало создание методологии, обеспечивающей промышленный способ проектирования и реализации автоматизированных систем и комплексов [1,2,3,11]. В основе такой методологии лежит последовательное отображение формально-языковых представлений [81]. Адекватность такого отображения обеспечивается наличием сформированного семантического видения или представления, которое позволяет установить смысловое единство при отображении. Формирование такого семантического видения в данной методологии обусловлено методом концептуального моделирования.

Основаниями концептуального моделирования являются: – положения и понятия методологии создания машин по Ю. М. Соломенцеву; – категории и законы диалектики; – структура процесса познания (в формулировке В.И.иЛенина); –результаты анализа философского понятия «производство» [11].

Теоретические положения концептуального моделирования включают фундаментальную и прикладную части. Фундаментальная часть в виде универсального концептуального представления (УКП) рассматривается как совокупность моделей на трех уровнях абстрагирования – абстрактном, объектном и конкретном. Прикладная часть теории рассматривается в виде концептуального представления предметных задач (КППЗ) как совокупность моделей на двух уровнях абстрагирования – объектном и конкретном.

Для УКП определены структура, состав моделей и закономерности формирования для всех составляющих моделей. Для КППЗ также определены структура, состав моделей и ограничения, накладываемые УКП. КППЗ - это модель системы знаний, которая учитывает специфику организации знаний в конкретной предметной области. При этом концептуальная модель на объектном уровне

определяет предметные задачи как методики их решения, в то время как модели конкретного уровня описывают знания в виде множества возможных решений [45, 46].

Каждая из вышеописанных моделей является сочетанием взаимосвязанных конструкций: динамических, статических и связей между ними. Для описания представлений и моделей сформирован специальный математический аппарат.

Статические отношения фиксируются при помощи понятий-категорий, а также бинарных и тернарных связей на них на разных уровнях абстрагирования. Тернарные связи используются для создания сложных элементов, таких как схемы категорий, между которыми существует еще и бинарная связь. Динамические отношения фиксируются на уровне зависимостей первого и второго рода.

Таким образом, система ограничений первого рода накладывается на основную концептуальную структуру, а ограничения второго рода накладываются на производную концептуальную структуру на разных уровнях абстрагирования. Именно так происходит формальное описание связей между конструкциями в рамках концептуальных представлений. Метод концептуального моделирования отличается, в первую очередь, наличием единых закономерностей в процессе формирования концептуальных моделей. Обоснованием такой закономерности являются категории и законы диалектики [1].

Методическое обеспечение концептуального моделирования представлено методикой, в которой рассмотрен процесс моделирования как набор процедур, правил их выполнения, правил оформления результатов моделирования в графической (диаграммы) и табличной (спецификации) формах для всех конструкций концептуальной модели объектного уровня.

Программная поддержка концептуального моделирования разрабатывалась по разным технологиям: полуавтоматизированная в виде системы Concept (2003г.), позволяющей формировать только табличные формы модели и визуализировать результаты по ним; полуавтоматизированная в виде системы Coda (2005), позволяющей формировать и графические, и табличные формы отдельно для каждой конструкции модели; автоматизированная в виде системы ВИС (2008),

позволяющей формировать и графические, и табличные формы в составе интегрированной среды; автоматизированная в виде системы ИС-2 (2018), позволяющей формировать взаимоувязанные графические и табличные формы в составе интегрированной среды.

В табл.1.2 приведены результаты сравнительного анализа методов семантического моделирования [79].

В результате сравнительного анализа можно сделать следующие выводы:

1. Методы 1-го и 2-го этапов автоматизации интеллектуального труда (АИТ) ориентированы на описание статических составляющих семантической модели. При этом методы 3-го и 4-го этапов АИТ – ориентированы на описание и статических, и динамических составляющих семантической модели.

2. В методах 1-го, 2-го и 3-го этапов АИТ механизм формирования составляющих модели – индивидуальный, «рецептурный». В методе 4-го этапа – выявлено наличие закономерностей формирования всех составляющих семантической модели.

3. В методах 1-го и 2-го этапов АИТ отсутствует формальный аппарат описания/интеграции семантических моделей. В методе 3-го этапа – наличие формального аппарата (известного). В методе 4-го этапа – наличие оригинального формального аппарата для описания/интеграции концептуальных представлений, концептуальных моделей, концептуальных конструкций.

4. В методах 1-го, 2-го и 3-го этапов АИТ отсутствует формальный аппарат для обработки/оптимизации самих моделей.

5. Методическое обеспечение для 1-го, 2-го, 3-го этапов – обусловлено индивидуальным опытом и квалификацией специалистов. Методическое обеспечение для 4-го этапа включает унифицированный набор процедур, правил их выполнения, правил оформления и фиксации результатов в графической и табличной форме.

Табл.1.2. Результаты сравнительного анализа методов семантического моделирования

Метод / Критерий	Тезаурусов	ERA-метод	Декларативные модели ИИ	Процедуральные модели ИИ	Метод онтологий	Методы в О-О подходе	Метод КМ в МАИТ
Этап	ЭОД (1)	ИТ (2)	НИТ (3)	НИТ 3)	НИТ (3)	НИТ (3)	КП (4)
Область деятельности – источник	Библиотечная деятельность	Информационные системы – моделирование данных. Системы орга-	Моделирование знаний	Моделирование знаний	Моделирование знаний (=концептуализация знаний)	Проектирование баз данных	Автоматизация проектной деятельности.
Начало разработки	Середина 60г. XX века	Середина 70-х годов XX века	80-е годы XX века.	80-е годы XX века.	00-е годы XXI века	10-е	Середина 80-х годов 20-го века.
Основания семантического моделирования / базовые элементы:	Понятия-элементы, семантические (парадигматические) отношения: род-вид; синонимия; ассоциация.	Entity, Attribute, Relation	для СС – понятия и связи; для фреймов – фрейм, слот, процедура.	Логический вывод; причина-следствие.	Построение онтологии - субъективно; концепты или объекты, их свойства, отношения, обработка	Объекты, отношения, ограничения, отображения	Основания концептуального моделирования-методологические, теоретические, практические. Категории и зависимости разных уровней абстрагирования, типы и виды связей

Продолжение табл.1.2.

Метод / Критерий	Тезаурусов	ERA-метод	Декларатив-ные модели ИИ	Процеду-ральные модели ИИ	Метод онтологий	Методы в О-О подходе	Метод КМ в МАИТ
Этап	ЭОД (1)	ИТ (2)	НИТ (3)	НИТ 3)	НИТ (3)	НИТ (3)	КП (4)
Виды и разнообразие статических отношений	Иерархия понятий. Алфавитный перечень. Классы условной эквивалентн .	Структуры сущностей на разных отношениях	для СС –сеть общих и частных понятий на разных свя-зях; для фрей-мов-структура	Структура предикатов	Структуры на концептах	Структуры объектов	Модели разных представлений и разных уровней абстрагирования. Статические структуры: основные, производные.
Виды и разнообразие динамичес-ких отношений	Нет.	Нет.	Структура процедур / вычислимых фреймов.	Структура логического вывода/ продукций.	Нет	Структуры отображений	Модели разных представлений и разных уровней абстрагирования. Динамические структуры: ограничения 1-го рода (на основные структуры), ограничения 2-го рода (на производные структуры).
Наличие за-кономерност-тей форми-рования	Нет.	Нет.	Нет.	Нет	Нет	Нет	Закономерности формиро-вания всех концептуальных конструкций в УКП

Окончание табл.1.2.

Метод / Критерий	Тезаурусов	ERA-метод	Декларатив-ные модели ИИ	Процеду-ральные модели ИИ	Метод онтологий	Методы в О-О подходе	Метод КМ в МАИТ
Этап	ЭОД (1)	ИТ (2)	НИТ (3)	НИТ 3)	НИТ (3)	НИТ (3)	КП (4)
Наличие формального аппарата для описания отношений	Нет.	Нет.	Теория графов.	Нет	Формальное описание структур	Частично	Оригинальный формальный аппарат для описания и интеграции всех составляющих концептуальных моделей.
Наличие формального аппарата для обработки	Нет. (Запрос как логическая формула).	Нет.	Нет.	Математическая логика.	Частично	Реляционная алгебра	Формальное описание содержательной связности
Методическое обеспечение	Нет.	Индивидуа-льные методики	Индивидуальные методики.	Индивиду-альные методики.	Индивидуаль-ные методики	Индивидуаль-ные методики	Методика концептуального моделирования (унифицированный набор процедур, правил их выполнения, правил оформления и фиксации результатов)
Программная поддержка	Информаци-онно-поисковые системы	ER Win, IDEF X1.	Языки FRL, KRL.	Система Пролог и др.	Имеется	В разработке	а) Concept (2003), б) Coda (2005), в) ВИС (2008), г) ИС-2 (2018).

1.2 Анализ существующих инструментальных средств для семантического моделирования предметных задач, подлежащих автоматизации

1.2.1 Общая характеристика инструментальных средств, используемых при автоматизации информационных и интеллектуальных процессов

Эволюция технологий и инструментов для разработки автоматизированных систем и комплексов обусловлена увеличением объемов данных и информации, используемых в производственной и управленческой деятельности [47,48,49,50,54,58]. Это требует от разработчиков таких систем более тщательно подходить к классификации и систематизации этой информации, к выбору технологии и средств реализации, к организации взаимодействия с пользователями, к размещению программных модулей в вычислительной среда и т.д. Разнообразие требований к автоматизированным системам и комплексам со стороны пользователей, заказчиков, программистов, аналитиков и других специалистов обусловило необходимость создания множества моделей, обеспечивающих согласованную разработку таких систем.

При этом из-за увеличения количества приложений, реализуемых разработчиками, становится сложно выявлять и фиксировать все функции в программной системе, выполнять организацию информации в вычислительной среде, учитывать влияние внешних воздействий на систему и т.д. Появление инструментов различного уровня при создании прикладных автоматизированных систем (ПАС) позволило сократить время для их разработки, реализации и отладки [47].

Развитие возможностей инструментов в дальнейшем позволило: - автоматически создавать программный код, - поддерживать отладку сгенерированного кода по моделям, - постоянно документировать действия, выполняемые в рамках определенного подхода к разработке ПАС, - обеспечивать скоординированное взаимодействие и эффективную работу групп специалистов.

Создание автономных программ повлекло за собой развитие технологий программирования. Как следствие, развивались инструменты автоматизации, осуществлялась их интеграция в комплексы инструментов [53]. Анализ инструментов ПАС проводился на основе следующих критериев:

- этапы создания ПАС (проектирование, тестирование, внедрение, анализ потребностей, анализ способов решения предметных задач и т.д.);
- подходы к разработке ПАС (объектно-ориентированный, когнитивный и структурный);
- разделение по ориентации на составляющие предметной области (информационную или функциональную);
- уровень сложности инструментального средства с учетом такой классификации:

1) отдельные вспомогательные программы, позволяющие решать лишь небольшие задачи по созданию одного модуля/компонента на определенном этапе жизненного цикла АС;

2) пакеты инструментов – это набор программных средств, которые поддерживают создание автоматизированного комплекса на отдельном этапе ЖЦ и используют централизованное хранилище данных на этапе;

3) инструментальный «верстак» – осуществляющий интеграцию функций разработки, системного хранилища, автономного функционирования в определенной программно-технической среде;

4) инструментальные системы (CASE-системы или «технологические линии») – это инструментальный комплекс, который реализован на основе общего хранилища данных, реализует выполнение последовательности этапов создании АС и обеспечивает управление разработкой АС;

- характерные черты инструментальной среды: наличие или отсутствие централизованной системы хранения данных и ее вид, количество поддерживаемых пользователей, возможность сопряжения с другими средствами, используемые методы и модели.

В качестве средств для анализа были выбраны наиболее известные, такие как CA BPwin; CA ERwin Data Modeling Suite; WorkFlow Analyzer; Silverrun; IBM Rational Software Architect; CASE.Аналитик; Power Designer; Oracle Designer; Visio; инструментальные средства МАИТ.

Практически каждый из инструментов имеет дополнительные модули, обеспечивающие коллективную работу с общим репозиторием.

1.2.2 Сравнительный анализ инструментальных средств, используемых в рамках методологии автоматизации интеллектуального труда

CA BPwin - это программный продукт, с помощью которого можно осуществлять моделирование, анализ, описание и последующую оптимизацию бизнес-процессов [48,50,54]. Удобен для применения внутри больших компаний, так как объединяет базы данных предприятия в единую информационную систему, отвечающую потребностям современного предприятия. Данный инструмент можно применять также и для ведения проектов.

BPwin основан на известных технологиях моделирования, таких как IDEF0, а также использует объединенные модели, содержащие все три поддерживаемых типа диаграмм. Широкая известность BPwin (AllFusion Process Modeler 7) помогает сопоставлять функциональные модели в электронном виде.

CA ERwin Data Modeler - это инструмент моделирования данных и проектирования баз данных, используемый для построения концептуальных, логических и физических моделей данных. Erwin может создавать реальные базы данных из физических моделей и создавать различные физические вставки из одних и тех же логических моделей. Модель данных может преобразовать существующую базу данных в график. Он подходит для большинства систем управления базами данных (СУБД). Выход инструмента включает ER-диаграмму, стандартные или пользовательские отчеты (таблицы, поля, отношения).

Он основан на методе IDEF1X и может отображать диаграммы с помощью вариантной нотации проектирования информационных технологий, а также нотации размерного моделирования.

Базы данных и приложения проектируются при помощи веб-версии ERwin ModelMart. ERwin/OPEN работает с инструментами улучшения приложений PowerBuilder, Delphi и SQLWindows и позволяет перенести схему разработанной базы данных непосредственно в информационное хранилище инструментов. Для ряда инструментов разработки программного обеспечения (PowerBuilder, SQLWindows, Delphi, Visual simple) создаются формы и прототипы приложений.

WorkFlow Analyzer компании Meta Software [50,51] - это продукт для анализа бизнес-процессов, который создает модели систем на базе технологии IDEF0, анализирует их производительность и моделирует работу программной системы в разных средах: UNIX, MS Windows environment. Данный пакет включает в себя ряд программных модулей. Например, Workflow Modeler помогает осуществить построение графика утилитарной структуры Фреймворка, при использовании систем IDEF0 и методологии IDEF1X. Полученная модель может быть проверена, благодаря модулям данного пакета Workflow Simulator.

В данном пакете можно визуально представить всю схему проекта в виде диаграмм, которые отображают все этапы работы над проектом, их составляющие и задачи, а далее описывает каждую задачу, образуя единое «древо» проекта. Чаще всего методы, которые применяют в диаграммах IDEF0, имеют связь с общей структурой построения структуры данных IDEF1X.

Готовая модель учитывает затраты на выполнение работ (ABC-анализ). Здесь можно использовать метод имитационного моделирования и построить модели с готовыми параметрами в виде видеоролика. Это позволяет заранее выявить слабые места и оценить трудозатраты и уровень загрузки исполнителей.

Система **Silverrun, Computer systems Advisors, Inc. (CSA)**. Данное инструментальное средство ориентировано на структурный подход и его методы [55,56,57]. Система предназначена для формирования информационных и функциональных составляющих при исследовании и разработке информационных

систем корпоративного уровня. Система дополнительно фокусируется на спиральной модели жизненного цикла продукта с учетом ограничений. Так как спиральная модель влечет за собой увеличение степени доступности создаваемого продукта в зависимости от времени, затраченного на разработку технического задания, все уровни моделирования, наряду с анализом и проектированием, повторяются при разработке следующей версии продукта.

Система Silverrun применима тогда, когда методологии разработки учитывают связи между информационными и функциональными моделями. С помощью графической нотации и набора правил для проверки можно настраиваться на конкретную задачу. Пакет имеет модульную структуру, куда входит три независимых модуля, без учета управления данными в разных моделях. Silverrun состоит из:

1.Business Process Modeler – модуль потоков данных внутри моделей бизнес-процессов [58]. Результат его работы оформляется в виде диаграммы. Данный модуль настраивает согласование внутри многофункциональных моделей. Диаграммы могут иметь две формы представления: по методу Jordan-De Marco и методу Gein-Sarson. Самый эффективный метод – это метод Datarun, который за небольшое время позволяет создать полноценную программу.

2.Entity-Relationship eXpert - модуль, используемый для разработки концептуальной структуры базы данных. Модуль позволяет создавать структуры данных, описывающие ресурсы и их связи внутри предприятия, которые не всегда связаны с текущей реализацией. Данный модуль включает в себя подход, позволяющий построить структуру данных на основе нормализованной модели. Вместе с этим появляется возможность из заданной структуры данных автоматически выстроить реляционную модель.

3. Data Modeler – модуль реляционного моделирования, выстраивает модели типа «сущность-отношения», которые могут существовать в реляционной базе данных. Данный модуль предназначен для организации баз данных с помощью учета множества ограничений (триггеров и индексов, и других процессов в том числе). Таким образом, имитируются точки обработки и профили пользователей.

Relational Data Modeler собирает весь комплект документации, которая описывает реляционную базу данных.

4.Workgroup Repository Manager – средства управления репозиторием и словарями данных, которые применяются для хранения стандартной информации обо всех моделях. Более того он обеспечивает интеграцию модулей Silverrun в одну систему.

Цена высокой гибкости и разнообразия инструментов визуального моделирования заключается в том, что Silverrun больше не имеет строгого взаимного управления компонентами различных моделей (например, автоматическое распространение изменений между диаграммами DFD различных уровней декомпозиции). Как следствие, при постоянном совершенствовании системы на основе спирального проектирования могут быть получены несогласованные модели. Таким образом, лучше использовать Silverrun при выполнении предметных задачи и создавать информационно-поисковые модули в хорошо формализованных предметных областях.

Система **Rational Rose** фирмы IBM Rational Software Architect является средством разработки на базе объектно-ориентированного подхода для двух этапов: проектирования и анализа [59,60,61]. Согласно исследованиям независимой Международной корпорации данных (IDC) эта система является одним из лучших инструментов объектно-ориентированного анализа.

UML (Unified Modeling Language) [61] – актуальный язык программирования, применяющийся в данной системе, используется для построения архитектуры программ. С помощью данного языка программисты могут не только определять данные, но и визуализировать их, делая видимыми, и описывать централизованно всю систему. Данный язык позволил разработчикам работать в единой среде.

Таким образом, концепция UML Rational стала стандартом и применятся в Object Management Group (OMG), которая возникла как следствие общего соглашения между разработчиками, учеными и пользователями.

Так как Rational Rose позволила создавать свои интерфейсы на основе готовых моделей, компонентов макета, эта система стала достаточно

распространенной в управлении бизнес-процессами. На основе общих взаимосвязей между компонентами модели стало возможным автоматически перемещать данные из информационных документов в готовые диаграммы объектов.

Единая система помогла упростить процесс работы с одновременным увеличением коммерческой ценности. Данную систему легко оптимизировать под собственные разработки, поэтому стало возможным приобретение отдельных компонентов и обособленное их применение.

Стоит отметить, что большие программные проекты создаются с помощью множества программных языков, но это не является препятствием к применению Rational Rose, так как система многоязычна.

Дополнительным преимуществом программной системы будет то, что она работает в условиях разных операционных сред (Windows и UNIX). Одно из основных преимуществ заключается в том, что был упрощен переход от разработки к анализу и наоборот, платформа связала оба процесса и вывела на качественно новый уровень.

Для каждой задачи формируется свой список требований и регламентов, которые учитываются на всех этапах разработки.

Такая система ознаменовала итеративный подход к разработке, автоматизировала процесс создания, анализа, соединения элементов друг с другом и фиксации полученных данных в ходе данных процессов, а также обеспечила автоматическое включение данных в макет.

CASE Аналитик (CASE-Analyst) –это российская разработка, которая вышла на уровень рынка [62]. Данная система стала использоваться для оптимизации работы предприятий. В основе CASE Аналитик лежит метод структурного анализа Гейн-Сарсона. Данная система обладает гибкостью и может использоваться с разнообразными типами данных. Используется для создания как информационно-вычислительных систем, организационных системы, так и систем автоматизации научных экспериментов. В результате применения формируется информационно-логическая модель создаваемой программной системы, которую

можно анализировать. Модель состоит из множества диаграмм, где каждая диаграмма имеет свой уровень значимости. Самым важным приоритетом обладают диаграммы, которые описывают главные функции системы с внешними входами и выходами, также они отображают файлы, необходимые для работы. Главные функции обуславливают набор диаграмм более низкого уровня декомпозиции. Данное разложение будет выполняться до тех пор, пока рабочий процесс не достигнет требуемой степени детализации. Этот подход хорошо работает в условиях небольших предметных задач, подлежащих автоматизации. Так как чем больше прикладная задача, тем больше разнообразных ресурсов она использует [47,62].

Детализировать логические функции системы не всегда рационально, поэтому на первое место выходит внутренняя логика с применением мини-спецификаций, которые описывают преобразование входных потоков данных в выходные потоки данных. Содержание каждой модели изображается уже не с помощью визуальных средств, а с помощью текстовой информации. Выполняется подготовка информационно-логической модели. Кроме того важно определить логическую функциональную спецификацию, которая показывает дополнительные возможности, используя при этом неформализованный язык.

Пакет состоит из:

1) База данных проекта включает контекстные диаграммы, диаграммы потоков данных, диаграммы управляющих потоков, структуры данных, описания логики процессов, спецификации элементов данных, сигналов и структурных объектов и так далее.

2) Графических редакторов потоковых диаграмм и структурограмм данных. Таким образом, каждый шаг отображается визуально при работе с диаграммами. Важно отметить, что помимо проверки правильности видимого изображения данных проходит проверка корректной работы всех взаимосвязанных частей проекта. При необходимости весь процесс изменения можно сохранить.

3) Набора инструментов, необходимых для отображения полученной информации как на экране, так и в печатной форме. Это диаграммы потоков

данных, контекстная диаграмма, перечень объектов словаря, отобранных по различным критериям. А также спецификации минилогики работы, а также отчеты и протоколы по проекту.

4) Документатора – набора документов по проекту, который формируется согласно определенным стандартам (ГОСТы на автоматизированные системы).

5) Верификатора, который представляет собой систему верификации проекта, которой обычно занимается аналитик проекта.

Основное преимущество данной инструментальной системы в том, что в ней удобно хранить большое количество документов и разного рода отчетов, так как она обеспечивает простую навигацию по схемам внутри информационной базы. В данном контексте важна разработка программного продукта, чтобы российский производитель вышел на новый уровень автоматизации управления предприятием. Однако, есть ограничения для применения данной системы в сфере машиностроения. Более эффективно данная система может применяться в сфере банковского дела.

Power Designer (ранее S-Designer) продукт Sybase, который специально создавался для работы с информационными системами [63,64]. Power Designer состоит из набора модулей, соединенных между собой общей системой разработки MetaWorks. Данный продукт создает диаграммы потоков данных для различных прикладных систем, физические и концептуальные модели данных. Также в рамках данного продукта имеется возможность создать информационную модель, которая состоит из уже существующих баз данных, а также создавать программные продукты современных средств разработки. Применяется в операционных средах UNIX, MS Windows. Данная инструментальная система функционирует на основе моделирования данных, структур предприятия и архитектуры предприятия. Power Designer состоит из шести модулей, а также из специального средства для моделирования и создания баз данных – DataArchitect. Также есть WarehouseArchitect для хранения данных и ProcessAnalys, которое позволяет смоделировать нужным образом потоки данных. Для моделирования на физическом уровне используется AppModeler.

MetaWorks - модуль обработки и работы с моделями, а модуль Viewer помогает настроить просмотр информации без возможности ее редактирования. Результатом работы с данной системой является готовые структуры, которые могут быть аналитически обработаны и данные о которых хранятся в специальном хранилище, Данная система обеспечивает хранение полного пакета документов по проекту.

Oracle Designer – программное обеспечение для выполнения анализа бизнес-процессов, формирования требований и генерации клиент-серверных систем, которые подходят под данные требования [65,66,67]. Дополнительно данный инструмент поддерживает разработку программного обеспечения и создание прикладных систем. Oracle Designer удобен тем, что имеет обширное хранилище данных с возможностью одновременного использования несколькими пользователями и имеет интеграцию с Developer/2000. С помощью Oracle Designer у компаний появляется возможность создавать обширные клиент-серверные системы, которые будут отвечать их индивидуальным требованиям.

Oracle Designer (ранее Designer/2000) – часть интегрированных средств разработки Oracle9i Developer Suite. Oracle Designer – это репозиторная CASE-система [68], предназначенная для бизнес-анализа, генерации кода программ и создания проектов. Oracle Designer/2000 предоставлял графический интерфейс для проектирования и моделирования баз данных и позволял разработчикам создавать и изменять модели данных, бизнес-правила и пользовательские интерфейсы. В данной системе применяются инструменты Oracle9i Forms и Oracle9i Reports, которые преобразовывают компоненты системы в объекты базы данных, отчеты и экранные формы. Финальный этап разработки проекта – это автоматизированное создание серверных компонентов, которое осуществляется не только для сервера баз данных Oracle, но и на других, например, DB/2, Microsoft SQL Server и других.

Тем не менее, возможности системы Oracle с ее инструментарием могут быть использованы в полной мере только в ее среде. Так, например, при условии наличия спецификаций для форм Oracle, выполненных ранее, можно сделать приложение,

которое будет функционировать в многослойной архитектуре с *Oracle Forms Services*.

CASE-система Oracle использует свою методологию проектирования и разделяет весь цикл работы системы на этапы, ориентируясь на разработку прикладной системы. В основе системы лежит единая база данных, включающая спецификации проектов на всех этапах разработки и спецификации необходимые для слаженной работы разработчиков по каждому проекту. Следует отметить удобство визуальной составляющей системы – графического интерфейса, который делает работу быстрее и проще. Oracle Designer может интегрироваться с методологиями UML и IDEF, позволяя моделировать бизнес-логику, данные и т.д.

Подводя итог, можно отметить, что CASE-инструменты крайне важны в начале разработки масштабных программных продуктов. Анализ рынка показывает, что CASE-система необходима и на этапе внедрения и сопровождения прикладных автоматизированных систем. Проект не может быть осуществлен без формализованной модели автоматизированного объекта. При автоматизации разработчики могут столкнуться с рядом трудностей, которые прежде всего связаны с материальной стороной работы над проектом. При постоянном обновлении инструментов, появлении новых инструментов растет риск непредусмотренных расходов, в том числе и на подготовку сотрудников к работе в новых условиях, постоянному обучению и развитию.

Среди наиболее важных проблем выделяются следующие:

- проблемы, возникающие при работе с CASE-системами, не поддаются огласке, однако можно привести несколько из них для примера: много усилий и средств, потраченных на интеграцию CASE может привезти к снижению производительности, и, как следствие, руководство предприятия перестанет поддерживать финансирование проектов, которые приносят ощутимый перерасход средств и трудовых ресурсов;
- в компании работают с уже сложившимися методами и средствами работы с проектами, и новая система не всегда будет с ними коррелировать;

- внутри самой системы инструменты и средства CASE не всегда совместимы друг с другом, одновременно это усложняет передачу информации между инструментами;
- некоторые CASE-системы требуют слишком больших усилий для их использования.

Таким образом, рассматриваемые CASE-инструменты наиболее полезны в областях, где доступна централизованная информация, преобладают процедуры поиска информации и простая статистическая обработка хранимых данных.

Visio - графический редактор, позволяющий создавать диаграммы из готовых элементов или отдельных элементов [69,70]. Этот продукт производится компанией Visio Corporation в текущей версии 5.0.

Существует ряд шаблонов, который объединяет разнообразные элементы диаграмм. Все эти элементы могут быть использованы для создания следующих диаграмм: диаграмм Бахмана, диаграмм статической структуры UML, диаграмм элементов интерфейса, сущностных отношений на основе нотаций Crow's Foot и IDEF1X. Тем не менее есть возможность создания новых шаблонов. Для программирования свойств элементов используется язык Си либо Visual Basic. С помощью драйверов ODBC происходит соединение диаграмм с базами данных, а также соединение с репозиторием Microsoft Windows.

В итоге получается схема реального объекта, созданного при помощи определенного набора документов, а также структура, в которой элементы расположены в соответствии с определенной иерархией, по сути это разложение модели на составные части.

Инструментальные средства МАИТ. На данный момент инструменты методологии автоматизированной интеллектуальной деятельности предоставляют возможность сделать автоматизированным процесс создания моделей на разных этапах разработки прикладных автоматизированных систем [4,50,71].

Методология автоматизации интеллектуального труда (МАИТ) является российской разработкой (МГТУ «СТАНКИН») и представляет собой процесс автоматизации [47,80], обеспечивающий промышленный способ создания

прикладных систем и комплексов. Процесс создания прикладных систем включает формирование и обработку следующих модельных представлений: -начального, -концептуального, -инфологического и даталогических представлений.

Статические, динамические и функциональные компоненты моделей фиксируются с помощью диаграмм и спецификаций, а также их взаимозависимостей. МАИТ позволяет моделировать разные аспекты задачи и постепенно собирать их в единый комплекс, который включает в себя компоненты модели [1,2,10,36,53].

На каждом из этапов создания использовалось свое средство поддержки (Model, Concept, Infology). Существует две основные технологии моделирования, которые основаны на способах создания моделей: полуавтоматизированная и автоматизированная. Первый способ основан на подготовке «ручных» диаграмм с последующим автоматизированным созданием спецификаций и формированием по ним контрольных диаграмм. Во втором методе используется графический редактор, который поддерживает формирование диаграмм и автоматически заполняет спецификации. Программный пакет «CODA» является графическим редактором, который может создавать графические и табличные описания, но только для концептуальных представлений.

В настоящее время разработана и эксплуатируется программная среда «ИС-2». Графический интерфейс и ядро программного обеспечения отделены от программного комплекса. При объединении средств поддержки процедур в рамках единой интегрированный среды необходимо учитывать особенности процесса создания информационных и автоматизированных систем и комплексов на основе МАИТ.

Моделирование на любом этапе выполняется на двух уровня сложности: комплексов предметных задач (КПЗ), и моделей отдельных предметных задач (ПЗ). Вначале эксперт формирует первичную обобщенную модель под КПЗ, в котором определяет взаимную компоновку отдельных задач, каждая задача в такой модели представлена в виде элементарного элемента структуры [5,46]. Локальные модели

разрабатываются системными аналитиками независимо друг от друга в параллельном, последовательном или последовательно-параллельном режиме.

Каждая локальная модель должна пройти процедуру анализа, в которой необходимо участие эксперта, а также процедуры аналитической обработки и документирования. По завершении разработки локальная модель встраивается (интегрируется) в обобщенную. Результатом разработки модели для комплекса предметных задач является единая обобщенная модель, включающая локальные модели всех задач комплекса.

При разработке программной среды, обеспечивающей интеграцию средств поддержки процедур МАИТ, главной задачей является обеспечение информационных потоков между средствами, которые могут функционировать одновременно на различных ЭВМ. Также необходимо учитывать, что одна процедура моделирования может иметь несколько средств поддержки в зависимости от конфигурации самой интегрированной среды.

Созданная на кафедре информационных технологий и вычислительных систем ФГБОУ ВО «МГТУ «СТАНКИН» реализация - программный комплекс «ИС-2» - строится по следующему принципу: серверная часть (сервер приложений и база данных) обеспечивает управление функционированием и информационным взаимодействием между отдельными клиентами. Клиентское приложение выступает в качестве «толстого» клиента и решает задачи формирования локальных и первичных обобщенных моделей, а в качестве «тонкого» – для решения задач управления и обработки обобщенной модели.

Каждое отдельное рабочее место может быть настроено на решения определенного перечня процедур в выбранных этапах. Пока еще не существует программных модулей для автоматической передачи результатов с одного этапа на другой, поскольку каждый этап поддерживается отдельным инструментом, который нельзя комбинировать с другими этапами, что ограничивает полезность разработки модели последовательности при окончательном проектировании автономной программной системы.

В результате анализа инструментальных средств можно сделать два основных вывода. Во-первых, существует ряд качественных программных систем таких, как Oracle Designer или Silverrun. Они в свою очередь имеют возможность полностью от начала до конца настроить пакеты данных. Но не все предприятия могут себе позволить такие большие трудозатраты и материальные расходы в том числе, не всегда такой подход себя оправдывает [50].

Во-вторых, есть более дешевые и простые по технологии пакеты ERwin, BPwin, Power Designer, однако при этом они работаю местно, то есть отдельно с каждым уровнем или элементом модели, что не позволяет наладить целостное управление всей структурой.

Общим для всех систем является то, что они не предоставляют возможность представления модели вне программного обеспечения. Более того и в одних и в других системах слабая привязка к программному и аппаратному обеспечению. Но при этом, если в программах и появится возможность индивидуальной настройки уровень издержек повысится – такое ПО вырастет в цене и потребует отдельного обучения сотрудников.

В целом, все вышесказанное представлено в табл.1.3 [50]. Каждая из систем проанализирована по нескольким критериям: универсальности, уровню сложности, переносимости, сопряжения, реализации, результатов и стоимости.

1.3 Особенности концептуального моделирования в рамках методологии автоматизации интеллектуального труда

Практическая разработка ПАС во многом опирается на теоретические основы методологии автоматизации интеллектуального труда (МАИТ) [1,2,4,5,6,72,73,80].

Как было уже отмечено выше, МАИТ обеспечивает промышленную разработку ПАС в машиностроении (рис.1.1). Следует отметить главные преимущества МАИТ:

Табл.1.3. Сравнение инструментальных средств

Наименование продукта	Назначение	Используемая методология	Переносимость	Связь с другими средствами	Сетевая реализация системы	Результат работы системы	Стоимость
Bpwin	Функциональное моделирование	IDEF0, DFD, IDEF3	Неполная	ERWin	Есть с ограничениями	Комплект документов, стоимостные оценки	$2500
Erwin	Концептуальное моделирование и реинжиниринг БД	IDEF1x, IE	Неполная	СУБД: Oracle, Informix, MS SQL Server, FoxPro, Ingress, Clipper, dBASE, MS Access, Paradox и др. Ср-ва разработки приложений: SQLWindows и др. Delphi	Есть	Информационная модель ПАС, таблицы, индексы и другие объекты базы данных	$250-$1000
WorkFlow Analyzer	Моделирование и симуляция деловых процессов (бизнес-анализ)	IDEF0, IDEF1x	Между средами полная, между версиями – в одном направлении	WorkFlow Simulator	Есть	Комплект документов, стоимостные оценки.Структуры данных, результаты симуляции	$35000
Silverrun	Полный цикл разработки ПО	DataRun + настройки пользователя	Неполная, имеются различия в версиях с разной целевой платформой	JAM, PowerBuilder, SQL Windows, Uniface, Delphi	Есть	Структуры данных, описание логики работы, базовые функции	$3000

Продолжение табл.1.3

Наименование продукта	Назначение	Используемая методология	Переносимость	Связь с другими средствами	Сетевая реализация системы	Результат работы системы	Стоимость
Rational Rose	Анализ, проектирование и создание ПО	UML	Неполная	На уровне текста программ: C++, Smalltalk, Ada, PowerBuilder, SQLWindows, и др.	Есть	Комплект документов. модель ПАС	$1200-$1700
CASE-Аналитик	Полный цикл разработки ПО	Набор нотаций UML для всего ЖЦ ПО	Неполная, имеются различия в версиях с разной целевой платформой	На уровне текста программ	Есть с ограничениями	Программный продукт	$500
Power Designer	Концептуальное моделирование БД	IDEF1x, IE	Неполная	PowerBuilder, Visual Basis, Delphi, Power++	Есть	Структуры данных ИС, заготовки форм и приложений, описание логики работы, базовые функции	$995-$6495
Oracle Designer	Полный цикл разработки ПО(без кодирования)	Oracle CASE Method. Набор стандартных нотаций для всего ЖЦ ПО	Неполная, имеются различия в версиях с разной целевой платформой	Developer 2000	Есть	Структуры данных, описание логики работы, базовые функции	$3000

Окончание табл.1.3

Наименование продукта	Назначение	Используемая методология	Переносимость	Связь с другими средствами	Сетевая реализация системы	Результат работы системы	Стоимость
Visio	Документирование	IDEF1x , Bachman, UML	Между версиями - в одном направлении	Нет	Нет	Комплект документов, макет данных	$150-$500
SoDA	Автоматизированное документирование	-	Между версиями - в одном направлении	Нет	Есть с ограничениями	Комплект документов, макет данных	$500-$5000
ИС-2	Автоматизированное моделирование начальных и проектных процессов	МАИТ	Между версиями - в одном направлении	Нет	Есть с ограничениями	Комплект документов начальной и проектных моделей автоматизированных систем	$150-$1000

1. Методология обеспечивает последовательное отображение моделей на трех уровнях абстрагирования.

2. Наличие модельных представлений, как независимых от программно-технических средств реализации ПАС, так и ориентированных на них.

3. Соответствие формально-языковых представлений обеспечивается единым семантическим описанием, воплощенным в концептуальной модели.

4. Возможность интеграции моделей разного вида и их аналитической обработки.

Рис.1.1. Процесс создания ПАС по МАИТ [2,72,80]

Концептуальное моделирование базируется на двух представлениях [1,4]:

1. Универсальное концептуальное представление определяет общую схему моделирования знаний на трех уровнях абстрагирования и является фундаментальной частью теории моделирования знаний.

2. Концептуальное представление предметных задач – определяет строение системы знаний отдельно для каждой из предметных областей на двух уровнях абстрагирования и является прикладной частью теории моделирования знаний.

Концептуальное моделирование предметных задач методически состоит из нескольких процедур:

- создание первичной обобщенной модели для комплекса задач (в соответствии с предметной областью);

- формирование модели отдельной предметной задачи из комплекса;

- анализ текущей модели;

- аналитическая обработка концептуальной модели;

-синтез обобщенной и текущей моделей;

-документирование модели (обобщенной и/или текущей).

Концептуальное представление прикладной задачи чрезвычайно важно для автоматизации, так как оно обеспечивает целостное содержательное представление предметной области для будущего автоматического решения этой задачи.

Концептуальное представление прикладной задачи включает одну модель объектного уровня, множество моделей конкретного уровня и их увязку.

Создание обобщенной и текущей концептуальных моделей на объектном уровне подразумевает формирование понятийных структур, систем предметных зависимостей и модели в целом. Методически это оформляется в виде специальных диаграмм и спецификаций.

Концептуальная модель объектного уровня есть совокупность конструкций: статических, динамических и их увязки. Статические отношения представлены элементами – предметными понятиями или категориями с бинарными и тернарными связями. Системы зависимостей первого рода выражают динамические отношения.

Увязка статических и динамических конструкций формирует концептуальную модель в целом.

Метод концептуального моделирования отличается от существующих методов семантического моделирования тем, что основан на проявлении законов диалектики, которое отражено в закономерностях формирования концептуальных моделей в универсальном представлении.

1.4 Выводы по главе 1

1. Основываясь на анализе семантических методов, используемых при разработке систем автоматизации приложений, было установлено, что эти методы затруднительно использовать при создании приложений для машиностроения. Кроме того, в CASE-средствах удобно, информативно и доступно используются визуальные средства для представления информации в этих методах, но они не имеют формального описания различных моделей и их обработки.

2. Анализ инструментальных средств для создания автоматизированных систем показывает, что существующие инструменты часто разрабатываются и обеспечивают работу только с одним из существующих методов. Наличие настраиваемого программного обеспечения затрудняет его разработку и использование и увеличивает его стоимость.

3. Анализ существующих методов, подходов и инструментальных средств их поддержки позволил установить, что по сравнению с методами других подходов, методология автоматизации интеллектуального труда имеет ряд преимуществ: полный набор компонентов модели и их увязка в модельном представлении; наличие закономерностей организации семантических модельных представлений в УКП; формирование КППЗ, как основы для единой интерпретации последующих формально-языковых представлений при создании прикладных автоматизированных систем.

4. Проблемы, выявленные по результатам анализа, показали, что при проектировании и реализации ПАС следует использовать не только методы моделирования знаний, но требуются способы и средства поддержки обработки концептуальных представлений предметных задач.

ГЛАВА 2. УТОЧНЕНИЕ ФОРМАЛЬНОГО ОПИСАНИЯ КОНЦЕПТУАЛЬНЫХ МОДЕЛЕЙ И РАЗРАБОТКА ФОРМАЛЬНОГО АППАРАТА ДЛЯ ПРОЦЕДУР ИХ ОБРАБОТКИ

2.1 Исследование особенностей процесса концептуального моделирования в рамках методологии автоматизации интеллектуального труда

Одной из актуальных задач, стоящих перед промышленными предприятиями, является задача сохранения их интеллектуального потенциала.

Интеллектуальный ресурс любой организации или предприятия включает как опыт и знания их специалистов и руководителей, так и разнообразные фонды, хранящиеся как в документной, так и в электронной форме в виде архивов технической документации, регламентов, стандартов, патентов, методик, типовых решений и др [79].

Для переноса этого ресурса в компьютерную среду необходимо выполнить моделирование всех его составляющих в формализованном виде.

Методологией, которая обеспечивает такой перенос на основе формального моделирования знаний (или концептуальное моделирование) является методология автоматизации интеллектуального труда, разработанная на кафедре ИТиВС МГТУ «СТАНКИН».

К особенностям процесса концептуального моделирования следует отнести:

- наличие теоретического аппарата для моделирования знаний в виде концептуальных представлений (универсального и предметного);
- наличие моделей разных уровней абстрагирования в концептуальных представлениях (три – в универсальном, две – в предметном);
- представления системы знаний в виде концептуальной модели как набора взаимоувязанных конструкций -статических, динамических и их увязки, как в универсальном, так и в предметном представлениях;

- наличие методического обеспечения концептуального моделирования предметных задач для процедуры формирования моделей в виде правил выполнения и средств представления результатов моделирования в графической и табличной формах, т.е. фиксации моделей, отражающих привычный для предметного специалиста процесс принятия решений;
- отсутствие методик и средств обработки концептуальных моделей и конструкций (оптимизации, расслоения, классификации).

Эти особенности определили необходимость дальнейшего развития теоретического аппарата и методического обеспечения.

2.2 Уточнение формального описания концептуальных представлений предметных задач

2.2.1 Общее описание

Аналитическое описание системы в общем виде может быть рассмотрено следующим образом [83,84]:

$$S_1 =< Y, X, q(Y), q(X), R_1 >, \quad (2.1)$$

где множество Y - описывает структуру процесса, выполняемого системой; множество X - описывает входные/выходные объекты, используемые в этих процессах; $q(Y)$ - множество разнородных свойств элементов структуры процесса; $q(X)$ - множество разнородных свойств входных/выходных объектов; R_1 - множество связей между перечисленными компонентами.

В представлениях систем могут быть использованы различные частные случаи их описания. Так, например, можно описывать или только процессы с их структурой, или только объекты с их структурой и т.д.

В концептуальном моделировании модель знаний можно рассматривать как систему, включающую различные объекты и их связи.

Большее число составляющих компонентов системы обуславливает необходимость разделения в определенных условиях видов элементов, их связей и т.д.

Так, объекты могут образовывать сложную структуру, построенную на различных связях- бинарных, тернарных.

Формально это можно описать в общем виде следующим образом [83,84]:

1) множество объектов представляет собой множество непересекающихся подмножеств, при этом элементы этих подмножеств имеют двойной индекс:

$$X = \bigcup_i X_i \,, X_i \cap X_j = \emptyset, X_i = \{x_{il}\}, \tag{2.2}$$

2) тогда множество бинарных связей на этих объектах является подмножеством декартового произведения (×) их двух множеств:

$$T \subset X \times X, \tag{2.3}$$

3) в силу разделения исходного множества объектов на подмножества получаем разделение бинарных связей на подмножества:

$$T = \{LX_i, LX_iX_j, LX_j/X_i\}, \tag{2.4}$$

4) тогда множество тернарных связей на этих объектах является подмножеством декартового произведения (×) их трех множеств:

$$H \subset X \times X \times X \tag{2.5}$$

5) в силу разделения исходного множества объектов на подмножества получаем также разделение тернарных связей на подмножества:

$$H = \{HX_iX_jX_k\}. \tag{2.6}$$

Прикладная часть концептуального моделирования предметных задач связана с идентификацией и выделением самой задачи из предметной области и последующим формированием моделей под задачу.

Поскольку концептуальное представление предметных задач определяет совокупность моделей, то формально это можно описать [3,74,78,79]:

$$KP(n) =< KP2(n), \overline{KP3(n)}, R_{23}^{KP}(n) >, \tag{2.7}$$

где $KP2(n)$ - модель объектного уровня (2) для выделенной (n-ой) предметной задачи, $\overline{KP3(n)} = \{KP3(nm)\}$ - совокупность моделей конкретного уровня для той же задачи. $R_{23}^{KP}(n)$ - отношения, описывающие взаимосвязи концептуальных моделей разных уровней абстрагирования для n-й предметной задачи.

При этом модель каждого уровня абстрагирования включает подмодели соответствующего ***i***-го уровня и ***j***-го рода $KPi.j\ (z)$,где для $i=2$, $z=n$, а для $i=3$, $z=nm$.

$$KPi(z) =< KPi.1(z), KPi.2(z), R_i^{KP}(z) >, \qquad (2.8)$$

Уточненная структура представления с учетом подмоделей приведена на рис.2.1.

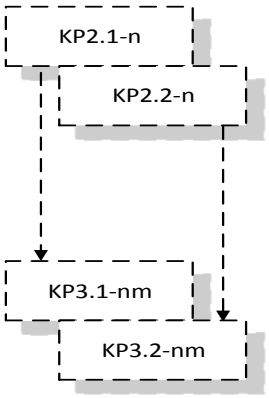


Рис. 2.1. Уточненная структура концептуального представления предметной задачи с учетом подмоделей [74,78,79]

Уточнение структуры концептуального представления 1-го рода приведена на рис. 2.2.

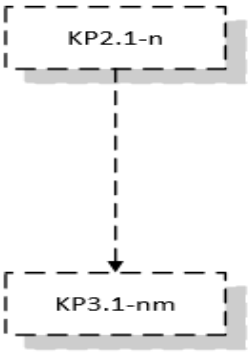


Рис.2.2. Уточнение структуры представления в общем виде [79]

$$KP.1(n) =< KP2.1(n), \overline{KP3.1(n)}, R_{23}^{KP1}(n) >, \qquad (2.9)$$

где $KP2.1(n)$ - модель 1-го рода объектного уровня (2) для выделенной (n-ой) предметной задачи, $\overline{KP3.1(n)} = \{KP3.1(nm)\}$ - совокупность моделей 1-го рода конкретного уровня (3) для той же задачи.

2.2.2 Уточнение формального описания концептуальной модели 1-го рода предметной задачи на объектном уровне

Концептуальная модель 1-го рода на объектном уровне для n- ой предметной задачи включает следующие совокупности: множество предметных категорий (ПК), множество статических (структурных) отношений на ПК и множество динамических отношений на ПК [74,75,78,79]:

$$KP2.1(n) = < M2(n), T2(n), H2(n), \overline{F2}(n), R_2^{KP1}(n) > \quad (2.10)$$

Множество ПК для n- ой предметной задачи разбивается на множество непересекающихся подмножеств [3,74,79]:

$$M2(n) \equiv A(n) \quad (2.11)$$

$$A(n) = \bigcup_i A_i\,(n), A_i(n) \cap\ A_j(n) = \emptyset, A_i(n) \subset A(n),$$

$$A_i(n) = \{a_{ik}\}$$

$T2(n)$- множество бинарных отношений и $H2(n)$- множество тернарных отношений на ПК, $\overline{F2}(n)$ – множество динамических отношений 1-го рода, $R_2^{KP1}(n)$-увязка статической и динамической составляющих.

$$T2(n)\ \subset\ A(n) \times A(n). \quad (2.12)$$

Множество $T2(n)$ можно представить как совокупность подмножеств различных видов [3]:

$$T2(n) = \{LA_i(n), LA_i(n)A_j(n), LA_j(n)/\ A_i(n)\} \quad (2.13)$$

$LA_i(n)$ описывает бинарные связи вида “упорядочивание” («У») между ПК одного подмножества:

$$LA_i(n) = \{(a_{il}, a_{is}) | a_{il}, a_{is} \in A_i(n)\} \quad (2.14)$$

$LA_i(n)A_j(n)$ описывает бинарные связи вида “состав” («С») между ПК разных подмножеств:

$$LA_i(n)A_j(n) = \{(a_{il}, a_{jm})|a_{il}, \in A_i(n), a_{jm} \in A_j(n)\} \quad (2.15)$$

$LA_j(n)/A_i(n)$ описывает бинарные связи вида “компоновка” («К») между ПК одного подмножества:

$$LA_j(n)/A_i(n) = (a_{jm}, a_{jt})|a_{jm}, a_{jt} \in A_j(n) \quad \& \quad (2.16)$$
$$(\exists a_{il} \in A_i(n): (a_{il}, a_{jm}), (a_{il}, a_{jt}) \in LA_i(n)A_j(n))\}$$

Множество тернарных отношений $H2(n)$ также описывает подмножества определенного вида:

$$H2(n) \subset A(n) \times A(n) \times A(n) \quad (2.17)$$
$$H2(n) = \{HA_i(n)A_j(n)A_k(n)\}$$
$$HA_i(n)A_j(n)A_k(n) = \{(a_{il}, a_{jm}, a_{kn})| \quad (2.18)$$
$$a_{il} \in A_i(n), a_{jm} \in A_j(n), a_{kn} \in A_k(n) \quad \&$$
$$(a_{il}, a_{jm}) \in LA_i(n)A_j(n), (a_{jm}, a_{kb}) \in LA_j(n)A_k(n) \quad \&$$
(a_{il}-контекстуальная ПК, a_{jm}- структурная ПК, a_{kn}-монадическая ПК)}

Динамические отношения включают систему предметных зависимостей первого рода $\overline{F2}(n)$ [3,74,79]:

$$\overline{F2}(n) = (\overline{V2}(n), \overline{FV2}(n)) \quad (2.19)$$

где $\overline{V2}(n)$ – множество предметных зависимостей 1-го рода (ПЗ-1) и $\overline{FV2}(n)$-множество бинарных отношений между ПЗ-1.

Множество ПЗ-1 разбивается на множество непересекающихся подмножеств разных уровней разложения (декомпозиции):

$$\overline{V2}(n) \equiv \overline{W}(n), \overline{W}(n) = \bigcup_s \overline{W_s}(n), \quad (2.20)$$
$$\overline{W_s}(n) = \{\overline{w_{sp}}\}$$
$$\overline{FV2}(n) \subset \overline{W}(n) \times \overline{W}(n), \quad (2.21)$$
$$\overline{FV2}(n) = \{L\overline{W_s}(n), L\overline{W_s}(n), \overline{W_t}(n), L\overline{W_t}(n)/\overline{W_s}(n)\} \quad (2.22)$$

$L\overline{W_s}(n)$ описывает бинарные связи вида «У» между ПЗ-1 одного подмножества (т.е. упорядоченность ПЗ-1 на одном уровне декомпозиции) [3]:

$$L\overline{W_s}(n) = \{(\overline{w_{sp}}, \overline{w_{sv}}) |\ \overline{w_{sp}}, \overline{w_{sv}} \in \overline{W_s}(n)\} \quad (2.23)$$

$L\overline{W_s}(n)\overline{W_t}(n)$ описывает бинарные связи вида «С» между ПЗ-1 разных подмножества (т.е. связи между ПЗ-1разных уровней декомпозиции:

$$L\overline{W_s}(n)\overline{W_t}(n) = \{\overline{(w_{sp}, \overline{w_{tn}})} | \overline{w_{sp}} \in \overline{W_s}(n), \overline{w_{tn}} \in \overline{W_t}(n)\} \quad (2.24)$$

$L\overline{W_t}(n)/\overline{W_s}(n)$ описывает бинарные связи вида «К» между ПЗ-1 одного подмножества (т.е. локальная компоновкаПЗ-1, обусловленная связью «состав»):

$$L\overline{W_t}(n)/\overline{W_s}(n) = \{(\overline{w_{tk}}, \overline{w_{tn}}) | \overline{w_{tk}}, \overline{w_{tn}} \in \overline{W_t}(n) \quad \& \quad (2.25)$$

$$(\exists \overline{w_{sp}} \in \overline{W_s}(n): (\overline{w_{sp}}, \overline{w_{tk}}), (\overline{w_{sp}}, \overline{w_{tn}}) \in L\overline{W_s}(n)/\overline{W_t}(n))\}$$

Так как ПЗ-1 всегда есть вариант определенного типа, то ПЗ-1 как вариант - фиксируется на подмножестве множества предметных категорий (в силу наличия ограничений универсального представления) [3,74,79]:

$$\overline{w_{sp}} = \overline{w_{sp}}(i\ , t, u_{sp}) \quad (2.26)$$

Отношения, описывающие взаимосвязи статистической и динамической составляющих модели предметной задачи объектного уровня имеют вид [3]:

$$R_2^{KP1}(n) \subset A(n) \times A(n) \ldots \times A(n) \times \overline{W}(n), \quad (2.27)$$

$$R_2^{KP1}(n) = \{r_l^{KP-1} | r_l^{KP-1}: (a_{j1}, \ldots, a_{jq}) \rightarrow \overline{w_{sp}}\}$$

ПЗ-1 для 1-го типа отражает ограничения на взаимосвязь структурных ПК в рамках одной контекстуальной ПК (рис. 2.3) [3,75]:

$$\overline{w_{sp}}(i, 1, u_{sp}) = (\ a_{j1}, \ldots, a_{jq})_{u_{sp}} \&(a_{il}, a_{j1}) \& \ldots \&(\ a_{il}, \ldots, a_{jq}) | \quad (2.28)$$

$$a_{j1}, \ldots, a_{jq} \in A_j(n)\ \&\ a_{il} \in A_i(n) \quad \&$$

$$(\ a_{il}, a_{j1}), \ldots, (a_{il}, a_{jq}) \in LA_i(n)A_j(n) \quad \&$$

$$(a_{il}, a_{j1}, a_{km}), \ldots, (a_{il}, a_{jq}, a_{kn}) \in HA_i(n)A_j(n)A_k(n)$$

где $(\ a_{j1}, \ldots, a_{jq})_{u_{sp}}$- множество структурных ПК, связанных с данной ПЗ-1, а $(a_{il}, a_{j1}), \ldots, (a_{il}, a_{jq})$- множество связей этих ПК с определенной контекстуальной ПК.

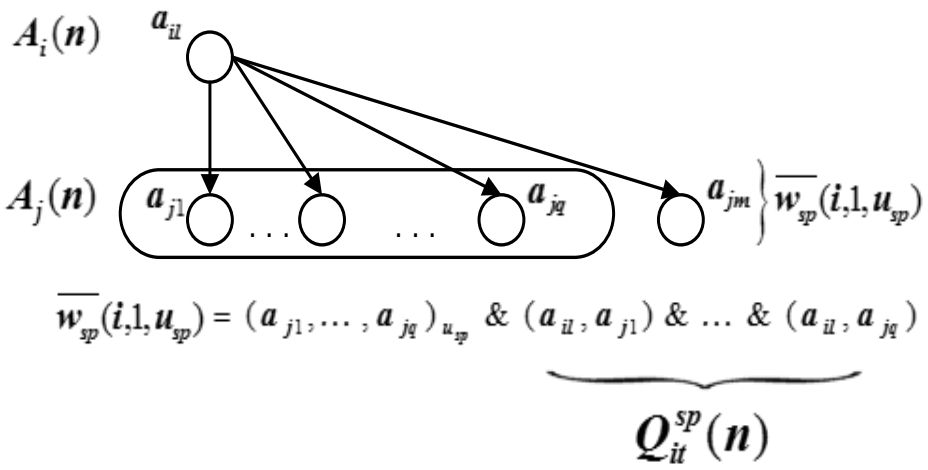


Рис.2.3. ПЗ-1 для 1-го типа [3,75]

ПЗ-1 для 2-го типа отражает ограничения на взаимосвязь структурных ПК, связанных с разными, но однородными контекстуальными ПК (рис. 2.4) [3,75]:

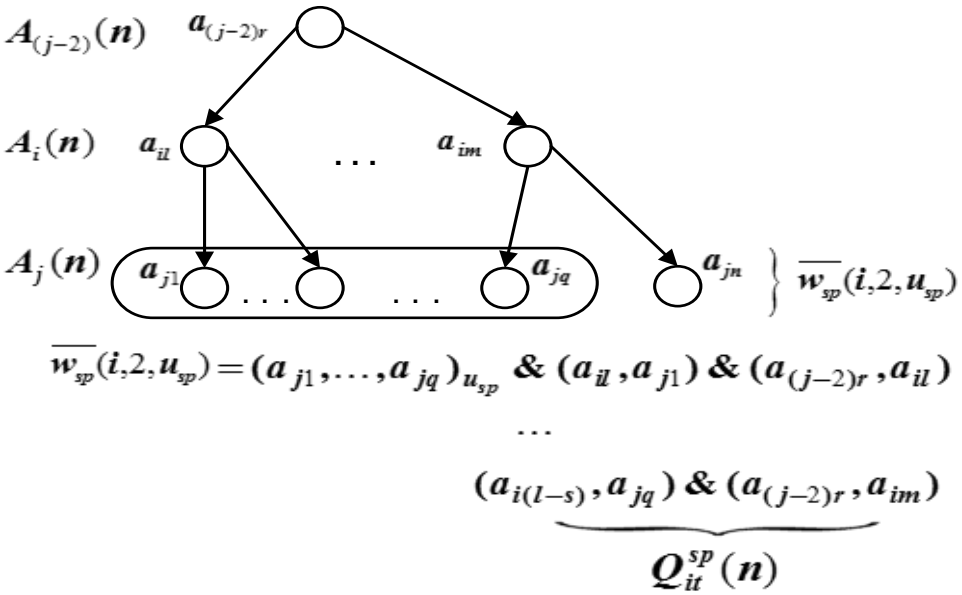


Рис.2.4. ПЗ-1 для 2-го типа [3,75]

$$\overline{w_{sp}}(i,2,u_{sp}) = (a_{j1},\dots,a_{jq})_{u_{sp}} \& (a_{il},a_{j1}) \& (a_{(j-2)r},a_{il}) \& \qquad (2.29)$$

$$\dots$$

$$(a_{im},a_{jq}) \& (a_{(j-2)r},a_{im}) |$$

$$a_{j1},\dots,a_{jq} \in A_j(n) \quad \&$$

$$a_{il},\dots,a_{im} \in A_i(n) \quad \&$$

$$a_{(j-2)r} \in A_{j-2}(n) \quad \&$$

$$(a_{il}, a_{j1}), \dots, (a_{il}, a_{jq}) \in LA_i(n)A_j(n)\ \&$$

$$(a_{(j-2)r}, a_{im}), \dots, (a_{(j-2)r}, a_{il}) \in LA_{j-2}(n)A_i(n)\ \&$$

$$(a_{il}, a_{j1,} a_{km}), \dots, (a_{il}, a_{jq}, a_{kn}) \in HA_i(n)A_j(n)A_k(n)\ \&$$

$$(a_{st}, a_{(j-2)r}, a_{il}), .., (a_{st}, a_{(j-2)r}, a_{im}) \in HA_s(n)A_{j-2}(n)A_i(n)$$

где $(a_{il}, \dots, a_{j1})_{u_{sp}}$- множество структурных ПК, связанных с данной ПЗ-1, а $(a_{il}, a_{j1})\&(a_{(j-2)r}, a_{il}), \dots, (a_{im}, a_{jq})\&(a_{(j-2)r}, a_{im})$ - множество связей этих структурных ПК с определенными однородными контекстуальными ПК.

ПЗ-1 для 3-го типа отражает ограничения на взаимосвязь структурных ПК, связанных с разными и разнородными контекстуальными ПК (рис.2.5) [3,75]:

$$\overline{w_{sp}}(i, 3, u_{sp}) = (a_{j1}, \dots, a_{jq})_{u_{sp}}\&(a_{il}, a_{j1})\&(a_{(j-2)r}, a_{il})\& \qquad (2.30)$$

…

$$(a_{im}, a_{jq})\&(a_{(j-2)t}, a_{im})|$$

$$a_{j1}, \dots, a_{jq} \in A_j(n)\ \&$$

$$a_{il}, \dots, a_{im} \in A_i(n)\ \&$$

$$a_{(j-2)t}, \dots, a_{(j-2)r} \in A_{j-2}(n)\ \&$$

$$(a_{il}, a_{j1}), \dots, (a_{im}, a_{jq}) \in LA_i(n)A_j(n)\ \&$$

$$(a_{(j-2)r}, a_{il}), \dots, (a_{(j-2)t}, a_{im}) \in LA_{j-2}(n)A_i(n)\ \&$$

$$(a_{il}, a_{j1,} a_{km}), \dots, (a_{im}, a_{jq}, a_{kn}) \in HA_i(n)A_j(n)A_k(n)\ \&$$

$$(a_{st}, a_{(j-2)r}, a_{il}), \dots,$$

$$(a_{sg}, a_{(j-2)t}, a_{im})\ \in HA_s(n)A_{j-2}(n)A_i(n)$$

где $(a_{il}, \dots, a_{j1})_{u_{sp}}$- множество структурных ПК, связанных с данным ограничанием 1-го рода, а $(a_{il}, a_{j1})\&(a_{(j-2)r}, a_{il}),\ \dots, (a_{im}, a_{jq})\&(a_{(j-2)t}, a_{im})$ - множество связей структурных ПК с определенными разнородными контекстуальными ПК.

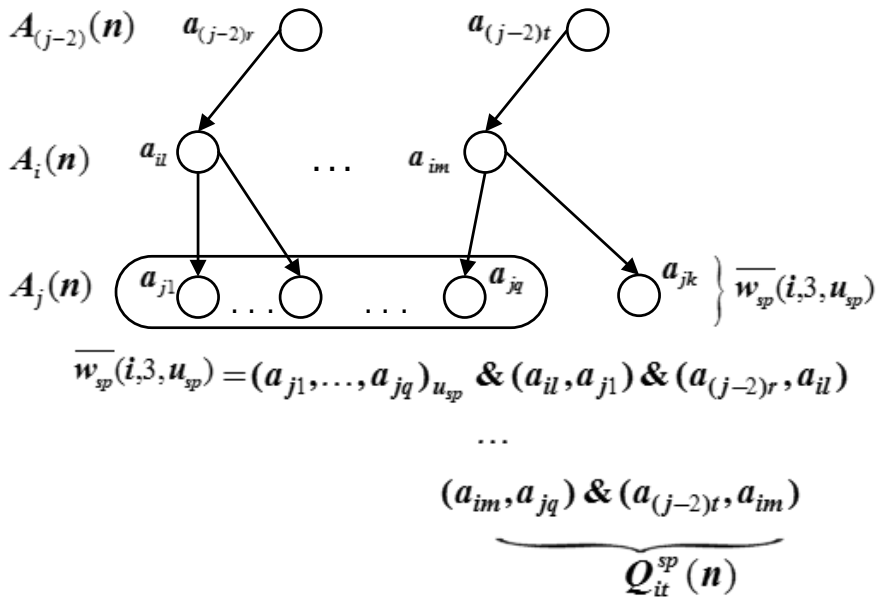


Рис. 2.5. ПЗ-1 для 3-го типа [3,75]

Таким образом, соотношения между ПЗ-1 и ПК можно описать в общем виде [3,74,75,79]:

$$\overline{w_{sp}}(i,t,u_{sp}) = (a_{j1},\ldots,a_{jq})_{u_{sp}} \& Q_{it}^{sp}(n) \qquad (2.31)$$

где $Q_{it}^{sp}(n)$ – описывает соотношения структурных и контекстуальных ПК и индекс u_{sp} размечает подмножество $A_j(n)$, которое характеризует $sp-$ е ограничение или ПЗ-1 для выделенной предметной задачи.

2.2.3 Уточнение формального описания концептуальной модели 1-го рода предметной задачи на конкретном уровне

Концептуальная модель конкретного уровня 1-го рода есть совокупность множества экземпляров предметных категорий (ЭПК), множества статических (структурных) отношений и множества динамических отношений [3,74,77,78,79]:

$$KP3.1(nm) = < M3(nm), T3(nm), H3(nm), \overline{F3}(nm), R_3^{KP1}(nm) > \qquad (2.32)$$

где каждое из множеств можно представить в следующем виде. Множество ЭПК для m-ой реализации n-ой предметной задачи разбивается на множество непересекающихся подмножеств.

$$M3(nm) \equiv B(nm) \tag{2.33}$$

$$B(nm) = \bigcup_{i,j} B_{ij}\,(nm), B_{ij}(nm) \cap\ B_{km}(nm) = \emptyset, \tag{2.34}$$

$$B_{ij}(nm) \subset B(nm), B_{ij}(nm) = \{b_r^{ij}\} \tag{2.35}$$

$T3(nm)$- множество бинарных отношений и $H3(nm)$- множество тернарных отношений между экземплярами ПК, $\overline{F3}(nm)$ – динамические отношения, $R_3^{KP-1}(nm)$- увязка статической и динамической составляющих.

$$T3(nm)\ \subset\ B(nm) \times B(nm). \tag{2.36}$$

Множество $T3(nm)$ описывается как совокупность подмножеств различных видов [3]:

$$T3(nm) = \{LB_{ip}(nm), LB_{ip}(nm)B_{iq}(nm), LB_{iq}(nm) \tag{2.37}$$
$$/\ B_{ip}(nm), LB_{il}(nm)B_{jm}(nm),$$
$$LB_{jn}(nm)LB_{jm}(nm)/\ B_{il}(nm)\}$$

$LB_{ip}(nm)$ описывает бинарные связи вида «У» между ЭПК одного подмножества (p) одного сорта(i):

$$LB_{ip}(nm) = \left\{\left(b_s^{ip}, b_t^{ip}\right)\middle| b_s^{ip}, b_t^{ip} \in B_{ip}(nm)\right\} \tag{2.38}$$

$LB_{ip}(nm)B_{iq}(nm)$ описывает бинарные связи вида «С» между ЭПК разных подмножеств (p, q) одного сорта(i):

$$LB_{ip}(nm)B_{iq}(nm) = \tag{2.39}$$
$$= \left\{\left(b_r^{ip}, b_s^{iq}\right)\middle| b_r^{ip}, \in B_{ip}(nm), b_s^{iq} \in B_{iq}(nm)\right\}$$

$LB_{iq}(nm)/B_{ip}(nm)$ описывает бинарные связи вида «К» между ЭПК одного подмножества (q) одного сорта(i):

$$LB_{iq}(nm)/B_{ip}(nm) = \{(b_t^{iq}, b_s^{iq}) | b_t^{iq}, b_s^{iq} \in B_{iq}(nm)\ \ \& \tag{2.40}$$
$$(\exists b_r^{ip} \in B_{ip}(nm): \left(b_r^{ip}, b_t^{iq}\right), (b_r^{ip}, b_s^{iq}) \in LB_{ip}(nm)B_{iq}(nm))\}$$

$LB_{in}(nm)B_{jm}(nm)$ описывает бинарные связи вида «С» между ЭПК разных подмножеств (l,m) разных сортов (i,j):

$$LB_{il}(nm)B_{jm}(nm) = \tag{2.41}$$

$$= \left\{\left(b_r^{il}, b_t^{jm}\right)\middle| b_r^{il}, \in B_{il}(nm), b_t^{jm} \in B_{jm}(nm)\right\}$$

$LB_{jn}(nm)B_{jm}(nm)/B_{il}(nm)$ описывает бинарные связи вида «К» между ЭПК разных подмножеств (n,m) одного сорта(j):

$$LB_{jn}(nm)B_{jm}(nm)/B_{ip}(nm) = (b_s^{jn}, b_t^{jm})|b_s^{jn} \in B_{jn}(nm), b_t^{jm} \tag{2.42}$$

$$\in B_{jm}(nm)\ \&$$

$$(\exists b_r^{il} \in B_{il}(nm): (b_r^{il}, b_s^{jn}) \in LB_{il}(nm)B_{jn}(nm),$$

$$(b_r^{il}, b_t^{jm}) \in LB_{il}(nm)B_{jm}(nm))\}.$$

$$H3(nm) \subset B(nm) \times B(nm) \times B(nm). \tag{2.43}$$

$H3(nm)$ также описывает как совокупность подмножеств определенного вида:

$$H3(nm) = \{HB_{il}(nm)B_{jm}(nm)B_{kn}(nm)\} \tag{2.44}$$

$$HB_{il}(nm)B_{jm}(nm)B_{kn}(nm) = \{(b_s^{il}, b_t^{jm}, b_r^{kn})|$$

$$b_s^{il} \in B_{il}(nm), b_t^{jm} \in B_{jm}(nm), b_r^{kn} \in B_{kn}(nm)\ \&$$

$$\left(b_s^{il}, b_t^{jm}\right) \in LB_{il}(nm)B_{jm}(nm)\ \&$$

$$\left(b_t^{jm}, b_r^{kn}\right) \in LB_{jm}(nm)B_{kn}(nm)\ \&$$

(b_s^{il}-контекстуальная ЭПК, b_t^{jm}- структурная ЭПК, b_r^{kn}-

монадическая ЭПК)}

Динамические отношения это система предметных зависимостей первого рода $\overline{F3}(nm)$ [3,74,79]:

$$\overline{F3}(nm) = (\overline{V3}(nm), \overline{FV3}(nm)), \tag{2.45}$$

где $\overline{V3}(nm)$ – множество экземпляров предметных зависимостей 1-го рода (ЭПЗ-1) и $\overline{FV3}(nm)$-множество бинарных отношений между ЭПЗ-1.

$$\overline{V3}(nm) \equiv \overline{P}(nm) \tag{2.46}$$

$$\overline{P}(nm) = \bigcup_{i,p} \overline{P_{ip}}\,(nm), \overline{P_{ip}}(nm) \cap\ \overline{P_{rt}}(nm) = \varnothing,$$

$$\overline{P_{rt}}(nm) \subset\ \overline{P}(nm), \overline{P_{rt}}(nm) = \{p_s^{rt}\}$$

$$\overline{FV3}(nm) \subset\ \overline{P}(nm) \times \overline{P}(nm) \qquad (2.47)$$

$\overline{FV3}(nm)$ описывает совокупность подмножеств определенного вида [3].

$$\overline{FV3}(nm) = \{L\overline{P_{ip}}(nm), L\overline{P_{ip}}(nm)\overline{P_{iq}}(nm), \qquad (2.48)$$

$$L\overline{P_{iq}}(nm)/\overline{P_{ip}}(nm),$$

$$L\overline{P_{il}}(nm)\overline{P_{kn}}(nm), L\overline{P_{kn}}(nm)\overline{P_{km}}(nm)/\overline{P_{il}}(nm)\}$$

$L\overline{P_{ip}}(nm)$ описывает бинарные связи вида «У» между ЭПЗ-1 одного подмножества (p) одного сорта(i)(или упорядоченность экземпляров одной ПЗ-1):

$$L\overline{P_{ip}}(nm) = \{(\overline{p_l^{ip}}, \overline{p_s^{ip}})|\ \overline{p_s^{ip}}, \overline{p_t^{ip}} \in \overline{P_{ip}}(nm)\} \qquad (2.49)$$

$L\overline{P_{ip}}(nm)\overline{P_{iq}}(nm)$ описывает бинарные связи вида «С» между ЭПЗ-1 разных подмножеств (p, q) одного сорта(i)(или связи экземпляров разных ПЗ-1 одного уровня разложения):

$$L\overline{P_{ip}}(nm)\overline{P_{iq}}(nm)\ = \{(\overline{p_r^{ip}}, \overline{p_l^{iq}})|\overline{p_r^{ip}} \in \overline{P_{ip}}(nm), \overline{p_l^{iq}} \in \overline{P_{iq}}(nm)\} \qquad (2.50)$$

$L\overline{P_{iq}}(nm)/\overline{P_{ip}}(nm)$ описывает бинарные связи вида «К» между ЭПЗ-1 одного подмножества (q) одного сорта(i)

$$L\overline{P_{iq}}(nm)/\overline{P_{ip}}(nm) = \{\overline{(p_s^{iq}, p_l^{iq})}|\overline{p_s^{iq}}, \overline{p_l^{iq}}\ \in \overline{P_{iq}}(nm)\ \ \& \qquad (2.51)$$

$$(\exists \overline{p_r^{ip}} \in \overline{P_{ip}}(nm)\colon \overline{(p_r^{ip}, p_s^{iq})}, \overline{(p_r^{ip}, p_t^{iq})}\ \in L\overline{P_{ip}}(nm)\overline{P_{iq}}(nm))\}$$

$L\overline{P_{in}}(nm)\overline{P_{jk}}(nm)$ описывает бинарные связи вида «С» между ЭПЗ-1 разных подмножеств (n, k) разных сортов(i, j)(или связи экземпляров разных ПЗ-1 разных уровней разложения):

$$L\overline{P_{in}}(nm)\overline{P_{jk}}(nm) = \{\overline{(p_r^{in}, p_l^{jk})}|\overline{(p_r^{in}} \in \overline{P_{in}}(nm)\ , \overline{p_l^{jk}}\ \in \qquad (2.52)$$

$$\overline{P_{jk}}(nm)\}$$

$L\overline{P_{jk}}(nm)\overline{P_{jm}}(nm)/\overline{P_{in}}(nm)$ описывает бинарные связи вида «К» между ЭПЗ-1 разных подмножеств (k,m) одного сорта (j) (или локальная компоновка экземпляров разных ПЗ-1, обусловленная связью «состав»):

$$L\overline{P_{jk}}(nm)\overline{P_{jm}}(nm)/\overline{P_{in}}(nm) = \{\overline{(p_l^{jk}, p_s^{jm})}| \quad (2.53)$$

$$\overline{p_l^{jk}} \in \overline{P_{jk}}(nm), \overline{p_s^{jm}} \in \overline{P_{jm}}(nm) \ \&$$

$$(\exists \overline{p_r^{in}} \in \overline{P_{in}}(nm): \overline{(p_r^{in}, p_l^{jk})} \in L\overline{P_{in}}(nm)\overline{P_{jk}}(nm),$$

$$\overline{(p_r^{in}, p_s^{jm})} \in L\overline{P_{in}}(nm)\overline{P_{jm}}(nm))\}$$

Так как ЭПЗ-1 всегда есть вариант определенного типа, то ЭПЗ-1 как вариант - фиксируется на подмножестве множества экземпляров предметных категорий (в силу ограничений универсального представления) [3,74,79]:

$$\overline{p_r^{sp}} = \overline{p_r^{sp}}(i,t,s,u_{spr}) \quad (2.54)$$

Отношения, описывающие взаимосвязи статической и динамической составляющих модели предметной задачи конкретного уровня [3]:

$$R_3^{KP1}(n) \subset B(nm) \times B(nm) \times \ldots \times B(nm) \times \overline{P}(nm) \quad (2.55)$$

$$R_3^{KP1}(n) = \left\{ r_l^{KP3-1} | r_l^{KP3-1}: \left(b_x^{j1}, \ldots, b_z^{jq}\right) \rightarrow \overline{p_k^{sp}} \right\}$$

ЭПЗ 1-го рода для 1-го типа отражает ограничения на взаимосвязь между ЭПК в рамках одного контекстуального ЭПК (рис.2.6) [3]:

$$\overline{p_r^{sp}}(i,1,s,u_{spr}) = \left(b_x^{j1}, \ldots, b_z^{jq} \right)_{u_{spr}} \& \quad (2.56)$$

$$(b_r^{il}, b_x^{j1}) \& \ldots \& (b_r^{il}, b_z^{jq})|$$

$$b_r^{il} \in B_{il}(nm) \ \&$$

$$b_x^{j1} \in B_{j1}(nm), \ldots, b_z^{jq} \in B_{jq}(nm) \ \&$$

$$\left(b_r^{il}, b_x^{j1}\right) \in LB_{il}(nm)B_{j1}(nm) \ \& \ldots$$

$$\left(b_r^{il}, b_z^{jq}\right) \in LB_{il}(nm)B_{jq}(nm) \ \&$$

$$\left(b_r^{il}, b_x^{j1}, b_t^{km}\right) \in LB_{il}(nm)B_{j1}(nm)B_{km}(nm) \ \& \ldots$$

$$\left(b_r^{il}, b_z^{jq}, b_r^{kn}\right) \in LB_{il}(nm)B_{jq}(nm)B_{kn}(nm),$$

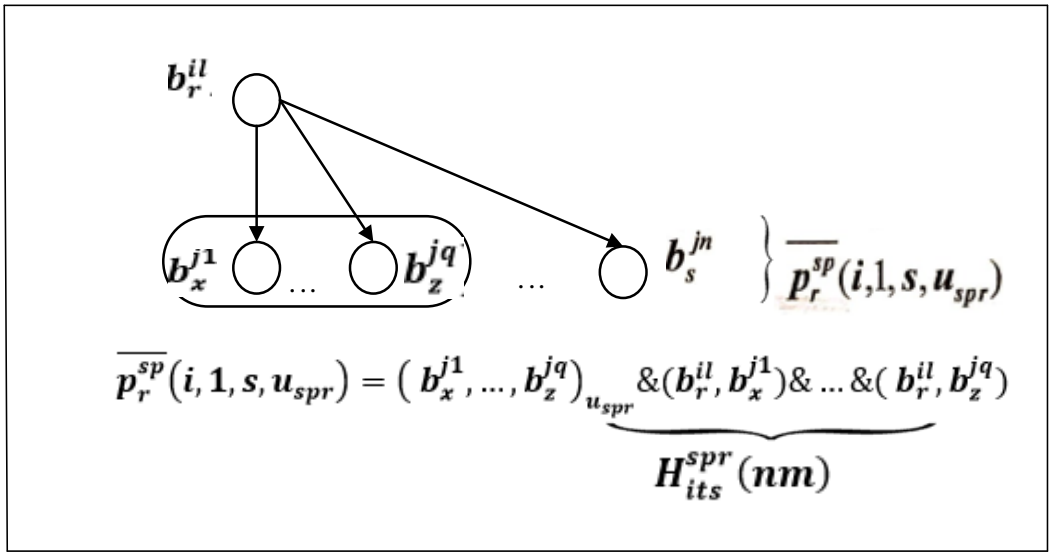


Рис.2.6. ЭПЗ-1 для 1-го типа [3]

где $(b_x^{j1},\dots,b_z^{jq})_{u_{spr}}$ - множество структурных ЭПК, связанных с данным экземпляром предметной зависимости 1-го рода [3], а $(b_r^{il},b_x^{j1}),\dots,(b_r^{il},b_z^{jq})$- множество связей этих ЭПК с определенной контекстуальной ЭПК.

ЭПЗ-1 для 2-го типа отражает ограничения на взаимосвязь между структурными ЭПК в рамках разных, но однородных контекстуальных ЭПК (рис. 2.7) [3]:

$$\overline{p_r^{sp}}(i,2,s,u_{spr}) = (b_x^{j1},\dots,b_z^{jq})_{u_{spr}}\& \quad (2.57)$$

$$(b_a^{il},b_x^{j1})\& \left(b_f^{(j-2)r},b_a^{il}\right)\&$$

$$\dots$$

$$(b_b^{ik},b_z^{jq})\&(b_f^{(j-2)r},b_b^{jk})|$$

$$b_x^{j1} \in B_{j1}(nm),\dots,b_z^{jq} \in B_{jq}(nm)\ \&$$

$$b_b^{ik} \in B_{ik}(nm),\dots,b_a^{il} \in B_{il}(nm)\ \&$$

$$b_h^{(j-2)r} \in B_{(j-2)r}(nm)\ \&$$

$$\left(b_a^{il},b_x^{j1}\right) \in LB_{il}(nm)B_{j1}(nm)\ \& \dots$$

$$\left(b_b^{ik},b_z^{jq}\right) \in LB_{ik}(nm)B_{jq}(nm)\ \&$$

$$(b_f^{(j-2)r}b_a^{il}) \in LB_{(j-2)r}(nm)B_{il}(nm)\ \&\dots$$

$$(b_f^{(j-2)r} b_b^{ik}) \in LB_{(j-2)r}(nm)B_{ik}(nm)\ \&$$

$$\left(b_a^{il}, b_x^{j1}, b_t^{km}\right) \in HB_{il}(nm)B_{j1}(nm)B_{km}(nm)\ \&\ldots$$

$$\left(b_b^{ik}, b_z^{jq}, b_r^{kn}\right) \in HB_{ik}(nm)B_{jq}(nm)B_{kn}(nm)\ \&$$

$$\left(b_c^{ab}, b_f^{(j-2)r} b_b^{ik}\right) \in HB_{ab}(nm)B_{(j-2)r}(nm)B_{ik}(nm)\ \&\ldots$$

$$\left(b_c^{ab}, b_f^{(j-2)r} b_a^{il}\right) \in HB_{ab}(nm)B_{(j-2)r}(nm)B_{il}(nm)$$

где $\left(b_x^{j1}, \ldots, b_z^{jq}\right)_{u_{spr}}$ - множество структурных ЭПК, связанных с данным экземпляром предметной зависимости 1-го рода, а $\left(b_r^{il}, b_x^{j1}\right)\&(\, b_f^{(j-2)r}, b_a^{il}), \ldots, \left(b_b^{ik}, b_z^{jq}\right)\&(\, b_f^{(j-2)r}, b_b^{ik})$- множество связей этих ЭПК с определенной контекстуальной ЭПК.

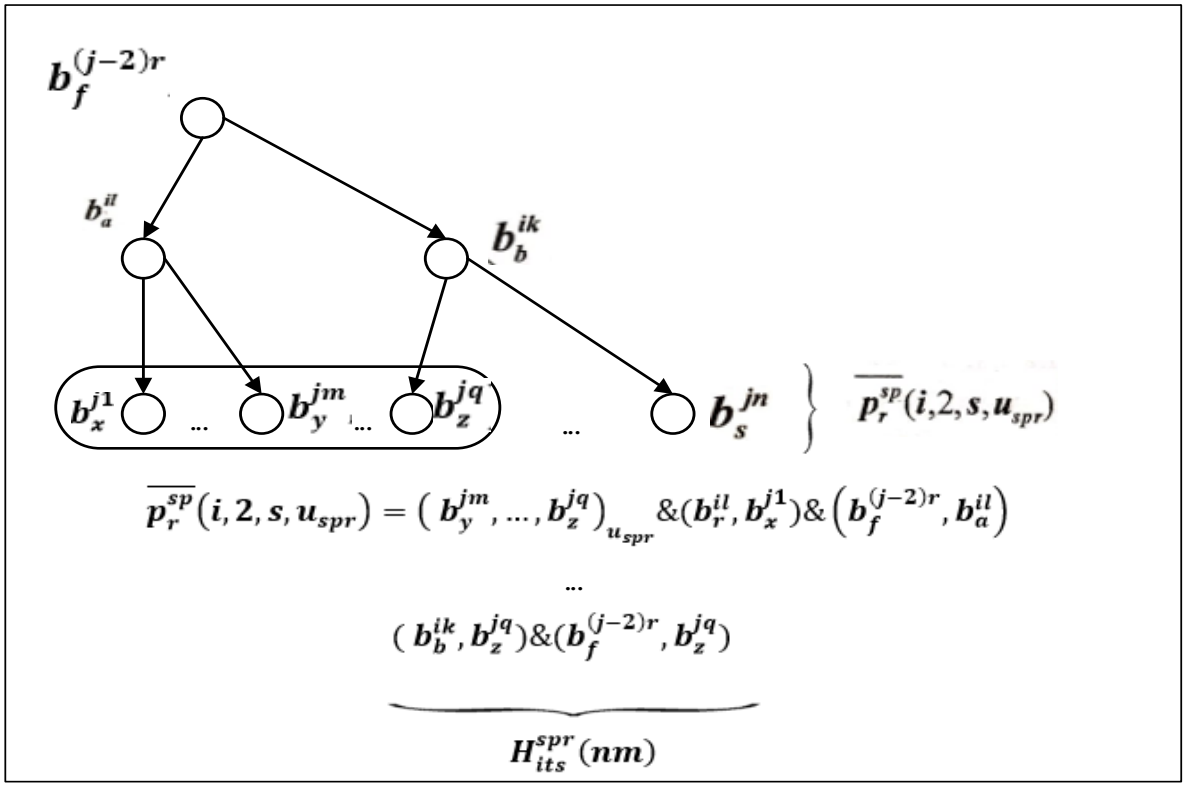


Рис.к 2.7. ЭПЗ-1 для 2-го типа [3]

ЭПЗ-1 для 3-го типа отражает ограничения на взаимосвязь между структурными ЭПК в рамках разных и разнородных контекстуальных ЭПК (рис.2.8) [3]:

$$\overline{p_r^{sp}}(i,3,s,u_{spr}) = \tag{2.58}$$

$$= \left(b_x^{j1},\dots,b_z^{jq}\right)_{u_{spr}} \&\left(b_r^{il},b_x^{j1}\right)\&\left(b_f^{(j-2)r},b_a^{il}\right)\&$$

$$\dots$$

$$\left(b_b^{ik},b_z^{jq}\right)\&\left(b_g^{(j-2)u},b_b^{ik}\right)|$$

$$b_x^{j1} \in B_{j1}(nm),\dots,b_z^{jq} \in B_{jq}(nm)\ \&$$

$$b_b^{ik} \in B_{ik}(nm),\dots,b_a^{il} \in B_{il}(nm)\ \&$$

$$b_g^{(j-2)u} \in B_{(j-2)u}(nm),\dots,b_f^{(j-2)r} \in B_{(j-2)r}(nm)\ \&$$

$$\left(b_a^{il},b_x^{j1}\right) \in LB_{il}(nm)B_{j1}(nm)\ \&\dots$$

$$\left(b_b^{ik},b_z^{jq}\right) \in LB_{ik}(nm)B_{jq}(nm)\ \&$$

$$\left(b_f^{(j-2)r}b_a^{il}\right) \in LB_{(j-2)r}(nm)B_{il}(nm)\ \&\dots$$

$$\left(b_g^{(j-2)u}b_b^{ik}\right) \in LB_{(j-2)u}(nm)B_{ik}(nm)\ \&$$

$$\left(b_a^{il},b_x^{j1},b_t^{km}\right) \in HB_{il}(nm)B_{j1}(nm)B_{km}(nm)\ \&\dots$$

$$\left(b_b^{ik},b_z^{jq},b_r^{kn}\right) \in HB_{ik}(nm)B_{jq}(nm)B_{kn}(nm)\ \&$$

$$\left(b_c^{ab},b_g^{(j-2)u}b_b^{ik}\right) \in HB_{ab}(nm)B_{(j-2)u}(nm)B_{ik}(nm)\ \&\dots$$

$$\left(b_c^{ab},b_f^{(j-2)r}b_a^{il}\right) \in HB_{ab}(nm)B_{(j-2)r}(nm)B_{il}(nm)$$

где $\left(b_x^{j1},\dots,b_z^{jq}\right)_{u_{spr}}$- множество структурных ЭПК, связанных с данным экземпляром предметной зависимости 1-го рода, а $\left(b_r^{il},b_x^{j1}\right)\&\left(b_f^{(j-2)r},b_a^{il}\right),\dots,\left(b_b^{ik},b_z^{jq}\right)\&\left(b_g^{(j-2)u},b_b^{ik}\right)$- множество связей этих ЭПК с определенной контекстуальной ЭПК.

Таким образом, эти соотношения можно представить в общем виде [3,74,79]:

$$\overline{p_r^{sp}}(i,t,s,u_{spr}) = \left(b_x^{j1},\dots,b_z^{jq}\right)_{u_{spr}} \& H_{its}^{spr}(nm), \tag{2.59}$$

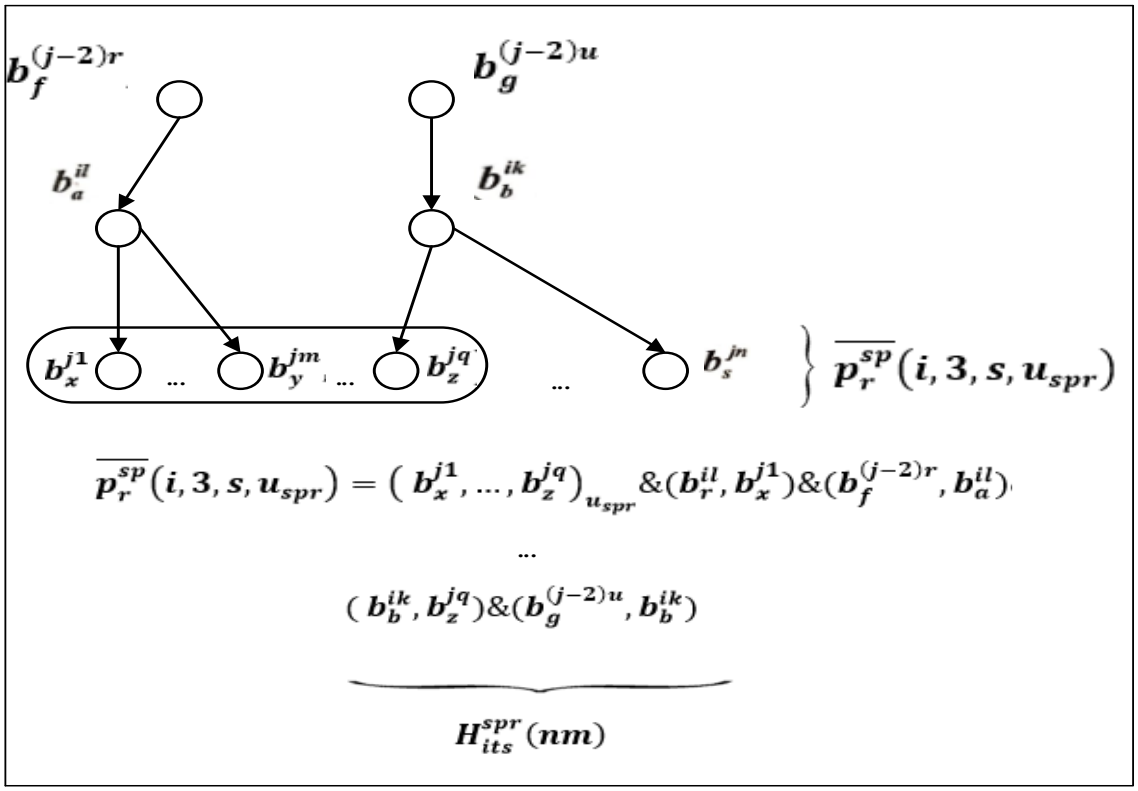


Рис. 2.8. ЭПЗ-1 для 3-го типа [3]

где $H_{its}^{spr}(nm)$ – условие, отражающее соотношения структурных и контекстуальных экземпляров предметных категорий и индекс u_{spr} отмечает подмножество ЭПК определенного класса (*j*), которое характеризует *spr*-ограничение для выделенной предметной задачи.

2.2.4 Уточнение формального описания взаимосвязей концептуальных моделей 1-го рода предметной задачи объектного и конкретного уровней

Отношения, описывающие взаимосвязи концептуальных моделей разных уровней абстрагирования для представления предметных задач $\mathrm{R}_{23}^{\mathrm{KP}}$ можно представить в следующем виде [3,74,76]:

$$R_{23}^{KP} \subset KP2(n) \times KP3(nm), \tag{2.60}$$

$$R_{23}^{KP} = \{r_l^{KP23} | r_l^{KP23} : KP2(n) \to KP3(nm)\}$$

Взаимосвязь моделей может быть представлена по-компонентно [3,74,79]:

$$KP2.1(n) \to KP3.1(nm): \quad (2.61)$$

$$A(n) \to B(nm); \quad (2.62)$$

$$T2(n) \to T3(nm), H2(n) \to H3(nm), \quad (2.63)$$

$$\overline{F2}(n) \to \overline{F3}(nm): \bar{W}(n) \to \bar{P}(nm), \overline{FV2}(n) \to \overline{FV3}(nm), \quad (2.64)$$

Или, по-другому, эту взаимосвязь концептуальных моделей можно представить в виде множеств и блочной матрицы [2,5,6,7]:

$$R_{23}^{KP1}(n) = \{R_{23}^{KP1-1}(n), R_{23}^{KP1-2}(n)\} \quad (2.65)$$

Табл.2.1. Взаимосвязи концептуальных моделей 1-го рода объектного и конкретного уровней [3,74,79]

$R_{23}^{KP1}(n)$	$B(nm)$	$T3(nm)$	$H3(nm)$	$\bar{P}(nm)$	$\overline{FV3}(nm)$
$A(n)$	$R_{23}^{KP1-1}(n)$	0	0	0	0
$T2(n)$	0	$R_{23}^{KP1-1}(n)$	0	0	0
$H2(n)$	0	0	$R_{23}^{KP1-1}(n)$	0	0
$\bar{W}(n)$	0	0	0	$R_{23}^{KP1-2}(n)$	0
$\overline{FV2}(n)$	0	0	0	0	$R_{23}^{KP1-2}(n)$

Таким образом, уточнение моделей осуществляется по каждому компоненту модели: элементам, статическим отношениям (бинарным и тернарным), динамическим отношениям (зависимостям 1-го рода и их связям).

При этом формальным обоснованием взаимосвязи служат следующие зависимости [3]:

$$\overline{GT3}(nm) = GA(n)B(nm) \bowtie GT2(n) \bowtie GA(n)B(nm), \quad (2.66)$$

где $\overline{GT3}(nm)$- допустимые бинарные связи на конкретном уровне для m-ой реализации n-ой предметной задачи, $GA(n)B(nm)$- отношение, описывающее уточнение ПК их экземплярами для реализации предметной задачи, $GT2(n)$-

отношение, описывающее бинарные связи ПК для предметной задачи, ⊳⊲ - знак операции реляционной алгебры «естественное соединение»;

$$\overline{\overline{GFV3}}(nm) = G\bar{W}(n)\bar{P}(nm) \ \triangleright\triangleleft \ G\overline{FV2}(n) \triangleright\triangleleft G\bar{W}(n)\bar{P}(nm), \quad (2.67)$$

где $\overline{\overline{GFV3}}(nm)$- допустимая система ограничений 1-го рода конкретного уровня для m-ой реализации n-ой предметной задачи, $G\bar{W}(n)\bar{P}(nm)$ - отношение, описывающее уточнение предметных ограничеий 1-го рода их экземплярами, $G\overline{FV2}(n)$- отношение, описывающее бинарные связи предметных ограничеий 1-го рода. ⊳⊲ - знак операции реляционной алгебры «естественное соединение».

Графическое представление правил перехода, в общем виде, приведено на рис. 2.9.

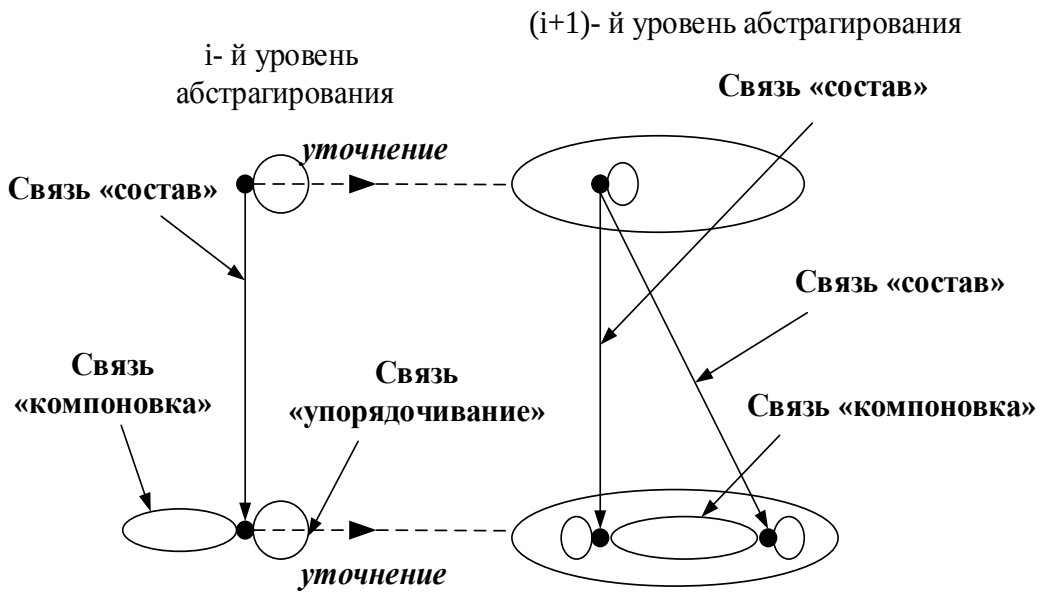


Рис.2.9. Графическая интерпретация правил перехода [3]

Это позволило сформулировать правила перехода от модели объектного уровня к модели конкретного уровня для выбранной предметной задачи [3]:

1) $R_{23}^{KP1-1}(n)$- для элементов (предметных категорий) моделей:

$$A_i(n) = \{a_{ij}\}, A_i(n) \subset A(n), a_{ij} \to B_{ij}(nm) = \{b_r^{ij}\}, \quad (2.68)$$

$$B_{ij}(nm) \subset B(nm);$$

2) $R_{23}^{KP1-1}(n)$- для бинарных связей на предметных категориях:

$$T2(n) \to T3(nm) \quad (2.69)$$

$$LA_i(n) = \{(a_{il}, a_{in})\}, LA_i(n) \subset T2(n)$$

$$(a_{ip}, a_{ip}) \to LB_{ip}(nm) = \{(b_l^{ip}, b_s^{ip})\}, LB_{ip}(nm) \subset T3(nm)$$

$$(a_{ip}, a_{iq}) \to LB_{ip}(nm)B_{iq}(nm) = \{(b_l^{ip}, b_s^{iq})\},$$

$$LB_{ip}(nm)B_{iq}(nm) \subset T3(nm)$$

$$(a_{iq}, a_{iq})\&(a_{ip}, a_{iq}) \to LB_{iq}(nm)/B_{ip}(nm) = \{(b_l^{iq}, b_s^{iq})\},$$

$$LB_{iq}(nm)/B_{ip}(nm) \subset T3(nm)$$

$$LA_i(n)A_j(n) = \{(a_{il}, a_{jm})\}, LA_i(n)A_j(n) \subset T2(n)$$

$$(a_{il}, a_{jm}) \to LB_{il}(nm)B_{jm}(nm) = \{(b_l^{il}, b_s^{jm})\},$$

$$LB_{il}(nm)B_{jm}(nm) \subset T3(nm)$$

$$LA_j(n)/A_i(n) = \{(a_{jn}, a_{jm})\}, LA_j(n)/A_i(n) \subset T2(n)$$

$$(a_{jn}, a_{jm})\&(a_{ik}, a_{jn})\&(a_{ik}, a_{jm}) \to$$

$$LB_{jn}(nm)B_{jm}(nm)/B_{ik}(nm) = \{(b_l^{jn}, b_s^{jm})\},$$

$$LB_{jn}(nm)B_{jm}(nm)/B_{ik}(nm) \subset T3(nm)$$

3) $R_{23}^{KP1-1}(n)$- для тернарных связей на предметных категориях:

$$H2(n) \to H3(nm) \quad (2.70)$$

$$HA_i(n)A_j(n)A_k(n) = \{(a_{il}, a_{jn}, a_{km})\},$$

$$HA_i(n)A_j(n)A_k(n) \subset H2(n)$$

$$(a_{il}, a_{jn}, a_{km}) \to HB_{il}(nm)B_{jn}(nm)B_{km}(nm) =$$

$$= \{(b_r^{il}, b_s^{jn}, b_t^{km})\},$$

$$HB_{il}(nm)B_{jn}(nm)B_{km}(nm) \subset H3(nm)$$

4) $R_{23}^{KP1-2}(n)$- для динамических отношений 1-го рода:

$$\overline{F2}(n) \to \overline{F3}(nm): \tag{2.71}$$

$R_{23}^{KP1-2}(n)$ - для предметных зависимостей 1-го рода:

$$\overline{W_i}(n) = \{\overline{w_{ij}}\}, \overline{W_i}(n) \subset \bar{W}(n), \tag{2.72}$$

$$\overline{w_{ij}} \to \overline{P_{ij}}(nm) = \left\{\overline{p_r^{ij}}\right\}, \overline{P_{ij}}(nm) \subset \bar{P}(nm);$$

5) $R_{23}^{KP1-2}(n)$- для бинарных связей предметных зависимостей 1-го рода:

$$\overline{FV2}(n) \to \overline{FV3}(nm) \tag{2.73}$$

$$L\overline{W_i}(n) = \{(\overline{w_{ij}}, \overline{w_{in}})\}, L\overline{W_i}(n) \subset \overline{FV2}(n)$$

$$\left(\overline{w_{ip}}, \overline{w_{ip}}\right) \to L\overline{P_{ip}}(nm) = \left\{\left(\overline{p_l^{ip}}, \overline{p_s^{ip}}\right)\right\},$$

$$L\overline{P_{ip}}(nm) \subset \overline{FV3}(nm)$$

$$\left(\overline{w_{ip}}, \overline{w_{iq}}\right) \to L\overline{P_{ip}}(nm)\overline{P_{iq}}(nm) = \left\{\left(\overline{p_l^{ip}}, \overline{p_s^{iq}}\right)\right\},$$

$$L\overline{P_{ip}}(nm)\overline{P_{iq}}(nm) \subset \overline{FV3}(nm)$$

$$\left(\overline{w_{iq}}, \overline{w_{iq}}\right)\&\left(\overline{w_{ip}}, \overline{w_{iq}}\right) \to L\overline{P_{iq}}(nm)/\overline{P_{ip}}(nm) = \left\{\left(\overline{p_l^{iq}}, \overline{p_s^{iq}}\right)\right\},$$

$$L\overline{P_{iq}}(nm)/\overline{P_{ip}}(nm) \subset \overline{FV3}(nm)$$

$$L\overline{W_i}(n)\overline{W_j}(n) = \{(\overline{w_{in}}, \overline{w_{jm}})\}, L\overline{W_i}(n)\overline{W_j}(n) \subset \overline{FV2}(n)$$

$$\left(\overline{w_{in}}, \overline{w_{jm}}\right) \to L\overline{P_{in}}(nm)\overline{P_{jm}}(nm) = \left\{\left(\overline{p_l^{in}}, \overline{p_s^{jm}}\right)\right\},$$

$$L\overline{P_{in}}(nm)\overline{P_{jm}}(nm) \subset \overline{FV3}(nm)$$

$$L\overline{W_j}(n)/\overline{W_i}(n) = \{(\overline{w_{jn}}, \overline{w_{jm}})\}, L\overline{W_j}(n)/\overline{W_i}(n) \subset \overline{FV2}(n)$$

$$\left(\overline{w_{jn}}, \overline{w_{jm}}\right)\&\left(\overline{w_{ik}}, \overline{w_{jn}}\right)\&\left(\overline{w_{ik}}, \overline{w_{jm}}\right) \to$$

$$L\overline{P_{jn}}(nm)\overline{P_{jm}}(nm)/\overline{P_{ik}}(nm) = \left\{\left(\overline{p_l^{jn}}, \overline{p_s^{jm}}\right)\right\},$$

$$L\overline{P_{jn}}(nm)\overline{P_{jm}}(nm)/\overline{P_{ik}}(nm) \subset \overline{FV3}(nm)$$

Таким образом, правила перехода задают методическую основу сопряжения моделей для представления знаний предметных задач.

2.3 Формальное описание типологии предметных ограничений 1-го рода

2.3.1 Общая характеристика типологии предметных ограничений 1-го рода

Поскольку любая ПЗ-1 может быть сопоставлена с определенным типом зависимостей 1-го рода, то можно для нее определить «замыкающую» вершину или ПК определенного класса в основной концептуальной структуре. Любой тип зависимости имеет двухпозиционный код, в котором 1-я позиция характеризует уровень семантической сложности, а 2-я позиция – определяет «замыкание» на одну или разные контекстуальные предметные категории данного уровня семантической сложности [74,76,79].

Все предметные категории (ПК) разделены на семь классов: Цикл, Процесс, Задача, Компонент, Объект, Признак, Значение. Эти классы образуют строгую иерархию и для уровней семантической сложности: контекстуальные предметные категории определяют классы семантической сложности (1- соответствует классу Цикл; 3- соответствует классу Задача; 5- соответствует классу Объект).

Исходя из вышеизложенного, можно определить совокупность правил определения типов предметных зависимостей 1-го рода:

- Тип 5.1: ограничение на взаимосвязь ПК класса «Признак» в рамках одной ПК класса «**Объект**»;

- Тип 5.2: ограничение на взаимосвязь ПК класса «Признак», связанных с разными, но однородными ПК класса «Объект», то есть принадлежащих одной ПК класса «**Компонент**»;

- Тип 3.1= тип 5.3: ограничение на взаимосвязь ПК класса «Признак», связанных с разными и разнородными ПК класса «Объект», принадлежащих различным ПК класса «Компонент», но в рамках одной ПК класса «**Задача**»;

- Тип 3.2: ограничение на взаимосвязь ПК класса «Признак», связанных с разными и разнородными ПК класса «Объект», принадлежащих различным ПК

класса «Компонент» разных, но однородных ПК класса «Задача», т.е. в рамках одной ПК класса «**Процесс**»;

- Тип 1.1 –тип 3.3: ограничение на взаимосвязь ПК класса «Признак», связанных с разными и разнородными ПК класса «Объект», принадлежащих различным ПК класса «Компонент» разных ПК класса «Задача», входящих в различные ПК класса «Процесс» в рамках одной ПК класса «**Цикл**»;

Таким образом, любая ПЗ-1 «замыкается» на одну ПК определенного класса в соответствии с их иерархией.

2.3.2 Формальное описание типологии предметных ограничений 1-го рода на объектном уровне

ПЗ-1 для типа 5.1 отражает ограничения на взаимосвязь структурных ПК класса «Признак» в рамках одной контекстуальной ПК класса «Объект» [76]:

$$\overline{w_{sp}}(5{,}1, u_{sp}) = \tag{2.74}$$

$$(a_{61}, \dots, a_{6q})_{u_{sp}} \& (a_{5l}, a_{61}) \& a_{5l}, a_{62}) \& \dots \& (a_{5l}, a_{6q}) |$$

$$a_{61}, \dots, a_{6q} \in A_6(n) \ \&$$

$$a_{5l} \in A_5(n) \ \&$$

$$(a_{5l}, a_{61}), \dots, (a_{5l}, a_{6q}) \in LA_5(n)A_6(n) \ \&$$

$$(a_{5l}, a_{61,} a_{7m}), \dots, (a_{5l}, a_{6q}, a_{7k}) \in HA_5(n)A_6(n)A_7(n)$$

ПЗ-1 для типа 5.2 отражает ограничения на взаимосвязь структурных ПК класса «Признак», связанных с разными, но однородными контекстуальными ПК класса «Объект» в рамках одной ПК класса «Компонент» [76]:

$$\overline{w_{sp}}(5{,}2, u_{sp}) = (a_{61}, \dots, a_{6q})_{u_{sp}} \& \ (a_{5l}, a_{61}) \& (a_{4r}, a_{5l}) \& \tag{2.75}$$

$$\& a_{5(l-1)}, a_{62}) \& (a_{4r}, a_{5(l-1)}) \&$$

…

$$\&(a_{5t}, a_{6q})\&(a_{4r}, a_{51})|$$

$$a_{61}, \dots, a_{6q} \in A_6(n)\ \&$$

$$a_{5l}, \dots, a_{5t} \in A_5(n)\ \&$$

$$a_{4r} \in A_4(n)\ \&$$

$$(a_{5l}, a_{61}), \dots, (a_{5t}, a_{6q}) \in LA_5(n)A_6(n)\ \&$$

$$(a_{4r}, a_{5l}), \dots, (a_{4r}, a_{5t}) \in LA_4(n)A_5(n)\ \&$$

$$(a_{5l}, a_{61,} a_{7m}), \dots, (a_{5t}, a_{6q}, a_{7n}) \in HA_5(n)A_6(n)A_7(n)\ \&$$

$$(a_{3k}, a_{4r}, a_{5l}), \dots, (a_{3k}, a_{4r}, a_{5t})\ \in HA_3(n)A_4(n)A_5(n).$$

ПЗ-1 для типа 3.1 или типа 5.3 отражает ограничения на взаимосвязь структурных ПК класса «Признак», связанных с разными и разнородными контекстуальными ПК класса «Объект», принадлежащих различным ПК класса «Компонент» в рамках одной ПК класса «Задача» [76]:

$$\overline{w_{sp}}(3{,}1, u_{sp}) = (a_{61}, \dots, a_{6q})_{u_{sp}}\& \tag{2.76}$$

$$(a_{5l}, a_{61})\&(a_{4r}, a_{5l})\&(a_{3k}, a_{4r})\&$$

$$a_{5(l-1)}, a_{62})\&(a_{4(r-1)}, a_{5(l-1)})\&(a_{3k}, a_{4(r-1)})\&$$

$$\dots$$

$$\&(a_{5t}, a_{6q})\&(a_{4(r-f)}, a_{5t})\&(a_{3k}, a_{4(r-f)})|$$

$$a_{61}, \dots, a_{6q} \in A_6(n)\ \&$$

$$a_{5l}, \dots, a_{5t} \in A_5(n)\ \&$$

$$a_{4(r-f)}, \dots, a_{4r} \in A_4(n)\ \&$$

$$a_{3k} \in A_3(n)\ \&$$

$$(a_{5l}, a_{61}), \dots, (a_{5t}, a_{6q}) \in LA_5(n)A_6(n)\ \&$$

$$(a_{4r}, a_{5l}), \dots, (a_{4(r-f)}, a_{5t}) \in LA_4(n)A_5(n)\ \&$$

$$(a_{3k}, a_{4r}), \dots, (a_{3k}, a_{4(r-f)}) \in LA_3(n)A_4(n)\ \&$$

$$(a_{5l}, a_{61,} a_{7m}), \dots, (a_{5t}, a_{6q}, a_{7n}) \in HA_5(n)A_6(n)A_7(n)\ \&$$

$$(a_{3k}, a_{4r}, a_{5l}), \dots, (a_{3k}, a_{4(r-f)}, a_{5t})\ \in HA_3(n)A_4(n)A_5(n).$$

ПЗ-1 для типа 3.2 отражает ограничения на взаимосвязь структурных ПК класса «Признак», связанных с разными и разнородными ПК класса «Объект», принадлежащих различным ПК класса «Компонент» разных ПК класса «Задача» в рамках одной ПК класса «Процесс» [76]:

$$\overline{w_{sp}}(3{,}2, u_{sp}) = \tag{2.77}$$

$$= (a_{61}, \dots, a_{6q})_{u_{sp}} \&(a_{5l}, a_{61})\&(a_{4r}, a_{5l})\&(a_{3k}, a_{4r})\&(a_{2p}, a_{3k})\&$$

$$\dots$$

$$(a_{5g}, a_{6q})\&(a_{4(r-f)}, a_{5g})\&(a_{3n}, a_{4(r-f)})\&(a_{2p}, a_{3n})|$$

$$a_{61}, \dots, a_{6q} \in A_6(n) \ \&$$

$$a_{5l}, \dots, a_{5g} \in A_5(n) \ \&$$

$$a_{4(r-f)}, \dots, a_{4r} \in A_4(n) \ \&$$

$$a_{3k}, \dots, a_{3n} \in A_3(n) \ \&$$

$$a_{2p} \in A_2(n) \ \&$$

$$(a_{5l}, a_{61}), \dots, (a_{5g}, a_{6q}) \in LA_5(n)A_6(n) \ \&$$

$$(a_{4r}, a_{5l}), \dots, (a_{4(r-f)}, a_{5g}) \in LA_4(n)A_5(n) \ \&$$

$$(a_{3k}, a_{4r}), \dots, (a_{3n}, a_{4(r-f)}) \in LA_3(n)A_4(n) \ \&$$

$$(a_{2p}, a_{3k}), \dots, (a_{2p}, a_{3n}) \in LA_2(n)A_3(n) \ \&$$

$$(a_{5l}, a_{61,} a_{7m}), \dots, (a_{5g}, a_{6q}, a_{7n}) \in HA_5(n)A_6(n)A_7(n) \ \&$$

$$(a_{3k}, a_{4r}, a_{5l}), \dots, (a_{3n}, a_{4(r-f)}, a_{5g}) \in HA_3(n)A_4(n)A_5(n) \ \&$$

$$(a_{1s}, a_{2p}, a_{3k}), \dots, (a_{1s}, a_{2p}, a_{3n}) \in HA_1(n)A_2(n)A_3(n).$$

ПЗ-1 для типа 1.1 или типа 3.3 отражает ограничения на взаимосвязь структурных ПК класса «Признак», связанных с разными и разнородными ПК класса «Объект», принадлежащих различным ПК класса «Компонент» разных ПК класса «Задача» различных ПК класса «Процесс» в рамках одной ПК класса «Цикл» [76]:

$$\overline{w_{sp}}(1{,}1, u_{sp}) = (a_{61}, \dots, a_{6q})_{u_{sp}} \& \tag{2.78}$$

$$(a_{5l}, a_{61})\&(a_{4r}, a_{5l})\&(a_{3k}, a_{4r})\&(a_{2p}, a_{3k})\& \ (a_{1s}, a_{2p})\&$$

$$\dots$$

$$(a_{5g}, a_{6q})\&\big(a_{4(r-f)}, a_{5g}\big)\&\big(a_{3n}, a_{4(r-f)}\big)\&(a_{2t}, a_{3n})\&(a_{1s}, a_{2t})|$$

$$a_{61}, \dots, a_{6q} \in A_6(n)\ \&$$

$$a_{5l}, \dots, a_{5g} \in A_5(n)\ \&$$

$$a_{4(r-f)}, \dots, a_{4r} \in A_4(n)\ \&$$

$$a_{3k}, \dots, a_{3n} \in A_3(n)\ \&$$

$$a_{2p}, \dots, a_{2t} \in A_2(n)\ \&$$

$$a_{1s} \in A_1(n)\ \&$$

$$(a_{5l}, a_{61}), \dots, \big(a_{5g}, a_{6q}\big) \in LA_5(n)A_6(n)\ \&$$

$$(a_{4r}, a_{5l}), \dots, \big(\ a_{4(r-f)}, a_{5g}\big) \in LA_4(n)A_5(n)\ \ \&$$

$$(a_{3k}, a_{4r}), \dots, \big(\ a_{3n}, a_{4(r-f)}\big) \in LA_3(n)A_4(n)\ \ \&$$

$$\big(a_{2p}, a_{3k}\big), \dots, (\ a_{2t}, a_{3n}) \in LA_2(n)A_3(n)\ \ \&$$

$$\big(a_{1s}, a_{2p}\big), \dots, (\ a_{1s}, a_{2t}) \in LA_1(n)A_2(n)\ \ \&$$

$$\big(a_{5l}, a_{61,}a_{7m}\big), \dots, \big(a_{51}, a_{6q}, a_{7n}\big) \in HA_5(n)A_6(n)A_7(n)\&$$

$$(a_{3k}, a_{4r}, a_{5l}), \dots, (a_{3n}, a_{4(r-f)}, a_{5g})\ \in HA_3(n)A_4(n)A_5(n)\ \ \&$$

$$\big(a_{1s}, a_{2p}, a_{3k}\big), \dots, (a_{1s}, a_{2t}, a_{3n})\ \in HA_1(n)A_2(n)A_3(n).$$

Для геометрической интерпретации ограничений 1-го рода можно использовать следующие понятия: абстрактное пространство, оси абстрактного пространства, метрики или шкалы осей абстрактного пространства [75].

При этом полагается, что оси имеют «разнородные» метрики и любая ось может иметь набор «неравномерных» шкал. Тогда абстрактное пространство будет определяться контекстуальной предметной категорией, его оси – определяются структурными предметными категориями, а метрики этих осей – набором монадических предметных категорий.

На рис. 2.10-2.12 приведена геометрическая интерпретация предметных зависимостей 1-года от1-го типа до 3-го типа [79].

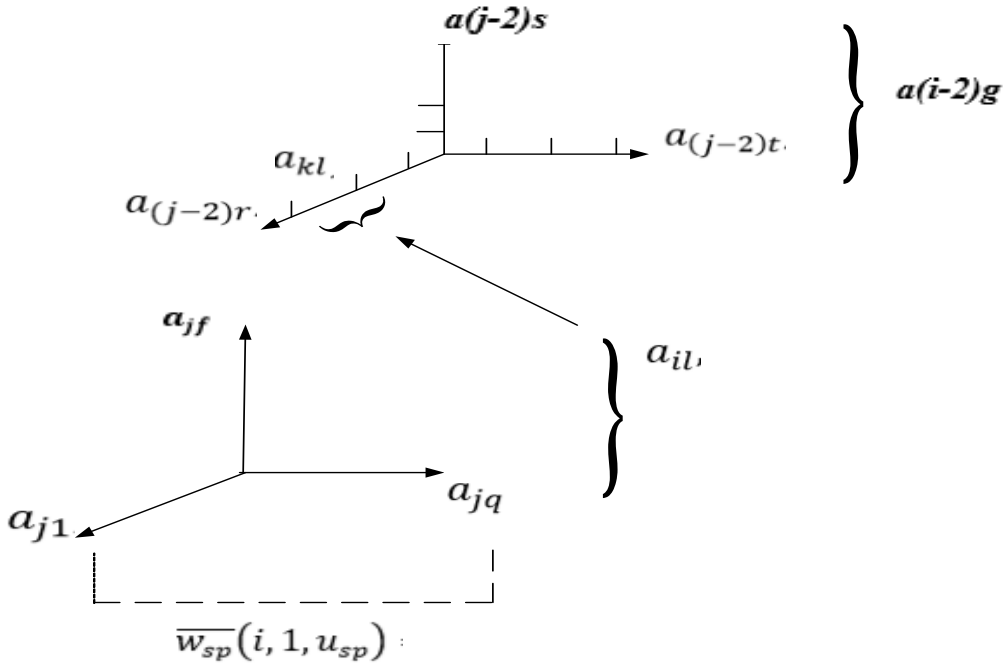


Рис.2.10. Геометрическая интерпретация предметной зависимости 1-го рода 1-го типа, $\overline{w_{sp}}(1,1,u_{sp})$ – кривая в $\boldsymbol{\alpha}$ – мерном пространстве, где a_{il} - определяет абстрактное пространство i-го уровня семантической сложности, $a_{j1},\dots,a_{jq},a_{(j-2)r},a_{(j-2)t}$ - определяют оси пространств разного уровня сложности, a_{kl} – определяет фрагмент шкалы на оси [79].

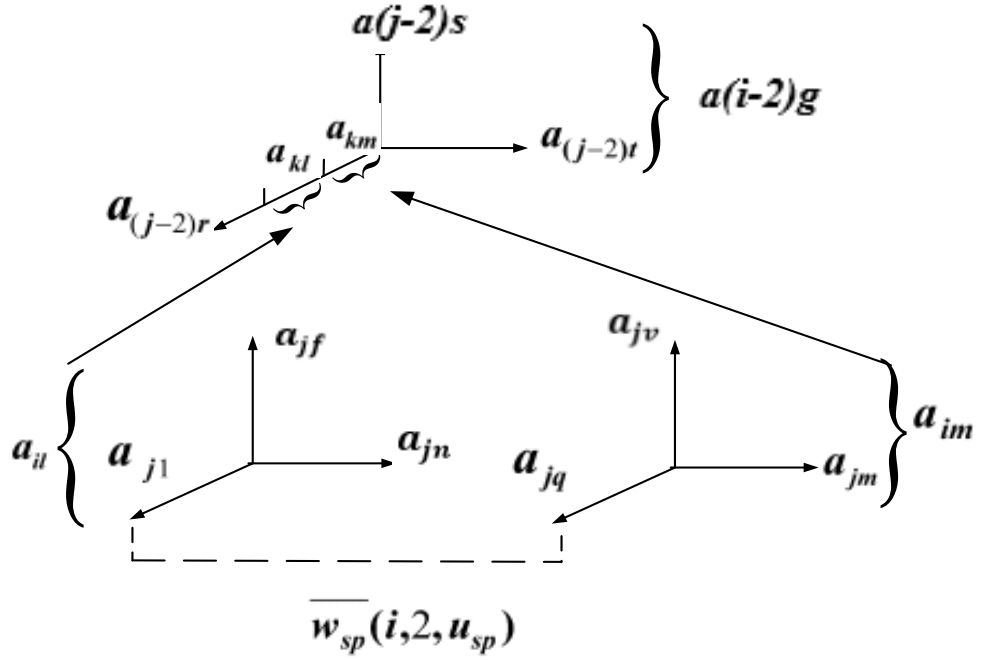


Рис. 2.11. Геометрическая интерпретация предметной зависимости 1-го рода 2-го типа, где a_{il},a_{im} - определяют абстрактные пространства, $a_{j1},\dots,a_{jq},a_{jm},a_{(j-2)r},a_{(j-2)t}$ - определяют оси пространств разного уровня семантической сложности, a_{kl},a_{km} – определяют фрагменты шкал на оси [74,79].

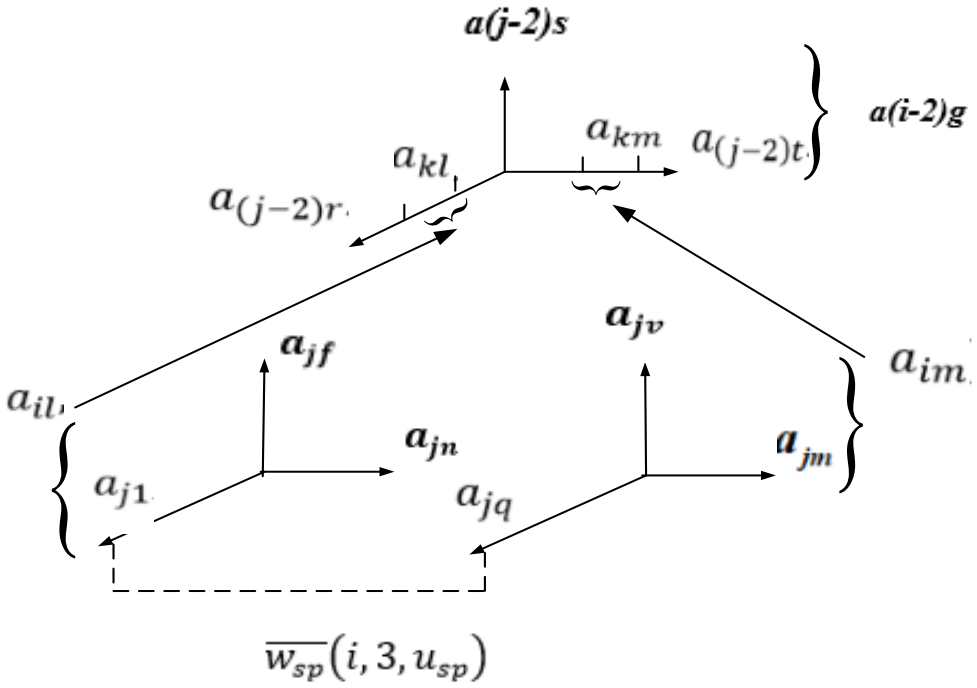


Рис.2.12. Геометрическая интерпретация предметной зависимости 1-го рода 3-го типа где a_{il}, a_{im} - определяют абстрактные пространства, $a_{j1}, \dots, a_{jq}, a_{jm}, a_{(j-2)r}, a_{(j-2)t}$ - определяют оси пространств разного уровня семантической сложности, a_{kl}, a_{km} – определяют фрагменты шкал на разных осях [74,79].

2.3.3 Формальное описание типологии предметных ограничений 1-го рода на конкретном уровне

ЭПЗ-1 для типа 5.1 отражает ограничения на взаимосвязь структурных ЭПК класса «Признак» в рамках одной контекстуальной ЭПК класса «Объект»:

$$\overline{p_r^{sp}}(5{,}1, s, u_{spr}) = \left(b_x^{61}, \dots, b_z^{6q} \right)_{u_{spr}} \&(b_r^{5l}, b_x^{61}) \& \dots \& \left(b_r^{5l}, b_z^{6q} \right) | \qquad (2.79)$$

$$b_x^{61} \in B_{61}(nm) \ \ \& \dots b_z^{6q} \in B_{6q}(nm) \ \ \&$$

$$b_r^{5l} \in B_{5l}(nm) \ \ \&$$

$$(b_r^{5l}, b_x^{61}) \in LB_{5l}(nm)B_{61}(nm) \ \ \& \dots$$

$$\left(b_r^{5l}, b_z^{6q} \right) \in LB_{5l}(nm)B_{6q}(nm) \ \ \&$$

$$(b_r^{5l}, b_x^{61}, b_t^{7m}) \in LB_{5l}(nm)B_{61}(nm)B_{7m}(nm) \ \ \& \dots$$

$$\left(b_r^{5l}, b_z^{6q}, b_r^{7n}\right) \in LB_{5l}(nm)B_{6q}(nm)B_{7n}(nm).$$

ЭПЗ-1 для типа 5.2 отражает ограничения на взаимосвязь структурных ЭПК класса «Признак», связанных с разными, но однородными контекстуальными ЭПК класса «Объект» в рамках одной ЭПК класса «Компонент»:

$$\overline{p_r^{sp}}(5{,}2, s, u_{spr}) = \left(\ b_x^{61}, \dots, b_z^{6q}\right)_{u_{spr}} \&(b_r^{5l}, b_x^{61}) \& (b_q^{4r}, b_r^{5l}) \& \tag{2.80}$$

...

$$(\ b_s^{5p}, b_z^{6q}) \& (b_q^{4r}, b_s^{5p})|$$

$$b_x^{61} \in B_{61}(nm) \ \ \& \dots b_z^{6q} \in B_{6q}(nm) \ \ \&$$

$$b_r^{5l} \in B_{5l}(nm) \ \ \& \dots b_s^{5p} \in B_{5p}(nm) \ \ \&$$

$$b_q^{4r} \in B_{4r}(nm) \ \ \&$$

$$(b_r^{5l}, b_x^{61}) \in LB_{5l}(nm)B_{61}(nm) \ \ \& \dots$$

$$\left(\ b_s^{5p}, b_z^{6q}\right) \in LB_{5p}(nm)B_{6q}(nm) \ \ \&$$

$$\left(b_q^{4r}, b_r^{5l}\right) \in LB_{4r}(nm)B_{5l}(nm) \ \ \& \dots$$

$$\left(b_q^{4r}, b_s^{5p}\right) \in LB_{4r}(nm)B_{5p}(nm) \ \ \&$$

$$(b_r^{5l}, b_x^{61}, b_t^{7m}) \in LB_{5l}(nm)B_{61}(nm)B_{7m}(nm) \ \ \& \dots$$

$$\left(b_s^{5p}, b_z^{6q}, b_s^{7n}\right) \in LB_{5p}(nm)B_{6q}(nm)B_{7n}(nm) \ \ \&$$

$$\left(b_p^{3k}, b_q^{4r} b_r^{5l}\right) \in LB_{3k}(nm)B_{4r}(nm)B_{5l}(nm) \ \ \& \dots$$

$$\left(b_p^{3k}, b_q^{4r} b_s^{5p},\right) \in LB_{3k}(nm)B_{4r}(nm)B_{5p}(nm).$$

ЭПЗ-1 для типа 3.1 или типа 5.3 отражает ограничения на взаимосвязь структурных ЭПК класса «Признак», связанных с разными и разнородными контекстуальными ЭПК класса «Объект», принадлежащих различным ЭПК класса «Компонент» в рамках одной ЭПК класса «Задача»:

$$\overline{p_r^{sp}}(3{,}1, s, u_{spr}) = \tag{2.81}$$

$$= \left(\ b_x^{61}, \dots, b_z^{6q}\right)_{u_{spr}} \&(b_r^{5l}, b_x^{61}) \&(b_q^{4r}, b_r^{5l}) \&\left(b_p^{3k}, b_q^{4r}\right) \&$$

...

$$(\ b_s^{5p}, b_z^{6q}) \&(b_n^{4(r-f)}, b_s^{5p}) \& (b_p^{3k}, b_n^{4(r-f)})|$$

$$b_x^{61} \in B_{61}(nm) \ \ \& \dots b_z^{6q} \in B_{6q}(nm) \ \ \&$$

$$b_r^{5l} \in B_{5l}(nm) \ \ \& \dots b_s^{5p} \in B_{5p}(nm) \ \ \&$$

$$b_n^{4(r-f)} \in B_{4(r-f)}(nm) \ \ \& \dots b_q^{4r} \in B_{4r}(nm) \ \ \&$$

$$b_p^{3k} \in B_3(nm) \ \ \&$$

$$(b_r^{5l}, b_x^{61}) \in LB_{5l}(nm)B_{61}(nm) \ \ \& \dots$$

$$\left(b_s^{5p}, b_z^{6q} \right) \in LB_{5p}(nm)B_{6q}(nm) \ \ \&$$

$$\left(b_q^{4r}, b_p^{5l}\right) \in LB_{4r}(nm)B_{5l}(nm) \ \ \& \dots$$

$$\left(b_q^{4(r-f)}, b_s^{5p}\right) \in LB_{4(r-f)}(nm)B_{5p}(nm) \ \ \&$$

$$\left(b_p^{3k}, b_q^{4r}\right) \in LB_{3k}(nm)B_{4r}(nm) \ \ \& \dots$$

$$(b_p^{3k}, b_n^{4(r-f)}) \in LB_{3k}(nm)B_{4(r-f)}(nm) \ \ \&$$

$$(b_r^{5l}, b_z^{61}, b_t^{7m}) \in LB_{5l}(nm)B_{61}(nm)B_{7m}(nm) \ \ \& \dots$$

$$\left(b_s^{5p}, b_z^{6q}, b_s^{7n}\right) \in LB_{5p}(nm)B_{6q}(nm)B_{7n}(nm) \ \ \&$$

$$\left(b_p^{3k}, b_q^{4r}, b_r^{5l}\right) \in LB_{3k}(nm)B_{4r}(nm)B_{5l}(nm) \ \ \& \dots$$

$$\left(b_p^{3k}, b_q^{4(r-f)}, b_s^{5p}\right) \in LB_{3k}(nm)B_{4(r-f)}(nm)B_{5p}(nm).$$

ЭПЗ-1 для типа 3.2 отражает ограничения на взаимосвязь структурных ЭПК класса «Признак», связанных с разными и разнородными ЭПК класса «Объект», принадлежащих различным ЭПК класса «Компонент» разных ЭПК класса «Задача» в рамках одной ЭПК класса «Процесс»:

$$\overline{p_r^{sp}}(3{,}2, s, u_{spr}) = \left(b_x^{61}, \dots, b_z^{6q} \right)_{u_{spr}} \& \tag{2.82}$$

$$(b_r^{5l}, b_x^{61}) \& \left(b_q^{4r}, b_r^{5l}\right) \& \left(b_p^{3k}, b_q^{4r}\right) \& \left(b_v^{2p}, b_p^{3k}\right) \ \&$$

$$\dots$$

$$\&\left(b_s^{5p}, b_z^{6q}\right) \& (b_n^{4(r-f)}, b_s^{5p}) \& \ (b_q^{3n}, b_n^{4(r-f)}) \& (b_v^{2p}, b_q^{3n}) |$$

$$b_x^{61} \in B_{61}(nm) \ \ \& \dots b_z^{6q} \in B_{6q}(nm) \ \ \&$$

$$b_r^{5l} \in B_{5l}(nm) \ \ \& \dots b_s^{5p} \in B_{5p}(nm) \ \ \&$$

$$b_n^{4(r-f)} \in B_{4(r-f)}(nm) \ \ \& \dots b_q^{4r} \in B_{4r}(nm) \ \ \&$$

$$b_p^{3k} \in B_{3k}(nm) \ \ \& \dots b_q^{3n} \in B_{3n}(nm) \ \ \&$$

$$b_v^{2p} \in B_{2p}(nm) \ \ \&$$

$$(b_r^{5l}, b_x^{61}) \in LB_{5l}(nm)B_{61}(nm) \ \ \& \dots$$

$$\left(b_s^{5p}, b_z^{6q}\right) \in LB_{5p}(nm)B_{6q}(nm) \ \ \&$$

$$\left(b_q^{4r}, b_r^{5l}\right) \in LB_{4r}(nm)B_{5l}(nm) \ \ \& \dots$$

$$\left(b_n^{4(r-f)}, b_s^{5p}\right) \in LB_{4(r-f)}(nm)B_{5p}(nm) \ \ \&$$

$$\left(b_p^{3k}, b_q^{4r}\right) \in LB_{3k}(nm)B_{4r}(nm) \ \ \& \dots$$

$$(b_p^{3k}, b_n^{4(r-f)}) \in LB_{3k}(nm)B_{4(r-f)}(nm) \ \ \&$$

$$\left(b_v^{2p}, b_p^{3k}\right) \in LB_{2p}(nm)B_{3k}(nm) \ \ \& \dots$$

$$(b_v^{2p}, b_q^{3n}) \ \in LB_{2p}(nm)B_{3n}(nm) \ \ \&$$

$$(b_r^{5l}, b_z^{61}, b_t^{7m}) \in LB_{5l}(nm)B_{61}(nm)B_{7m}(nm) \ \ \& \dots$$

$$\left(b_s^{5p}, b_z^{6q}, b_s^{7n}\right) \in LB_{5p}(nm)B_{6q}(nm)B_{7n}(nm) \ \ \&$$

$$\left(b_p^{3k}, b_q^{4r}, b_r^{5l}\right) \in LB_{3k}(nm)B_{4r}(nm)B_{5l}(nm) \ \ \& \dots$$

$$\left(b_q^{3n}, b_n^{4(r-f)}, b_s^{5p}, \right) \in LB_{3n}(nm)B_{4(r-f)}(nm)B_{5p}(nm) \ \ \&$$

$$\left(b_v^{1s}, b_g^{2p}, b_p^{3k}\right) \in LB_{1s}(nm)B_{2p}(nm)B_{3k}(nm) \ \ \& \dots$$

$$\left(b_v^{1s}, b_g^{2p}, b_q^{3n}\right) \in LB_{1s}(nm)B_{2p}(nm)B_{3n}(nm).$$

ЭПЗ-1 для типа 1.1 или типа 3.3 отражает ограничения на взаимосвязь структурных ЭПК класса «Признак», связанных с разными и разнородными ЭПК класса «Объект», принадлежащих различным ЭПК класса «Компонент» разных ЭПК класса «Задача» различных ЭПК класса «Процесс» в рамках одной ЭПК класса «Цикл»:

$$\overline{p_r^{sp}}(1,1,s,u_{spr}) = \left(b_x^{61}, \dots, b_z^{6q}\right)_{u_{spr}} \& \tag{2.83}$$

$$(b_r^{5l}, b_x^{61})\&(b_q^{4r}, b_r^{5l})\&\left(b_p^{3k}, b_q^{4r}\right)\&\left(b_v^{2p}, b_p^{3k}\right)\&\left(b_g^{1s}, b_v^{2p}\right)\&$$

$$\dots$$

$$\&\left(b_s^{5p}, b_z^{6q}\right)\&(b_n^{4(r-f)}, b_s^{5p})\& \ (b_q^{3n}, b_n^{4(r-f)})\&(b_y^{2t}, b_q^{3n})\&\left(b_g^{1s}, b_y^{2t}\right)|$$

$$b_x^{61} \in B_{61}(nm) \ \& \ldots \ \& \ b_z^{6q} \in B_{6q}(nm) \ \&$$

$$b_r^{5l} \in B_{5l}(nm) \ \& \ . \ . \ . \ \& \quad b_s^{5p} \in B_{5p}(nm) \ \&$$

$$b_n^{4(r-f)} \in B_{4(r-f)}(nm) \ \& . \ . \ . \ \& \quad b_q^{4r} \in B_{4r}(nm) \ \&$$

$$b_p^{3k} \in B_{3k}(nm) \ \& . \ . \ . \ \& \quad b_q^{3n} \in B_{3n}(nm) \ \&$$

$$b_v^{2p} \in B_{2v}(nm) \ \& . \ . \ . \ \& \quad b_y^{2t} \in B_{2t}(nm) \ \&$$

$$b_g^{1s} \in B_{1s}(nm) \ \&$$

$$(b_r^{5l}, b_x^{61}) \in LB_{5l}(nm)B_{61}(nm)\& \ldots.$$

$$(\ b_s^{5p}, b_z^{6q}) \in LB_{5p}(nm)B_{6q}(nm)\&$$

$$\left(b_q^{4r}, b_r^{5l}\right) \in LB_{4r}(nm)B_{5l}(nm) \ \& \ldots..$$

$$\left(b_n^{4(r-f)}, b_s^{5p}\right) \in LB_{4(r-f)}(nm)B_{5p}(nm) \ \&$$

$$\left(b_p^{3k}, b_q^{4r}\right) \in LB_{3k}(nm)B_{4r}(nm) \ \& \ . \ . \ .$$

$$(b_q^{3n}, b_n^{4(r-f)}) \in LB_{3n}(nm)B_{4(r-f)}(nm) \ \&$$

$$\left(b_v^{2p}, b_p^{3k}\right) \in LB_{2p}(nm)B_{3k}(nm) \ \& \ldots$$

$$(b_y^{2t}, b_q^{3n}) \ \in LB_{2t}(nm)B_{3n}(nm) \ \&$$

$$\left(b_g^{1s}, b_v^{2p}\right) \in LB_{1s}(nm)B_{2p}(nm) \ \& \ldots$$

$$\left(b_g^{1s}, b_y^{2t}\right) \in LB_{1s}(nm)B_{2t}(nm) \ \&$$

$$(b_r^{5l}, b_z^{61}, b_t^{7m}), \in LB_{5l}(nm)B_{61}(nm)B_{7m}(nm) \ \& \ldots$$

$$\left(b_s^{5p}, b_z^{6q}, b_s^{7n}\right) \in LB_{5p}(nm)B_{6q}(nm)B_{7n}(nm) \ \&$$

$$\left(b_p^{3k}, b_q^{4r}, b_r^{5l}\right) \in LB_{3k}(nm)B_{4r}(nm)B_{5l}(nm) \ \& \ldots$$

$$\left(b_q^{3n}, b_n^{4(r-f)}, b_s^{5p}, \right) \in LB_{3n}(nm)B_{4(r-f)}(nm)B_{5p}(nm) \ \&$$

$$\left(b_g^{1s}, b_v^{2p}, b_p^{3k}\right) \in LB_{1s}(nm)B_{2p}(nm)B_{3k}(nm) \ \& \ldots$$

$$\left(b_g^{1s}, b_y^{2t}, b_q^{3n}\right) \in LB_{1s}(nm)B_{2t}(nm)B_{3n}(nm).$$

Для геометрической интерпретации ограничений 1-го рода на конкретном уровне можно использовать следующие понятия: абстрактное пространство как совокупность точек, оси абстрактного пространства, метрики или шкалы осей абстрактного пространства и их значения.

При этом полагается, что оси имеют «разнородные» метрики и любая ось может иметь набор «неравномерных» шкал. Само абстрактное пространство будет определяться контекстуальной предметной категорией (ПК), а его точка – экземпляром этой ПК, оси пространства – определяются структурными ПК и глобальным значением на оси экземпляром этой предметной категории, а метрики этих осей – набором монадических ПК и их экземпляров [79].

На конкретном уровне монадический экземпляр ПК определяет локальную точку на фрагменте оси (или глобально - структурный экземпляр предметной категории).

Совокупность точек на осях определяет точку в абстрактном пространстве (или контекстуальный экземпляр предметной категории).

На рис. 2.13 приведена геометрическая интерпретация экземпляров предметных зависимостей 1-года 1-го типа.

На рис. 2.14-2.15 приведена геометрическая интерпретация экземпляров предметных зависимостей 1-года 2-го типа и 3-го типа.

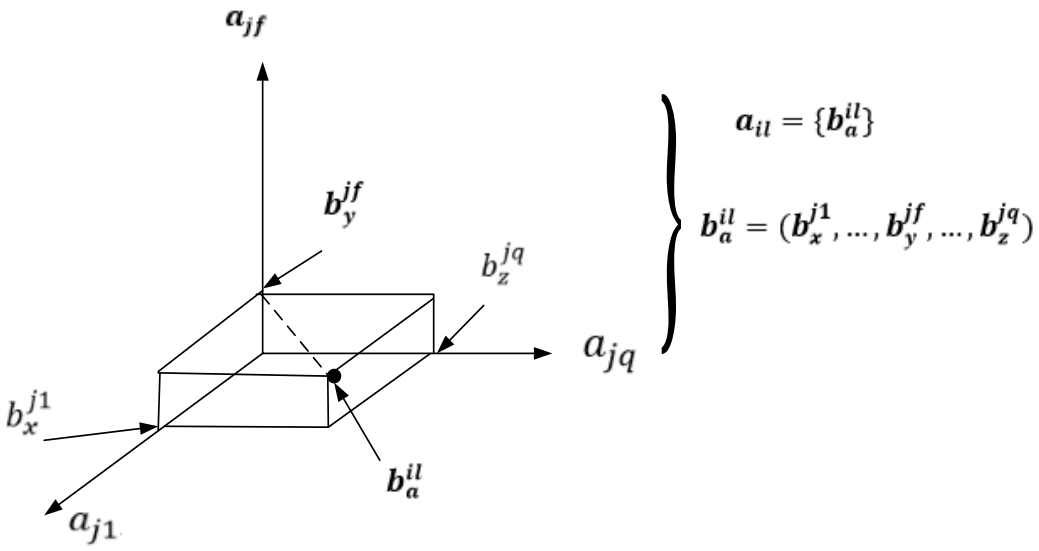


Рис.2.13. Геометрическая интерпретация экземпляров предметной зависимости 1-рода 1-го типа, a_{il}- пространство с осями $a_{j1}, \dots, a_{jf}, \dots, a_{jq}$, b_a^{il} - точка в пространстве a_{il} [79].

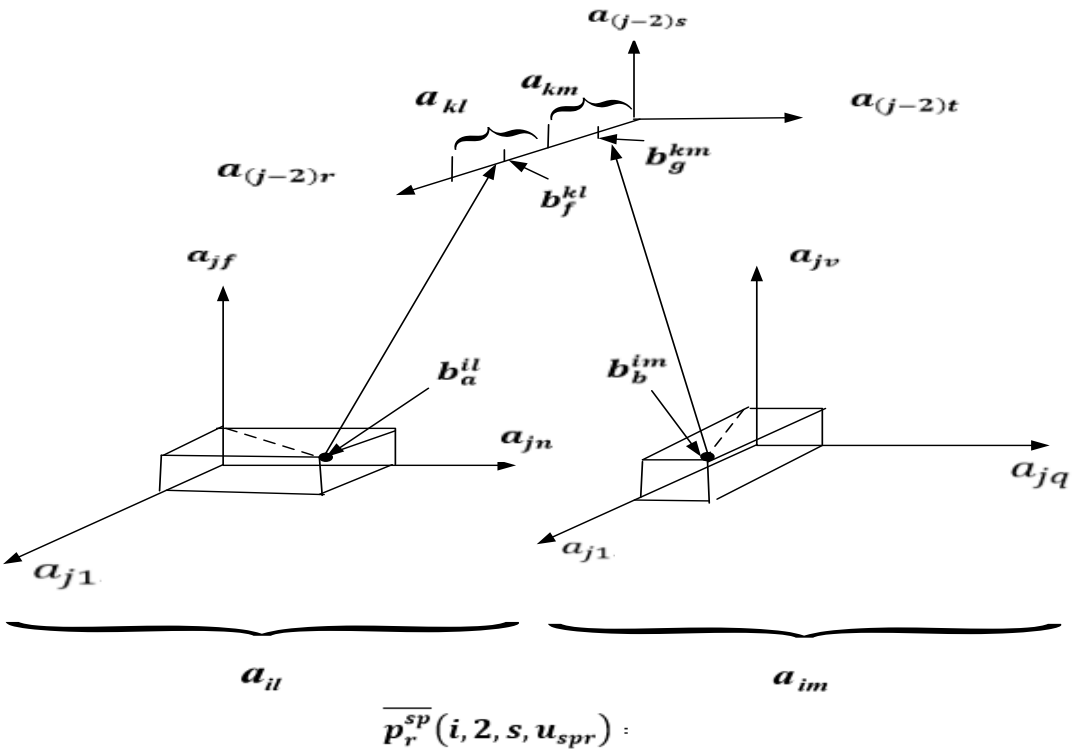


Рис.2.14. Геометрическая интерпретация экземпляра предметной зависимости 1-рода 2-го типа, где b_b^{km}- точка во фрагменте a_{km} оси $a_{(j-2)r}$, b_a^{kl}- точка во фрагменте a_{kl} оси $a_{(j-2)r}$, b_a^{il}- точка в пространстве a_{il}, b_b^{im}- точка в пространстве a_{im} [79].

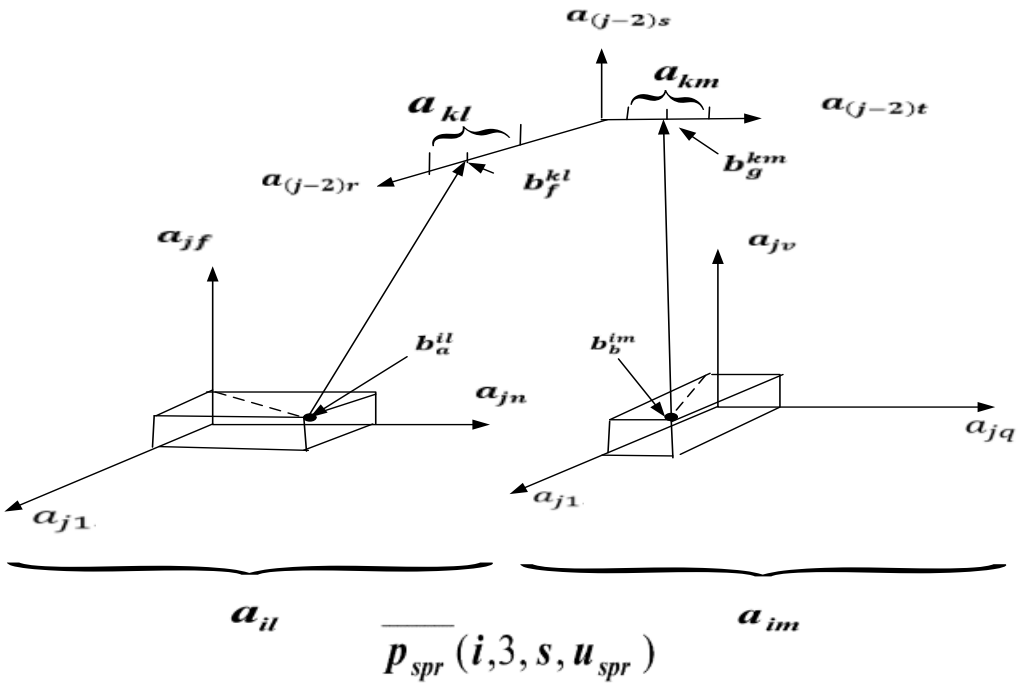


Рис.2.15. Геометрическая интерпретация экземпляров предметной зависимости 1-рода 3-го типа, где b_b^{km}- точка во фрагменте a_{km} оси $a_{(j-2)t}$, b_a^{kl}- точка во фрагменте a_{kl} оси $a_{(j-2)r}$, b_a^{il}- точка в пространстве a_{il}, b_b^{im}- точка в пространстве a_{im} [79].

2.4 Разработка формального описания статистических предметных ограничений 1-го рода

2.4.1 Общая характеристика статистических предметных ограничений 1-го рода

Статистические ПЗ-1 определяют фрагмент концептуальной модели предметной задачи объектного уровня, в котором одна ПЗ-1 связана с фрагментом основной концептуальной структуры.

Исходя из этого, было выполнено формальное описание процедуры расслоения концептуальной модели предметной задачи под переменную и постоянную информацию [3,74,78,79].

Каждая простая ПЗ-1 по степени формализации может быть аналитической, статистической и эмпирической.

Требования к формальному описанию статистических предметных ограничений: выделение и описание категориальных структур на различных уровнях абстрагирования с учетом выделенных категорий предметной области; определение и описание статистических правил и ограничений, действующих в рассматриваемой предметной области; описание взаимосвязей концептуальных конструкций на различных уровнях абстрагирования; представление существенных черт моделируемых процессов и ограничений на них на различных уровнях абстрагирования.

Формальное описание концептуальных представлений позволяет рассмотреть и систематизировать все ограничения, применяемые в задачах проектирования на основе типологий ограничений в единой концептуальной структуре. Данная обработка заданий является основой для их последующей методологической идентификации и выявления при разработке метода выделения фрагментов устойчивой информационной концептуальной модели для проектно-конструкторских задач.

Концептуальная модель 1-го рода любого уровня абстрагирования для n-ой предметной задачи с учетом выполненного процесса расслоения представляется набором фрагментов модели в виде [74,76,79]:

$$KPi.1(z) =< KPi_v.1(z), \overline{KPi_c.1}(z) >, \quad (2.84)$$

где $z = n$ описывает концептуальную модель n-ой предметной задачи объектного уровня $(i = 2), z = nm$ - концептуальную модель m-ой реализации n-ой предметной задачи конкретного уровня ($i = 3$).

Рассмотрим эти наборы более подробно.

Тогда $KP2_v.1(n)$ - фрагмент концептуальной модели 1-го рода под переменную информацию; $\overline{KP2_c.1}(n)$ – множество фрагментов концептуальной модели 1-го рода под постоянную информацию для объектного уровня.

$$\overline{KP2_c.1}(n) = \{KP2_c^w.1(n)\} \quad (2.85)$$

где $KP2_c^w.1(n)$- множество фрагментов концептуальной модели 1-го рода под постоянную информацию на объектном уровне абстрагирования; индекс w - идентифицирует статистическую предметную зависимость 1-го рода [79].

Для конкретного уровня $KP3_v.1(nm)$ - фрагмент концептуальной модели 1-го рода под переменную информацию; $\overline{KP3_c.1}(nm)$ – множество фрагментов концептуальной модели 1-го рода под постоянную информацию для конкретного уровня.

$$\overline{KP3_c.1}(nm) = \{KP3_c^{wp}.1(nm)\} \quad (2.86)$$

где $KP3_c^{wp}.1(nm)$- фрагмент концептуальной модели 1-го рода под постоянную информацию на конкретном уровне абстрагирования; индекс wp - идентифицирует экземпляр статистический предметной зависимости 1-го рода.

2.4.2 Формальное описание статистических предметных ограничений 1-го рода

Расслоение предметных зависимостей 1-го рода на простые (***el***) и сложные (***com***) формально можно представить [79]:

$$\overline{W}(n) = \overline{W^{el}}(n) \cup \overline{W^{com}}(n) \quad (2.87)$$

Все простые предметные зависимости 1-го рода по степени формализации можно классифицировать на: аналитические (*an*), статистические (*st*), эмпирические (*em*). Формально это можно представить [74,79]:

$$\overline{W^{el}}(n) = \overline{W^{el-an}}(n) \cup \overline{W^{el-st}}(n) \cup \overline{W^{el-em}}(n) \quad (2.88)$$

Поскольку типология зависимостей обусловлена уровнем семантической сложности, то необходимо для предметных зависимостей определять ее тип с учетом уровня сложности.

Статистические предметные зависимости 1-го рода соответственно можно классифицировать по типам и по содержанию (по общим аргументам).

Фрагмент концептуальной модели 1-го рода под постоянную информацию для выделенной статистической предметной зависимости 1-го рода на объектном уровне может быть представлен следующим образом [74,76,79]:

$$KP2_c^{sp}.1(n) =< A^{sp}(n), T2^{sp}(n), \overline{w_{sp}}, R_2^{KP1-\overline{w_{sp}}} > \quad (2.89)$$

Где $A^{sp}(n)$- множество предметных категорий, характеризующих ПЗ 1-го рода; $T2^{sp}(n)$-множество бинарных отношений на предметных категориях, характеризующих sp -ую ПЗ 1 -го рода; $\overline{w_{sp}}$ - статистическая предметная зависимость 1-го рода, $R_2^{KP1-\overline{w_{sp}}}$- увязка ПЗ-1 и основной концептуальной структуры.

Экземпляры статистических предметных зависимостей 1-го рода на простые (*el*) и сложные (*com*) формально можно представить [79]:

$$\bar{P}(n) = \overline{P^{el}}(n) \cup \overline{P^{com}}(n) \quad (2.90)$$

Все предметные зависимости по степени формализации можно классифицировать на: аналитические (*an*), статистические (*st*), эмпирические (*em*). Формально это можно представить [74,79]:

$$\overline{P^{el}}(n) = \overline{P^{el-an}}(n) \cup \overline{P^{el-st}}(n) \cup \overline{P^{el-em}}(n) \quad (2.91)$$

Экземпляры статистических предметных зависимостей 1-го рода соответственно можно классифицировать по типам и по содержанию (по общим аргументам и по их значениям).

Фрагмент концептуальной модели под постоянную информацию для выделенного статистического экземпляра предметной зависимости 1-го рода на конкретном уровне может быть представлен следующим образом [74,76,79]:

$$KP3_c^{spl}.1(nm) =< B^{spl}(nm), T3^{spl}(nm), \overline{p_l^{sp}}, R_3^{KP2-\overline{p_l^{sp}}} > \quad (2.92)$$

где $B^{spl}(nm)$ - множество экземпляров предметных категорий, характеризующих $spl-$ый ЭПЗ 1-го рода; $T3^{spl}(nm)$ - множество бинарных отношений на ЭПК, характеризующих spl –ый ЭПЗ 1-го рода; $\overline{p_l^{sp}}$- статистический экземпляр предметной зависимости 1-го рода, $R_3^{KP2-\overline{p_l^{sp}}}$ - увязка ЭПЗ-1 и концептуальной структуры на конкретном уровне.

Увязка фрагментов концептуальной модели под постоянную информацию для выделенной статистической предметной зависимости 1-го рода (sp) и для выделенного статистического экземпляра предметной зависимости 1-го рода (spl) [74,76,79].

$$KP2_c^{sp}.1(n) \to \{KP3_c^{spl}.1(nm)\} \quad (2.93)$$

2.5 Выводы по главе 2

1. Выполнено уточнение формального описания концептуального представления предметных задач, позволяющего разделить концептуальное представление на модели 1-го и 2-го родов.

2. Разработан метод обработки статистических предметных ограничений 1-го рода при концептуальном моделировании прикладных задач, включающий формальное описание концептуальных моделей 1-го рода на объектном и конкретном уровнях абстрагирования и их взаимосвязи; формальное описание

расслоения концептуальных моделей 1-го рода под переменную и постоянную информацию; формальное описание типологии статистических предметных зависимостей 1-го рода и их геометрическую интерпретацию.

3. Установлены связи между характеристиками моделей статистических (таблично оформленных) знаний и их аналитической интерпретацией.

4. Разработанный метод позволяет систематизировать и классифицировать таблично оформленные знания, зафиксированные в технических нормативно-справочных документах машиностроительных организаций.

5. Это служит основой для их возможной идентификации и описания при создании методических основ для формирования статистических предметных ограничений при концептуальном моделировании прикладных задач.

ГЛАВА 3. РАЗРАБОТКА МЕТОДИЧЕСКИХ ОСНОВ ВЫДЕЛЕНИЯ, ОПИСАНИЯ И ОБРАБОТКИ СТАТИСТИЧЕСКИХ ПРЕДМЕТНЫХ ОГРАНИЧЕНИЙ 1-ГО РОДА ПРИ КОНЦЕПТУАЛЬНОМ МОДЕЛИРОВАНИИ

3.1 Общая характеристика процесса концептуального моделирования

Под концептуальным моделированием будем понимать процесс объективирования (выявления) системы знаний предметной области выделенной прикладной задачи и фиксации полученных результатов в определенной форме. Структура процесса концептуального моделирования представлена на рис.3.1 и 3.2. Процесс концептуального моделирования включает следующие основные процедуры [3,82]:

- формирование обобщенной модели (под комплекс предметных задач);
- формирование модели текущей предметной задачи;
- анализ модели текущей задачи;
- дополнительная обработка модели текущей задачи;
- синтез модели текущей задачи с обобщенной (начальной) моделью;
- дополнительная обработка обобщенной модели комплекса задач;
- документирование модели текущего комплекса задач.

Исходной информацией для концептуального моделирования предметных задач являются результаты предыдущего этапа, а именно начального модельного представления проектно-конструкторской задачи, включающего [3,82]:

- диаграммы и спецификации декомпозиции комплекса предметных задач, диаграмма и спецификации декомпозиции предметной задачи комплекса (для каждой задачи);
- классификацию информации для комплекса и для каждой задачи;
- спецификации параметров (для каждой задачи);

- спецификации элементарных действий и потоков данных (для каждой задачи);
- матричную диаграмму взаимосвязей действий и параметров (для каждой задачи).

Концептуальное моделирование предметных задач выполняет специалист представлению знаний (инженер-когнитолог) [3].

На этапе формирования концептуальных моделей (обобщенной и текущей) предметных задач выполняется последовательно формирование систем предметных ограничений 1-го рода, основных концептуальных структуры, формирование концептуальной модели в целом. Основанием для формирования концептуальной модели предметной задачи является формальное описание концептуального представления предметной задачи (на объектном уровне).

На этапе анализа концептуальных моделей выполняется проверка структур и модели на полноту и связность, анализ правильности их описания, наличие в них противоречий и т.д., по результатам анализа вырабатываются рекомендации по коррекции модели. При отсутствии ошибок данная версия модели является окончательной и подлежит дальнейшей доработке.

На этапе дополнительной обработки концептуальной (обобщенной и локальной) модели выполняются процедуры определения типологии для каждой зависимости и расслоения модели на две части: под постоянную и переменную информацию.

На этапе синтез концептуальных моделей выполняется интеграция или объединение обобщенной и текущей моделей по виду информации (постоянная, переменная), по компонентам модели (предметным категориям, их структурам, предметным зависимостям и их структурам), по моделям в целом.

На этапе документирования моделей выполняется визуализация результатов моделирования в виде отчетов (полных или сокращенных) по каждой составляющей модели.

Основанием синтеза является формальное описание интеграции концептуальных представлений предметных задач [3,82].

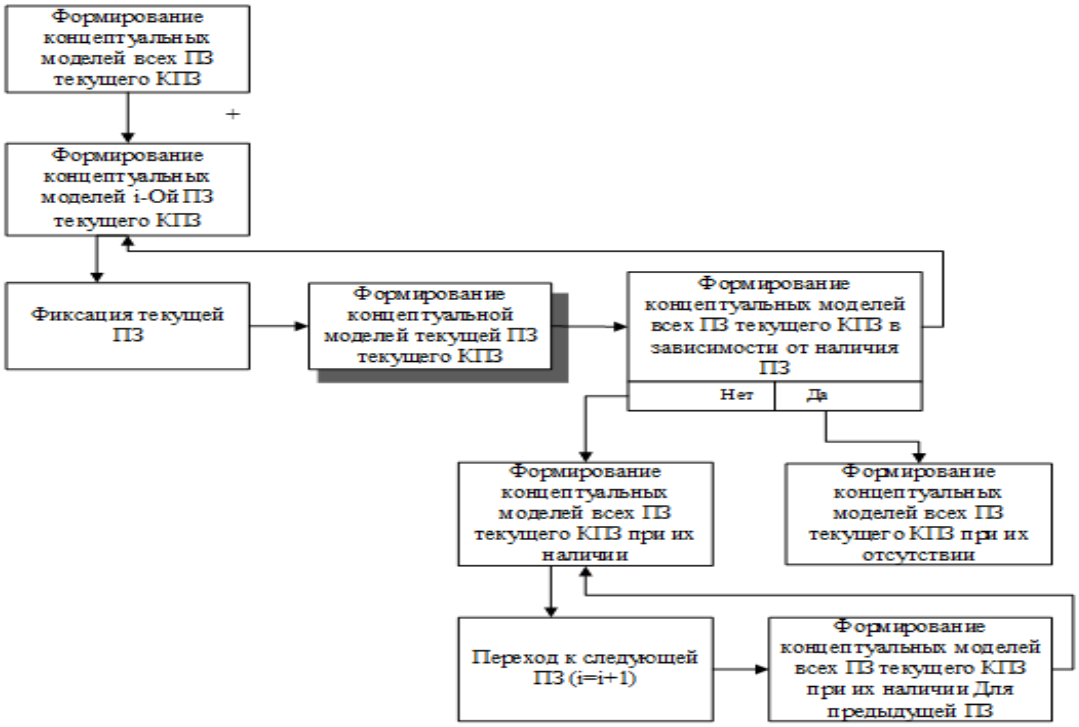


Рис.3.1. Укрупненный алгоритм концептуального моделирования текущего комплекса как цикла по предметным задачам [3]

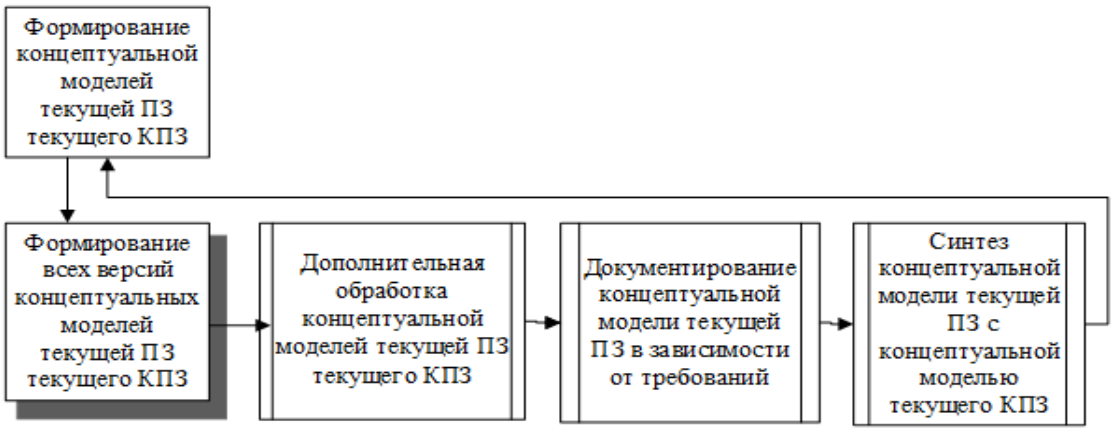


Рис. 3.2. Укрупненный алгоритм концептуального моделирования текущей предметной задачи [3]

В результате формируются: а) концептуальные модели комплекса предметных задач под проектную и постоянную информацию; б) концептуальные модели предметной задачи под переменную и постоянную информацию (для каждой предметной задачи комплекса).

Под обработкой статистических предметных ограничений при концептуальном моделировании будем понимать процедуры выделения и

определения статичтических предметных зависимостей 1-го рода и их предварительной обработки, зафиксированных при описании основных компонентов модели в рамках концептуальных представлений предметных задач.

В процессе дополнительной обработки концептуальной модели включает в себя следующие основные процедуры на рис. 3.3:

- выделение статистических предметных зависимостей 1-го рода и их предварительная обработка;
- определение содержания для всех статистических ПЗ 1-го рода;
- формирование концептуальных моделей под постоянную информацию для всех статистических ПЗ 1-го рода;
- определение типологии для всех статистических ПЗ- 1-го рода;
- документирование моделей для всех статистических ПЗ 1-го рода.

В основе дополнительной обработки лежат ограничения, налагаемые универсальным представлением на представления предметных задач.

Исходной информацией для обработки концептуального представления предметных задач является описания основных конструкций концептуальной модели: - спецификации (формы F1, F2) и диаграмма основной концептуальной структуры предметной задачи; - спецификации (формы F3, F4) и структурная диаграмма системы предметных зависимостей 1 -го рода; - спецификация (форма F6) и матричная диаграмма концептуальной модели в целом.

Концептуальное моделирование предметных задач выполняет специалист представлению знаний (инженер-когнитолог).

При выделении статистических предметных зависимостей 1-го рода и их предварительной обработки выявляются элементарные предметные зависимости 1-го рода, которые имеет степень формализации как «статистическую».

Далее выполняется упорядочение или ранжирование всех выделенных зависимостей.

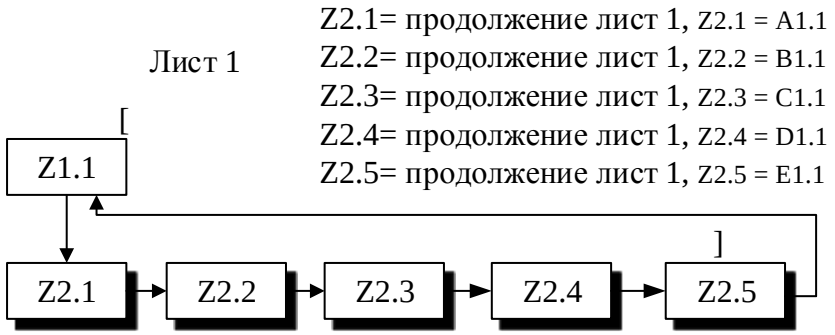


Рис.3.3. Декомпозиция проектной задачи «обработки концептуальных моделей – Z1.1» где Z2.1- Выделение статистических предметных зависимостей и их предварительная обработка, Z2.2- Определение содержания для всех статистических ПЗ 1-го рода, Z2.3- Формирование концептуальных моделей для всех статистических ПЗ 1-го рода, Z2.4- Определение типологии для всех статистических ПЗ- 1-го рода, Z2.5- Документирование моделей для всех статистических ПЗ 1-го рода

При определении содержания для всех статистических ПЗ 1-го рода выполняется фиксация множества ПК и их ролевых свойств в рамках этих зависимостей 1-го рода.

При формировании концептуальных моделей для всех статистических ПЗ 1-го рода выполняется выделение подструктуры основной статической составляющей, увязанной с данной ПЗ-1.

При определении типологии статистических ПЗ 1-го рода выполняется выделение замыкающей вершины в «родословной» ПК и соотнесение ее с определенным типом зависимости 1-го рода.

На этапе документирования выполняется формирование документов, представляющие описания концептуальных моделей 1-го рода под постоянную информацию, которые хранятся в табличной форме.

Результатом данных моделей формируются: множество концептуальных моделей предметной задачи под постоянную информацию (для каждой статистической ПЗ-1).

Результирующая информация сохраняется в общих формах (табл.3.1 – табл.3.5) и формах $F1^c$-$F6^c$ (табл.3.6 –табл.3.9).

Табл.3.1. Предметная задача

Код ПрЗ	Наименование ПрЗ	Колич. стат.ПЗ-1	Колич. стат.КМ	Массив кодов стат.ПЗ-1

Табл.3.2. Модели Задачи

Код ПрЗ	Код КМ

Табл.3.3. Концептуальные модели

Код КМ	Род КМ	Вид КМ

Табл.3.4. Состав концептуальных моделей

Код КМ	Код стат. структуры	Код дин. структуры	Код структуры увязки

Табл.3.5. Структуры концептуальных моделей

Код структуры	Тип структуры	Род структуры	Вид структуры	Количество элементов структкры

Табл.3.6. Форма $F1^c$ -описание ПК

Код структуры	Код ПК	Наименование ПК	Класс ПК	Тип ПК	Статус ПК	Примечания

Табл.3.7. Форма $F2^c$ - описание бинарных связей ПК

Код струк-туры	Код связи ПК	Класс связи ПК	Код ПК_1	Код ПК_2	Код ПК_3	Наименова-ние связи ПК	Тип связи ПК	Оцен-ка

Табл.3.8. Форма F3^c -описание предметных ограничений 1-го рода

Код структуры	Код ПЗ-1	Наименование ПЗ-1	Степень формализации	Статус ПЗ-1	Структурное свойство	Тип ПЗ-1	Прим.

Табл.3.9. Форма F6^c -описание содержания ПЗ

Код структуры	Код ПЗ	Структурное свойство ПЗ-1	Код ПК	Роль ПК	Структурное свойство ПК	Объемное свой-во ПК	Особая роль ПК

3.2 Разработка начального модельного представления задачи «Обработка статистических предметных ограничений 1-го рода при концептуальном моделировании прикладных задач»

3.2.1 Общая характеристика задачи в целом

Предметная задача «Обработка статистических предметных ограничений 1-го рода при концептуальном моделировании прикладных задач» включает несколько подзадач или процедур.

Процесс расслоения концептуальных моделей под постоянную информацию предполагает выделение предметных зависимостей 1-го рода из исходного списка предметных зависимостей 1-го рода, которые имеют степень формализации как «статистическую» [78].

Поскольку каждая зависимость имеет двухпозиционный код, отражающий уровень разложения и позицию на уровне, то следует выполнить упорядочение или ранжирование всех выделенных зависимостей (процедура А).

И затем для каждой зависимости (из множества упорядоченных) определить их содержание, т.е. выявить множество предметных категорий класса «Признак» с учетом их роли в предметной зависимости 1-го рода (процедура B).

Полученные предметные категории имеют в соответствии с исходной информацией свою «родословную» или бинарные связи предметных категорий в основной концептуальной структуре. Это позволит сформировать для всех статистических предметных зависимостей 1-го рода локальные концептуальные структуры, которые будут выделенными подструктурами основной статической составляющей концептуальной модели предметной задачи (процедура C). Заключительной процедурой является процедура определения типологии статистических предметных зависимостей 1-го рода (процедура D).

Результатом формирования начального модельного представления является графическое и табличное представление набора структур: – действий задачи, -используемых в них параметров, -содержательной увязки параметров и действий в виде матричной диаграммы.

3.2.2 Разработка начальной модели для процедуры A

3.2.2.1 Формирование диаграмм системы предметных действий для процедуры A

На рис. 3.4 показано, что для выделения статистических предметных зависимостей и их предварительной обработки (Z2.1= A1.1) организуется последовательность, включающая 4 действия. В первом из них (A2.1) определяется код динамической структуры концептуальной модели (КМ) предметной задачи (ПрЗ), после чего (A2.2) определяется количество и список статистических ПЗ 1-го рода (ПЗ-1). В третьем действии (A2.3) выполняется перекодировка всех выделенных статистических ПЗ 1-го рода и их ранжирование в действии (A2.4).

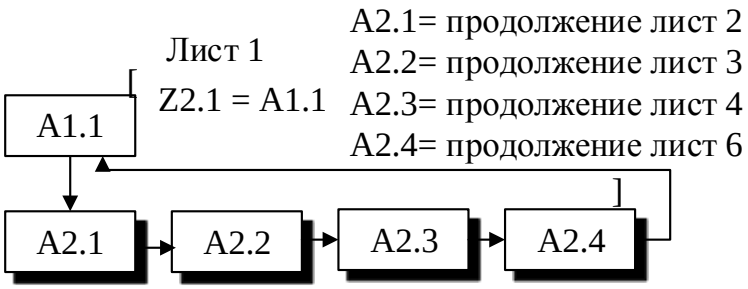


Рис. 3.4. Декомпозиция задачи «Выделение статистических предметных зависимостей и их предварительная обработка – А1.1»: А2.1- Определние кода динамической структуры концептуальной модели (КМ) предметной задачи (ПрЗ), А2.2- Определение количества и списка стат.ПЗ 1-го рода, А2.3- Перекодировка всех выделенных стат.ПЗ 1-го рода, А2.4- Ранжирование стат. ПЗ 1-го рода.

Позиция (А2.1) раскрывается как последовательность (рис. 3.5). Для определения кода динамической структуры КМ для ПрЗ необходимо определить код КМ (А3.1) и далее в (А3.2) определяется код ее динамической структуры.

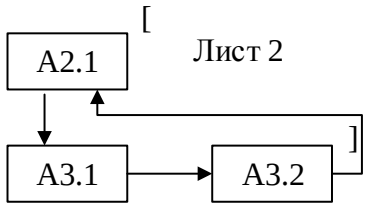


Рис. 3.5. Декомпозиция проектной задачи «определения кода дин. структуры ПрЗ - А2.1» где А3.1- Определение кода модели, А3.2- Определение кода динамической структуры КМ.

На рис. 3.6 действие А2.2 раскрывается как последовательность. В первом действии (А3.3) необходимо определить количество статистических ПЗ-1 для текущего кода динамической структуры, затем указать (А3.4а) количество моделей в задаче под постоянную информацию и в последнем действии (А3.4) определить список статистических ПЗ-1 для текущей задачи.

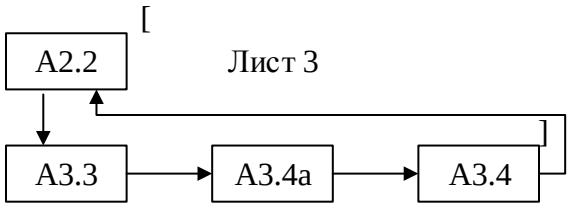


Рис. 3.6. Декомпозиция задачи «Определение количества и списка статистических ПЗ-1 – А2.2» где А3.3 – Определение количества ПЗ-1 для текущего кода динамической структуры, А3.4а - Определение количества стат. КМ в задаче, А3.4 – Определение множества стат. ПЗ-1 для текущей задачи.

Позиция А2.3 раскрывается как цикл, в котором происходит перекодировки всех статистических ПЗ 1-го рода (рис. 3.7). В действии А3.5 происходит перекодировка i-ой статистической ПЗ-1.

В позиции А4.1 происходит фиксация статистической ПЗ-1 и перекодировка текущей ПЗ-1 выполняется в действии А4.2.

Действие А4.3 раскрывается как альтернатива. При наличии следующей статистической ПЗ-1 происходит переход к следующей итерации i=i+1 (А6.1) и фиксируется перекодировка для предыдущей статистической ПЗ-1 А6.2. На выходе действия А5.4 получаем массив перекодировок всех статистических ПЗ-1.

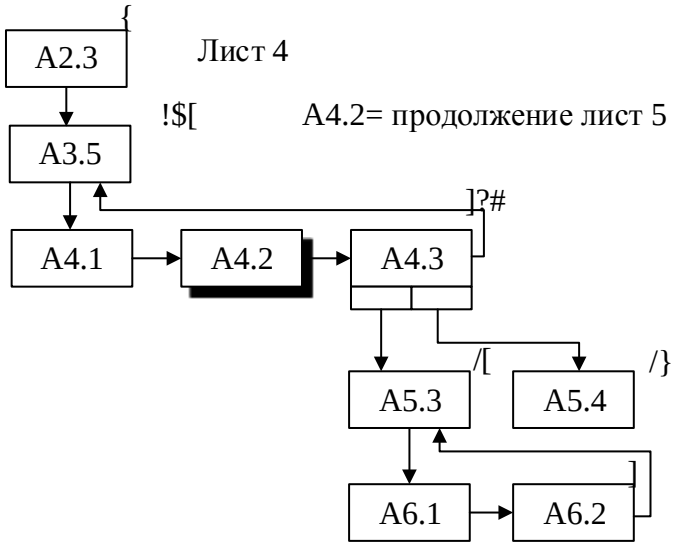


Рис.3.7. Декомпозиция задачи «Перекодировка всех статистических ПЗ 1-го рода – А2.3» где А3.5 – Перекодировка i-ой стат. ПЗ-1, А4.1 - Фиксация стат. ПЗ-1, А4.2 – Перекодировка для текущей ПЗ-1**,** А4.3 - Перекодировка всех стат. ПЗ-1 в зависимости от их наличия, А5.3 - Перекодировка всех стат. ПЗ при их наличии, А6.1 - Переход к следующей стат. ПЗ-1 i=i+1, А6.2 – Перекодировка всех стат. ПЗ-1 при их наличии для предыдущей стат. ПЗ-1, А5.4 - Перекодировка всех стат. ПЗ-1 при их отсутствии.

На рис. 3.8 показано действие для определения новых кодов для текущей ПЗ-1 (А4.2).

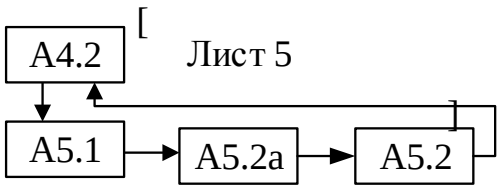


Рис. 3.8. Декомпозиция задачи «Перекодировка для текущей ПЗ-1- А4.2» где А5.1 – Определение код уровня декомпозиции, А5.2а– Запись во временную таблицу, А5.2 – Определение код позиции на уровне.

Особенность данного действия является то, что код ПЗ-1 включает код уровня декомпозиции (А5.1), запись во временную таблицу (А5.2а) и код позиции на уровне (А5.2).

На рис. 3.9 показано, что для ранжирования всех статистических ПЗ -1 (А2.4) выполняется последовательность из 3-х сложных действий. В первом (А3.6) выполняется «Формирование и обработка массива уровней декомпозиции для множества ПЗ-1». Следующее действие (А3.7) определяет формирование и обработку массивов позиций всех уровней декомпозиции для множества ПЗ-1. В последнем действии (А3.8) - формируется общий ранг для множества ПЗ-1.

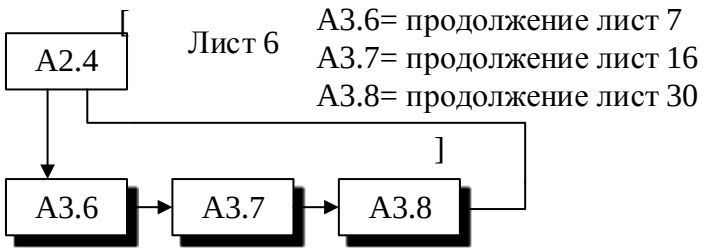


Рис. 3.9. Декомпозиция задачи «Ранжирование всех статистических ПЗ-1- А2.4» где А3.6 - Формирование и обработка массива уровней декомпозиции для множества ПЗ 1-го рода, А3.7- Формирование и обработка массивов позиций всех уровней декомпозиции для множества ПЗ 1-го рода, А3.8-Формирование общего ранга для множества ПЗ 1-го рода

Процесс ранжирования всех статистических включает следующие действия. На рис. 3.10 показано, что формирование и обработка массива уровней декомпозиции для множества ПЗ 1-го рода (А3.6) является последовательностью, включающая три сложных действия.

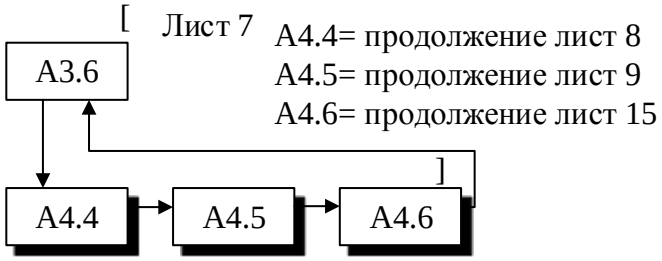


Рис. 3.10. Декомпозиция задачи «Формирование и обработка массива уровней декомпозиции для множества ПЗ 1-го рода - А3.6» где А4.4 - Формирование исходного массива уровней декомпозиции, А4.5 - Упорядочение по убыванию исхдного массива по всем основным элементам, А4.6 - Ранжирование всех уровней декомпозиции.

В первом действии (А4.4) формируется исходный массив уровней декомпозиции. В следующем действии (А4.5) выполняется упорядочение по убыванию исходного массива по всем основным элементам. В действии (А4.6) ранжируются все уровни декомпозиции

На рис. 3.11 показано, что формирование исходного массива уровней декомпозиции (А4.4) – это последовательность. В первом действии (А5.5а) определяется количество уникальных кодов уровней декомпозиции, и формируется список уникальных кодов уровней (А5.5).

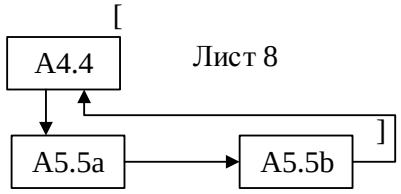


Рис. 3.11. Декомпозиция проектной задачи «Формирование исходного массива уровней декомпозиции - А4.4 » где А5.5а - Определение количества уникальных кодов уровней, А5.5b - Определение списка уникальных кодов уровней.

Для упорядочения по убыванию исходного массива по всем основным элементом (А4.5) используется циклическая процедура (рис. 3.12).

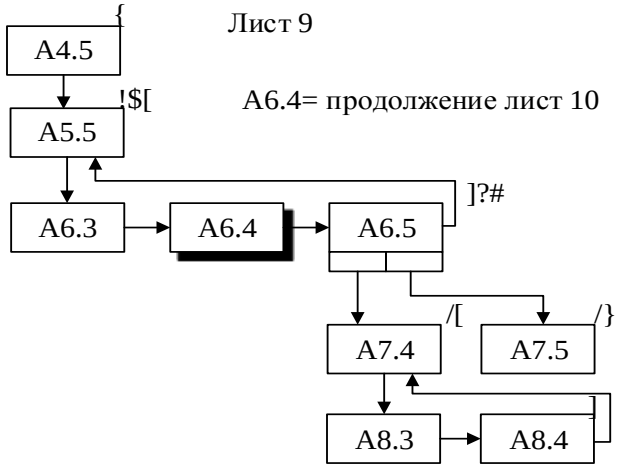


Рис.3.12. Декомпозиция задачи «Упорядочение по убыванию исходного массива по всем основным элементам – А4.5» где А5.5- Упорядочение массива по i-му основному элементу, А6.3- Фиксация i-ой элемента, А6.4- Упорядочение массива для текущего основного элемента, А6.5- Упорядочение по убыванию исходного массива по всем основным элементам в зависимости от их наличия, А7.4- Упорядочение по убыванию исходного массива по всем основным элементам при их наличии, А8.3- Переход к следующему основному элементу i=i+1, А8.4- Упорядочение по убыванию исходного массива по всем основным элементам при их наличии для предыдущего основного элемента, А7.5- Упорядочение по убыванию исходного массива по всем основным элементам при их отсутствии.

В действии (А5.5) происходит упорядочение массива i-му основному элементу. Далее i-ой элемент фиксируется (А6.3) и выполняется действие (А6.4). Действие по продолжению цикла раскрывается как альтернатива. При наличии основного элемента происходит переход к следующему i=i+1 (А8.3) и он вставляется в упорядоченный массив (А8.4).

А в случае отсутствия следующего основного элемента, он вставляется в упорядоченный массив и происходит выход из цикла (А7.5).

На рис. 3.13 показано, что для упорядочения массива для текущего основного элемента А6.4 является последовательность, раскрывающаяся на 3 действия. В первом действии (А7.1) необходимо определить начальное значение максимального элемента, следующее действие (А7.2) позволяет определить количество элементов для сравнения, и в последнем действии (А7.3) - определяется максимальный элемент путем сравнения.

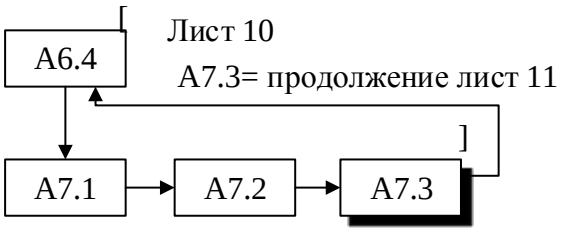


Рис. 3.13. Декомпозиция задачи «Упорядочение массива для текущего основного элемента - А6.4» где А7.1- Определение начального значения максимального элемента, А7.2- Определение количества элементов для сравнения, А7.3- Определение максимального элемента путем сравнения.

На рис. 3.14 действие (А7.3) раскрывается как последовательность.

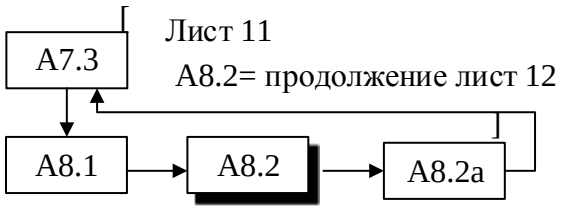


Рис. 3.14. Декомпозиция задачи «Определение максимального элемента путем сравнения - А7.3 » где А8.1- Определение начального индекса, А8.2- Определение максимального элемента путем сравнение текущего основного элемента со всеми другими элементами, А8.2а- Переопределение максимального элемента.

Определяется начальный индекс (A8.1) и действие (A8.2) позволяет определить максимальный элемент путем сравнения текущего основного элемента со всеми другими элементами, затем выполняется переопределение максимального элемента (A8.2a).

Действие (A8.2) раскрывается как цикл (рис. 3.15). В действии A9.1 происходит определение максимального элемента путем сравнения с j-ом элементом. Далее j -ой элемент фиксируется (A10.1) и выполняется альтернатива (A10.2). Действие по продолжению цикла раскрывается как альтернатива (A10.3). При наличии элемента для сравнения (A11.3) происходит переход к следующему j=j+1 (A12.5) и он вставляется в упорядоченный массив (A12.6).

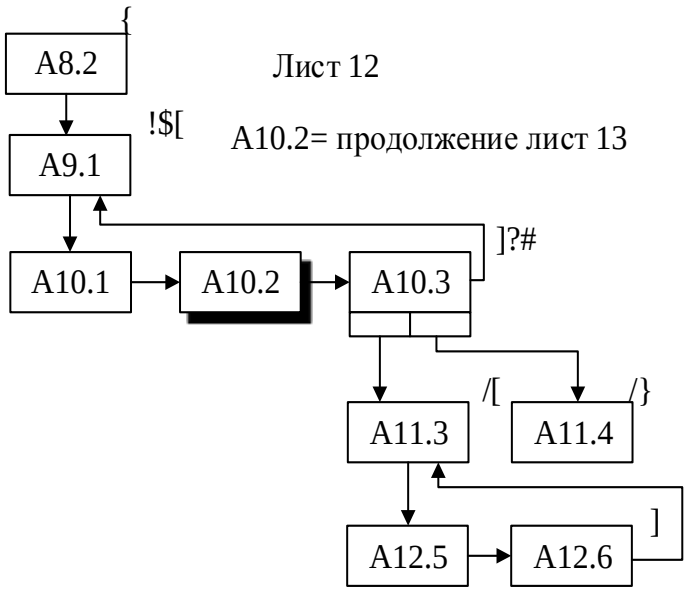


Рис.3.15. Декомпозиция задачи «Определение максимального элемента путем сравнения со всеми другими элементами – A8.2 » где A9.1- Определение максимального элемента путем сравнения с j-ом элементом, A10.1- фиксация элемента, A10.2- Определение максимального элемента в зависимости от сравнения текущего основного элемента с текущим элементом, A10.3- Определение максимального элемента путем сравнения текущего основного элемента со всеми другими в зависимости от их наличии, A11.3- Определение максимального элемента путем сравнения текущего основного элемента со всеми другими элементами при их наличии, A12.5- Переход к следующему сравниваемому элементу j=j+1, A12.6- Определение максимального элемента путем сравнения текущего основного элемента со всеми другими элиметами при их наличии для предыдущего элементов, A11.4- Определение максимального элемента путем сравнения текущего основного элемента со всеми другими элементами при их отсутствии.

А в случае отсутствия следующего элемента для сравнения, он вставляется в упорядоченный массив и происходит выход из цикла (А11.4).

На рис. 3.16 показано, что действие (А10.2) - альтернатива для определения максимального элемента в зависимости от сравнения текущего основного элемента с текущим элементом. Если текущий основной элемент меньше, чем текущий элемент, то выполняется действие (А11.1). Если текущий основной элемент больше, чем текущий элемент, то выполняется действие (А11.2).

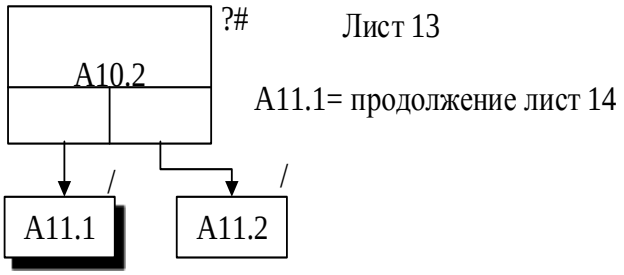


Рис.3.16. Декомпозиция задачи «Определение максимального элемента в зависимости от сравнения текущего основного элемента с текущим элементом - А10.2 » где А11.1- Определение максимального элемента при условии что текущий основной элемент меньше чем текущей элемент, А11.2- Определение максимального элемента при условии что текущий основной элемент больше чем текущей элемент.

Действие А11.1 является последовательностью, раскрывающейся на 4 действия (рис. 3.17). А именно: «Текущий основной элемент сохраняется в новой переменной А12.1», «Текущий элемент сохраняется в текущем основном элементе А12.2», «Новая переменная сохраняется в текущем элементе А12.3» и «Текущий изменённый основной элемент сохраняется как максимальный А12.4».

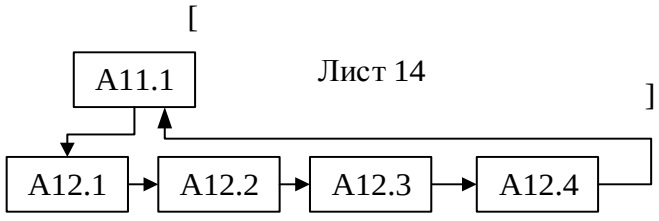


Рис.3.17. Декомпозиция задачи «Текущий основной элемент больше чем текущей элемент - А11.1» где А12.1- Текущий основной элемент сохраняется в новой переменной, А12.2- Текущий элемент сохраняется в текущим основном элементе, А12.3- Новая переменная сохраняется в текущим элементе, А12.4- Текущий изменённый основной элемент сохраняется как максимальный.

Для ранжирования всех уровней декомпозиции (А4.6) используется циклическая процедура (рис. 3.18). В действии (А5.6) происходит ранжирование i-ого уровня декомпозиции. Далее фиксируется уровень декомпозиции (А6.6). В действии А6.7 выполняется присваивание ранга текущему уровню декомпозиции. Окончание цикла раскрывается как альтернатива (А6.8). При наличии уровня декомпозиции происходит переход к следующему уровню i=i+1 (А8.5) и присваивается ранг текущему уровню в массиве уровней (А 8.6). А в случае отсутствия уровня декомпозиции присваивается ранг и происходит выход из цикла (А7.7).

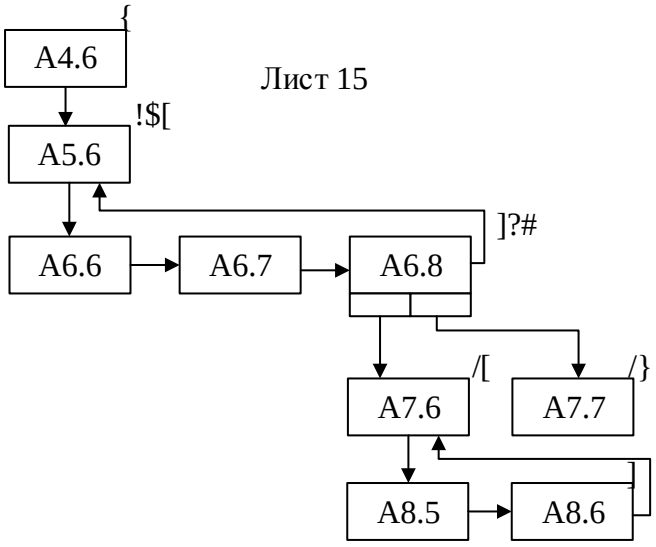


Рис.3.18. Декомпозиция задачи «Ранжирование всех уровней декомпозиции – А4.6» где А5.6- Ранжирование i-ой уровней декомпозиции, А6.6- Фиксация i-ой уровней декомпозиции, А6.7- Присвоение ранга текущий позиции уровней декомпозиции, А6.8- Ранжирование всех уровней декомпозиции в зависимости от их наличия, А7.6- Ранжирование всех уровней декомпозиции при их наличии, А8.5- Переход к следующему уровню декомпозицию i=i+1, А8.6- Ранжирование всех уровней декомпозиции при их наличии для предыдущего уровневого декомпозиции, А7.7- Ранжирование всех уровней декомпозиции при их отсутствии.

На рис. 3.19 показано, что для формирования и обработки массивов позиций всех уровней декомпозиции для множества ПЗ-1, действие А3.7 является последовательностью, определяющей обработку множества позиций на уровнях декомпозиции ПЗ-1.

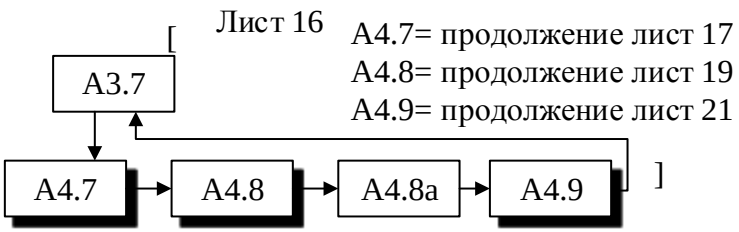


Рис.3.19. Декомпозиция задачи «Формирование и обработка массивов позиций всех уровней декомпозиции для множества ПЗ 1-го рода – А3.7» где А4.7- Определение количества позиций для каждого уровня декомпозиции ПЗ 1-го рода, А4.8- Определение статусов всех уровней декомпозиции, А4.8а- Определение массивов кодов позиций для всех уровней декомпозиции со статусом=1, А4.9- Упорядочение позиций для всех уровней декомпозиции со статусом=1.

Для определения количества позиций для каждого уровня декомпозиции ПЗ-1 А4.7 используется циклическая процедура (рис.3.20). В действии А5.7 происходит определение количества позиций для i-ого ранга уровня декомпозиции. Далее фиксируется ранг уровня декомпозиции А6.9.

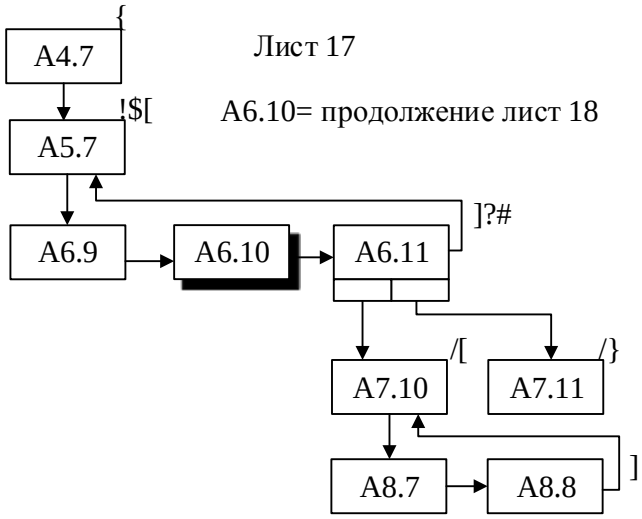


Рис.3.20. Декомпозиция задачи «Определение количества позиций для всех рангов уровней декомпозиции ПЗ-1 – А4.7 » А5.7- Определение количества позиций для i-го ранга уровня декомпозиции ПЗ 1-го рода, А6.9- Фиксация ранга уровня декомпозиции, А6.10: Определение количества позиций для текущего ранга уровня декомпозиции, А6.11- Определение количества позиций для всех рангов уровней декомпозиции ПЗ 1-го рода в зависимости от их наличия, А7.10- Определение количества позиций для всех рангов уровней декомпозиции ПЗ 1-го рода при их наличии, А8.7- Переход к следующему рангу уровня декомпозицию i=i+1, А8.8- Определение количества позиций для всех рангов уровней декомпозиции ПЗ 1-го рода при их наличии для предыдущего ранга уровня декомпозиции, А7.11- Определение количества позиций для всех рангов уровней декомпозиции ПЗ 1-го рода при отсутствии.

Действие А6.10 определяет количество позиций для текущего ранга уровня декомпозиции. Завершение/продолжение цикла (А6.11) раскрывается как альтернатива. При наличии следующего ранга уровня происходит переход к следующему i=i+1 (А8.7) и заносится в массив количество позиций для предыдущего ранга уровня декомпозиции (А 8.8). А в случае отсутствия ранга количество позиций заносится в массив и происходит выход из цикла (А7.11).

На рис. 3.21 показано, что последовательность для определения количества позиций для текущего ранга уровня декомпозиции (А6.10) раскрывается на 2 действия. В действии А7.8 считывается количество позиций для текущего ранга и в А7.9 выполняется присвоение количества позиций по текущему рангу.

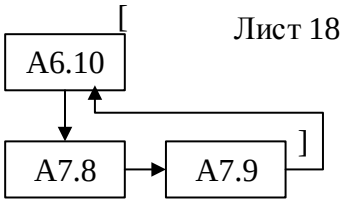


Рис. 3.21. Декомпозиция задачи «Определение количества позиций для текущего ранга уровня декомпозиции - А6.10» где А7.8- Определение количества позиций для текущего ранга, А7.9- Присвоение количества позиций по рангу для каждого уровня.

Следующее действие А4.8 раскрывается как цикл (рис. 3.22). В действии А5.8 происходит определение i-ого уровня декомпозиции. Далее фиксируется уровень декомпозиции А6.12 и присваивается статус текущему уровню декомпозиции А6.13. В действии А5.8 происходит определение i-ого уровня декомпозиции. Далее фиксируется уровень декомпозиции А6.12 и присваивается статус текущему уровню декомпозиции А6.13. Действие «Определение статусов всех уровней декомпозиции в зависимости от их наличия» (А6.14) раскрывается как альтернатива. При наличии уровня происходит переход к следующему уровню декомпозицию i=i+1 (А8.9) и определяют статусы всех уровней декомпозиции (А8.10) для предыдущего уровня декомпозиции. На выходе процедуры А7.13 получаем размеченное множество статусов всех уровней декомпозиции.

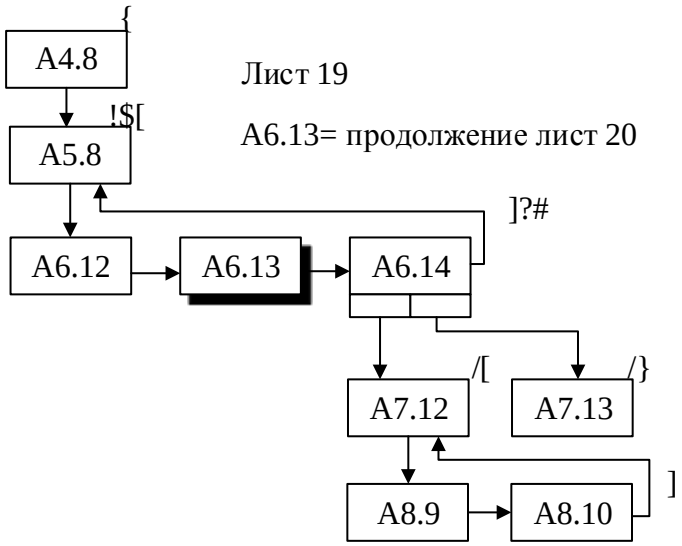


Рис.3.22. Декомпозиция задачи «Определение статусов всех уровней декомпозиции - A4.8 » A5.8- определение статуса i-ого уровня декомпозиции, A6.12- Фиксация i-ого уровня декомпозиции, A6.13- Определение статуса текущего уровня декомпозиции, A6.14- Определение статусов всех уровней декомпозиции в зависимости от их наличия A7.12- Определение статусов всех уровней декомпозиции при наличии, A8.9- Переход к следующему уровню декомпозиции i=i+1, A8.10- Определение статусов всех уровней декомпозиции при наличии для предыдущего уровня декомпозиции, A7.13- Определение статусов всех уровней декомпозиции при отсутствии.

На рис. 3.23 показано, что для определения статуса текущего уровня декомпозиции действие (A6.13) является последовательностью, раскрывающегося на 2 действия.

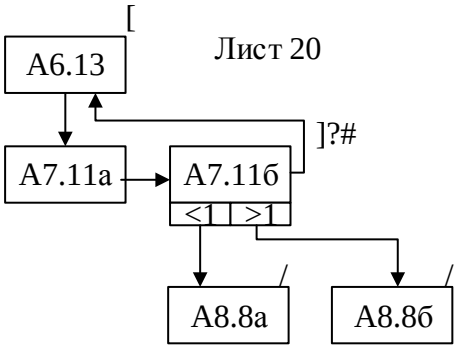


Рис. 3.23. Декомпозиция задачи «Определение статуса текущего уровня декомпозиции - A6.13» где A7.11a- Определение количества позиций текущего уровня декомпозиции, A7.11б- Определение статуса уровня в зависимости от количества позиций, A8.8a- Определение статуса уровня при количестве позиций равном 1 (статус =0), A8.8б- Определение статуса уровня при количестве позиций > 1 (статус =1).

В действии А7.11а необходимо определить количество позиций и в действии А7.11б определить статус уровня в зависимости от количества позиций. А7.11б раскрывается как альтернатива из двух действий: А8.8а - статус уровня (=0) при количестве позиций равном 1 и А8.8б - статус уровня (=1) количестве позиций большем 1.

Для упорядочения позиций для всех уровней декомпозиции со статусом = 1 используется циклическая процедура (рис. 3.24). В действии А5.9 происходит выполнение i-ой итерации. Далее в действии А6.15 фиксируется уровень со статусом = 1. Выполнение итераций цикла (А6.17) осуществляется в зависимости от наличия уровней декомпозиции и раскрывается как альтернатива. При наличии уровня происходит переход к следующему уровню декомпозицию i=i+1 (А8.13) и упорядочивает позиций всех уровней декомпозиции со статусом = 1 (А8.14) для предыдущего уровня декомпозиции. На выходе процедуры (А7.18) получаем упорядоченное множество позиций всех уровней декомпозиции со статусом = 1.

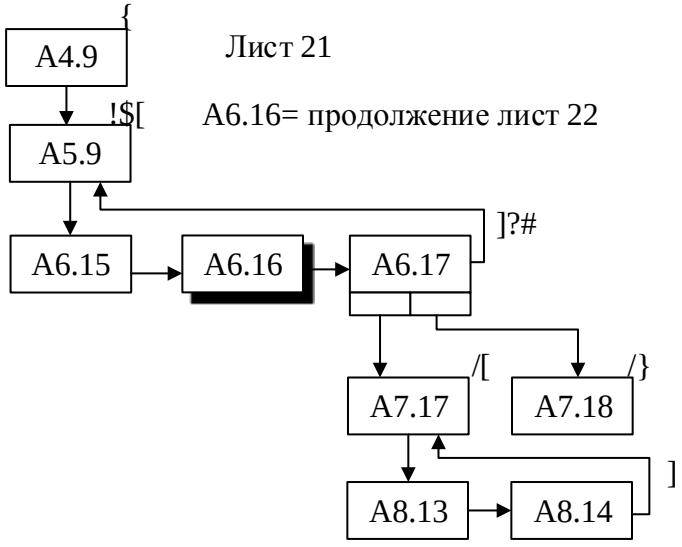


Рис.3.24. Декомпозиция задачи «Упорядочение позиций для всех уровней декомпозиции со статусом = 1 – А4.9» где А5.9- Упорядочение позиций для i-го уровня декомпозиции со статусом=1, А6.15- фиксация уровня, А6.16- Упорядочение позиций для текущего уровня декомпозиции со статусом=1, А6.17- Упорядочение позиций для всех уровней декомпозиции со статусом=1 в зависимости от их наличия, А7.17- Упорядочение позиций для всех уровней декомпозиции со статусом=1 при наличии уровней, А8.13- Переход к следующему уровню декомпозицию i=i+1, А8.14- Упорядочение позиций для всех уровней декомпозиции со статусом=1 при наличии для предыдущего уровня декомпозиции, А7.18- Упорядочение позиций для всех уровней декомпозиции со статусом=1 при отсутствии уровней.

На рис. 3.25 показано, что для упорядочения позиций для текущего уровней декомпозиции со статусом = 1 (А6.16) требуется последовательность, раскрывающаяся на 3 сложных действия.

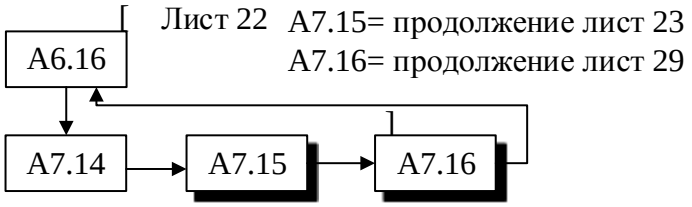


Рис. 3.25. Декомпозиция задачи «Упорядочение позиций для текущего уровней декомпозиции со статусом = 1 – А6.16» где А7.14- Формирование исходного массива позиций для текущего уровня декомпозиции с количеством позиций > 1, А7.15- Упорядочение исходного массива позиций по убыванию по всем основным элементам для текущего уровня декомпозиции с количеством позиций > 1, А7.16- Ранжирование всех позиций при фиксированном уровне.

В действии А7.14 необходимо формировать исходный массив позиций для текущего уровня декомпозиции с количеством позиций >1, в действии А7.15 упорядочивать исходный массив позиций по убыванию для текущего уровня декомпозиции с количеством позиций >1(А7.15) и действии «ранжирование всех позиций при фиксированном уровне» А7.16 раскрываются как цикл.

Следующее действие «Упорядочение исходного массива позиций по убыванию по всем основным элементам для текущего уровня декомпозиции с количеством позиций >1» А7.15 раскрывается как цикл (рис. 3.26). В действии А8.11 происходит упорядочение массива по i-му основному элементу для текущего уровня декомпозиции с количеством позиций >1. Далее фиксируется i-й основной элемент А9.2 и выполняется сложное действие А9.3. Действие «Упорядочение исходного массива позиций по убыванию по всем основным элементам для текущего уровня декомпозиции с количеством позиций >1 в зависимости от их наличия А9.4» раскрывается как альтернатива.

При наличии следующего уровня происходит переход к следующему основному элементу i=i+1 А11.7 и упорядочивает позиций по убыванию по всем основным элементам для текущего уровня декомпозиции с количеством позиций >1 А11.8 для предыдущего уровня декомпозиции.

А в случае отсутствия упорядочивает массива позиций по убыванию по всем основным элементам для текущего уровня декомпозиции с количеством позиций >1 и происходит выход из цикла А10.8.

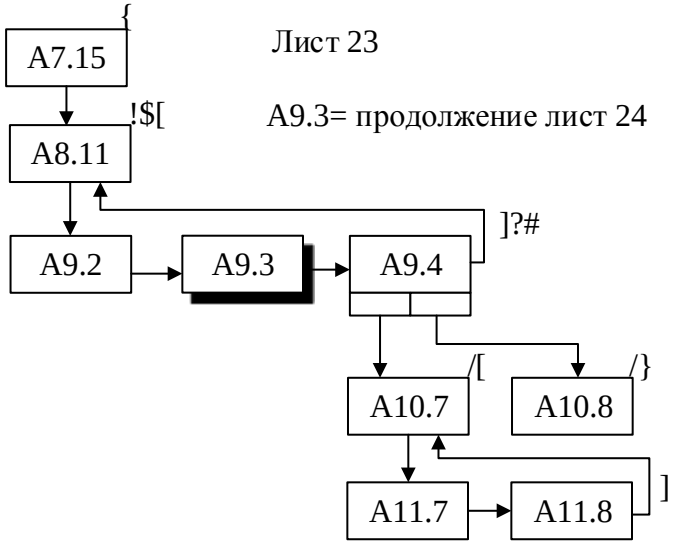


Рис.3.26. Декомпозиция задачи «Упорядочение исходного массива позиций по убыванию по всем основным элементам для текущего уровня декомпозиции с количеством позиций >1 - А7.15» где А8.11- Упорядочение массива по i-ому основному элементу, А9.2- Фиксация i-ого элемента, А9.3- Упорядочение массива для тек. основного элемента текущего уровня декомпозиции, А9.4- Упорядочение исходного массива позиций по убыванию по всем основным элементам для текущего уровня декомпозиции с количеством позиций > 1 в зависимости от их наличия, А10.7- Упорядочение исходного массива позиций по убыванию по всем основным элементам для текущего уровня декомпозиции с количеством позиций > 1 при наличии, А11.7: Переход к следующему элементу i=i+1, А11.8-Упорядочение исходного массива позиций по убыванию по всем основным элементам для текущего уровня декомпозиции с количеством позиций > 1 при их наличии для предыдущего уровня, А10.8- Упорядочение исходного массива позиций по убыванию по всем основным элементам для текущего уровня декомпозиции с количеством позиций > 1 при отсутствии.

На рис. 3.27 приведено, что упорядочение массива для текущего основного элемента текущего уровня декомпозиции А9.3 является последовательность, раскрывающаяся на 3 действия.

В действии А10.4 необходимо определить начальное значение максимального элемента, в действии А10.5 определить количество элементов для сравнения и следующее действие «Определение максимального элемента путем сравнения». Действие А10.6 является последовательностью, раскрывающейся на 3 действия (рис. 3.27).

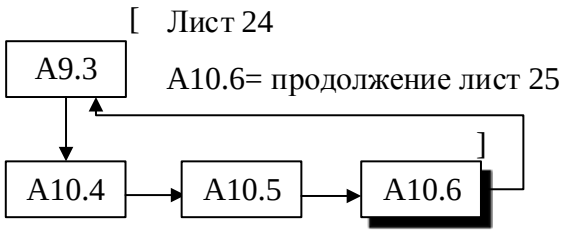


Рис. 3.27. Декомпозиция задачи «Упорядочение массива для текущего основного элемента текущего уровня декомпозиции с количеством позиций >1 – A9.3» где A10.4- Определение начального значения максимального элемента, A10.5- Определение количества элементов для сравнения, A10.6- Определение максимального элемента путем сравнения.

На рис. 3.28 показано, что для определения максимального элемента путем сравнения (A10.6) необходимо определить начальный индекс A11.5 и определить максимального элемента путем сравнения текущего основного элемента со всеми другими элементами (A11.6).

В последнем действии выполняеься «Переопределение максимального элемента (A11.6a)».

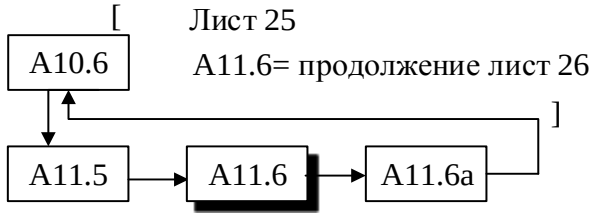


Рис. 3.28. Декомпозиция задачи «Определение максимального элемента путем сравнения - A10.6» где A11.5- Определение начального индекса, A11.6- Определение максимального элемента путем сравнения текущего основного элемента со всеми другими элементами, A11.6a- Переопределение максимального элемента.

Действие A11.6 раскрывается как цикл (рис. 3.29). В действии A12.7 происходит определение максимальной позиции путем сравнения с j-ым элементом.

Далее фиксируется основной элемент A13.1 и раскрывается как альтернатива A13.2. Действие «Определение максимальной позиции путем сравнения текущего основного элемента со всеми другими элементами в зависимости от их наличия A13.3» раскрывается как альтернатива.

При наличии следующего элемента происходит переход к следующему сравниваемому элементу j=j+1 A15.5 и определяет максимальный элемент путем сравнения текущего основного элемента со всеми другими элементами при их наличии для предыдущего элемента A15.6. А в случае отсутствия определяет максимальный элемент путем сравнения текущего основного элемента со всеми другими элементами и происходит выход из цикла A14.4.

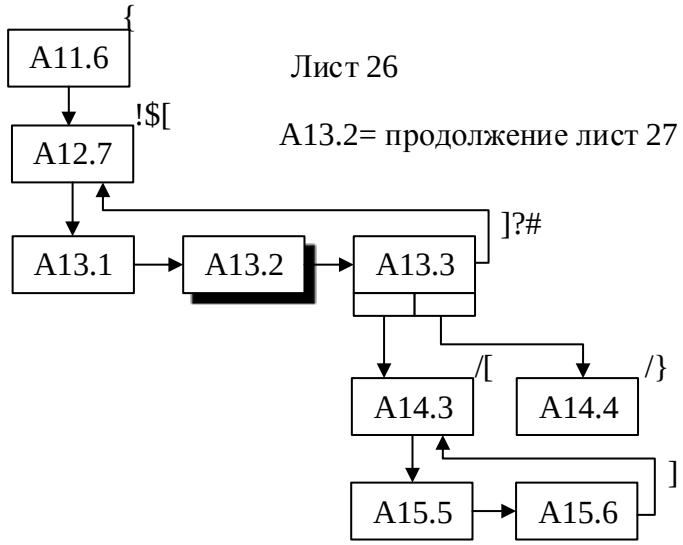


Рис.3.29. Декомпозиция задачи «Определение максимального элемента путем сравнения текущего основного элемента со всеми другими элементами - A11.6 » где A12.7- Определение максимального элемента путем сравнения с j-ым элементом, A13.1- Фиксация основного элемента, A13.2- Определение максимального элемента в зависимости от сравнения текущего основного элемента с другим элементом, A13.3- Определение максимального элемента путем сравнения текущего основного элемента со всеми другими элементами в зависимости от их наличия, A14.3- Определение максимального элемента путем сравнения текущего основного элемента со всеми другими элементами при наличии, A15.5- Переход к следующему элементу j=j+1, A15.6- Определение максимального элемента путем сравнения текущего основного элемента со всеми другими элементами при их наличии для предыдущего элемента, A14.4- Определение максимального элемента путем сравнения текущего основного элемента со всеми другими элементами при отсутствии.

Действие на рис. 3.30 раскрывается как альтернатива A13.2. Действие A14.1 является последовательность, раскрывающаяся на 4 действия. В действии A14.2 выполняется определение максимального элемента при условии, что текущий основной элемент больше, чем текущий элемент.

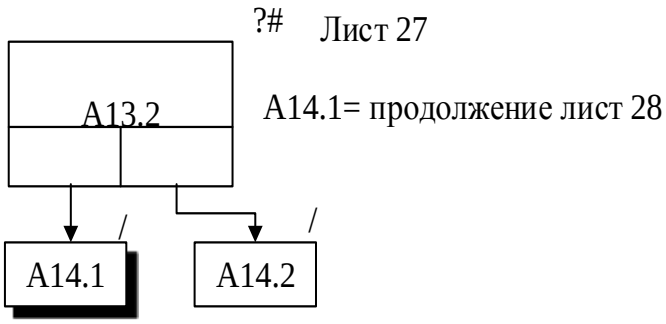


Рис. 3.30. Декомпозиция задачи «Определение максимального элемента в зависимости от сравнения текущего основного элемента с текущим элементом - A13.2» где A14.1- Определение максимального элемента при условии что текущий основной элемент меньше чем текущий элемент, A14.2- Определение максимального элемента при условии что текущий основной элемент больше, чем текущий элемент.

На рис. 3.31 показано, что для определения максимального элемента при условии, что текущий основной элемент меньше, чем текущий элемент A14.1,выполняются следующие действия: текущий основной элемент сохраняется в новой переменной (A15.1); текущий элемент сохраняется в текущем основном элементе (A15.2); новая переменная сохраняется в текущем элементе (A15.3); текущий изменённый основной элемент сохраняется как максимальный (A15.4).

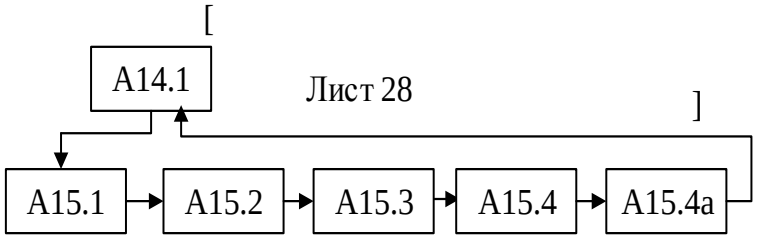


Рис.3.31. Декомпозиция задачи «Определение максимальной позиции при условии, что текущая основная позиция больше чем текущая позиция - A14.1» где A15.1- Текущий основной элемент сохраняется в новой переменной, A15.2- Текущий элемент сохраняется в текущем основном элементе, A15.3- новая переменная сохраняется в текущем элементе, A15.4- Текущий изменённый основной элемент сохраняется как максимальный.

Действие «Ранжирование всех позиций при текущем уровне» раскрывается как цикл (рис. 3.32). В действии A8.12 происходит ранжирование i-ой позиций при текущем уровне.

Далее позиция фиксируется А9.5 и ей присваивается ранг при текущем уровне А9.6. «Ранжирование всех позиций при фиксированном уровне в зависимости от их наличия» (А9.7) раскрывается как альтернатива.

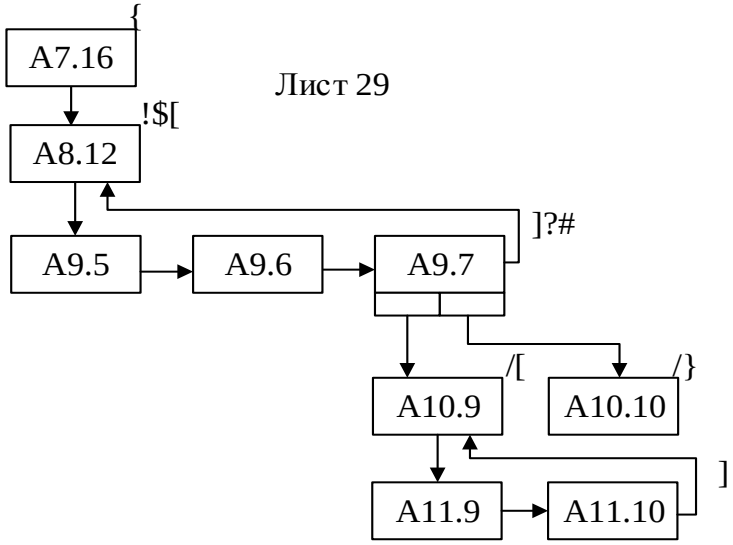


Рис.3.32. Декомпозиция задачи «Ранжирование всех позиций при текущем уровне – А7.16» где А8.12 - Ранжирование i-ой позиций при текущем уровне, А9.5- Фиксация позиции при текущем уровне, А9.6- Присвоение ранга текущей позиции при текущем уровне, А9.7- ранжирование всех позиций при текущем уровне в зависимости от их наличия, А10.9- Ранжирование всех позиций при текущем уровне при наличии, А10.10- Ранжирование всех позиций при текущем уровне при их отсутствии, А11.9- Переход к следующей позиции i=i+1, А11.10- Ранжирование всех позиций при текущем уровне при их наличии для предыдущей позиции.

При наличии позиции происходит переход к следующей i=i+1 (А11.9) и ранжируется все позиции при текущем уровне для предыдущей позиции (А11.10). А в случае отсутствия, ранжируются все позиции при текущем уровне и происходит выход из цикла (А10.10).

Действие «Формирование общего ранга для множества ПЗ 1-го рода» раскрывается как цикл (рис. 3.33). В действии А4.10 происходит определение ранга i-ой ПЗ-1. Далее фиксируется ПЗ 1-го рода (А5.10) и выполняется присвоение общего ранга текущей ПЗ 1-го рода (А5.11). Действие «Формирование общего ранга для множества ПЗ 1-го рода в зависимости от их наличия» (А5.12) раскрывается как альтернатива. При наличии ПЗ-1 происходит переход к следующей ПЗ-1 (i=i+1) (А7.19) и формируется общий ранг для множества ПЗ 1-го

рода для предыдущей ПЗ 1-го рода (А7.20). А в случае отсутствия формируется общий ранг для множества ПЗ 1-го рода и происходит выход из цикла (А6.19).

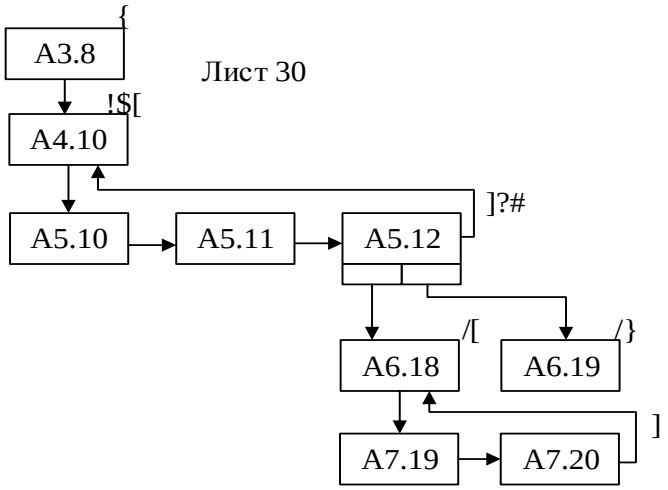


Рис.3.33. Декомпозиция задачи «Формирование общего ранга для множества ПЗ-1 – А3.8» где А4.10- Определение ранга i-ой ПЗ-1, А5.10- Фиксация i-ой ПЗ-1, А5.11- Присвоение ранга текущий ПЗ-1, А5.12- Формирование общего ранга для множества ПЗ 1-го рода в зависимости от их наличия, А6.18- Формирование общего ранга для множества ПЗ 1-го рода при наличии, А6.19- Формирование общего ранга для множества ПЗ 1-го рода при отсутствии, А7.19- Переход к следующей ПЗ-1 i=i+1, А7.20- Формирование общего ранга для множества ПЗ 1-го рода при их наличии для предыдущего уровня декомпозиции.

Таким образом полностью сформирована система предметных действий для первой процедуры «Выделение статистических предметных зависимостей 1-го рода и их предварительная обработа» (процедура А) при начальном моделировании задачи «Обработка статистических предметных ограничений при концептуальном моделировании».

3.2.2.2 Формирование диаграммы системы предметных действий в целом для процедуры А

В целом диаграмма системы предметных действий для процедуры А приведена на рис.3.34.

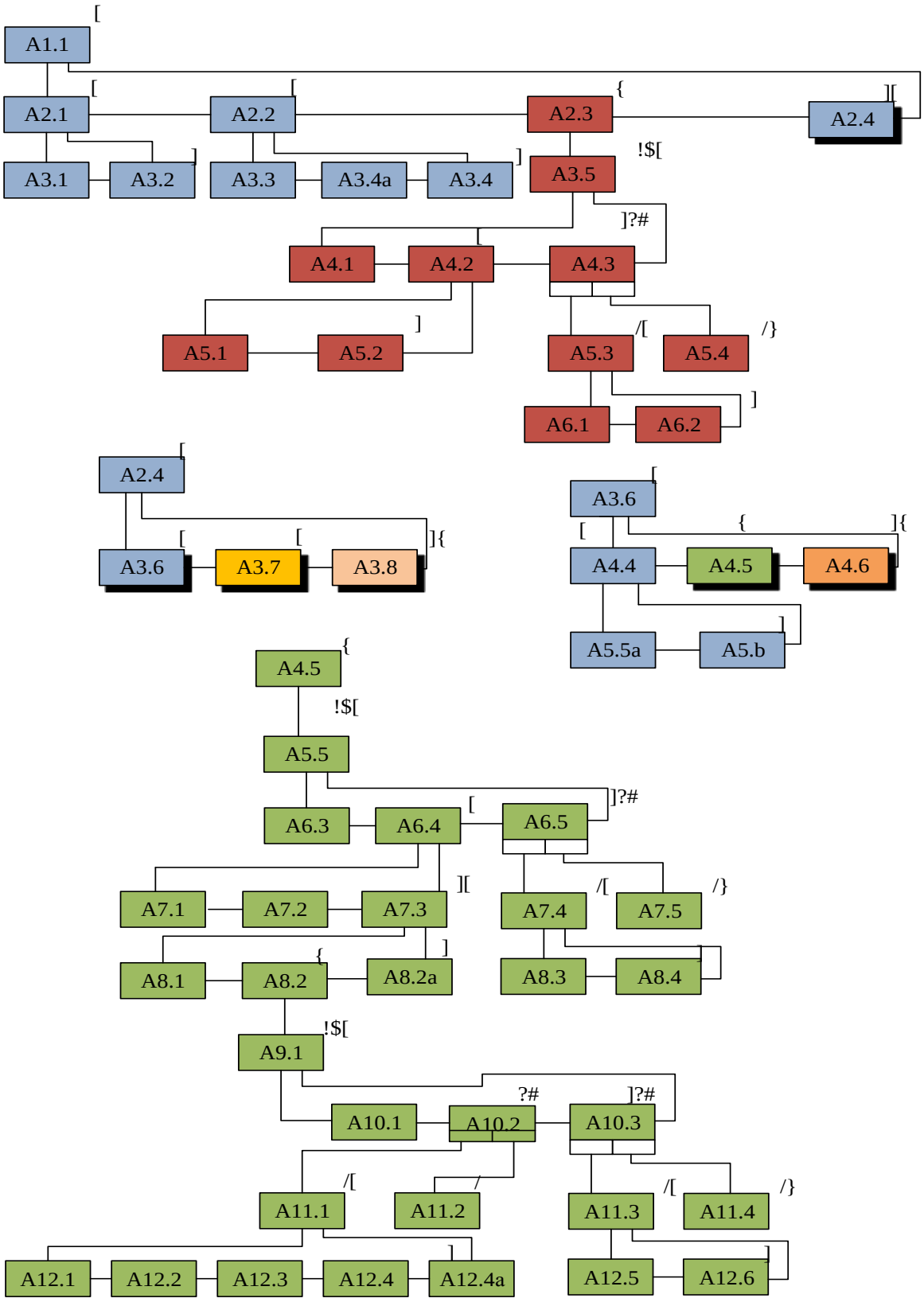


Рис.3.34. Диаграмма процедуры А

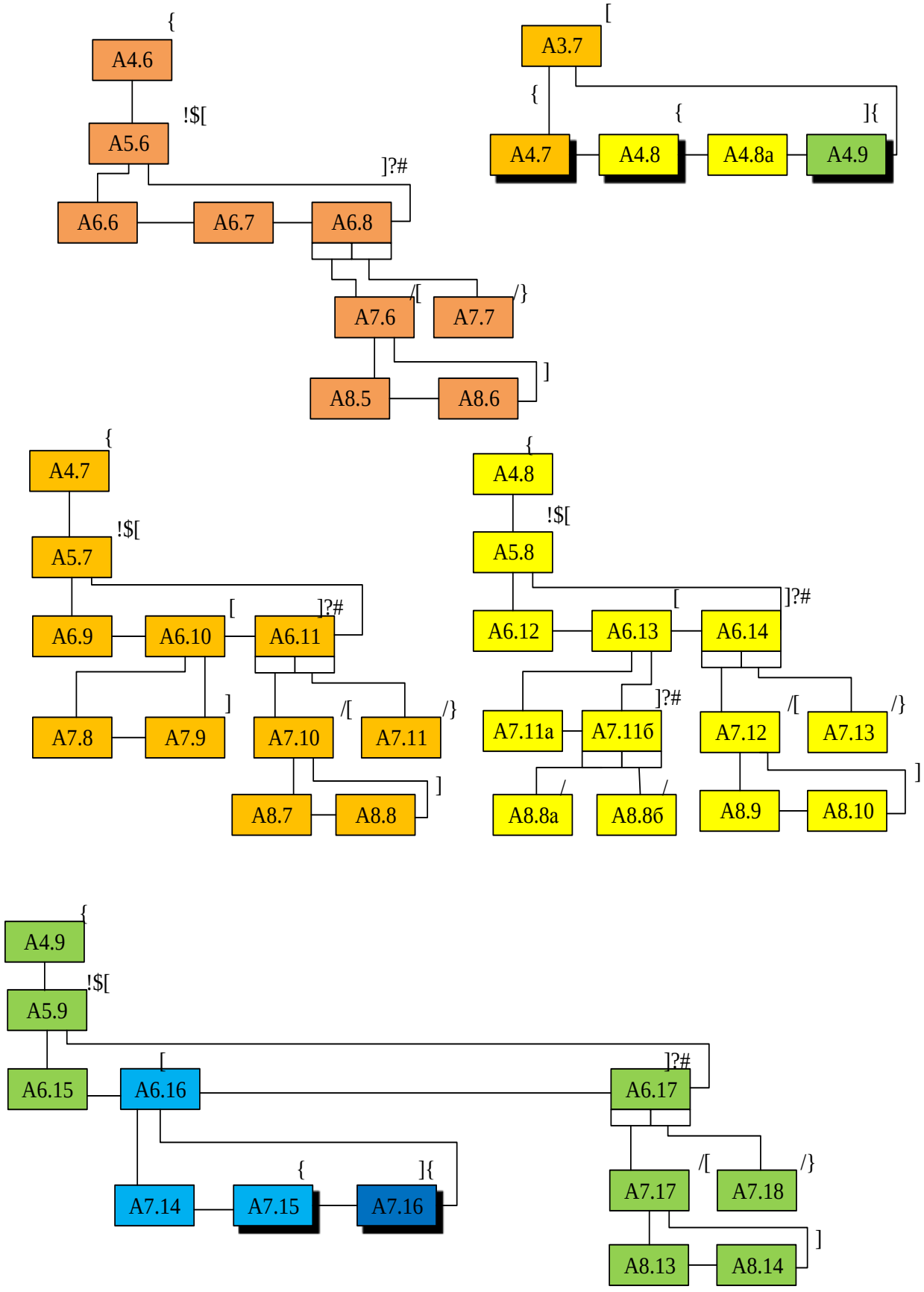


Рис.3.34. Диаграмма процедуры А (продолжение 1)

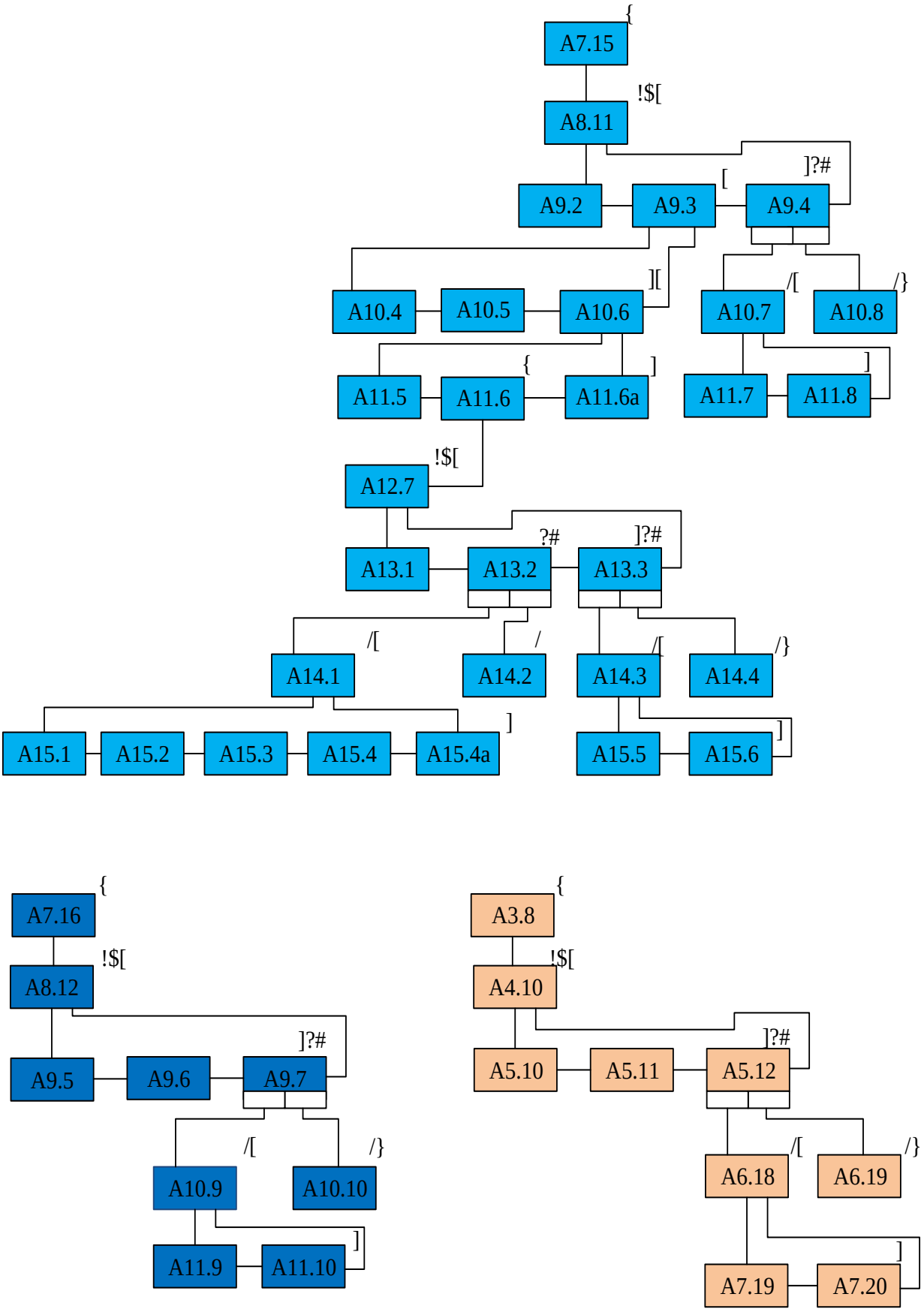


Рис.3.34. Диаграмма процедуры A (продолжение 2)

3.2.2.3 Формирование множества параметров для процедуры А

В целом структура организации данных для процедуры А приведена на рис.3.35 и рис.3.36.

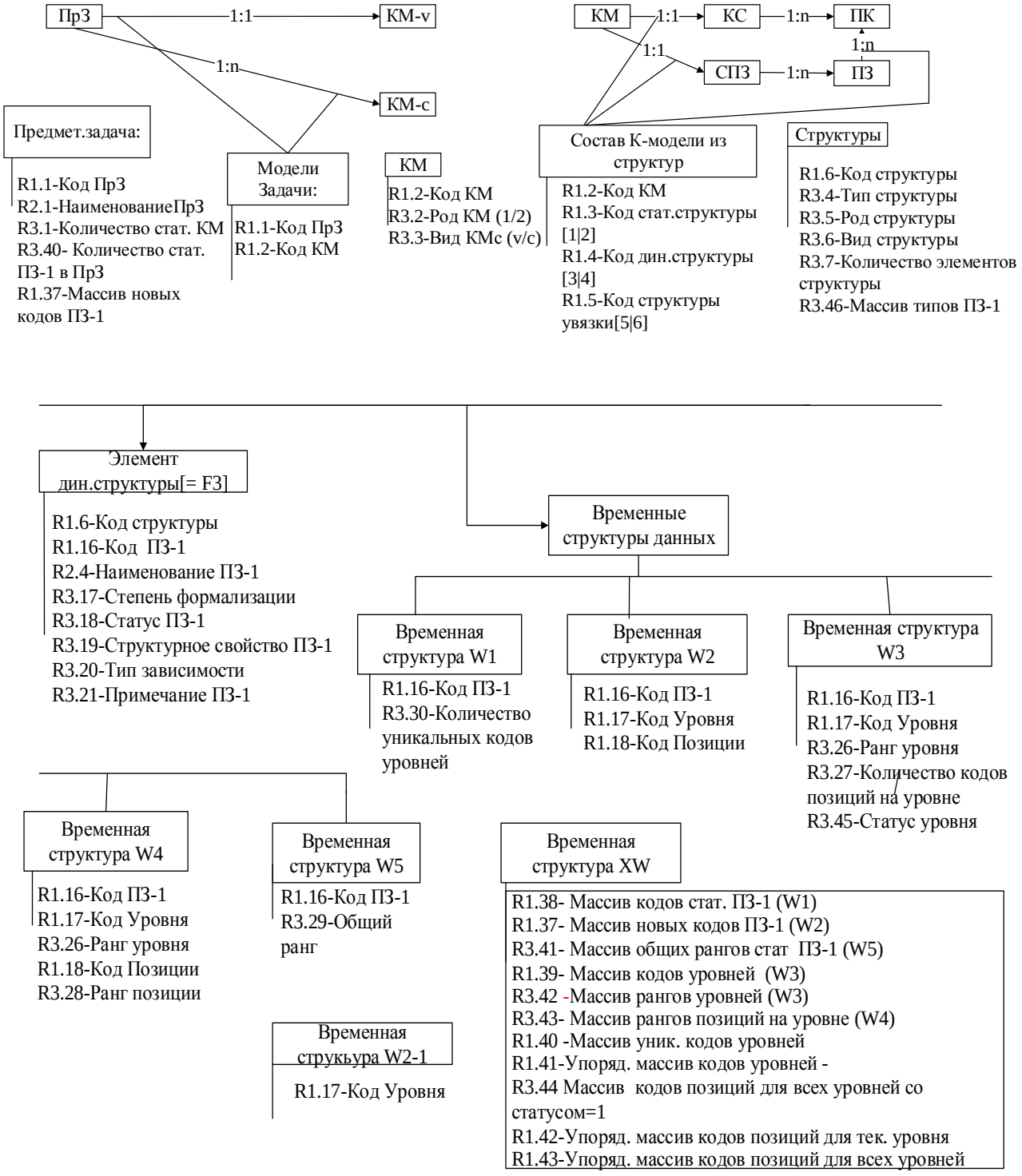


Рис. 3.35. Структура организации данных для процедуры А (часть 1).

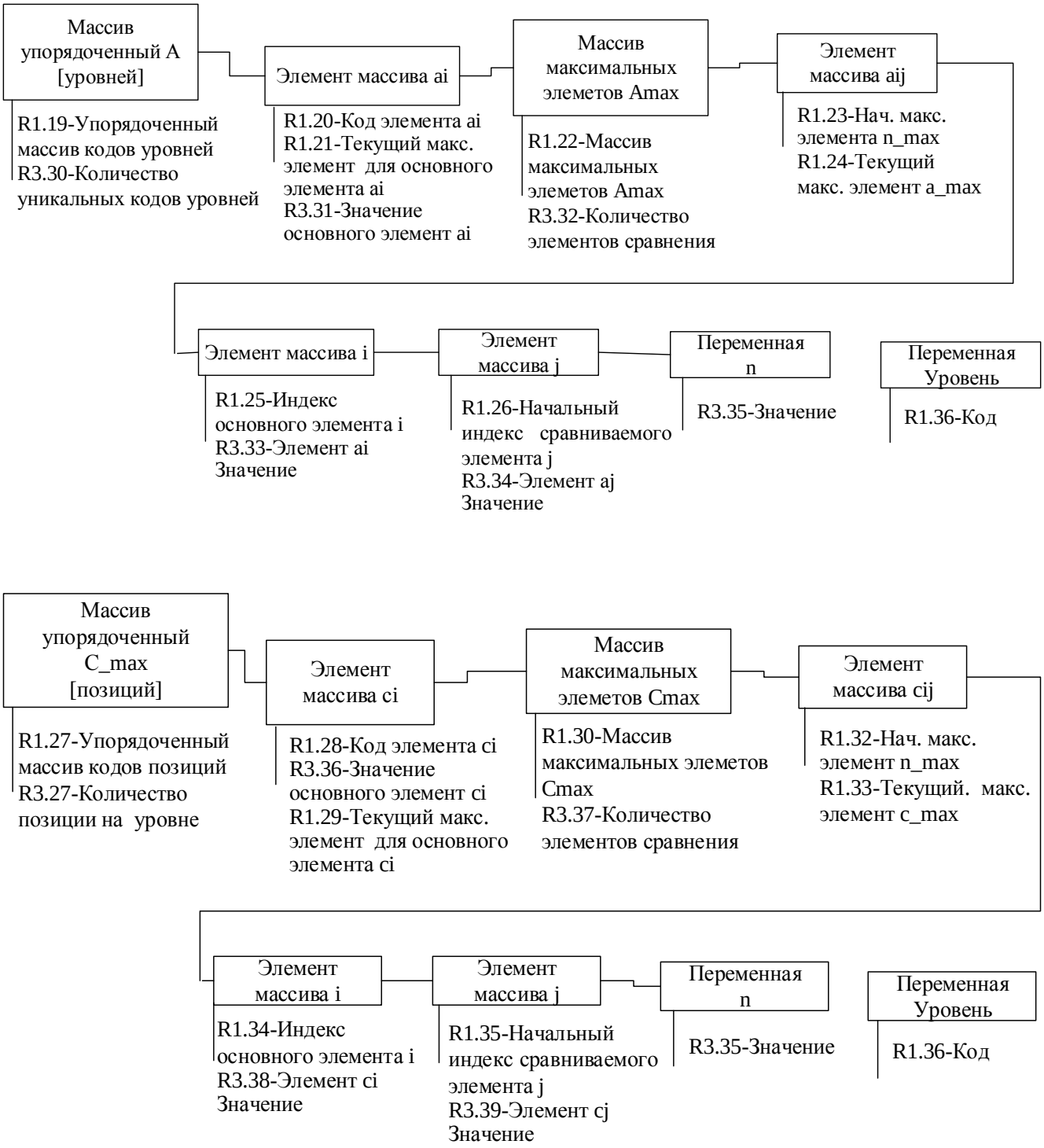


Рис. 3.36. Структура структура организации данных для процедуры A (часть 2).

Таким образом, разработана информационная составляющая начальной модели для процедуры A.

3.2.2.4 Формирование матричных диаграмм для процедуры А

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.4) представлена на рис.3.37. Основным параметром является: ФП –массив общих рангов стат. ПЗ-1- R3.41.

								ФП	……
Пози-ция ПД	R1.1	R3.2	R3.3	R1.2	R1.4	R1.38	R1.37	R3.41	
A1.1								**[**	
A2.1	+	+	+	**.**	*****				Из табл. Модели задачи, КМ, Состав К-стр.
A2.2					+	*****			Во временную табл.W1
A2.3						+	*****		Во временную табл.W2
A2.4							+	***]**	Во временную табл.W5

Рис.3.37. Диаграмма конструкции «Выделение стат. ПЗ-1 и их предварительная обработка» где R1.1- Код ПрЗ, R3.2- Род КМ, R3.3- Вид КМ, R1.2- код КМ, R1.4- Код дин. Структуры, R1.38- Массив кодов стат. ПЗ-1, R1.37- Массив новых кодов ПЗ-1.

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.5) представлена на рис.3.38. Основным параметром является: ФП - код динамической структуры R1.4.

					ФП	Пояснения
Позиция ПД	R1.1	R3.2	R3.3	R1.2	R1.4	
A2.1					**[**	
A3.1	+	+	+	*****		Из таблиц «модели задачи» , «КМ» по ест.соед с усдовием Род КМ =1, Вид КМ =v
A3.2				+	***]**	Из таблицы «состав КМ из структур»

Рис. 3.38. Диаграмма конструкции «Определение кода дин. структуры КМ» где R1.1- Код ПрЗ, R3.2 - Род КМ, R3.3- Вид КМ, R1.2- Код КМ.

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.6) представлена на рис.3.39. Основным параметром является: ФП - Массив кодов стат. ПЗ-1 R1.38.

						ФП	……
Позиция ПД	R1.6	R3.17	R1.1	R3.1	R3.40	R1.38	
A2.2						**[**	
A3.3	+	+			*		Из таблицы F3 ст.форм=стат
A3.4a			+	*	+		Запись в табл. «Пр.задача»
A3.4	+	+		+		*]	Во временную табл. W1

Рис.3.39. Диаграмма конструкции «Определение количество и списка статистических ПЗ 1-го рода» где R1.6- Код структуры, R3.17- ст.форм. (=стат), R1.1- Код ПрЗ, R3.1- Количество стат. КМ, R3.40- Количество стат. ПЗ-1 в задаче.

Матричная диаграмма для типовой конструкции «цикл» (рис. 3.7) представлена на рис. 3.40. Основными параметрами являются: СЦ– Количество стат. ПЗ-1 в задаче R3.40; АИ- Код ПЗ-1 R1.16; ФИ - Код уровня R1.17, Код позиции-R 1.18; ФЦ – Массив новых кодов ПЗ-1 -R 1.37).

	СЧ	АИ	ФИ	ФИ	ФЦ	
Позиция ПД	R3.40	R1.16	R1.17	R 1.18	R 1.37	
A2.3					{	
A3.5	!		$	$	[	
A4.1	.	*				из временной таблицы W1
A4.2	**.**	**+**	**[***	**[***		
A4.3	?				#]	
A5.3	/ .				[	
A6.1	. *					
A6.2	.	+	+	+	*]	
A5.4	/ .	+	+	+	*}	Табл. W2

Рис.3.40. Диаграмма конструкции «Перекодировка всех стат. ПЗ-1»

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.8) представлена на рис.3.41. Основным параметром является: ФП кортеж вр. Табл. w2(Код ПЗ- 1 R1.16, Код уровня-R1.17, Код позиции -R1.18).

				ФП	
Позиция ПД	R1.16	R1.17	R1.18	w2	
A4.2				**[**	
A5.1	+	*			
A5.2a	+	+	*		
A5.2	+	+	+	*]	Запись кортежа в W2

Рис. 3.41. Диаграмма конструкции «Перекодировка для текущей ПЗ-1» где R1.16 - Код ПЗ- 1, R1.17- Код уровня, R1.18- Код позиции

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.9) представлена на рис.3.42. Основным параметром является: ФП – Массив общих рангов (R3.41).

					ФП	
Позиция ПД	R1.16	R1.37	R3.42	R3.43	Массив общих рангов R3.41	
A2.4					**[**	
A3.6	**.**	+	**[***			Табл.W3
A3.7	**.**	+	+	**[***		Табл.W4
A 3.8	**.**	+	+	+	***]**	Табл.W5

Рис.3.42. Диаграмма конструкции «Ранжирование стат. ПЗ-1» где R1.16- Код ПЗ-1, R1.37- Массив новых кодов ПЗ-1, R3.42- Массив рангов уровней, R3.43- Массив рангов позиций на уровне

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.10) представлена на рис.3.43. Основным параметром является: ФП – Массив рангов уровней R3.42.

					ФП	
Позиция ПД	R1.16	R1.35	R1.40	R1.41	R3.42	……
A3.6					**[**	
A4.4	**.**	+	**[***			Табл.W1
A4.5	**.**	+	+	**{***		
A4.6	**.**	+		+	***]**	

Рис. 3.43. Диаграмма конструкции «Формирование и обработка массива уровней декомпозиции для множества ПЗ-1» где R1.16- Код ПЗ-1, R1.35- Массив новых кодов ПЗ-1, R1.40- Массив уник. кодов уровней, R1.41- Упоряд. массив кодов уровней.

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.11) представлена на рис. 3.44.

				ФП	……
Позиция ПД		R 1.37	R3.30	R1.40	……
A4.4				**[**	
A5.5a		+	*****		Из Табл. W1
A5.5		+	+	***]**	В Табл. W2-1

Рис.3.44. Диаграмма конструкции «Формирование исходного массива уровней декомпозиции» где R 1.37- Массив новых кодов ПЗ-1, R3.30- Количество уникальных кодов уровней.

Основным параметром является: ФП - Массив уник. кодов уровней W2-1. R1.40.

Матричная диаграмма для типовой конструкции «цикл» (рис. 3.12) представлена на рис. 3.45. Основными параметрами являются: СЦ– Количество уник.кодов уровней -R3.30; АИ - Значение основного элемента ai –R3.31; ФИ – Текущий макс. элемент для основного элемента ai R1.21; ФЦ - Упорядоченный массив кодов уровней (массив A)-R1.19.

	СЧ	АИ	ФИ	ФЦ	
Позиция ПД	R3.30	R3.31	R1.21	-R1.19	
A4.5				{	
A5.5	!		$	[	
A6.3	.	*			
A6.4	**.**	**+**	**[***		
A6.5	?			#]	
A7.4	/ .			[	
A8.3	. *				
A8.4	.		+	*]	
A7.5	/ .		+	*}	Табл. B

Рис.3.45. Диаграмма конструкции «Упорядочение по убыванию исходного массива по всем основным элементом»

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.13) представлена на рис. 3.46.

Основным параметром является ФП - Текущий макс. элемент для основного элемента ai R1.21.

				ФП			
Позиция ПД	R1.20	R3.31	R1.23	R1.21	R3.32	R3.30	
A6.4				**[**			
A7.1	+	+	*		.		
A7.2	+				*	+	
A7.3	+		.	***][**			

Рис.3.46. Диаграмма конструкции «Упорядочение массива для текущего основного элемента» где R1.20- Код элемента ai, R3.31- Значение основного элемента ai, R1.23- Нач. макс. элемента n_max, R3.32- Количество элементов для сравнения, R3.30- Количество уникальных кодов уровней.

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.14) представлена на рис. 3.47. Основным параметром является: ФП - Текущий макс. элемент для основного элемента ai = Текущий макс. элемент a_max-R1.24.

				ФП		
Позиция ПД	R3.31	R1.25	R1.26	R1.24	R1.22	
A7.3				**[**		
A8.1		+	*****			
A8.2		+	+		**{***	{ Из временной таблицы Amax
A8.2a			+	***]**	+	

Рис. 3.47. Диаграмма конструкции «Определение максимального элемента путем сравнения» где R3.31- Количество элементов для сравнения, R1.25- Индекс основного элемента i, R1.26- Начальный индекс сравниваемого элемента j, R1.22- Массив максимальных элеметов Amax.

Матричная диаграмма (МД) для типовой конструкции «цикл» (рис. 3.15) представлена на рис. 3.48.

Основными параметрами являются: СЦ– Количество элементов сравнения - R3.32; АИ – Элемент ai значение -R3.33, Элемент для срав-нения значение aj - R3.34; ФИ – Максимальный элемент R1.24; ФЦ - Массив максимальных элементов Код-R1.22.

	СЧ	АИ	АИ	ФИ	ФЦ	
Позиция ПД	R3.32	R3.33	R3.34	R1.24	R1.22	
A8.2					{	
A9.1	!			$	[	
A10.1	.		*			
A10.2	**.**		+	*****		**?#**
A10.3	?				#]	
A11.3	/ .				[	
A12.5	. *					
A12.6	.			+	*]	
A11.4	/ .			+	*}	во врем. Табли Bi

Рис.3.48. Диаграмма конструкции «Определение максимального элемента путем сравнения со всеми другими элементами»

Локальная МД для конструкции «альтернатива» (рис. 3.16) представлена на рис. 3.49. Основными параметрами являются: ФА - Максимальный элемент-a_max Код R1.24, АА - Элемент ai Значение - R3.33, Элемент aj Значение - R3.34.

	АА	АА	ФА	Пояснения
Позиция ПД	R3.33	R3.34	R1.24	
A10.2	?	?	#	ai > aj
A11.1	**/**	**/**	**[***	**[a_max= ai**
A11.2	/	/	*	a_max= aj

Рис. 3.49. Диаграмма конструкции «Определение максимального элемента в зависимости сравнения текущего основного элемента с текущим элементом»

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.17) представлена на рис. 3.50. Основным параметром является ФП - Максимальный элемента_max Код -R1.24.

				ФП	А-у	
Позиция ПД	R3.33	R3.34	R3.35	R1.24	R1.36	
A11.1				**[**		
A12.1	+		*		.	x=ai
A12.2	*	+			.	ai= aj
A12.3		*	+		.	aj= x
A12.4		+		***]**	.	a_max= ai

Рис. 3.50. Диаграмма конструкции «Текущий основной элемент больше, чем текущей элемент» где R3.33- Элемент ai значение, R3.34- Элемент aj значение, R3.35- Переменая значение, R1.36- Переменая код уровня

Матричная диаграмма (МД) для типовой конструкции «цикл» (рис. 3.18) представлена на рис. 3.51.

	СЧ	АИ	ФИ	ФЦ	
Позиция ПД	R3.30	R 1.17	R 3.26	R 3.39	
A4.6				{	
A5.6	!		$	[	
A6.6	.	*			
A6.7	.	+	*		
A6.8	?			#]	
A7.6	/ .			[	
A8.5	. *				
A8.6	.		+	*]	
A7.7	/ .		+	*}	во врем. Табл W3

Рис. 3.51. Диаграмма конструкции «Ранжирование всех уровней декомпозиции»

Конструкция представляет собой. Основными параметрами являются: СЦ– Количество уникальных кодов-R3.30; АИ - Код уровня-R1.17; ФИ - Ранг уровня - R 3.26; ФЦ - Массив рангов уровней Код-R3.39.

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.19) представлена на рис. 3.52. Основным параметром является: ФП - Упорядоченный массив кодов позиций для всех уровней - R1.43.

						ФП	
Позиция ПД	R1.16	R3.30	W3	W3	R3.44	R1.43	
A3.7						**[**	
A4.7	+	+	**{***				**{**
A4.8	+	+		**{***			**{**
A4.8a				+	*****		
A4.9		+		+	+	***]**	

Рис.3.52. Диаграмма конструкции «Формирование и обработка массивов позиций всех уровней декомпозиции для множества ПЗ 1-го рода» где R1.16- Код ПЗ-1, R3.30- Количество уникальных кодов уровней декомпозиции, W3 (Код ПЗ-1- R1.16, Код уровня- R1.17, Количество позиций на уровне- R3.27), W3 (Код ПЗ-1- R1.16, Код уровня- R1.17, Количество позиций на уровне- R3.27, Статус уровня- R3.45), R3.44- Массив кодов позиций для всех уровней со статусом уровня=1.

Матричная диаграмма (МД) для типовой конструкции «цикл» (рис. 3.20) представлена на рис. 3.53. Основными параметрами являются: Счетчик итераций (СЦ)– Количество уникальных кодов уровней-R3.30; Аргумент итераций (АИ) - Код Код уровня- R1.17; Функция итераций (ФИ) - Количество кодов позиций на уровне R3.27, Функция цикла (ФЦ) - W3 (Код ПЗ-1- R1.16, Код Уровня- R1.17, Количество позиций на уровне- R3.27).

	СЧ	АИ	ФИ	А-у	ФЦ	
Позиция ПД	R3.30	R1.17	R3.27	R1.16	W3	
A4.7					{	
A5.7	!		$		[	
A6.9	.	*				
A6.10	**.**	+	*****			**[**
A6.11	?				#]	
A7.10	/ .				[	
A8.7	. *					
A8.8	.		+		*]	
A7.11	/ .		+		*}	во врем. Табл W3

Рис.3.53. Диаграмма конструкции «Определение количества позиций для всех рангов уровней декомпозиции ПЗ-1» где R1.16- Код ПЗ-1.

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.21) представлена на рис. 3.54. Основным параметром является: ФП - Количество кодов позиций на уровне R3.27.

		ФП		
Позиция ПД	R3.26	R3.27	R1.17	
A6.10		**[**		
A7.8	*****		+	
A7.9	**+**	***]**	**+**	

Рис.3.54. Диаграмма конструкции «Определение количества позиций для текущего ранга уровня декомпозиции» где R3.26- Ранг уровня, R1.17- Код уровня.

Матричная диаграмма (МД) для типовой конструкции цикл (рис. 3.22) представлена на рис. 3.55.

Основными параметрами являются: СЦ– Количество уникальных кодов уровней R3.30; АИ - Код уровня R 1.17, Количество позиции на уровне R 3.27; ФИ - Статус уровня R3.45; ФЦ - Массив новых кодов в Табл. W4 Код-R1.16.

	СЧ	АИ	АИ	ФИ	ФЦ	
Позиция ПД	R3.30	R 1.17	R3.27	R3.45	R1.16	
A4.8					{	
A5.8	!			$	[	
A6.12	.	*				
A6.13	**.**	**+**	**+**	*****		
A6.14	?				#]	
A7.12	/ .				[	
A8.9	. *					
A8.10	.			+	*]	
A7.13	/ .			+	*}	во врем Табл. W4

Рис. 3.55. Диаграмма конструкции «Определение статусов всех уровней декомпозиции»

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.23) представлена на рис. 3.56. Основным параметром является: ФП- Статус уровня R3.45.

			ФП	
Позиция ПД	R3.27	R 1.17	R3.45	……
A6.13			**[**	
A7.11a	*****	+		
A7.11б	+	+	***]**	

Рис.3.56. Диаграмма конструкции «Определение статуса текущего уровня декомпозиции» где R3.27- Количество позиции на уровне, R 1.17- Код уровня.

МД для конструкции «альтернатива» (рис. 3.24) представлена на рис.3.57. Основными параметрами являются: ФА - Статус уровня R3.45; АА- Количество позиции на уровне- R3.27, Код уровня R 1.17.

	АА	АА	ФА	Пояснения
Позиция ПД	R3.27	R 1.17	R3.45	
A7.11б	?	?	#	
A8.8a	/ .	/ .	*	
A8.8б	/.	/.	*	

Рис.3.57. Диаграмма конструкции «Присвоение ранга текущий позиции уровней декомпозиции с количеством позиций >1»

Матричная диаграмма (МД) для типовой конструкции «цикл» (рис. 3.25) представлена на рис. 3.58. Основными параметрами являются: СЦ– Количество позиции на уровне R 3.27; АИ - Код уровня R 1.17; ФИ Упорядоченный массив кодов позиций для тек. уровней R 1.42; ФЦ - Упорядоченный массив кодов позиций для всех уровней R 1.43.

	СЧ	АИ	ФИ	ФЦ	
Позиция ПД	R 3.27	R 1.17	R 1.42	R 1.43	
A4.9				{	
A5.9	!		$	[	
A6.15	.	*			
A6.16	**.**	**+**	*****		**[**
A6.17	?			#]	
A7.17	/ .			[	
A8.13	. *				
A8.14	.		+	*]	
A7.18	/ .		+	*}	во врем Табл. W4

Рис.3.58. Диаграмма конструкции «Упорядочение позиций для всех уровней декомпозиции со статусом=1»

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.26) представлена на рис. 3.59. Основным параметром является: ФП - Табл Код уровня R 1.17.

			ФП	
Позиция ПД	R3.26	R 3.27	R 1.17	……
A6.16			**[**	
A7.14	+	*****		
A7.15				**{ массив C_max)**
A7.16		**+**	***]**	**{ врем. Табл W5**

Рис. 3.59. Диаграмма конструкции «Упорядочение позиций для текущего уровней декомпозиции со статусом=1» где R3.26- Ранг уровня, R 3.27- Количество позиции на уровне.

Матричная диаграмма (МД) для типовой конструкции «цикл» (рис. 3.27) представлена на рис. 3.60. Основными параметрами являются: СЦ– Количество позиции на уровне R 3.27; АИ - Значение основного элемента ci -R3.36, Код уровня R1.17; ФИ- Текущий макс. элемент для основного элемента ci - R1.29-); ФЦ - Упорядоченный массив кодов позиций для тек. уровня (массив C_max) -R1.27.

	СЧ		АИ	ФИ	ФЦ	
Позиция ПД	R 3.27	R1.17	R3.36	R1.29	R1.27	
A7.15					{	
A8.11	!			$	[	
A9.2	.		*			
A9.3	**.**	+	+	*****		**[**
A9.4	?				#]	
A10.7	/ .				[	
A11.7	. *					
A11.8	.			+	*]	
A10.8	/ .			+	*}	во врем. Табл C,

Рис. 3.60. Диаграмма конструкции «Упорядочение исходного массива позиций по убыванию для текущего уровня декомпозиции с количеством позиций >1»

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.28) представлена на рис. 3.61. Основным параметром является: ФП - Текущий макс. элемент для основного элемента ci R1.29.

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.29) представлена на рис. 3.62. Основным параметром является: ФП- Текущий макс. элемент для основного элемента ci = Текущий макс. элемент c_max R1.33.

				ФП			
Позиция ПД	R1.28	R1.32	R3.36	R1.29	R3.37	R3.27	
A9.3				**[**			
A10.4	+	*****	+		.		
A10.5	+				*****	+	
A10.6				***][**			

Рис. 3.61. Диаграмма конструкции «Упорядочение массива для текущего уровня декомпозиции с количеством позиций >1» где R1.28- Код элемента ci, R1.32- Нач. макс. элемент n_max, R3.36- Элемент ci значение, R3.37- Количество элементов для сравне-ния (для позиций), R3.27- Количество кодов позиций.

				ФП		
Позиция ПД	R3.35	R1.34	R1.35	R1.33	R1.30	
A10.6				**[**		
A11.5		+	*****			
A11.6					**{***	{ Во временную табл.Di
A11.6a			+	***]**	+	во врем. Табл ai

Рисунок 3.62. Диаграмма конструкции «Определение максимального элемента позиции путем сравнения» где R3.35- Количество элементов для сравнения, R1.34- Индекс основногоэлемента i, R1.35- Начальный индекс сравниваемого элемента j, R1.30- Массив максимальных элементов C_max.

Матричная диаграмма (МД) для типовой конструкции «цикл» (рис. 3.30) представлена на рис. 3.63. Основными параметрами являются: СЦ– Количество элементов сравнения -R3.37; АИ - Элемент ci значение – R3.38, Элемент для сравнения значение cj R3.39; ФИ – Тек.максимальный элемент c_max-R1.33; ФЦ - Массив максимальных элеметов Cmax -R1.30.

	СЧ	АИ		ФИ	ФЦ	
Позиция ПД	R3.37	R3.38	R3.39	R1.33	R1.30	
A11.6					{	
A12.7	!			$	[	
A13.1	.	*				
A13.2	**.**	+		*****		**?#**
A13.3	?				#]	
A14.3	/ .				[	
A15.5	. *					
A15.6	.			+	*]	
A14.4	/ .			+	*}	

Рис. 3.63. Диаграмма конструкции «Определение максимальной позиции путем сравнение текущей основной позиции со всеми другими позициями»

Матричная диаграмма для конструкции «альтернатива» (рис. 3.31) представлена на рис.3.64.

Основными параметрами являются: ФА - Максимальный элемент–c_max R1.33; АА - Элемент ci Значение -R3.38, Элемент cj Значение -R3.39.

	АА	АА	ФА	Пояснения
Позиция ПД	R3.38	R3.39	R1.33	
A10.2	?	?	#	ci > cj
A11.1	/	/	**[***	**[c_max= ci**
A11.2	/	/	*	c_max= cj

Рис. 3.64. Диаграмма конструкции «Определение максимальной позиции в зависимости от сравнения текущей основной позиции с текущей позицией»

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.32) представлена на рис. 3.65.

Основным параметром является ФП - Максимальный элемент c_max R1.33.

				ФП	А-у	
Позиция ПД	R3.38	R3.39	R3.35	R1.33	R1.17	
A14.1				**[**		
A15.1	+		*		.	y=ci
A15.2	*	+			.	ci= cj
A15.3		*	+		.	cj= y
A15.4		+		***]**	.	c_max= ci

Рис.3.65. Диаграмма конструкции «Определение максимальной позиции при условии, что текущая основная позиция больше чем текущая позиция» где R3.38- Элемент ci Значение, R3.39- Элемент cj Значение, R3.35- Переменая значение, R1.17- Код уровня.

Матричная диаграмма (МД) для типовой конструкции «цикл» (рис. 3.33) представлена на рис. 3.66.

Основными параметрами являются: СЦ–Количество позиции на уровне - R3.27); АИ- Код уровня R 1.17, Код позиции R 1.18; ФИ - Ранг позиции уровня R 3.28; ФЦ- Массив рангов позиций R3.43.

Матричная диаграмма для типовой конструкции «цикл» (рис. 3.34) представлена на рис. 3.67. Основными параметрами являются: СЦ– Количество стат. ПЗ-1 (R3.1); АИ - Код ПЗ-1 (R1.16); Ранг уровня (R 3.26), Ранг позиции на уровне (R 3.28), ФИ – Общий ранг (R 3.29); ФЦ - Массив общих рангов (R 3.41).

	СЧ		АИ	ФИ	ФЦ	
Позиция ПД	R3.27	R 1.17	R 1.18	R3.28	R3.43	
A7.16					{	
A8.12	!			$	[	
A9.5	.		*			
A9.6	.	+	+	*		
A9.7	?				#]	
A10.9	/ .				[	
A11.9	. *					
A11.10	.	+	+	+	*]	
A10.10	/ .	+	+	+	*}	во врем. Табл W4

Рис.3.66. Диаграмма конструкции «Ранжирование всех позиций при фиксированном уровне»

	СЧ	АИ	АИ	АИ	ФИ	ФЦ	
Позиция ПД	R 3.1	R 1.16	R 3.26	R 3.28	R3.29	R3.41	
A3.8						{	
A4.10	!				$	[	
A5.10	.	*	*	*			
A5.11	.	+	+	+	*		
A5.12	?					#]	
A6.18	/ .					[	
A7.19	. *						
A7.20	.				+	*]	
A6.19	/ .				+	*}	во врем. Табл W5

Рис.3.67. Диаграмма конструкции «Формирование общего ранга для множества ПЗ 1-го рода»

Таким образом, полностью сформирована начальная модель для процедуры А как увязка системы предметных действий и множества параметров исходной задачи.

3.2.3 Разработка начальной модели для процедуры В

3.2.3.1 Формирование диаграмм системы предметных действий для процедуры В

Определение содержания для всех статистических ПЗ 1-го рода заключается в выявлении подмножества ПК для каждой ПЗ-1, которые связанны с этой ПЗ-1. Действие (В1.1) раскрывается как цикл (рис. 3.68).

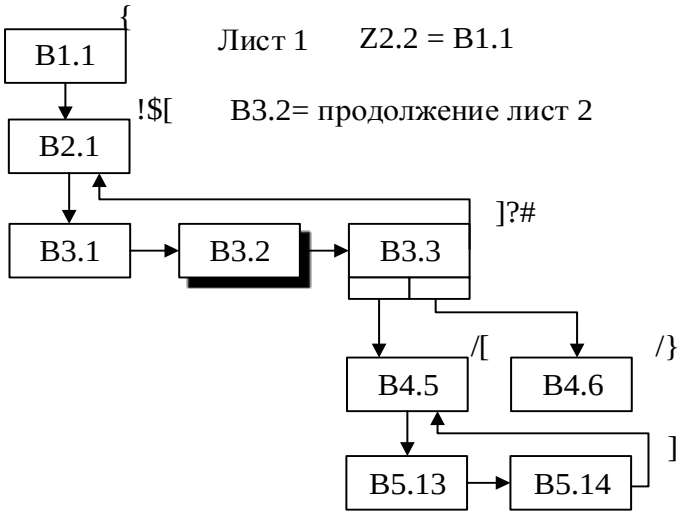


Рис. 3.68. Декомпозиция задачи «Определение содержания для всех статистических ПЗ 1-го рода – B1.1 » где B2.1- Определение содержания для i-й ПЗ 1-го рода, B3.1- Фиксация ПЗ 1-го рода, B3.2- Определение содержания для текущей ПЗ 1-го рода B3.3- Определение содержания всех статистических ПЗ 1-го рода в зависимости от их наличия, B4.5- Определение содержания всех статистических ПЗ 1-го рода при наличии B5.13- Переход к следующей ПЗ 1-го рода i=i+1, B5.14- Определение содержания всех статистических ПЗ 1-го рода при их наличии для предыдущей статистической ПЗ 1-го рода, B4.6- Определение содержания всех статистических ПЗ 1-го рода при их отсутствии.

Действие B3.3 раскрывается как альтернатива. При наличии следующей ПЗ 1 происходит переход к следующей итерации i=i+1 B5.5 и определяют содержания всех статистических ПЗ 1-го рода при их наличии для предыдущей статистической ПЗ 1-го рода B5.6. А в случае отсутствия определяют содержание всех статистических ПЗ 1-го рода и происходит выход из цикла B4.5.

На рис. 3.69 показано, что для определения содержания для текущей ПЗ-1 (B3.2) выполняется последовательность, раскрывающаяся на 4 сложных действия.

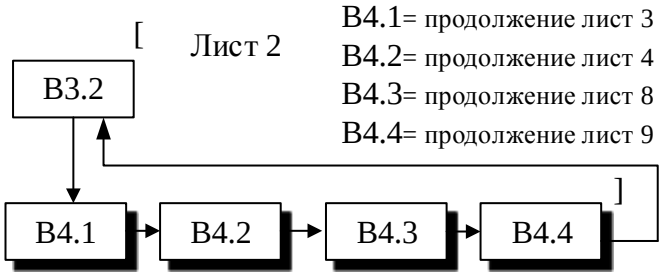


Рис.3.69. Декомпозиция задачи «Определение содержания для текущей ПЗ 1-го рода» где B4.1- Формирование кодов структур, B4.2-Запись кодов структуры в табл.«Структуры »для текущей ПЗ 1-го рода, B4.3 Запись по коду ПЗ-1 из таблицы F3 в табл.«F3c» с текущим кодом структуры, B4.4-Определение количества и список ПК, связанных с текущей ПЗ 1-го рода для текущего кода структуры увязки.

В первом действии (В4.1) необходимо формировать коды структур, в втором действии (В4.2) выполняется запись кодов структур в табл. «Структуры» для текущей ПЗ-1 и в действии (В4.3) запись по коду ПЗ-1 из табл. «FЗ» в табл. «FЗс» с новым кодом динамической структуры. В последнем действии (В4.4) необходимо определить количество и список ПК, связанных с текущей ПЗ-1 для текущего кода структуры увязки.

На рис. 3.70 показано, что формирование кодов структуры (В4.1) является последовательностью, раскрывающейся на 4 действия. В первом действии необходимо формировать код статической структуры (В5.1), в действии (В5.2) формировать код динамической структуры и в действии (В5.3) формировать код структуры увязки. В действии (В5.4) необходимо запись в таблицу «Состав концептуальных моделей из структур» (все в строке по коду КМ).

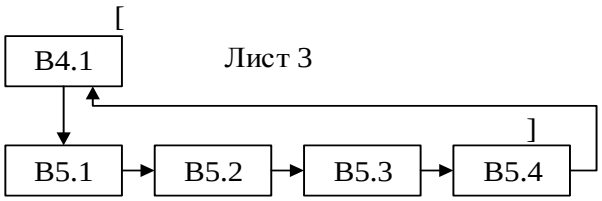


Рис. 3.70. Декомпозиция проектной задачи «Формирование кодов структур» где В5.1- Формирование кода статической структуры, В5.2- Формирование кода динамической структуры, В5.3-Формирование кода структуры увязки, В5.4-Запись в табл. «Состав концептуальных моделей из структур» (все в строке по коду КМ).

На рис. 3.71 показано, что для фиксации кодов структуры в табл. «Структуры» для текущей ПЗ 1-го рода (В4.2) необходимо описать статическую структуру (В5.5), динамическую структуру (В5.6) и структуру увязки (В5.7).

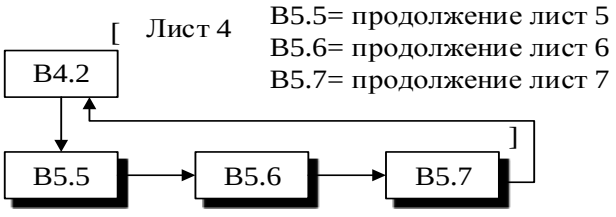


Рис.3.71. Декомпозиция задачи «Запись кодов структуры в табл. «Структуры» для текущей ПЗ 1-го рода- В4.2» где В5.5-Описание статической структуры, В5.6- Описание динамической структуры, В5.7-Описание структуры увязки.

Действие (В5.5) является последовательностью на рис.3.72. В действии (В6.1) необходимо считать из табл. «Состав концептуальных моделей из структур», переопределение кода структуры (В6.2) и запись в табл. «Структуры» (В6.3).

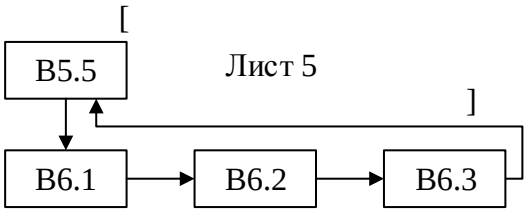


Рис.3.72. Декомпозиция задачи «Описание статической структуры- В5.5» где В6.1- Считывание из табл.«Состав концептуальных моделей из структур», В6.2- Переопределение кода структуры, В6.3- Запись в табл. «Структуры»

На рис. 3.73 показано, что для описания динамической структуры (В5.6) необходимо считать из табл. «Состав концептуальных моделей из структур» (В6.4), переопределение кода структуры (В6.5) и запись в табл. «Структуры» (В6.6).

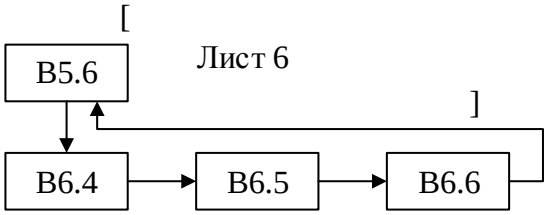


Рис.3.73. Декомпозиция задачи «Описание динамической структуры- В5.6» где В6.4- Считывание из табл.«Состав концептуальных моделей из структур», В6.5- Переопределение кода структуры, В6.6- Запись в табл. «Структуры».

На рис. 3.74 показано, что для описания структуры увязки (В5.7) необходимо считать из табл. «Состав концептуальных моделей из структур» (В6.7), переопределение кода структуры (В6.8) и запись в табл. «Структуры» (В6.9).

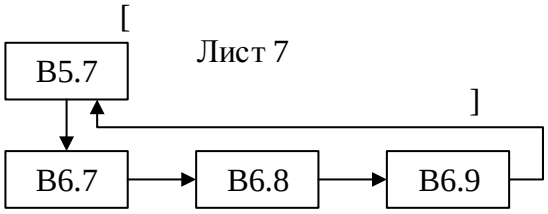


Рис.3.74. Декомпозиция задачи «Описание структуры увязки- В5.7» где В6.7- Считывание из табл.«Состав концептуальных моделей из структур», В6.8- Переопределение кода структуры, В6.9- Запись в табл. «Структуры».

Действие (В4.3) является последовательностью (рис. 3.75). В действии (В5.8) необходимо считать из табл. F3, описание текущей ПЗ-1 по коду динамической структуры для переменной информации и добавить эту запись в таблицу F3с с текущим ПЗ-1 по коду новой динамической структуры для постоянной информации (В5.9).

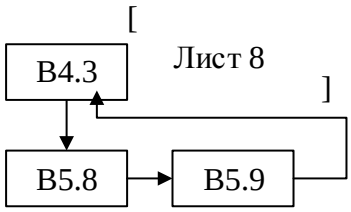


Рис.3.75. Декомпозиция задачи «Запись по коду ПЗ-1 из таблицы F3 в табл. «F3c» с текущим кодом структуры- B4.3» где B5.8- Считание из табл. F3, Описание текущий ПЗ-1 по коду динамической структуры (Var), B5.9- Запись в табл. F3c с текущим ПЗ-1 по коду новой динамической структуры (const).

На рис. 3.76 показано, что для определения количества и списка ПК, связанных с текущей ПЗ-1 для текущего кода структуры увязки (B4.4) необходимо выполнить последовательность, раскрывающуюся на 3 действия. В первом действии нужно определить количество ПК (из таблицы F6) (B5.10), потом формировать список ПК. связаныых с текущей ПЗ-1 (из таблицы F6 в таблицу W6) в действии (B5.11) и в последнем действии (B5.12) определяется содержание текущей ПЗ-1 (раскрывается как цикл на рис.3.77).

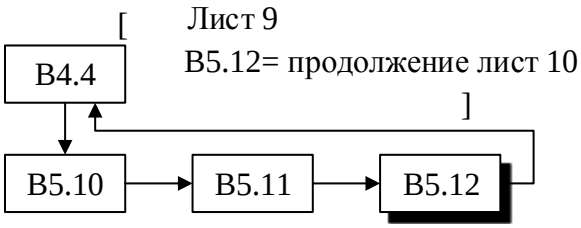


Рис. 3.76. Декомпозиция задачи «Определение количества и список ПК, связанных с текущей ПЗ 1-го рода для текущего кода структуры увязки» где B5.10- Определение количества ПК(Nув) (из таблицы F6), B5.11- Формировние списка ПК связаныых с текущим ПЗ-1 (из таблицы F6 в таблицу W6), B5.12- Определение содержания множества ПК для текущей ПЗ 1-го рода

Следующее действие «Определение множества ПК для текущей ПЗ 1-го рода» раскрывается как цикл (рис. 3.77). В действии (B6.10) происходит определение i-ой ПК для текущей ПЗ-1, фиксируется код ПК (B7.1) и выполняется сложное действие (B7.2). Альтернатива B7.3 определяет продолжение цикла. При наличии следующей ПК для текущей ПЗ-1 выполняется переход к следующей ПК i=i+1 (B9.1) и определяется множество ПК для текущей ПЗ-1 при их наличии для предыдущей ПК (B9.2). А в случае отсутствия, определяется множество ПК для текущей ПЗ-1 и происходит выход из цикла (B8.4).

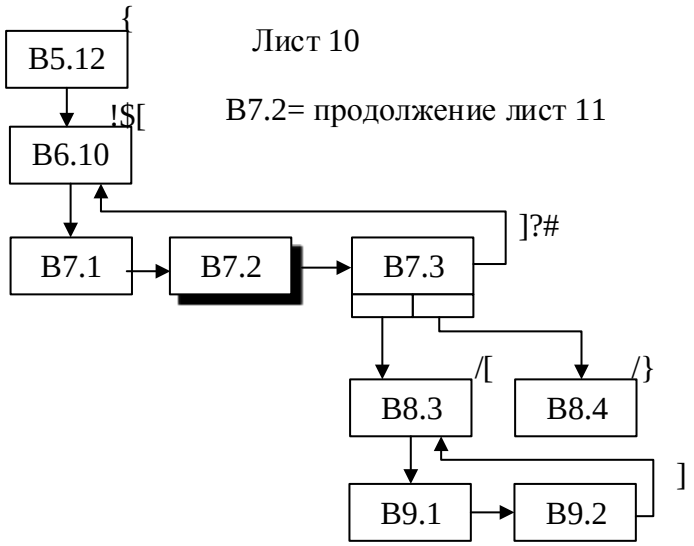


Рис.3.77. Декомпозиция задачи «Описание множества ПК для текущей ПЗ 1-го рода- B5.12» где B6.10- Определение i-ой ПК для текущей ПЗ 1-го рода, B7.1- Фиксация i-ой ПК, B7.2- определение текущей ПК для текущей ПЗ 1-го рода, B7.3- Определение множества ПК для текущей ПЗ 1-го рода в зависимости от их наличия, B8.3- Определение множества ПК для текущей ПЗ 1-го рода при наличии, B9.1- Переход к следующей ПК i=i+1, B9.2- Определение множества ПК для текущей ПЗ 1-го рода при их наличии для предыдущей ПК, B8.4- определение множества ПК для текущей ПЗ 1-го рода при их отсутствии.

На рис. 3.78 показано, что определение текущей ПК для текущей ПЗ-1– действие (B7.2) является последовательностью, раскрывающейся на 2 действия. В первом действии необходимо считать из временной табл. W6 по ключам (тек. код ПК и тек. код ПЗ-1) запись (B8.1) и во втором действии необходимо добавлять эту запись в табл. F6c с новым текущим кодом структуры (B8.2).

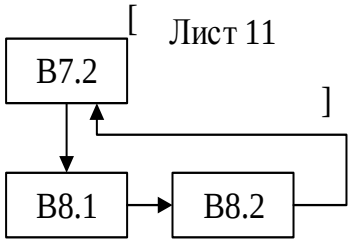


Рис.3.78. Декомпозиция задачи «Описание текущей ПК для текущей ПЗ 1-го рода- B7.2» где B8.1- Считание из табл. W6 по ключам(тек.ПК и тек.ПЗ-1) запись, B8.2- Добавление эту запись в табл. F6c с текущим кодом структуры и ключам

Таким образом полностью сформирована система предметных действий для второй процедуры «Определение содержания для всех статистических ПЗ 1-го рода» (процедура B) при начальном моделировании задачи «Обработка статистических предметных ограничений при концептуальном моделировании».

3.2.3.2 Формирование диаграммы системы предметных действий в целом для процедуры B

В целом диаграмма системы предметных действий для процедуры B приведена на рис.3.79.

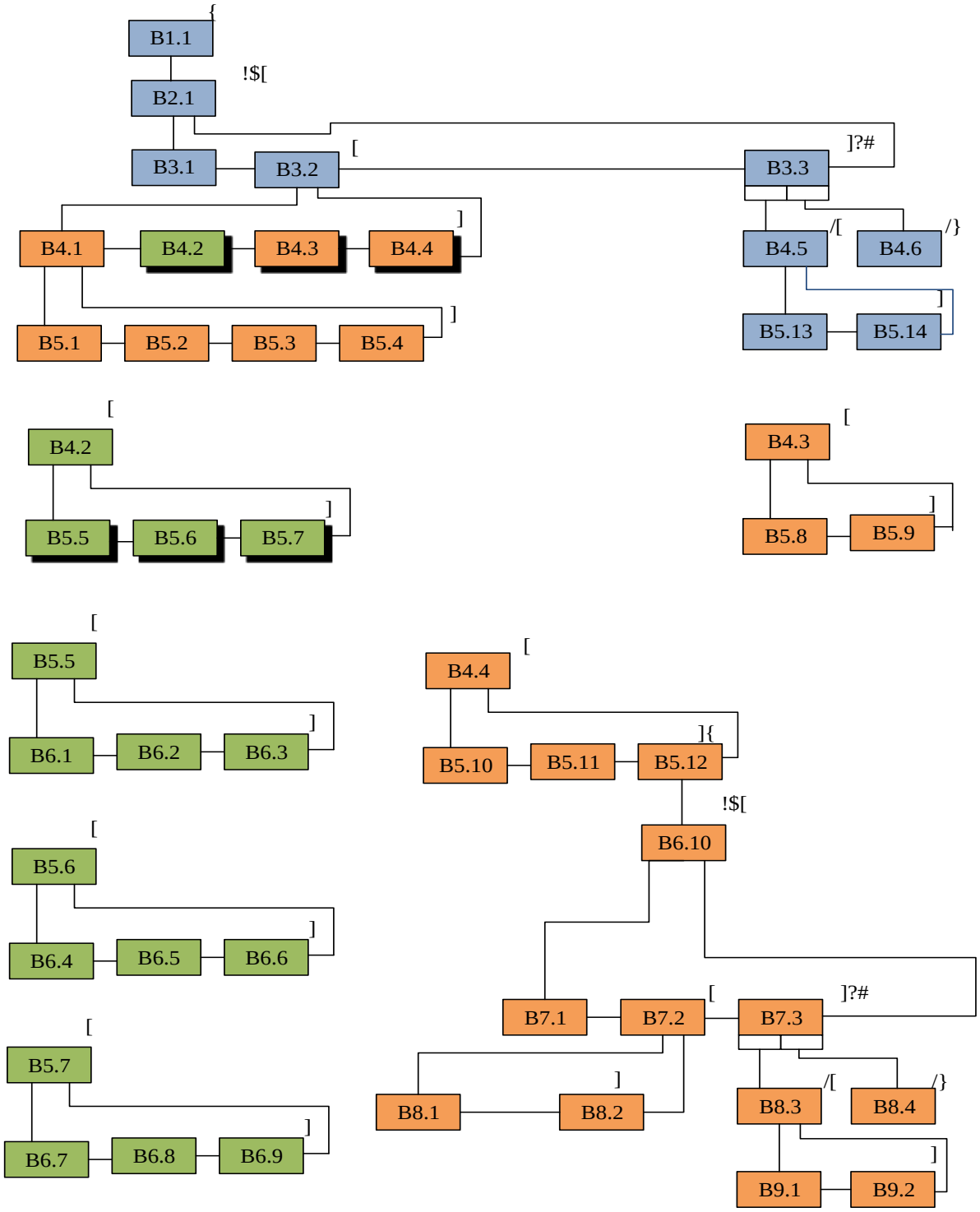


Рис. 3.79. Диаграмма для процедуры B

3.2.3.3 Формирование множества параметров для процедуры B

В целом структура организации данных для процедуры B приведена на рис.3.80.

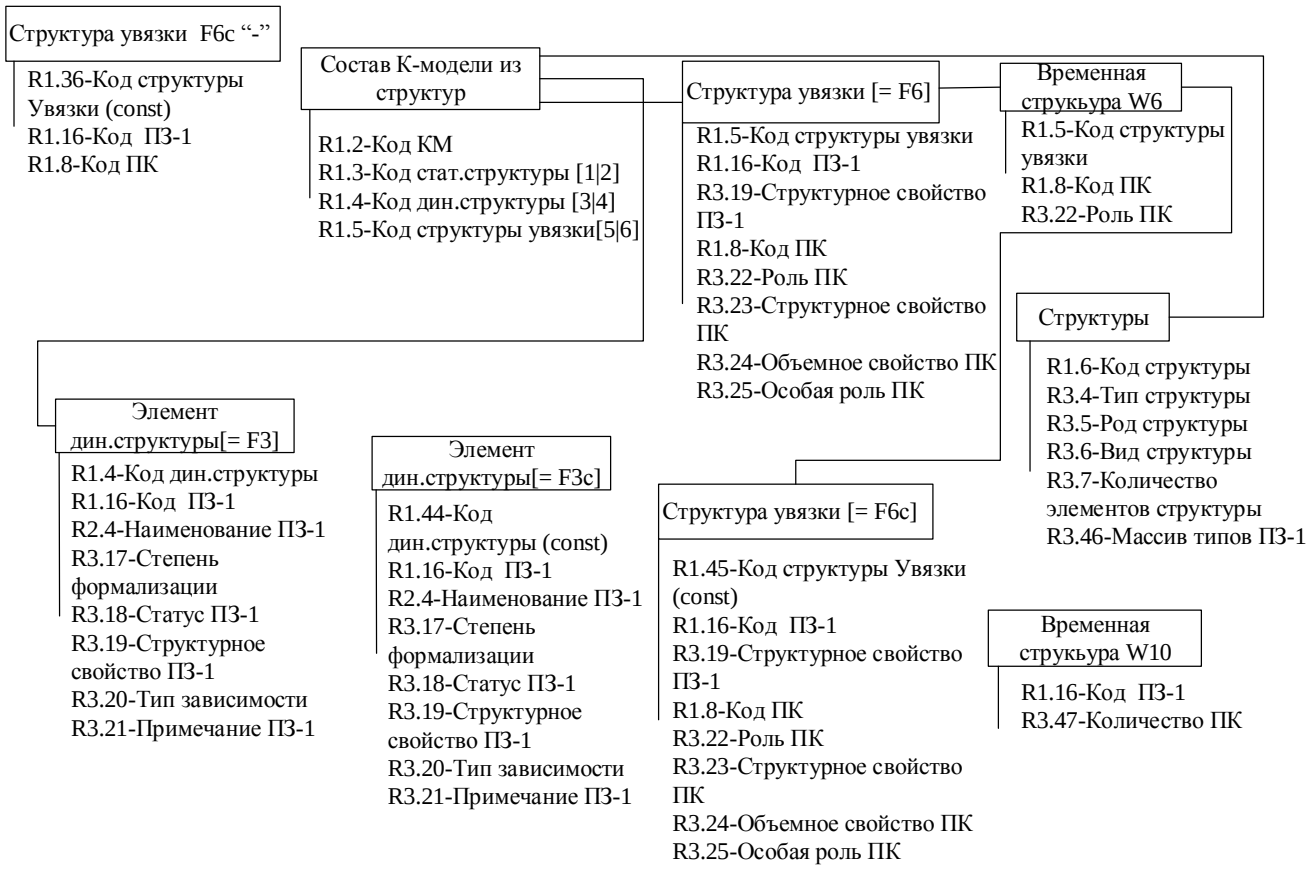


Рис.3.80. Структура организации данных для процедуры B

Таким образом, разработана информационная составляющая начальной модели для процедуры B.

3.2.3.4 Формирование матричных диаграмм для всех конструкций процедуры B

Матричная диаграмма (МД) для типовой конструкции «цикл» (рис. 3.68) представлена на рис. 3.81.

Основными параметрами являются следующие: СЦ– Количество стат. ПЗ-1 (R3.1); АИ - код стат. ПЗ-1 (R1.16); ФИ - Массив кодов ПК для текущего кода стат. ПЗ-1(R1.8); ФЦ - Массив кодов ПК для всех кодов стат. ПЗ-1 (R1.8).

	СЧ	АИ	ФИ	ФЦ	Пояснения
Позиция ПЗ	R3.1	R1.16	R1.8	R1.8	
B1.1				{	
B2.1	!		$	[	
B3.1	.	*			
B3.2	**.**	**+**	*****		**[**
B3.3	?			#]	
B4.5	/ .			[	
B5.13	. *				
B5.14	.	+	+	*]	
B4.6	/ .	+	+	*}	В табл.F6c “-”

Рис. 3.81. МД конструкции «Определение содержания для всех статистических ПЗ-1»

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.69) представлена на рис. 3.82.

Основным параметром является ФП - Массив кодов ПК для текущего кода стат. ПЗ-1(R1.8).

			ФП	Пояснения
Позиция ПД		R1.16	R1.8	
B3.2			**[**	
B4.1		**,**		**В табл «Состав КМ из структур»**
B4.2		**,**		**В табл «структуры»**
B4.3		**,**		**В табл «F3c»**
B4.4		**,**	***]**	**В табл.F6c “-”**

Рис. 3.82. Матричная диаграмма конструкции «Определение содержания для текущей ПЗ-1» где R1.16- Код стат. ПЗ-1

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.70) представлена на рис. 3.83.

Основным параметром является ФП – запись/кортеж в табл. «Состав КМ из структур» (Код КМ R1.2, Код стат. структуры R1.3, Код дин. структуры R1.4, Код структуры увязки R1.5).

						ФП	Пояснения
Позиция ПД	R1.16	R1.3	R1.4	R1.5	R1.2	Запись «Состав КМ из структур»	
B4.1						**[**	
B5.1	+	*			+		
B5.2	+		*		+		
B5.3	+			*	+		
B5.4		+	+	+	+	***]**	Запись в табл. «Состав КМ из структур»

Рис. 3.83. Матричная диаграмма конструкции «Формирование кодов структур КМ» где R1.16- Код стат. ПЗ- 1, R1.3- Код стат. стр-ры, R1.4- Код динам. стр-ры, R1.5- Код стр-ры Увязки, R1.2- Код КМ.

Матричная диаграмма для сложной конструкции вложенной «последовательности» (рис. 3.71-3.74) представлена на рис. 3.84. Основным параметром является ФП – Кортеж/запись в табл. «Структуры» (Код структуры- R1.6, Тип структуры- R3.4, Род структуры- R3.5, Вид структуры- R3.6).

									ФП	Поя-снения
Позиция ПД	R1.2	R1.3	R3.4	R3.5	R3.6	R1.4	R1.5	R1.6,	Кортеж в Табл. «структуры»	
B4.2									**[**	#
B5.5									**[**	
B6.1	+	*								(1)
B6.2		+						*		
B6.3			+	+	+			+	***]**	(2)
B5.6									**[**	
B6.4	+					*				(1)
B6.5						+		*		
B6.6			+	+	+			+	*]	(2)
B5.7									[	
B6.7	+						*			(1)
B6.8							+	*		
B6.9			+	+	+			+	***]]**	(2)

Рис.3.84. Матричная диаграмма конструкции «Описание структур для текущей ПЗ-1», где (1) Из табл. Состав считывание (2)В табл. Структуры запись, R1.2- Код КМ, R1.3- Код стат. струк-туры, R3.4- Тип структуры, R3.5- Род структуры, R3.6- Вид структуры, R1.4- Код дин. Структуры, R1.5- Код структуры увязки, R1.6- Код структуры,

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.75) представлена на рис. 3.85. Основным параметром является ФП - Кортеж $f3^c$ в табл.F3c.

						ФП	Поясне-ния
Позиция ПД	R1.4	R1.16	R3.17	Кортеж в табл. f3	Код дин.структ-уры (const)- R1.44	Кортеж в $f3^c$	
B4.3						**[**	
B5.8	+	+	+	*****			из табл «F3» считыван ие
B5.9				+	+	***]**	в табл «F3c» запись

Рис. 3.85. Матричная диаграмма конструкции «Запись по коду ПЗ-1 из таблицы F3 в табл. «F3c» с текущим кодом структуры» где R1.4- Код дин.структуры, R1.16- Код стат.ПЗ-1, R3.17- Степень формализации.

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.76) представлена на рис. 3.86. Основным параметром является ФП - Массив кортеж в Табл. $f6^c$.

					ФП	Пояснения
Позиция ПД	R1.5	R1.16	R3.47	W6	Массив кортежей в Табл. $f6^c$	
B4.4					**[**	
B5.10	+	+	*****			Из табл.F6
B5.11	+	+		*****		Из табл. F6 в табл. W6 (список ПК)
B5.12				+	***]**	

Рис. 3.86. Матричная диаграмма конструкции «Определение количества и списка ПК, связанных с текущей ПЗ-1 для текущего кода структуры увязки» где R1.5- Код структуры увязки, R1.16- Код стат.ПЗ-1, R3.47- Количество кодов ПК, W6(Код структуры увязки R1.5, Код стат.ПЗ-1-R1.16, Код ПК - R1.8, Роль ПК - R3.22)

Матричная диаграмма для типовой конструкции «цикл» (рис. 3.77) представлена на рис. 3.87. Конструкция представляет собой цикл. Основными

параметрами являются следующие: СЦ–Количество ПК R3.47; АИ - Код ПК R1.8; ФИ - Кортеж в табл. $f6^c$; ФЦ- Массив кортежей в табл. $f6^c$.

.	СЧ	АИ	ФИ	ФЦ	Пояснения
Позиция ПД	R3.47	R1.8	Кортеж в табл.$f6^c$	Массив кортежей в табл. $F6^c$	
B5.12				{	
B6.10	!		$	[	
B7.1	.	*			
B7.2	**.**	**+**	*****		**[**
B7.3	?			#]	
B8.3	/ .			[	
B9.1	. *				
B9.2	.	+	+	*]	Массив кортежей табл. $F6^c$
B8.4	/ .	+	+	*}	Массив кортежей табл. $F6^c$

Рис. 3.87. Матричная диаграмма конструкции «Определение множества ПК для текущей ПЗ-1»

Матричная диаграмма для базовой конструкции «последовательность» (рис. 3.78) представлена на рис. 3.88. Основным параметром является ФП - Кортеж $f6^c$ с кодом структуры увязки R1.45 (const).

					ФП	Пояснения
Позиция ПД	R1.8	R1.16	Кортеж w6	R1.5	Кортеж $f6^c$ (const)	
B7.2					**[**	
B8.1	+	+	*****			Из табл.W6 считывание
B8.2			+	+	*]	в табл «F6c» запись

Рис. 3.88. Матричная диаграмма конструкции «Определение текущей ПК для текущей ПЗ-1» где R1.8- Код ПК, R1.16- Код ПЗ-1, w6 (Код ПЗ-1-R1.16, Код ПК - R1.8,Роль ПК - R3.22), R1.5- Код структры увязки

Таким образом, полностью сформирована начальная модель для процедуры В как увязка системы предметных действий и множества параметров исходной задачи.

3.2.4 Разработка начальной модели для процедуры С

3.2.4.1 Формирование диаграмм системы предметных действий для процедуры С

Процесс формирование концептуальных моделей (КМ) под постоянную информацию (или для всех статистических ПЗ 1-го рода) заключается в построении «родословных» для всех ПК, связанных со статистической ПЗ-1, определяющих соответствующую КМ 1-го рода.

Действие «Формирование КМ для всех статистических ПЗ-1-го рода» раскрывается как цикл (на рис. 3.89). В действии (С2.1) происходит Формирование КМ для i-ой статистической ПЗ 1-го рода. Далее она фиксируется в действии (С3.1) и выполняется сложное действие (С3.2).

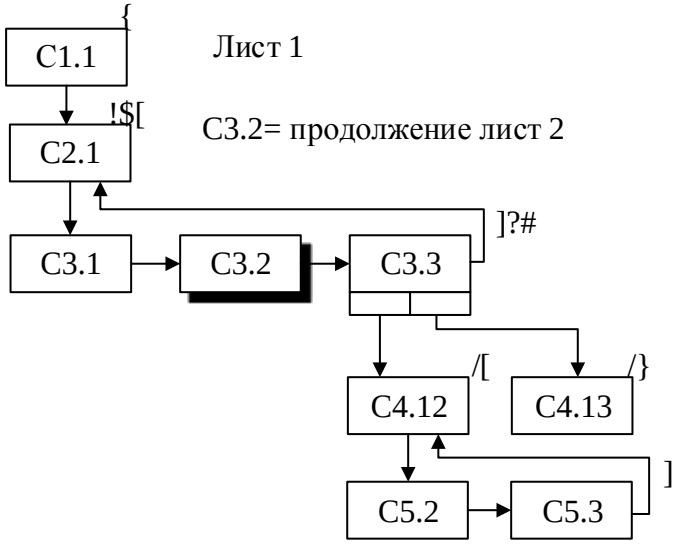


Рис. 3.89. Декомпозиция проектной задачи «Формирование “родословных” структур для всех статистических ПЗ-1» где С2.1- Формирование “родословных” структур для i -ой стат.ПЗ-1, С3.1- Фиксация i -го стат.ПЗ-1, С3.2-Формирование “родословных” структур для текущей статистических ПЗ-1-го рода (формирование множества цепочек), С3.3- Формирование “родословных” структур для всех статистических ПЗ-1-го рода в зависимости от их наличии, С4.12- Формирование “родословных” структур для всех статистических ПЗ-1-го рода при их наличии, С5.2- Переход к следующему код ПЗ-1i=i+1, С5.3- Формирование “родословных” структур для всех статистических ПЗ-1-го рода при их наличии для предыдущего стат.ПЗ-1, С4.13- Формирование “родословных” структур для всех статистических ПЗ-1-го рода при их отсутствии.

Действие «Формирование КМ для всех статистических ПЗ 1-го рода в зависимости от их наличия (С3.3)» раскрывается как альтернатива. При наличии следующей статистической ПЗ-1 происходит переход к следующей ПЗ 1-го рода i=i+1 (С5.2) и формирует КМ для всех статистических ПЗ 1-го рода при их наличии для предыдущего статистического ПЗ 1-го рода (С5.3).

А в случае отсутствия, формирует КМ для всех статистических ПЗ-1-го рода и происходит выход из цикла (С4.13).

На рис. 3.90 показано, что «Формирование “родословных” структур для текущей статистических ПЗ-1-го рода (формирование множества цепочек)» (С3.2) является последовательностью, раскрывающейся на 2 действия.

В первом действии (С4.1) необходимо определить список предметных категорий для текущей статистической ПЗ 1-го рода и во втором действии необходимо сформировать цепочки “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней (С4.2).

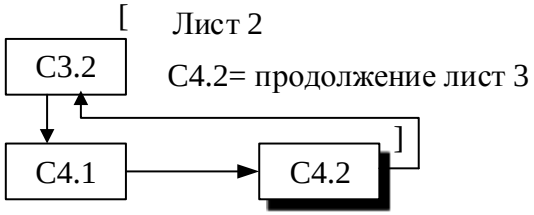


Рис. 3.90. Декомпозиция задачи «Формирование “родословных” структур для текущей статистических ПЗ-1-го рода (формирование множества цепочек)» где С4.1-Определение списка ПК, С4.2- Формирование цепочки “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней.

В действии «Формирование цепочек “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней» раскрывается как цикл (на рис. 3.91). В действии (С5.1) происходит формирование цепочки для i -ой ПК. Далее ПК фиксируется (С6.1) и выполняется сложное действие (С6.2). Действие «Формирование цепочки “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней» (С6.3) раскрывается как альтернатива. При наличии следующей ПК происходит переход к к ней i=i+1 (С8.26) и формируют цепочку “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней при их наличии для

предыдущей ПК (С8.27). А в случае отсутствия, формируют цепочку “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней, и происходит выход из цикла (С7.7).

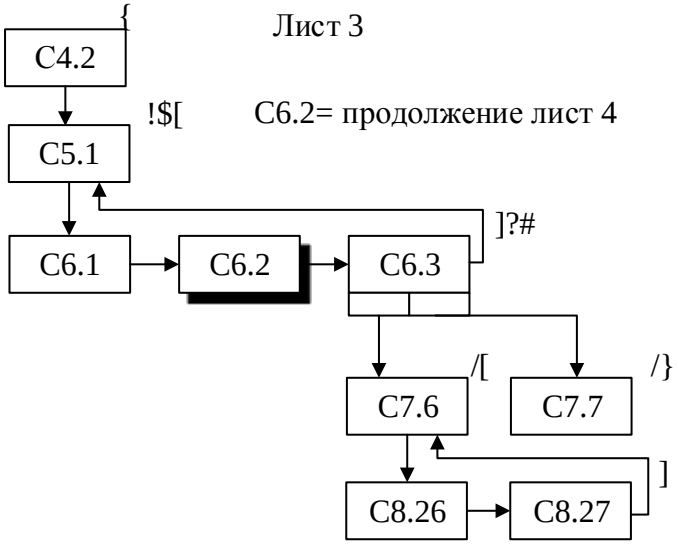


Рис. 3.91. Декомпозиция задачи «Формирование цепочек “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней» где С5.1- Формирование цепочки для i -ой ПК, С6.1- Фиксация цепочка i -ой ПК, С6.2-Формирование цепочки “родословных” структур для текущей ПЗ-1 для текущей ПК класса «R», С6.3- Формирование “i”-ой цепочки “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней в зависимости от их наличии, С7.6- Формирование “i”-ой цепочки “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней при их наличии, С8.26- Переход к следующему ПК i=i+1, С8.27- Формирование “i”-ой цепочки “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней при их наличии для предыдущей ПК, С7.7- Формирование “i”-ой цепочки “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней при их отсутствии.

На рис. 3.92 показано, что формирование цепочки “родословных” структур для текущей ПЗ-1 для текущей ПК класса «R» (С6.2) является последовательностью, раскрывающейся на 5 действий.

В первом действии необходимо формировать фрагмент цепочки по классу связи «OR» (С7.1), в действии (С7.2) выполняется «Формирование фрагмента цепочки по классу связи «KO», в действии (С7.3), (С7.4) и (С7.5) выполняются «Формирование фрагмента цепочки по классу связи, соответственно, «ZK», «PZ», «SP»».

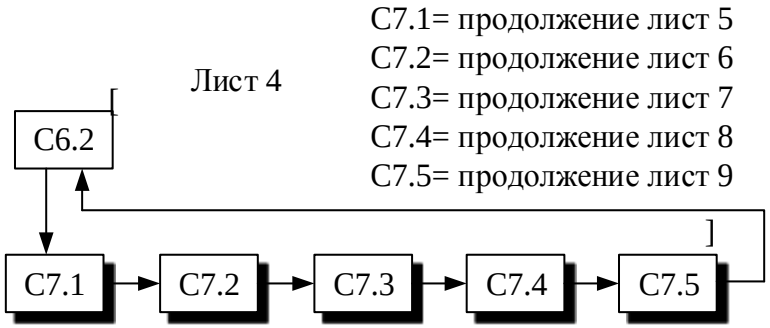


Рис. 3.92 Декомпозиция задачи «Формирование цепочки “родословных” структур для текущей ПЗ-1 для текущей ПК класса «R»» где С7.1-Формирование фрагмента цепочки по классу связи «OR», С7.2-Формирование фрагмента цепочки по классу связи «KO», С7.3-Формирование фрагмента цепочки по классу связи «ZK», С7.4-Формирование фрагмента цепочки по классу связи «PZ», С7.5-Формирование фрагмента цепочки по классу связи «SP».

Действие (С7.1) является последовательностью, раскрывающейся на 5 действий (рис. 3.93).

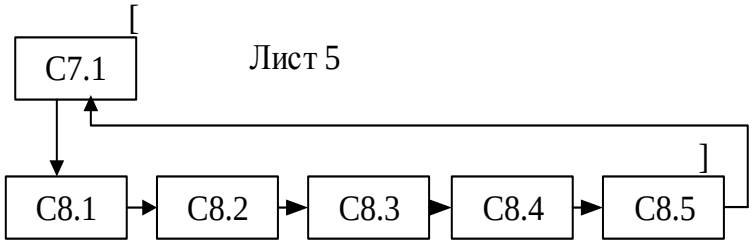


Рис. 3.93. Декомпозиция задачи «Формирование фрагмента цепочки по классу связи «OR»» где С8.1-Считывание из табл. F1 запись по тек.коду ПК класса «R» (Описание ПК), С8.2-Добавление этой записи в табл. F1c с новым кодом структуры, С8.3- Считывание из табл. F2 код.ПК-1=x в связи, в которой класс связи= «OR», Тип связи=C, код ПК-2= текущий код ПК класса «R» (Описание связи ПК), С8.4- Запись в табл. F2c с новым кодом структуры класс связи= «OR», Тип связи=C, код.ПК-1=x, код ПК-2= текущий код ПК класса «R», С8.5- Запись в табл. W7, A=текущий код ПК класса «R», B=x=Код ПК класса «O».

В первом действии выполняется «Считывание из табл. F1 записи по тек.коду ПК класса «R» (Описание ПК) (С8.1)», в действии (8.2) необходимо добавить этой записи в табл. F1c с новым кодом структуры, в действии (8.3) выполняется «Считывание из табл. F2 код ПК-1 (x) в связи, в которой класс связи= «OR», тип связи=C, код ПК-2= текущий код ПК класса «R» (Описание связи ПК) (С8.3)». А именно: «Запись в табл. F2c с новым кодом структуры класс связи= «OR», Тип связи=C, код ПК-1=x, код ПК-2= текущий код ПК класса «R» (С8.4)» и «Запись в

временную табл. W7, W7.A=текущий код ПК класса «R», и в последнем действии выполняется W7.B=x= код ПК класса «O» (C8.5)».

На рис. 3.94 показано, что формирование фрагмента цепочки по классу связи «KO»» (C7.2) является последовательностью, раскрывающейся на 5 действий. А именно: «Считывание из табл. F1 запись по тек.коду ПК класса «O» (Описание ПК) (C8.6)», «Добавление этой записи в табл. F1c с новым кодом структуры (C8.7)», «Считывание из табл. F2 код.ПК-1(x) в связи, в которой класс связи= «KO», Тип связи=C, код ПК-2= текущий код ПК класса «O» (Описание связи ПК) (C8.8)», «Запись в табл. F2c с новым кодом структуры класс связи= «KO», Тип связи=C, код.ПК-1=x, код ПК-2= текущий код ПК класса «O» (C8.9)» и «Запись в табл. W7, W7.C=x= Код ПК класса «K» (C8.10)».

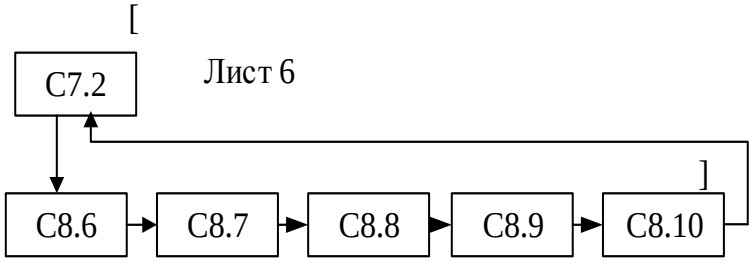


Рис. 3.94. Декомпозиция задачи «Формирование фрагмента цепочки по классу связи «KO»»
где C8.6-Считывание из табл. F1запись по тек.коду ПК класса «O» (Описание ПК),
C8.7-Добавление этой записи в табл. F1c с новым кодом структуры, C8.8- Считывание из табл. F2 код.ПК-1=x в связи, в которой класс связи= «KO», Тип связи=C, код ПК-2= текущий код ПК класса «O» (Описание связи ПК), C8.9- Запись в табл. F2c с новым кодом структуры класс связи= «KO», Тип связи=C, код.ПК-1=x, код ПК-2= текущий код ПК класса «O»,
C8.10- Запись в табл. W7, B=текущий код ПК класса «O», C=x= Код ПК класса «K».

Действие (C7.3) является последовательностью, раскрывающейся на 5 действий (рис. 3.99). В первом действии выполняется «Считывание из табл. F1 запись по тек.коду ПК класса «K» (Описание ПК) (C8.11)», в действии (8.12) необходимо добавить этой записи в табл. F1c с новым кодом структуры, в действии (8.13) выполняется «Считывание из табл. F2 код.ПК-1(x) в связи, в которой класс связи= «ZK», Тип связи=C, код ПК-2= текущий код ПК класса «K» (Описание связи ПК)». А именно: «Запись в табл. F2c с новым кодом структуры класс связи= «ZK», Тип связи=C, код.ПК-1=x, код ПК-2= текущий код ПК класса «K» (C8.14)» и в

последнем действии выполняется «Запись в табл. W7, W7.D=x= Код ПК класса «Z» (C8.15)».

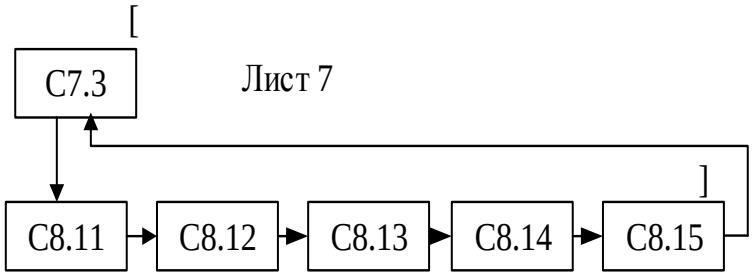


Рис. 3.95. Декомпозиция задачи «Формирование фрагмента цепочки по классу связи «ZK»» где C8.11-Считывание из табл. F1запись по тек.коду ПК Класса «K» (Описание ПК), C8.12-Добавление этой записи в табл. F1c с новым кодом структуры, C8.13- Считывание из табл. F2 код.ПК-1→x в связи, в которой класс связи= «ZK», Тип связи=C, код ПК-2= текущий код ПК класса «K» (Описание связи ПК), C8.14- Запись в табл. F2c с новым кодом структуры класс связи= «ZK», Тип связи=C, код.ПК-1=x, код ПК-2= текущий код ПК класса «K», C8.15-Запись в табл. W7, C=текущий код ПК класса «K», D=x= Код ПК класса «Z».

На рис. 3.96 показано, что формирование фрагмента цепочки по классу связи «PZ»» (C7.4) является последовательностью, раскрывающейся на 5 действий.

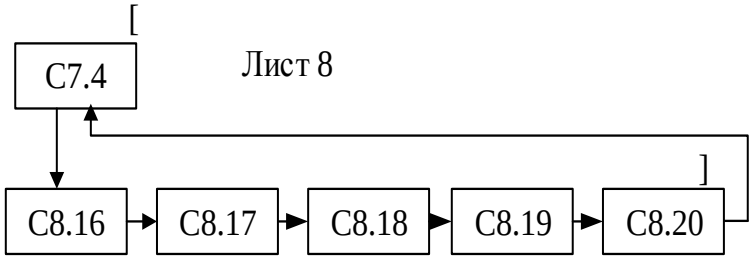


Рис. 3.96. Декомпозиция задачи «Формирование фрагмента цепочки по классу связи «PZ»» где C8.16-Считывание из табл. F1 запись по тек.коду ПК класса «Z» (Описание ПК), C8.17-Добавление этой записи в табл. F1c с новым кодом структуры, C8.18- Считывание из табл. F2 код.ПК-1→x в связи, в которой класс связи= «PZ», Тип связи=C, код ПК-2= текущий код ПК класса «Z» (Описание связи ПК), C8.19- Запись в табл. F2c с новым кодом структуры класс связи= «PZ», Тип связи=C, код.ПК-1=x, код ПК-2= текущий код ПК класса «Z», C8.20- Запись в табл. W7, D=текущий код ПК класса «Z», E=x= Код ПК класса «P».

В первом действии выполняется «Считывание из табл. F1 запись по тек.коду ПК класса «Z» (Описание ПК) (C8.16)», в действии (C8.17) выполняется «Добавление этой записи в табл. F1c с новым кодом структуры». А именно: «Считывание из табл. F2 код.ПК-1(x) в связи, в которой класс связи= «PZ», Тип связи=C, код ПК-2= текущий код ПК класса «Z» (Описание связи ПК) (C8.18)», «Запись в табл. F2c с новым кодом структуры класс связи= «PZ», Тип связи=C,

код.ПК-1=x, код ПК-2= текущий код ПК класса «Z» (C8.19)» и в последнем действии выполняется «Запись в табл. W7, W7.E=x= Код ПК класса «P» (C8.20)».

Действие (C7.5) является последовательностью, раскрывающейся на 5 действий (рис. 3.97). А именно: «Считывание из табл. F1 запись по тек.коду ПК класса «P» (Описание ПК) (C8.21)», «Добавление этой записи в табл. F1c с новым кодом структуры (C8.22)», «Считывание из табл. F2 код.ПК-1(x) в связи, в которой класс связи= «SP», Тип связи=C, код ПК-2= текущий код ПК класса «P» (Описание связи ПК) (C8.23)», «Запись в табл. F2c с новым кодом структуры класс связи= «SP», Тип связи=C, код.ПК-1=x, код ПК-2= текущий код ПК класса «P» (C8.24)» и «Запись в табл. W7, W7.F=x= Код ПК класса «S» (C8.25)».

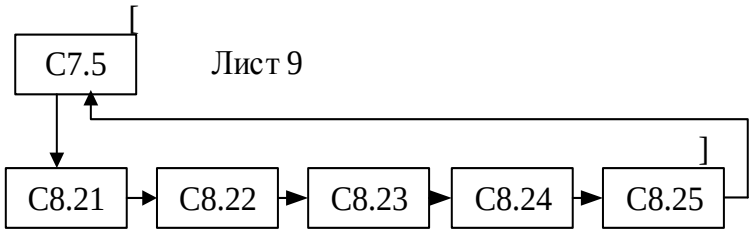


Рис. 3.97. Декомпозиция задачи «Формирование фрагмента цепочки по класса связи «SP»» где C8.21-Считывание из табл. F1 запись по тек.коду ПК Класса «P» (Описание ПК), C8.22-Добавление этой записи в табл. F1c с новым кодом структуры, C8.23- Считывание из табл. F2 код.ПК-1→x в связи, в которой класс связи= «SP», Тип связи=C, код ПК-2= текущий код ПК класса «P» (Описание связи ПК), C8.24- Запись в табл. F2c с новым кодом структуры класс связи= «SP», Тип связи=C, код.ПК-1=x, код ПК-2= текущий код ПК класса «P», C8.25- Запись в табл. W7, E=текущий код ПК класса «P», F=x= Код ПК класса «S».

Таким образом полностью сформирована система предметных действий для первой процедуры «формировании концептуальных моделей для всех статистических ПЗ 1-го рода» (процедура C) при начальном моделировании задачи «Обработка статистических предметных ограничений при концептуальном моделировании»

3.2.4.2 Формирование диаграммы системы предметных действий в целом для процедуры C

В целом диаграмма системы предметных действий для процедуры C приведена на рис.3.98.

Рис.3.98. Диаграмма для процедуры C

3.2.4.3 Формирование множества параметров для процедуры С

В целом структура организации данных для процедуры В приведена на рис.3.99.

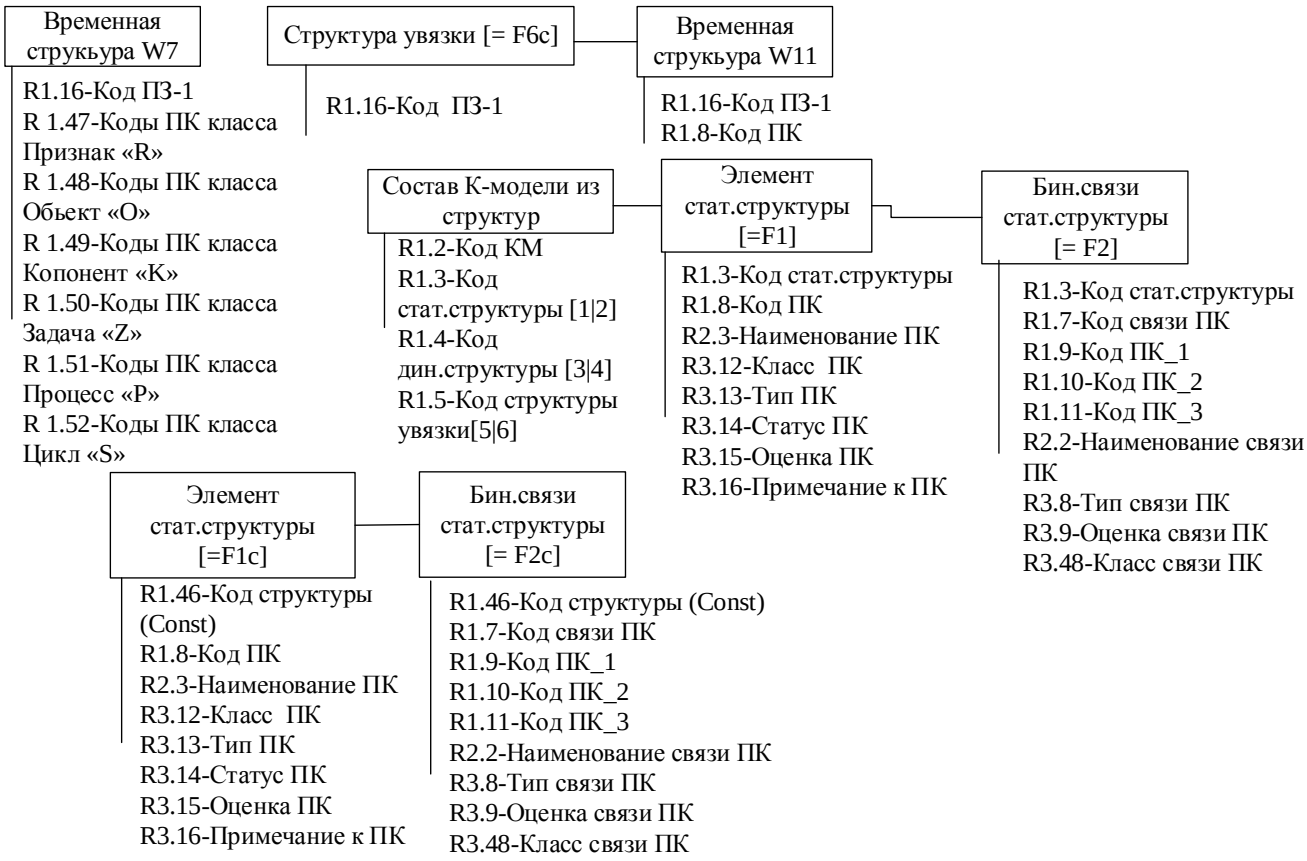


Рис. 3.99. Структура организации данных для процедуры С

Таким образом, разработана информационная составляющая начальной модели для процедуры С.

3.2.4.4 Формирование матричных диаграмм для процедуры С

Матричная диаграмма (МД) для типовой конструкции «цикл» (рис. 3.89) представлена на рис.3.100. Основными параметрами являются: Счетчик итераций (СЦ)– Количество стат. стат. ПЗ-1 - R3.1; Аргумент итераций (АИ) - Код ПЗ-1

R1.16; Функция итераций (ФИ) - w7(Массив цепочек для тек.ПЗ-1); Функция цикла (ФЦ) - W7(Массив цепочек для всех ПЗ-1).

	СЧ	АИ	ФИ	ФЦ	
Позиция ПД	R3.1	R1.16	w7(Массив цепочек для тек.ПЗ-1)	W7(Массив цепочек для всех ПЗ-1)	
C1.1				{	из временной таблицы
C2.1	!		$	[	
C3.1	.	*			
C3.2	**.**	**+**	*****		**[**
C3.3	?			#]	
C4.12	/ .			[	
C5.1	. *				
C5.2	.	+	+	*]	
C4.13	/ .	+	+	*}	W7

Рис. 3.100. Матричная диаграмма конструкции «Формирование “родословных” структур для всех статистических ПЗ-1»

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.90) представлена на рис.3.101. Основным параметром является ФП - w7(Массив цепочек для тек.ПЗ-1).

				ФП	Пояснения
Позиция ПД	R1.16	R3.47	W11	w7(Массив цепочек для тек.ПЗ-1)	
C3.2				**[**	
C4.1	+	*****	*****		
C4.2			+	***]**	

Рис. 3.101. Матричная диаграмма конструкции «Формирование “родословных” структур для текущей статистических ПЗ-1 (формирование множества цепочек)» где R1.16-($F6^c$) Код ПЗ-1, R3.47- Количество ПК ($N_{пк}$), W11 (Код ПЗ-1- R1.16, Код ПК- R1.8).

Матричная диаграмма (МД) для типовой конструкции «цикл» (рис.3.91) представлена на рис.3.102.

Основными параметрами являются следующие: СЦ Количество ПК R3.47; АИ - Коды ПК класса Признак «R»- R 1.47; ФИ - w7(“Цепочка” для кода ПК класса «R» Массив кодов ПК всех классов); ФЦ – w7 (Массив “цепочек” кодов ПК всех классов для всех ПК класса «R»).

	СЧ	АИ	ФИ	ФЦ	
Позиция ПД	R3.47	R 1.47	w7(“Цепочка “ для кода ПК класса «R»)	w7 (Массив “цепочек” кодов ПК всех классов для всех ПК класса «R»)	
C4.2				{	
C5.1	!		$	[	
C6.1	.	*			
C6.2	**.**	**+**	*****		**[**
C6.3	?			#]	
C7.6	/ .			[	
C8.26	. *				
C8.27	.	+	+	*]	
C7.7	/ .	+	+	*}	w7

Рис. 3.102. Матричная диаграмма конструкции «Формирование “i”-ой цепочки “родословных” структур для текущей ПЗ-1 для всех ПК, связанных с ней»

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.92) представлена на рис.3.103.

							ФП	Пояснения
Позиция ПД	R1.16	R1.47	R1.48	R1.49	R1.50	R1.51	R1.52	
C6.2							**[**	
C7.1	**+**	**+**	*****					**[**
C7.2	**+**	**+**	**+**	*****				**[**
C7.3	**+**	**+**	**+**	**+**	*****			**[**
C7.4	**+**	**+**	**+**	**+**	**+**	*****		**[**
C7.5	**+**	**+**	**+**	**+**	**+**	**+**	***]**	**[**

Рис. 3.103. Матричная диаграмма конструкции «Формирование фрагмента цепочки по классу связи «OR»» где R1.16- Код ПЗ-1, R1.47- Код ПК- Кл.R, R1.48- Код ПК- Кл.O, R1.49- Код ПК- Кл.K, R1.50- Код ПК- Кл.Z, R1.51- Код ПК- Кл.P, R1.52- Код ПК- Кл.S.

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.93) представлена на рис.3.104. Основным параметром является ФП w7 (Код ПЗ-1- R1.16, Коды ПК класса Признак «R»- R 1.47= Код ПК_2, Код ПК класса «O»- R1.48= Код ПК_1).

											ФП	Поясн-ения
Позиция ПД	R1.16	R1.47	R1.3	R1.46	R1.48	f1	$f1^c$	R3.48	R3.8	$f2^c$	w7	
C7.1											**[**	
C8.1		+	+			*****						из табл. F1
C8.2				+		**+**	*****					запись в табл. F1c
C8.3		+	+		*			+	+			из табл. F2
C8.4		+		+	+			+	+	*****		запись в табл. F2c
C8.5	+	+			+						***]**	Запись в таб. W7

Рис. 3.104. Матричная диаграмма конструкции «Формирование фрагмента цепочки по классу связи «OR»», где R1.16- Код ПЗ-1, R1.47- Код ПК- Кл.R, R1.3- Код стат.структуры- (var), R1.46- Код стат.структуры- (const), R1.48- Код ПК- Кл.O, Кортеж из табл.f1, Корт-еж в табл.$f1^c$ с кодом стат.структуры- R1.46 (const),R3.48- Кл.связи (=OR), R3.8- Тип связи (=C), Кортеж в табл.$f2^c$ с кодом стат.структуры- (const).

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.94) представлена на рис. 3.105. Основным параметром является ФП - w7 (Код ПЗ-1- R1.16, Коды ПК класса Признак «O»- R 1.48= Код ПК_2, Код ПК класса «K»- R1.49= Код ПК_1).

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.95) представлена на рис. 3.106. Основным параметром является ФП-w7 (Код ПЗ-1- R1.16, Коды ПК класса Признак «K»- R 1.49= Код ПК_2, Код ПК класса «Z»- R1.50= Код ПК_1).

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.96) представлена на рис. 3.107. Основным параметром является ФП - w7 (Код ПЗ-1- R1.16, Коды ПК класса Признак «Z»- R 1.50= Код ПК_2, Код ПК класса «P»- R1.51= Код ПК_1).

											ФП	Поясн-ения
Позиц-ия ПД	R1.16	R1.48	R1.3	R1.46	R1.49	f1	$f1^c$	R3.48	R3.8	$f2^c$	w7	
C7.2											**[**	
C8.6		+	+			*****						из табл.F1
C8.7				+		**+**	*****					запись в табл. F1c
C8.8		+	+		*			**+**	**+**			Из табл. F2
C8.9		+		+	+			**+**	**+**	*****		запись в табл. F2c
C8.10	+	+			+						***]**	Запись в таб. W7

Рис. 3.105. Матричная диаграмма конструкции «Формирование фрагмента цепочки по классу связи «KO»» где R1.16- Код ПЗ-1, R1.48- Код ПК- Кл.O, R1.3- Код стат.структуры- (var), R1.46- Код стат.структуры- (const), R1.49- Код ПК- Кл.K, Кортеж из табл.f1, Корт-еж в табл.$f1^c$ с кодом стат.структуры- R1.46 (const),R3.48- Кл.связи (=KO), R3.8- Тип связи (=C), Кортеж в табл.$f2^c$ с кодом стат.структуры- (const).

											ФП	Поясне-ния
Позиц-ия ПД	R1.16	R1.49	R1.3	R1.46	R1.50	f1	$f1^c$	R3.48	R3.8	$f2^c$	w7	
C7.3											**[**	
C8.11		+	+			*****						из табл. F1
C8.12				+		**+**	*****					запись в табл. F1c
C8.13		+	+		*			**+**	**+**			из табл. F2
C8.14		+		+	+			**+**	**+**	*****		запись в табл. F2c
C8.15	+	+			+						***]**	Запись в таб. W7

Рис. 3.106. Матричная диаграмма конструкции «Формирование фрагмента цепочки по классу связи «ZK»» где R1.16- Код ПЗ-1, R1.49- Код ПК- Кл.K, R1.3- Код стат.структуры- (var), R1.46- Код стат.структуры- (const), R1.50- Код ПК- Кл.Z, Кортеж из табл.f1, Корт-еж в табл.$f1^c$ с кодом стат.структуры- R1.46 (const),R3.48- Кл.связи (=ZK), R3.8- Тип связи (=C), Кортеж в табл.$f2^c$ с кодом стат.структуры- (const).

											ФП	Поясне-ния
Позиц-ия ПД	R1.16	R1.50	R1.3	R1.46	R1.51	f1	$f1^c$	R3.48	R3.8	$f2^c$	w7	
C7.4											**[**	
C8.16		+	+			*****						из табл. F1
C8.17				+		**+**	*****					запись в табл. F1c
C8.18		+	+		*			**+**	**+**			из табл. F2
C8.19		+		+	+			**+**	**+**	*****		запись в табл. F2c
C8.20	+	+			+						***]**	Запись в таб. W7

Рис. 3.107. Матричная диаграмма конструкции «Формирование фрагмента цепочки по классу связи «PZ»» где R1.16- Код ПЗ-1, R1.50- Код ПК- Кл.Z, R1.3- Код стат.структуры- (var), R1.46- Код стат.структуры- (const), R1.51- Код ПК- Кл.P, Кортеж из табл.f1, Корт-еж в табл.$f1^c$ с кодом стат.структуры- R1.46 (const),R3.48- Кл.связи (=PZ), R3.8- Тип связи (=C), Кортеж в табл.$f2^c$ с кодом стат.структуры- (const).

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.97) представлена на рис. 3.108. Основным параметром является ФП - w7 (Код ПЗ-1- R1.16, Коды ПК класса Признак «P»- R 1.51= Код ПК_2, Код ПК класса «S»- R1.52= Код ПК_1).

											ФП	Поясне-ния
Позиц-ия ПД	R1.16	R1.51	R1.3	R1.46	R1.52	f1	$f1^c$	R3.48	R3.8	$f2^c$	w7	
C7.5											**[**	
C8.21		+	+			*****						из табл. F1
C8.22				+		**+**	*****					запись в табл. F1c
C8.23		+	+		*			+	+			из табл. F2
C8.24		+		+	+			+	+	*****		запись в табл. F2c
C8.25	+	+			+						***]**	Запись в таб. W7

Рис.3.108. Матричная диаграмма конструкции «Формирование фрагмента цепочки по классу связи «SP»» где R1.16- Код ПЗ-1, R1.51- Код ПК- Кл.P, R1.3- Код стат.структуры- (var), R1.46- Код стат.структуры- (const), R1.52- Код ПК- Кл.S, Кортеж из табл.f1, Корт-еж в табл.$f1^c$ с кодом стат.структуры- R1.46 (const),R3.48- Кл.связи (=SP), R3.8- Тип связи (=C), Кортеж в табл.$f2^c$ с кодом стат.структуры- (const).

Таким образом, полностью сформирована начальная модель для процедуры C как увязка системы предметных действий и множества параметров исходной задачи.

3.2.5 Разработка начальной модели для процедуры D

3.2.5.1 Формирование диаграмм системы предметных действий для процедуры D

Процесс определение типологии для всех статистических ПЗ 1-го рода заключается в определении замыкающей вершины в «родословной» основной концептуальной структуры для каждой ПЗ-1.

Действие «Определение типологии для всех статистических ПЗ 1-го рода» раскрывается как цикл (рис. 3.109). В действии D2.1 происходит определение типологии для i-ой статистической ПЗ 1-го рода. Далее она фиксируется в действии D3.1 и выполняется сложное действие D3.2. Действие «Определение типологии для всех статистических ПЗ 1-го рода в зависимости от их наличия» (D3.3) раскрывается как альтернатива.

При наличии статистической ПЗ 1-го рода происходит переход к следующей i=i+1 в действии D5.4 и определяется типология для всех статистических ПЗ 1-го рода при их наличии для предыдущей статистической ПЗ 1-го рода в действии D5.5. А в случае отсутствия определяется типология для всех статистических ПЗ 1-го рода и происходит выход из цикла в действии D4.5.

На рис. 3.110 показано, что определение типологии для текущей ПЗ 1-го рода (D3.2) является последовательностью, раскрывающаейся на 4 действия. В действии (D4.1) необходимо определить количества уникальных ПК класса “O” для текущей ПЗ-1-го рода и в действии (D4.2) необходимо определить тип текущей ПЗ-1 в зависимости от количества ПК класса “O”.

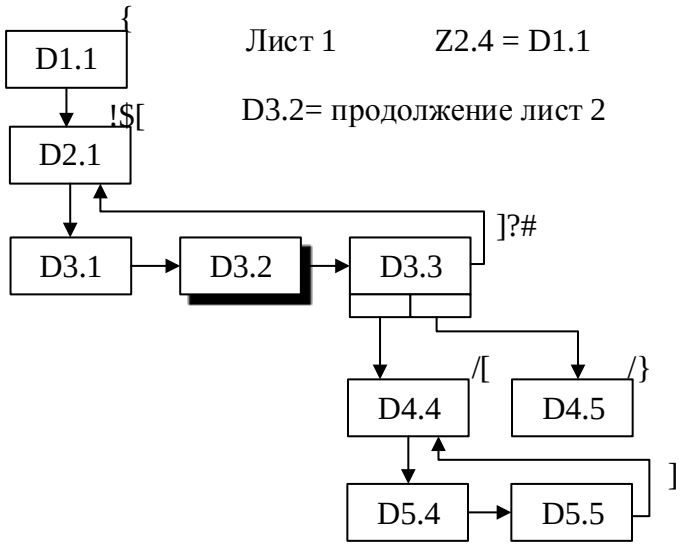


Рис.3.109. Декомпозиция задачи «Определение типологии для всех статистических ПЗ-1» где D2.1- Определение типологии i-ой статистической ПЗ-1-го рода, D3.1- Фиксация i-ой статистических ПЗ-1-го рода, D3.2- Определение типологии для текущей ПЗ-1-го рода, D3.3- Определение типологии для всех статистических ПЗ-1-го рода в зависимости от их наличия, D4.4- Определение типологии для всех статистических ПЗ-1-го рода при их наличии, D5.4- Переход к следующей типологии статистическому ПЗ-1-го рода i=i+1, D5.5- Определение типологии для всех статистических ПЗ-1-го рода при их наличии для предыдущего статистического ПЗ-1-го рода, D4.5- Определение типологии для всех статистических ПЗ-1-го рода при их отсутствии.

И в следующем действии нужно соотносить замыкающую вершину ПК с типом зависимостей 1-го рода (D4.3) и после этого нужно зафиксировать тип ПЗ-1-го рода (D4.4).

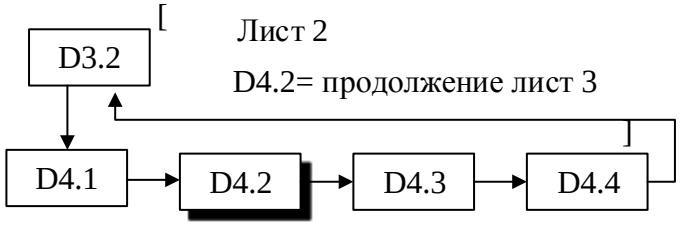


Рис. 3.110. Декомпозиция проектной задачи «Определение типологии для текущей ПЗ-1-го рода» где D4.1- Определение количества уникальных ПК класса «О» для текущей ПЗ-1-го рода, D4.2- Определение типа текущей ПЗ-1 в зависимости от количества ПК класса “О”, D4.3-Соотнесение замыкающей вершины ПК с типом зависимостей 1 -го рода, D4.4-Фиксация типа ПЗ 1-го рода.

На рис. 3.111 показано, что действие (D4.2) является альтернативой для определения типа текущей ПЗ-1 в зависимости от количества ПК класса “О”. Определение типа текущей ПЗ-1 при условии, что количество = 1, то выполняется действие (D5.1). Определение типа текущей ПЗ-1 при условии, что количество > 1, то выполняется действие (D5.2).

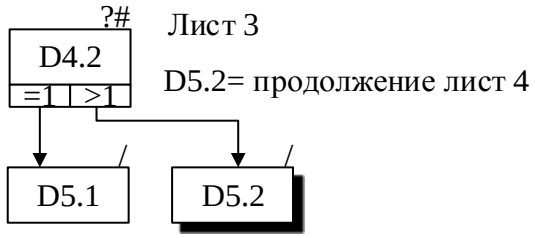


Рис. 3.111. Декомпозиция задачи «Определение типа текущей ПЗ-1 в зависимости от количества ПК класса “О”» где D5.1- Определение типа текущей ПЗ-1, если количество = 1, D5.2- Определение типа текущей ПЗ-1, если количество > 1.

На рис. 3.112 действие (D5.2) раскрывается как последовательность. В первом действии (D6.1) сначала нужно определить количество уникальных ПК класса “ K ” для текущей ПЗ 1-го рода и в последнем действии (D6.2) определить тип текущей ПЗ-1 в зависимости от количества ПК класса “ K ”.

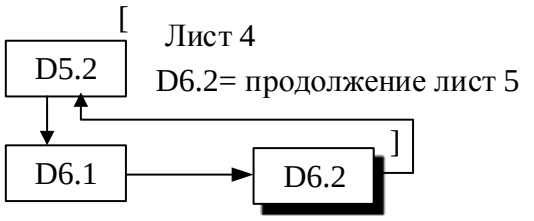


Рис. 3.112. Декомпозиция задачи «Определение типологии для текущей ПЗ-1-го рода» где D6.1- Определение количества уникальных ПК класса «K» для текущей ПЗ-1-го рода, D6.2- Определение типа текущей ПЗ-1-го рода в зависимости от количества ПК класса “K”.

На рис. 3.113 показано, что для определения типа текущей ПЗ-1 в зависимости от количества ПК класса “ K ” (D6.2) выполняется альтернатива. Определение типа текущей ПЗ-1 при условии, что количество = 1, то выполняется действие (D7.1). Определение типа текущей ПЗ-1 при условии, что количество > 1, то выполняется действие (D7.2).

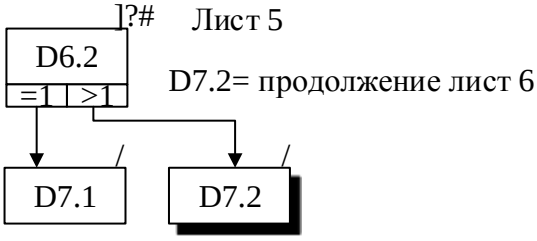


Рис. 3.113. Декомпозиция задачи «Определение типа текущей ПЗ-1 в зависимости от количества ПК класса “ K ”», где D7.1- Определение типа текущей ПЗ-1, если количество = 1, D7.2- Определение типа текущей ПЗ-1, если количество > 1.

На рис. 3.114 показано, что для определении типологии для текущей ПЗ-1-го рода (D7.2) является последовательность, раскрывающаяся на 2 действия. В первом действии (D8.1) необходимо определить количество уникальных ПК класса “Z”

для текущей ПЗ-1-го рода и в последнем действии (D8.2) определить тип текущей ПЗ-1 в зависимости от количества ПК класса “Z”.

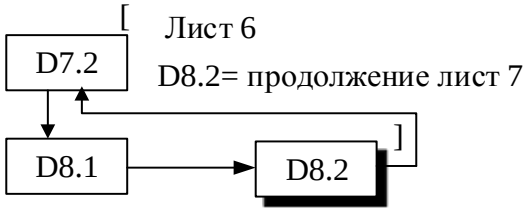


Рис. 3.114. Декомпозиция задачи «Определение типологии для текущей ПЗ-1-го рода» где D8.1- Определение количества уникальных ПК класса «Z» для текущей ПЗ-1-го рода, D8.2- Определение типа текущей ПЗ-1-го рода в зависимости от количества ПК класса “Z”.

На рис. 3.115 показано, что для определения типа текущей ПЗ-1 в зависимости от количества ПК класса “K” (D8.2) выполняется альтернатива. Определение типа текущей ПЗ-1 при условии, что количество = 1, то выполняется действие (D9.1). Определение типа текущей ПЗ-1 при условии, что количество > 1, то выполняется действие (D9.2).

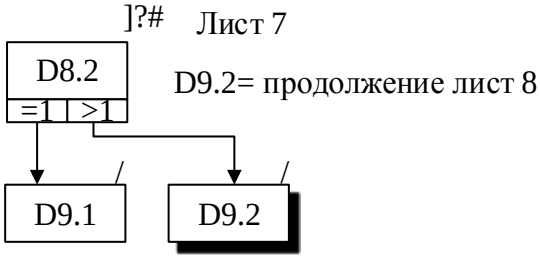


Рис. 3.115. Декомпозиция задачи «Определение типа текущей ПЗ-1 в зависимости от количества ПК класса “ Z ”» где D9.1- Определение типа текущей ПЗ-1, если количество = 1, D9.2- Определение типа текущей ПЗ-1, если количество > 1

На рис. 3.116 действие (D9.2) - последовательность. В действии D10.1 определяют количество уникальных ПК класса “P” для ПЗ-1 и в действии D10.2 определяют тип текущей ПЗ-1 в зависимости от их количества.

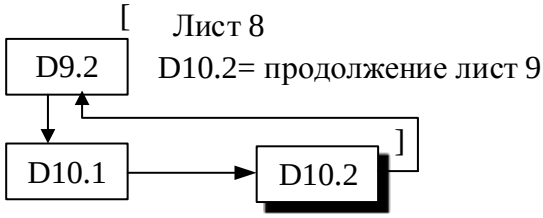


Рис. 3.116. Декомпозиция задачи «Определение типологии для текущей ПЗ-1» где D10.1- Определение количества уникальных ПК класса «P» для текущей ПЗ-1-го рода, D10.2- Определение типа текущей ПЗ-1-го рода в зависимости от количества ПК класса “P”.

На рис. 3.117 показано, что для определения типа текущей ПЗ-1 в зависимости от количества ПК класса “P” (D10.2) выполняется альтернатива. Определение типа ПЗ-1 при условии, что количество = 1, то выполняется действие (D11.1).

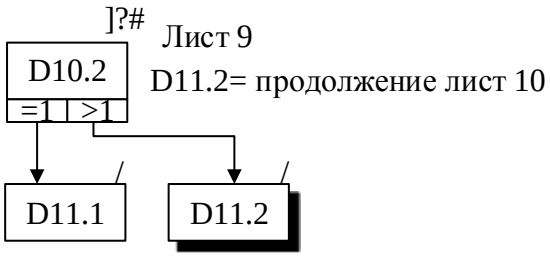


Рис. 3.117. Декомпозиция задачи «Определение типа текущей ПЗ-1 в зависимости от количества ПК класса “ P”» где D11.1- Определение типа текущей ПЗ-1, если количество = 1, D11.2- Определение типа текущей ПЗ-1, если количество > 1

Определение типа ПЗ-1 при условии, что количество > 1, то выполняется действие (D11.2). На рис. 3.118 действие (D11.2) - последовательность.

В действии (D12.1) определяют количество уникальных ПК класса “S” для текущей ПЗ-1 и в действии (D12.2) определяют тип текущей ПЗ-1 при количестве ПК класса “S”=1.

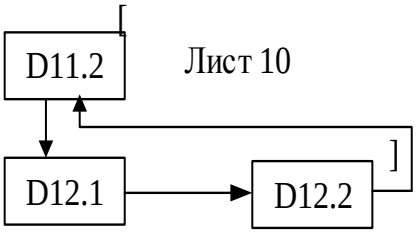


Рис. 3.118. Декомпозиция задачи «Определение типологии для текущей ПЗ-1» где D12.1- Определение количества уникальных ПК класса «S» для текущей ПЗ-1-го рода, D12.2- Определение типа текущей ПЗ-1-го рода количества ПК класса “S”=1.

Таким образом полностью сформирована система предметных действий для первой процедуры «Определение типологии статистических ПЗ 1-го рода» (процедура D) при начальном моделировании задачи «Обработка статистических предметных ограничений при концептуальном моделировании»

3.2.5.2 Формирование диаграммы системы предметных действий в целом для процедуры D

В целом диаграмма системы предметных действий для процедуры D приведена на рис.3.119.

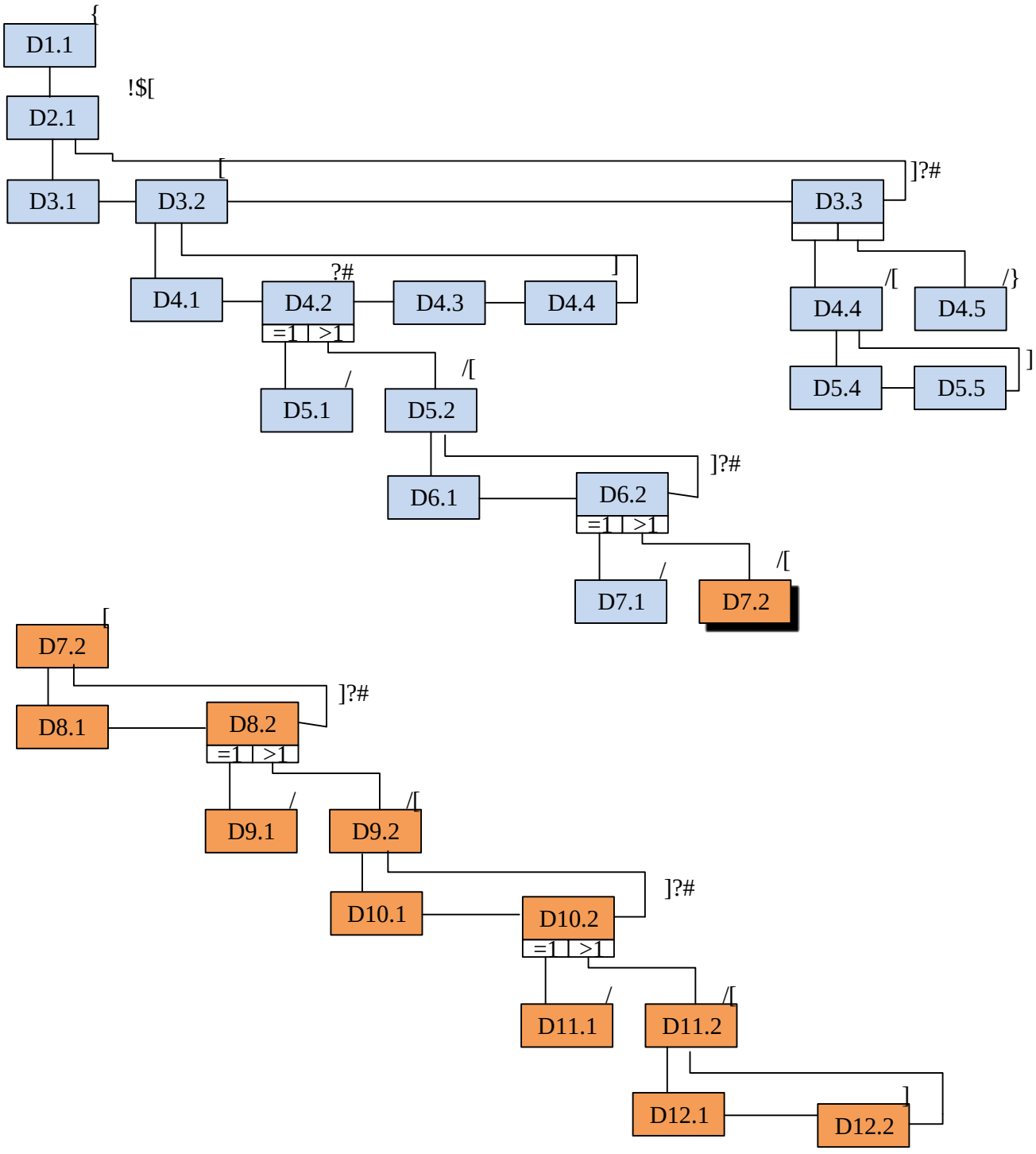


Рис.3.119. Диаграмма для процедуры D

3.2.5.3 Формирование множества параметров для процедуры D

В целом структура организации данных для процедуры D приведена на рис.3.120.

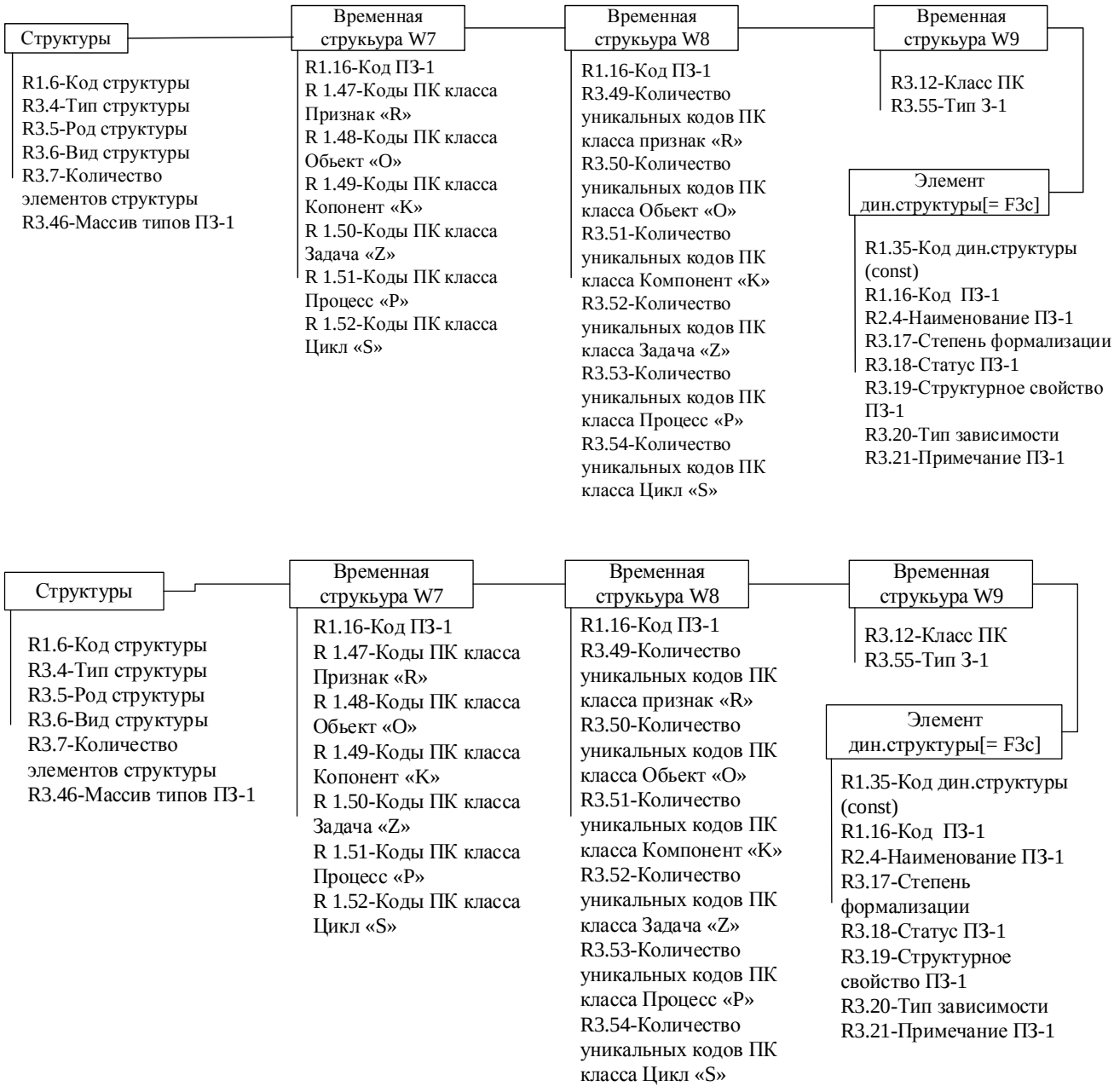


Рис. 3.120. Структура организации данных для процедуры D

Таким образом, разработана информационная составляющая начальной модели для процедуры D.

3.2.5.4 Формирование матричных диаграмм для процедуры D

Матричная диаграмма (МД) для типовой конструкции «цикл» (рис. 3.109) представлена на рис. 3.121. Основными параметрами являются: СЦ– Количество стат. ПЗ-1 R3.1; АИ - Код стат. ПЗ-1 R1.16; ФИ - Тип стат.ПЗ-1- R3.20; ФЦ – Массив типов ПЗ-1- R3.46.

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.110) представлена на рис. 3.122. Основным параметром является ФП – В табл. F3c (Код стат. ПЗ-1 R1.16, Код дин. структуры - R1.44, Тип стат ПЗ-1 R3.20).

	СЧ	АИ	ФИ	ФЦ	
Позиция ПД	R3.1	R1.16	R3.20	R3.46	
D1.1				{	
D2.1	!		$	[	
D3.1	.	*			
D3.2	**.**	**+**	*****		**[**
D3.3	?			#]	
D4.5	/ .			[	
D5.4	. *				
D5.5	.	+	+	*]	
D4.6	/ .	+	+	*}	

Рис.3.121. Матричная диаграмма конструкции «Определение типологии для всех статистических ПЗ-1»

					ФП	Пояснения
Позиция ПД	R1.16	R3.50	R1.48	R3.55	В табл. F3c	
D3.2					**[**	
D4.1	+	*****				
D4.2		+	*****			#
D4.3			+	*****		
D4.4	+			+	***]**	

Рис. 3.122. Матричная диаграмма конструкции «Определение типологии для текущей ПЗ-1» где R1.16- Код ПЗ-1, R3.50- Количество уникальных кодов ПК класса Обьект «О», R1.48- Код ПК R1.8 = Код ПК класса «О», R3.55- Тип Зависимости-1.

Матричная диаграмма для конструкции «альтернатива» (рис.3.111) представлена на рис. 3.123. Основными параметрами являются: ФА - Код ПК- R1.8 = Код ПК класса (О) R1.48; АА - Количество уникальных кодов ПК класса «О»- R3.50.

	АА		ФА=ФП	
Позиция ПД	R3.50	…	R 1.8	Пояснения
D4.2	?		#	W7
D5.1	/ .		*	
D5.2	**/ .**		*****	

Рис. 3.123. Матричная диаграмма конструкции «Определение типа текущей ПЗ-1 в зависимости от количества ПК класса “ О ”»

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.112) представлена на рис. 3.124. Основным параметром является ФП – Код ПК-R 1.8.

				ФП	Пояснения
Позиция ПД	R1.16	R1.49	R3.51	R 1.8	
D5.2				**[**	W7
D6.1	+	+	*****		
D6.2			+	***]**	#

Рис. 3.124. Матричная диаграмма конструкции «Определение типологии для текущей ПЗ-1» где R1.16- Код ПЗ-1, R1.49- Код ПК класса «К», R3.51- Количество уникальных кодов ПК класса «К».

Матричная диаграмма для конструкции «альтернатива» (рис.3.113) представлена на рис. 3.125. Основными параметрами являются: ФА - Код ПК- R1.8 = Код ПК класса (К) R1.49; АА - Количество уникальных кодов ПК класса «К»- R3.51.

	АА		ФА=ФП	
Позиция ПД	R3.51	…	R 1.8	Пояснения
D6.2	?		#	W7
D7.1	/ .		*	
D7.2	**/ .**		*****	

Рис. 3.125. Матричная диаграмма конструкции «Определение типа текущей ПЗ-1 в зависимости от количества ПК класса “ К ”»

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.114) представлена на рис. 3.126. Основным параметром является ФП – Код ПК-R 1.8.

				ФП	Пояснения
Позиция ПД	R1.16	R1.50	R3.52	R 1.8	
D7.2				**[**	W7
D8.1	+	+	*		
D8.2			+	***]**	

Рис.3.126. Матричная диаграмма конструкции «Определение типологии для текущей ПЗ-1» где R1.16- Код ПЗ-1, R1.50- Код ПК класса «Z», R3.52- Количество уникальных кодов ПК класса «Z».

Матричная диаграмма для конструкции «альтернатива» (рис.3.115) представлена на рис. 3.127. Основными параметрами являются: ФА - Код ПК -R 1.8 = Код ПК класса «Z» R1.50; АА - Количество уникальных кодов ПК класса «Z»- R3.52.

	АА		ФА=ФП	
Позиция ПД	R3.52	…	R 1.8	Пояснения
D8.2	?		#	W7
D9.1	/ .		*	
D9.2	**/ .**		*	

Рис.3.127. Матричная диаграмма конструкции «Определение типа текущей ПЗ-1 в зависимости от количества ПК класса “ Z ”»

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.116) представлена на рис. 3.128. Основным параметром является ФП – Код ПК-R 1.8.

				ФП	Пояснения
Позиция ПД	R1.16	R1.51	R3.53	R 1.8	
D9.2				**[**	W7
D10.1	+	+	*		
D10.2			+	***]**	

Рис.3.128. Матричная диаграмма конструкции «Определение типологии для текущей ПЗ-1» Где R1.16- Код ПЗ-1, R1.51- Код ПК класса «Р», R3.53- Количество уникальных кодов ПК класса «Р».

Матричная диаграмма для конструкции «альтернатива» (рис.3.117) представлена на рис. 3.129. Основными параметрами являются: ФА - Код ПК -R 1.8 = Код ПК класса «Р» R1.51; АА - Количество уникальных кодов ПК класса «Р» - R3.53.

	АА		ФА=ФП	
Позиция ПД	R3.53	…	R 1.8	Пояснения
D10.2	?		#	W7
D11.1	/ .		*	
D11.2	**/ .**		*****	

Рис. 3.129. Матричная диаграмма конструкции «Определение типа текущей ПЗ-1 в зависимости от количества ПК класса “ P ”»

Матричная диаграмма для базовой конструкции «последовательность» (рис.3.118) представлена на рис. 3.130. Основным параметром является ФП – Код ПК-R 1.8.

				ФП	Пояснения
Позиция ПД	R1.16	R1.52	R3.54	R 1.8	
D11.2				**[**	W7
D12.1	+	+	*****		
D12.2			+	***]**	

Рис. 3.130. Матричная диаграмма конструкции «Определение типологии для текущей ПЗ-1» где R1.16- Код ПЗ-1, R1.52- Код ПК класса «S», R3.54- Количество уникальных кодов ПК класса «S».

Таким образом, полностью сформирована начальная модель для процедуры D как увязка системы предметных действий и множества параметров исходной задачи.

3.3 Выводы по главе 3

1. На основе формального описания концептуальных моделей 1-го рода и их конструкций была разработано методическое обеспечение в виде совокупности начальных моделей, обеспечивающее детальную проработку алгоритмов выполнения процедур.

2. Разработка данной методики в виде начальной модели, как совокупности взаимоувязанных структур (системы предметных действий, множества параметров, матричной диаграммы), позволила перейти к созданию инструментальных средств поддержки обработки статистических предметных ограничений при концептуальном моделировании прикладных задач.

ГЛАВА 4. РАЗРАБОТКА ПРОГРАММНЫХ СРЕДСТВ ПОДДЕРЖКИ ОБРАБОТКИ СТАТИСТИЧЕСКИХ ПРЕДМЕТНЫХ ОГРАНИЧЕНИЙ ПРИ КОНЦЕПТУАЛЬНОМ МОДЕЛИРОВАНИИ ПРИКЛАДНЫХ ЗАДАЧ

4.1 Общее описание программного комплекса

Разработанные автоматизированные процедуры являются программными модулями, расширяющими существующие возможности инструментальных средств поддержки обработки статистических предметных ограничений при концептуальном моделировании прикладных задач.

Разработаны процедуры дополнительной обработки концептуальных моделей:

– выделения всех статистических ПЗ-1-го рода и их предварительной обработки; - определения содержания для всех статистических ПЗ-1-го рода;

– формирования концептуальных структур для всех статистических ПЗ-1-го рода;

– определения типологии для всех статистических ПЗ-1;

– документирования моделей для всех статистических предметных зависимостей 1-го рода.

При создании и разработке данных программных средств учитывались следующие требования:

– упростить работу и использование программного обеспечения путем автоматизации рутины формирования спецификаций, заложенных в исходные правила кодирования, а также создать и успешно применять модели концептуальных конструкций при выполнении установленного набора типовых алгоритмических конструкций при создании ПЗ-1, соответствующего внутренним закономерностям. Во время выполнения вышеуказанных задач подразумевается не пренебрегать следующими требованиями:

– последовательность автоматического описания составляющих модели должны выполняться по процедурам, которые применяются при ручном способе формирования таких данных;

– возможность сохранения корректного смыслового единства при отображении данных в тандеме с возможностями вносить корректировки, изменения и дополнения в ходе формирования компонентов моделей;

– разделение полномочий разных пользователей программного обеспечения на следующие роли: аналитик, эксперт и менеджер-администратор.

Деятельность менеджера подразумевает администрирование и сопровождение базы данных: распределение пользователей по различным задачам, контроль и ведение этих пользователей и различных задач. Помимо этого, менеджер должен своевременно обновлять базу данных, после внесения в не изменений или устранения неполадок, а также следить за корректной работой модулей.

Основной пользователь данного программного обеспечения – это Аналитик. Работа аналитика заключается в вводе данных для формирования концептуальных моделей, создание которых обусловлено текущими предметными задачами.

Эксперт – специалист более высокого уровня, чем аналитик; его обязанностью является контроль за выполнением работ коллективом аналитиков. Также эксперт занимается обработкой локальных концептуальных моделей

Для разработки графического интерфейса пользователя с использованием языков программирование PHP, HTML, CSS и JavaScript, а также была выбрана среда разработки Laravel Framework и СУБД MySQL, которая позволяет организовать работу с базой данных [85-91]. Интерфейсный компонент инструментов для выполнения дополнительной обработки концептуальной модели реализован с помощью экранного меню, которое запускает каждую отдельную процедуру.

Создание автоматизированных инструментов для выполнения вышеуказанных задач подразумевает воплощение определенной структуры и

определенного состава информационной базы, описания концептуальной модели в виде таблиц, содержащих описание модели; набор программных модулей, обеспечивающих выполнение описанных выше процедур; инструкции по эксплуатации автоматизированного комплекса для пользователя и администратора системы.

4.2 Структура информационной базы программного комплекса

4.2.1 Описание табличной структуры базы данных

Логически информационная структура базы данных (БД) разделена на несколько взаимодействующих друг с другом блоков таблиц:

– системные (служебные) таблицы для администрирования БД;

– спецификации для формирования выделения всех статистических ПЗ-1-го рода;

– спецификации для определения содержания всех статистических ПЗ-1-го рода;

– спецификации для формирования концептуальных моделей для всех статистических ПЗ-1-го рода;

– спецификации для типологии для всех статистических ПЗ-1-го рода

– спецификации для документирования моделей для всех статистических ПЗ-1-го рода

Системные таблицы (на рис. 4.1) отображены служебную информацию и служат для управления основными данными внутри программы будут выражены следующими сущностями: ведение пользователя – Users, ведение задания – Tasks, Назначение – для разделения заданий между пользователями, в столбце «Задание» зафиксируем сеансы взаимодействия пользователей с базой данных. Помимо этого, во внимание будут включены таблицы, в которых будут отражены данные о

сообщениях при работе с программой.

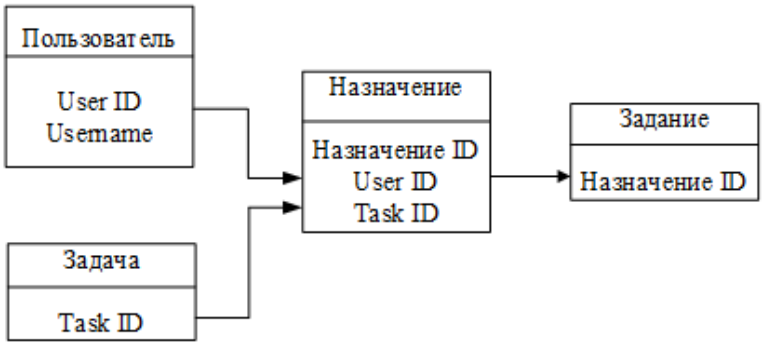


Рис. 4.1. Структура данных для управление системой (функции администратора)

Структура данных для выделения всех статистических предметных зависимостей 1-го рода приведен на рис. 4.2.

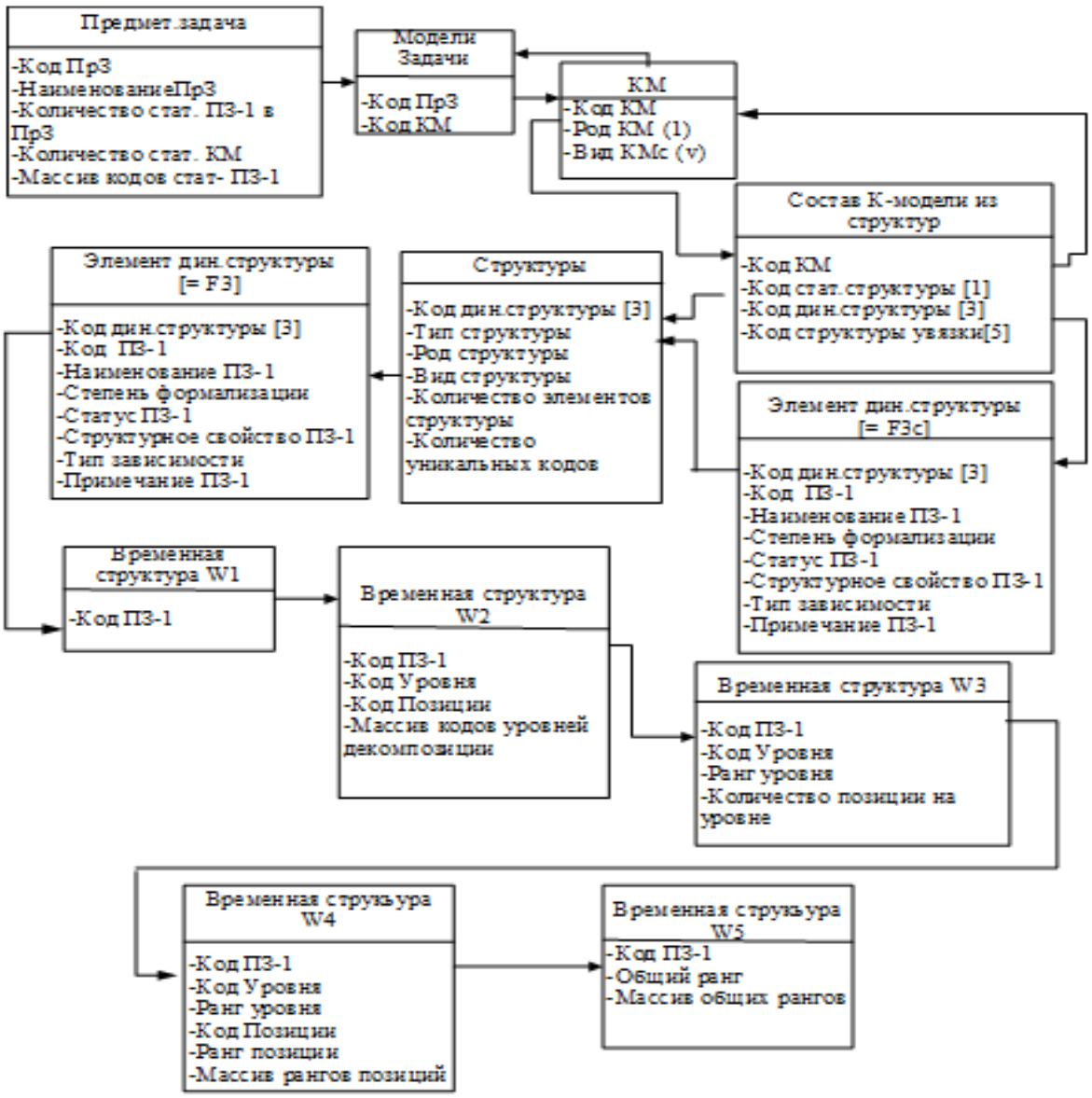


Рис. 4.2. Структура данных для выделения всех статистических предметных зависимостей 1-го рода

Структура данных для определения содержания всех статистическипредметных зависимостей 1-го рода приведен на рис. 4.3.

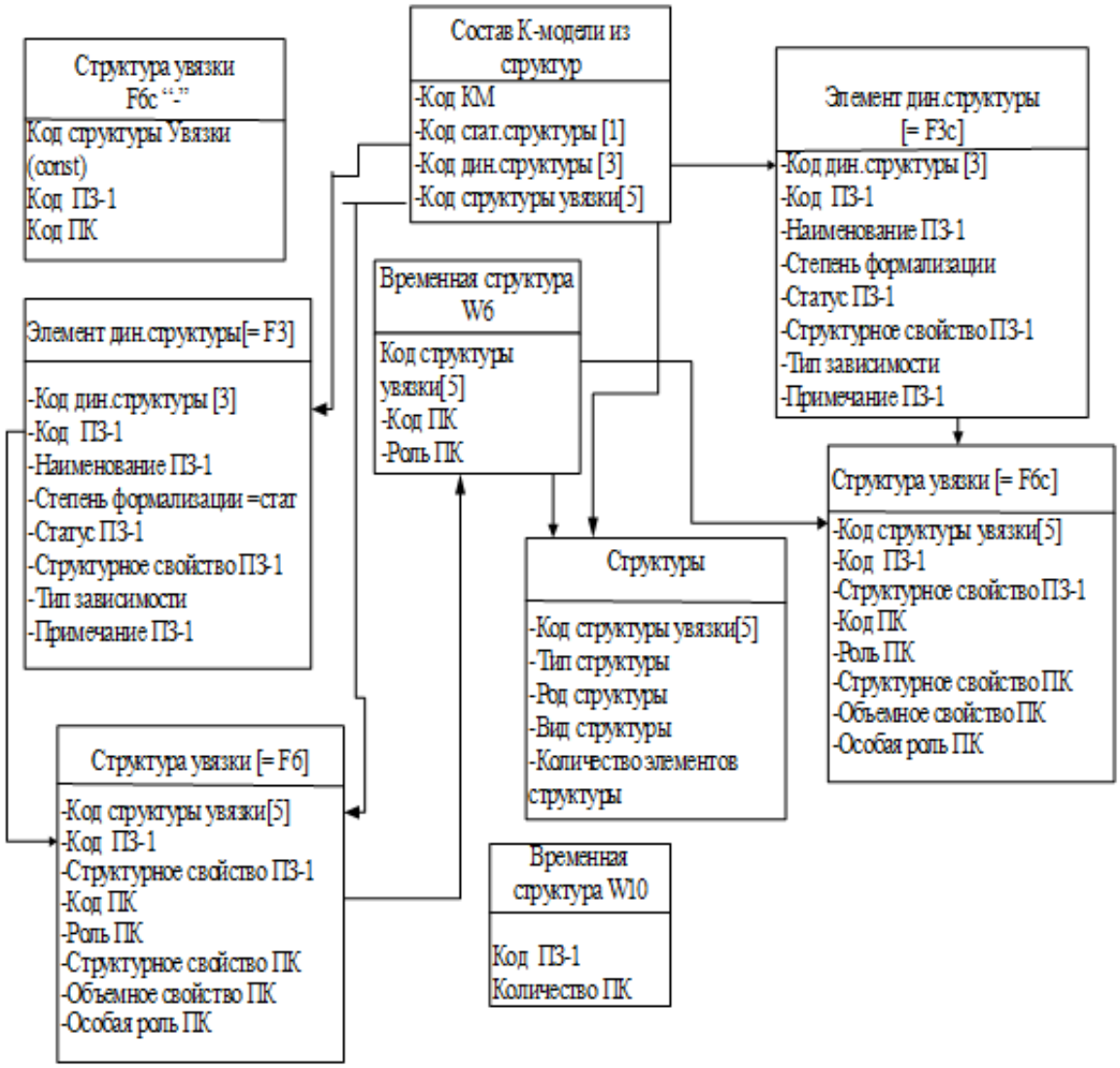


Рис.4.3. Структура данных для определения содержания всех статистических предметных зависимостей 1-го рода

Структура данных для формирования концептуальных структур для всех статистически предметных зависимостей 1-го рода приведен на рис. 4.4.

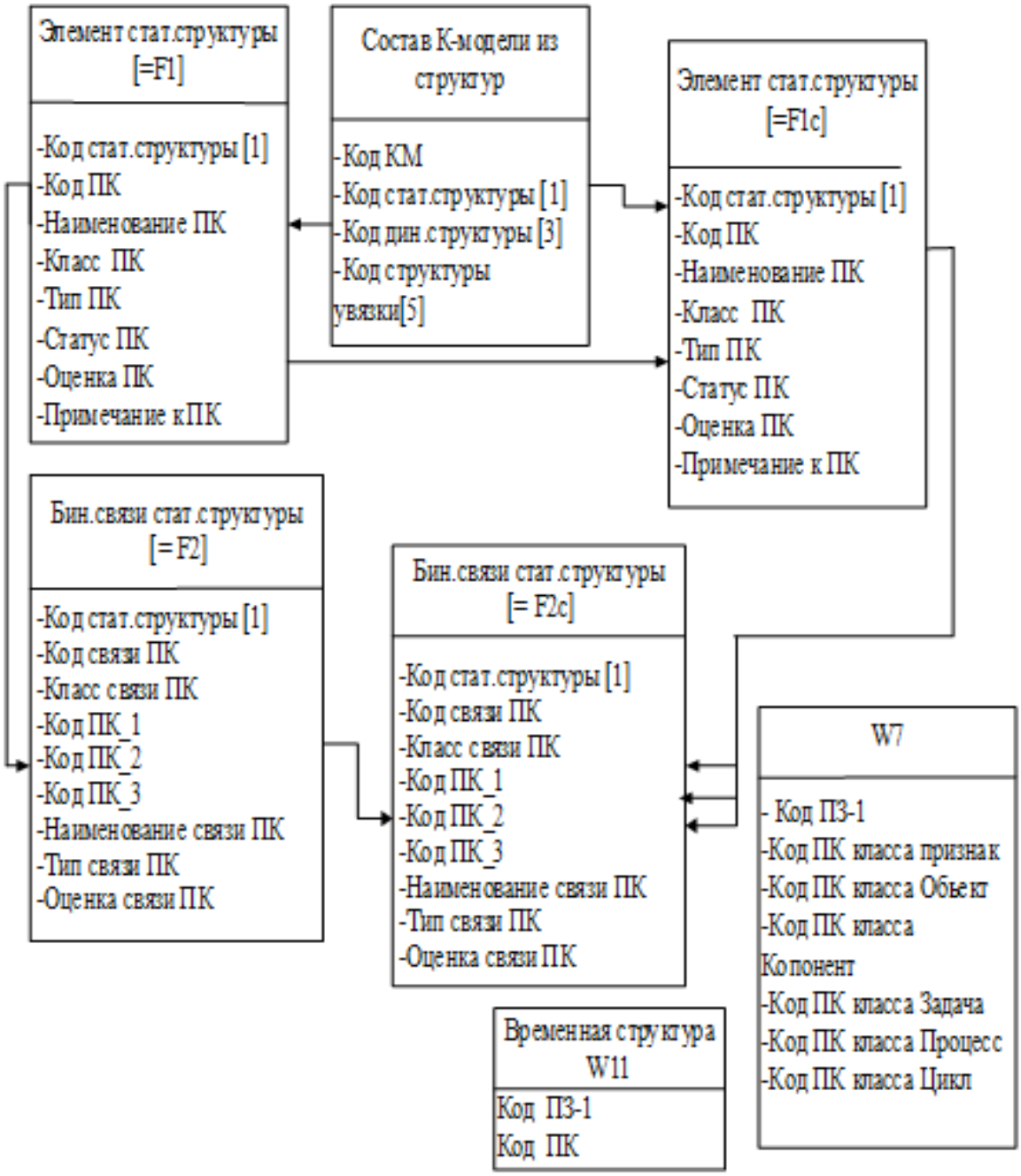


Рис.4.4. Структура данных для формирования концептуальных структур для всех статистически предметных зависимостей 1-го рода

Структура данных для определения типологии всех статистических ПЗ-1-го рода приведен на рис.4.5.

Рис.4.5. Структура данных для типологии для сформированных концептуальных структур всех статистических ПЗ-1-го рода

4.2.2 Описание содержания таблиц базы данных

Таблицы базы данных программного комплекса с описанием наименованиями полей, их типов данных и краткого комментария о хранящихся в них данных приведены в табл.4.1-4.30.

Табл. 4.1. User – Введение пользователя:

Наименование параметра	Тип данных	Формат	Ключевой элемент
Пользователя ID	int	5	PK
Имя	Varchar	20	-
Пароль	Varchar	20	-
Подтвердите пароль	Varchar	20	-
Адрес	Varchar	50	-
Статус	Varchar	20	-

Данные, отображающие введение Задачи, заполняются в виде табл. 4.2.

Табл. 4.2. Task - Введение Задачи:

Наименование параметра	Тип данных	Формат	Ключевой элемент
ЗадачаID	int	5	PK
Наименование Задача	Varchar	50	-

Данные, отображающие назначение пользователя на выполнение задания, заполняются в виде табл. 4.3.

Табл.4.3. Назначение пользователя на выполнение задания:

Наименование параметра	Тип данных	Формат	Ключевой элемент
Назначение ID	int	5	PK
ЗадачаID	int	5	FK
ПользователяID	int	5	FK

Данные, отображающие задание, заполняются в виде табл. 4.4.

Табл. 4.4. Задание:

Наименование параметра	Тип данных	Формат	Ключевой элемент
Назначение ID	int	5	FK
Дата_Время	Date/Time	50	-

Данные, отображающие предмет задачи, заполняются в виде табл. 4.5.

Табл. 4.5. Предметные задачи (ПрЗ):

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код ПрЗ	Varchar	5	PK
Наименование ПрЗ	Varchar	50	-
Колич.стат.ПЗ-1	Varchar	10	-
Количество стат. КМ	Varchar	10	-
Массив кодов стат. ПЗ-1	Varchar	10	-

Данные, отображающие модели Задачи, заполняются в виде табл.4.6.

Табл. 4.6. Модели Задачи:

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код ПрЗ	Varchar	5	PK
Код КМ	Varchar	8	FK

Данные, отображающие КМ, заполняются в виде табл. 4.7.

Табл. 4.7. КМ: Описание концептуальных моделей

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код КМ	Varchar	5	PK
Род КМ (1/2)	Number	5	-
Вид КМс (v/c)	Varchar	10	-

Данные, отображающие состав концептуальной модели (К-модели) из структур, заполняются в виде табл. 4.8.

Табл. 4.8. Состав К-модели из структур:

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код КМ	Varchar	5	PK
Код стат. структуры	Number	10	FK
Код дин. структуры	Varchar	10	FK
Код структуры увязки	Varchar	10	FK

Данные, отображающие структуры, заполняются в виде табл. 4.9.

Табл. 4.9. Структуры

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры	Varchar	10	PK
Тип структуры	Text	20	-
Род структуры	Number	10	-
Вид структуры	Varchar	10	-
Количество элементов структуры	Number	10	-

Исходные данные для дополнительной обработки представлены в табл. 4.10 – 4.14. Данные, отображающие элементы статической структуры [=F1], заполняются в виде табл. 4.10.

Табл.4.10. Элемент стат. структуры [=F1]

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры	Varchar	10	FK
Код ПК	Varchar	10	FK
Наименование ПК	LongText		-
Класс ПК	Text	10	-
Тип ПК	Text	10	-
Статус ПК	Text	10	-
Оценка ПК	LongText		-
Примечание к ПК	LongText		-

Данные, отображающие бинарные связи статической структуры [= F2], заполняются в виде табл. 4.11.

Табл. 4.11. Бин. связи стат. структуры [= F2]

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры	Varchar	10	FK
Класс связи ПК	Varchar	10	-
Код ПК_1	Varchar	10	FK
Код ПК_2	Varchar	10	FK
Код ПК_3	Varchar	10	FK
Наименование связи ПК	LongText		-
Тип связи ПК	Text	5	-
Оценка связи ПК	Number	5	-
Код связи ПК	Varchar	20	-

Данные, отображающие элементы динамической структуры [= F3], заполняются в виде табл. 4.12.

Табл. 4.12. Элемент дин. структуры [= F3]

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры	Varchar	10	FK
Код ПЗ-1	Varchar	10	PK
Наименование ПЗ-1	LongText		-
Степень формализации	Text	10	-
Статус ПЗ-1	Text	5	-
Структурное свойство ПЗ-1	Text	5	-
Тип зависимости	Text	5	-
Примечание ПЗ-1	LongText		-

Данные, отображающие бинарные связи динамической структуры [=F4], заполняются в виде табл. 4.13.

Табл. 4.13. Бин. связи дин. структуры [=F4]

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры	Varchar	10	FK
Код ПЗ-1_1	Varchar	10	FK
Код ПЗ-1_2	Varchar	10	FK
Код ПЗ-1_3	Varchar	10	FK
Оценка связи ПЗ-1	Text	5	-
Тип связи ПЗ-1	Text	5	-
Код связи ПЗ-1	Varchar	20	-

Данные, отображающие структуру увязки [= F6], заполняются в виде табл. 4.14.

Табл. 4.14. Структура увязки [= F6]

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры	Varchar	10	FK
Код ПЗ-1	Varchar	10	FK
Структурное свойство ПЗ-1	Text	5	-
Код ПК	Varchar	10	FK
Роль ПК	Text	5	-
Структурное свойство ПК	Varchar	5	-
Объемное свойство ПК	Text	5	-

Результирующие данные для дополнительной обработки представлены в табл. 4.15 – 4.18.

Данные, отображающие элемент стат. структуры [=F1c], заполняются в виде табл. 4.15.

Данные, отображающие бин. связи стат. структуры [= F2c], заполняются в виде табл. 4.16.

Табл. 4.15. Элемент стат. структуры [=F1c]

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры	Varchar	10	FK
Код ПК	Varchar	10	FK
Наименование ПК	LongText		-
Класс ПК	Text	10	-
Тип ПК	Text	10	-
Статус ПК	Text	10	-
Оценка ПК	LongText		-
Примечание к ПК	LongText		-

Табл. 4.16. Бин. связи стат. структуры [= F2c]

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры	Varchar	10	FK
Класс связи ПК	Varchar	10	-
Код ПК_1	Varchar		FK
Код ПК_2	Varchar	10	FK
Код ПК_3	Varchar	10	FK
Наименование связи ПК	LongText		-
Тип связи ПК	Text	5	-
Оценка связи ПК	Number	5	-
Код связи ПК	Varchar	20	-

Данные, отображающие элемент дин структуры [= F3c], заполняются в виде табл. 4.17.

Данные, отображающие структура увязки [= F6c], заполняются в виде табл. 4.18.

Табл. 4.17. Элемент дин. структуры [= F3c]

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры	Varchar	10	FK
Код ПЗ-1	Varchar	10	PK
Наименование ПЗ-1	LongText		-
Степень формализации	Text	10	-
Статус ПЗ-1	Text	5	-
Структурное свойство ПЗ-1	Text	5	-
Тип зависимости	Text	5	-
Примечание ПЗ-1	LongText		-

Табл. 4.18. Структура увязки [= F6c]

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры	Varchar	10	FK
Код ПЗ-1	Varchar	10	FK
Структурное свойство ПЗ-1	Text	5	-
Код ПК	Varchar	10	PK
Роль ПК	Text	5	-
Структурное свойство ПК	Varchar	5	-
Объемное свойство ПК	Text	5	-

Промежуточные данные для дополнительной обработки представлены в табл. 4.19 – 4.30. Данные, отображающие список статистических ПЗ-1, заполняются в виде табл. 4.19.

Табл. 4.19. Временная таблица W1

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код ПЗ-1	Varchar	10	FK

Данные, отображающие перекодировку стат. ПЗ-1, заполняются в виде табл. 4.20.

Табл. 4.20. Временная таблица W2

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код ПЗ-1	Varchar	10	FK
Код Уровня	Varchar	10	PK
Код позиции	Varchar	10	FK

Данные, отображающие ранжирование уровней для всех стат. ПЗ-1, заполняются в виде табл. 4.21.

Табл. 4.21. Временная таблица W3

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код ПЗ-1	Varchar	10	FK
Код Уровня	Varchar	10	FK
Ранг уровня	Varchar	10	-
Количество позиции на уровне	Varchar	10	-

Данные, отображающие ранжирование позиций с учетом ранга уровней для всех стат. ПЗ-1, заполняются в виде табл. 4.22.

Табл. 4.22. Временная таблица W4

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код ПЗ-1	Varchar	10	FK
Код Уровня	Varchar	10	FK
Ранг уровня	Varchar	10	-
Код Позиции	Varchar	10	PK
Ранг позиции	Varchar		-

Данные, отображающие общий ранг для всех стат. ПЗ-1, заполняются в виде табл. 4.23.

Табл. 4.23. Временная таблица W5

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код ПЗ-1	Varchar	10	FK
Общий ранг	Varchar	10	

Данные, отображающие ПК из табл. F6 с текущим кодом структуры, заполняются в виде табл. 4.24.

Табл. 4.24. Временная таблица W6

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры	Varchar	10	FK
Код ПЗ-1	Varchar	10	FK
Код ПК	Varchar	10	FK
Роль ПК	Text	5	-

Данные, отображающие «родословные» структуры для тек. ПЗ-1, заполняются в виде табл. 4.25.

Табл. 4.25. Временная таблица W7

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код ПЗ-1	Varchar	10	FK
Код ПК класса признак	Varchar	10	FK
Код ПК класса объект	Varchar	10	FK
Код ПК класса компонент	Varchar	10	FK
Код ПК класса задача	Varchar	10	FK
Код ПК класса процесс	Varchar	10	FK
Код ПК класса цикл	Varchar	10	FK

Данные, отображающие замыкание на определенный класс ПК для всех статистических ПЗ-1, заполняются в виде табл. 4.26.

Табл. 4.26. Временная таблица W8

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код ПЗ-1	Varchar	10	FK
Количество уникальных кодов ПК класса признак «R»	Varchar	10	
Количество уникальных кодов ПК класса признак «O»	Varchar	10	
Количество уникальных кодов ПК класса признак «K»	Varchar	10	
Количество уникальных кодов ПК класса признак «Z»	Varchar	10	
Количество уникальных кодов ПК класса признак «P»	Varchar	10	
Количество уникальных кодов ПК класса признак «S»	Varchar	10	

Данные, отображающие соотношение классов ПК с типом зависимости 1-го рода, заполняются в виде табл. 4.27.

Табл. 4.27. Временная таблица W9

Наименование параметра	Тип данных	Формат	Ключевой элемент
Класс ПК	Text	10	-
Тип З-1	Text	5	-

Данные, отображающие количество ПК для каждой ПЗ-1, заполняются в виде табл. 4.28.

Табл. 4.28. Временная таблица W10

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код ПЗ-1	Varchar	10	FK
Количество ПК	Number	10	-

Данные, отображающие список всех ПК для текущей ПЗ -1, заполняются в виде табл. 4.29.

Табл. 4.29. Временная таблица W11

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код ПЗ-1	Varchar	10	FK
Код ПК	Varchar	10	FK

Данные, отображающие все соотношения стат. ПЗ-1 и ПК с новым кодом структуры увязки, заполняются в виде табл. 4.30.

Табл. 4.30. Временная таблица F6c “-”

Наименование параметра	Тип данных	Формат	Ключевой элемент
Код структуры увязки (const)	Varchar	10	FK
Код ПЗ-1	Varchar	10	FK
Код ПК	Varchar	10	FK

4.3 Состав и структура программных модулей

Данного программного комплекса обеспечения входят следующие подсистемы: подсистема управления доступом к системе; подсистема главного меню, в котором определены процедуры дополнительной обработки концептуальных моделей, предусматривающие функции просмотра и документирования; подсистема выделения всех статистических предметных ограничение 1-го рода и их предварительной обработки; подсистема определения содержания всех статистических ПЗ-1-го рода и их предварительной обработки ; подсистема формирования концептуальных моделей для всех статистических ПЗ-1-го рода; подсистема определения типологии статистических ПЗ 1-го рода; подсистема документирования моделей для всех статистических предметных ограничение 1-го рода

Учетные записи, которые подразумевают деление пользователей на разные роли в зависимости от их полномочий помогают разделять доступы к информации программного комплекса (рис. 4.6). Поддержка разделения пользователей системы на рабочие группы по категориям: менеджер-администратор, аналитик, эксперт.

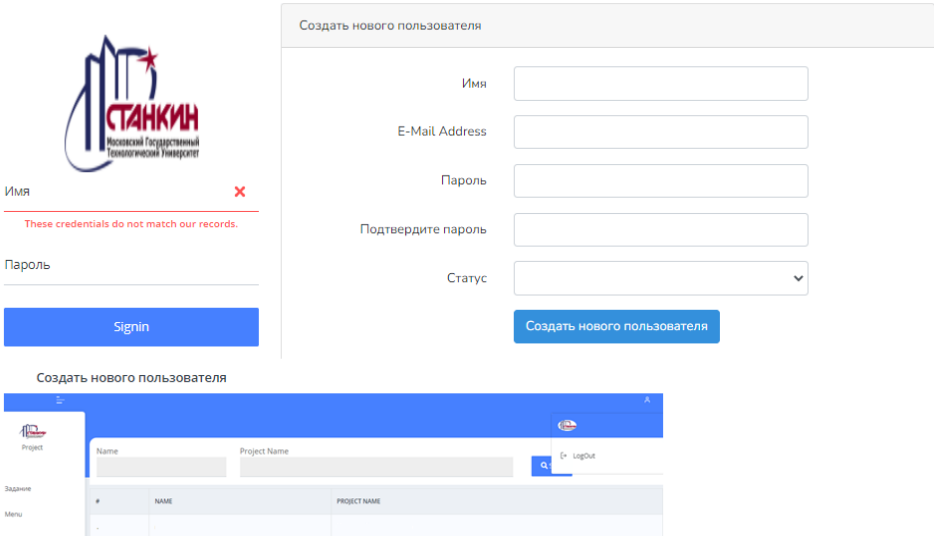


Рис. 4.6. Фрагмент экрана «Управление» для работы администратора

Также на форме «Управления доступом» пользователь выбирает вкладку, определяющую порядок работы: либо формирование исходной информации, либо обработку концептуальных моделей.

Разработка автоматизированных систем и обработка концептуальных моделей – задачи, которые выполняют основные пользователи программного комплекса. Если понадобится дополнительная обработка пользователи смогут воспользоваться следующей формой (рис. 4.7). Подробно функции пользователей будут рассмотрены в инструкции по эксплуатации (подраздел 4.4.2).

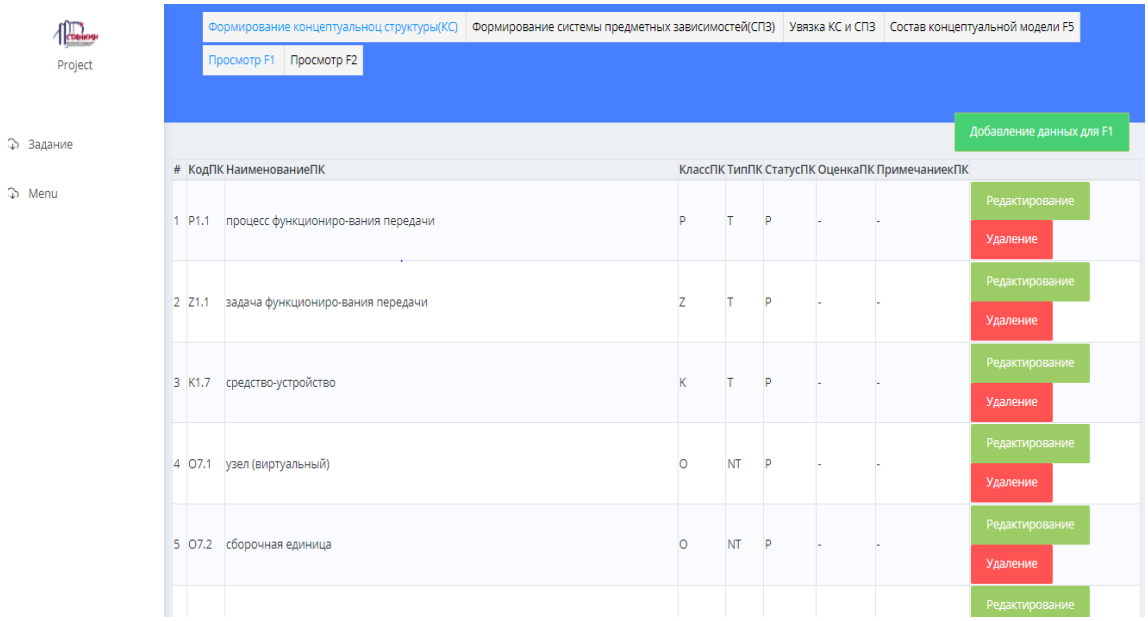


Рис. 4.7. Фрагмент экрана «Ввод данных» для работы предметного специалиста

Программная реализация формы занимает много времени и трудозатрат, так как включает в себя множество форм взаимодействий таблиц, функций а также структур. С учетом этого, с целью ускорения работы комплекса в целом, облегчения понимания текста программы, обеспечения удобства и сокращения времени работы, увеличение функциональности и а также сохранение возможности переноса в другие приложения, упрощения понимания текста программного обеспечения помимо непосредственного модуля формы в ее состав были включены модули классов.

Модули разделены по функциональному назначению и выполняют:

– формирование системы предметных зависимостей 1-го рода (СПЗ-1):

1) ввод данных;
2) редактирование данных;
3) удаление данных;
4) просмотр СПЗ-1;

– формирование основной концептуальной структуры (КС):

1) ввод данных;
2) редактирование данных;
3) удаление данных;
4) просмотр КС;

– формирование модели в целом:

1) ввод данных;
2) редактирование данных;
3) удаление данных;
4) просмотр увязки СПЗ-1 и КС;

– описание функций общего назначения Functions.

Каждый из перечисленных программных модулей отвечают за определенную процедуру дополнительной обработки данных и имеют свой набор функций.

Для формирования концептуальной модели используются спецификации с детальным описанием основных компонентов в качестве исходных данных. В рамках общепринятых правил, которые подразумевают что переход к обработке сематических конструкций возможен только после проверки корректности введенных ранее описаний пользователем-экспертом. Каждый модуль находится под защитой специального пароля, а обновить или как-либо изменить заполненные спецификации может только администратор. Затем осуществляется переход к обработке семантически конструкций.

Интерфейсные компоненты инструментов, используемые при анализе и обработке концептуальных моделей, реализуются через экранное меню для запуска каждой отдельной процедуры (рис. 4.8).

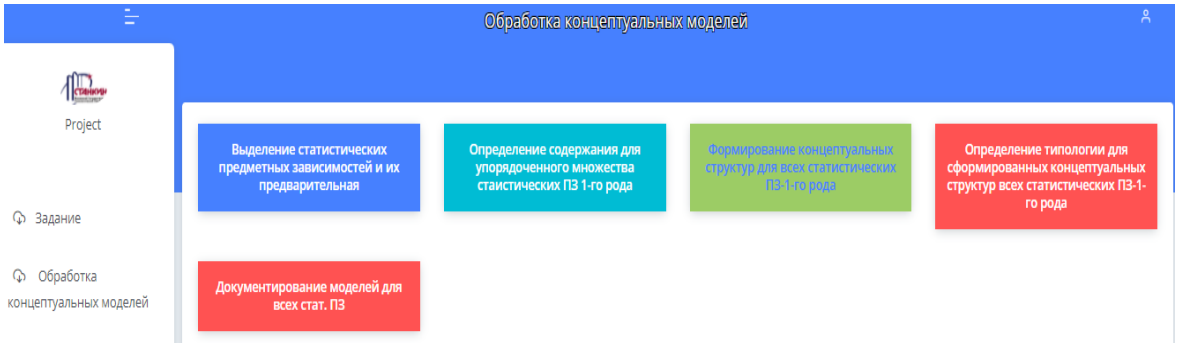


Рис. 4.8. Фрагмент экрана для дополнительной обработки концептуальной модели

4.4 Инструкции по установке и по эксплуатации программного обеспечения

4.4.1 Функции администратора

В рамках данного исследования дополнительной обработке концептуальных моделей осуществляется при использовании инструментария реализованного в системе управления базами данных MySQL. Поэтому пользователя необходимо установить XAMPP для работы программы на компьютере [85-91]. После установки XAMPP, необходимо нажать кнопку "Start" из Apache и MySQL для открытия PhpMyAdmin на рис.4.9.

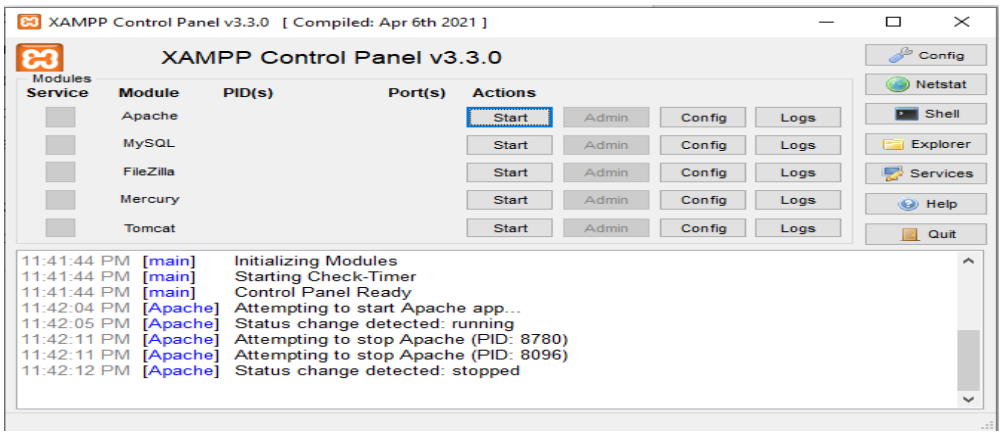


Рис.4.9. Фрагмент экрана «XAMPP Control Panel» для подключения PhpMyAdmin и проекта

И необходимо сохранить проект файл в htdocs из XAMPP для создания нового проекта. Администрирование данных программного комплекса – это

управление базой данных, то есть оно постоянно заполняется и обновляется данными, используемыми для ввода и обработки процессов для хранения описаний среды на рис.4.10.

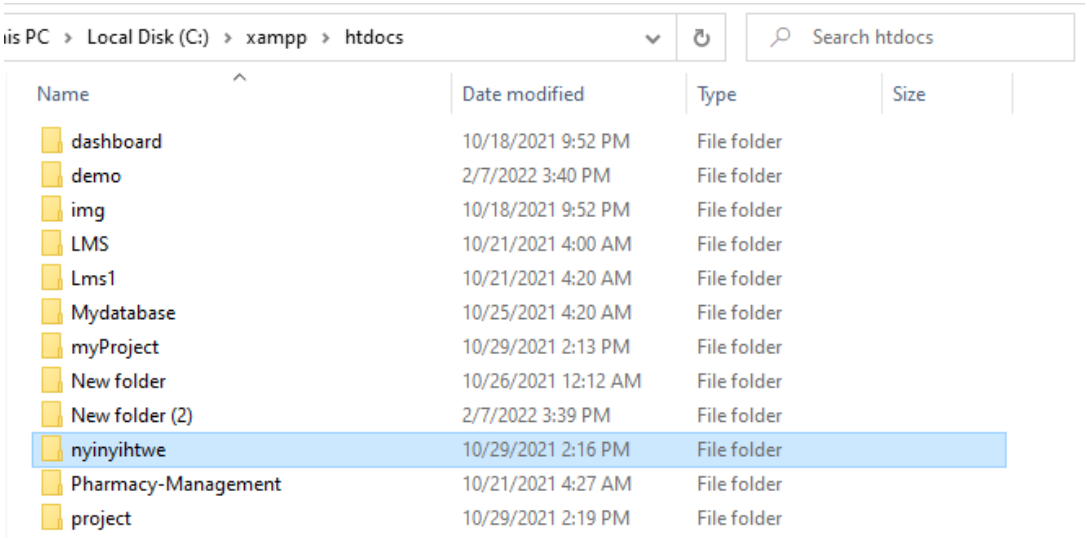


Рис. 4.10. Фрагмент экрана «Диалоговое окно References» при подключении проекта

В самом начале взаимодействия с программой и созданием концептуальных моделей администратор(менеджер) подразумевает заполнение данных по каждому пользователю с расшифровкой их заданий и назначить пользователей на выполнение предметных задач. Для выполнения этих действий необходимо выбрать проект и открыть на браузере (На пример http://localhost/demo/nyinyihtwe/public/ на рис. 4.11).

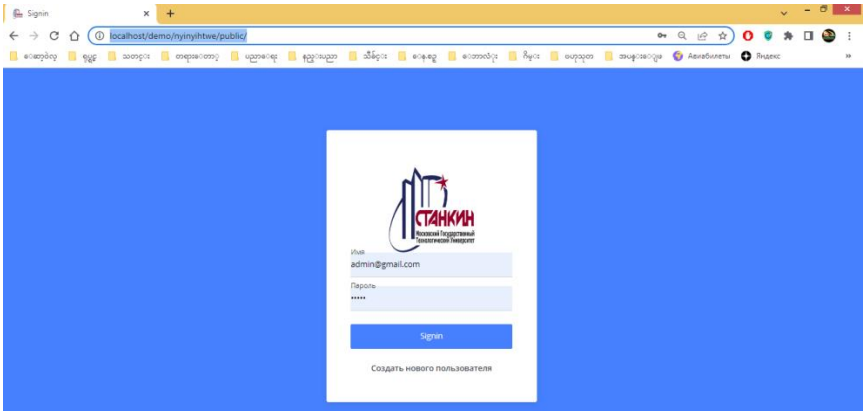


Рис. 4.11. Фрагмент экрана «Диалоговое окно References» при подключении проекта

В обязанности системного администратора входит ведение пользователей, выполнение задания по концептуальному моделированию и назначение

пользователя для выполнения задачи. Все эти операции подробно рассмотрены ниже.

Каждому пользователю присущ свой набор информации, которая сохранена в базе данных, благодаря этому происходит своевременная идентификация пользователя в системе. В результате должны заполнить следующие поля формы, чтобы отправить информацию о новых пользователях системы: Ввод данных нового пользователя: Имя, Email Address, Пароль, подтвердите пароль, Статус (на рис. 4.12).

На изображении показано поле с раскрывающимся списком **Статус**, перед входом в систему можно выбрать: роль «**Аналитик**» — обработку исходных данных, роль «**Эксперт**» предполагает обработку КМ, **Менеджер** (администратор). Концептуальные модели анализируются и формируются аналитиком. Эксперт обрабатывает данные модели в рамках предметных задач, необходимых для выполнения в ходе взаимодействия с программным комплексом.Каждому пользователю предлагается ввести определенные данные для защиты его аккаунта, после этого нужно нажать кнопку **Создать нового пользователя.**

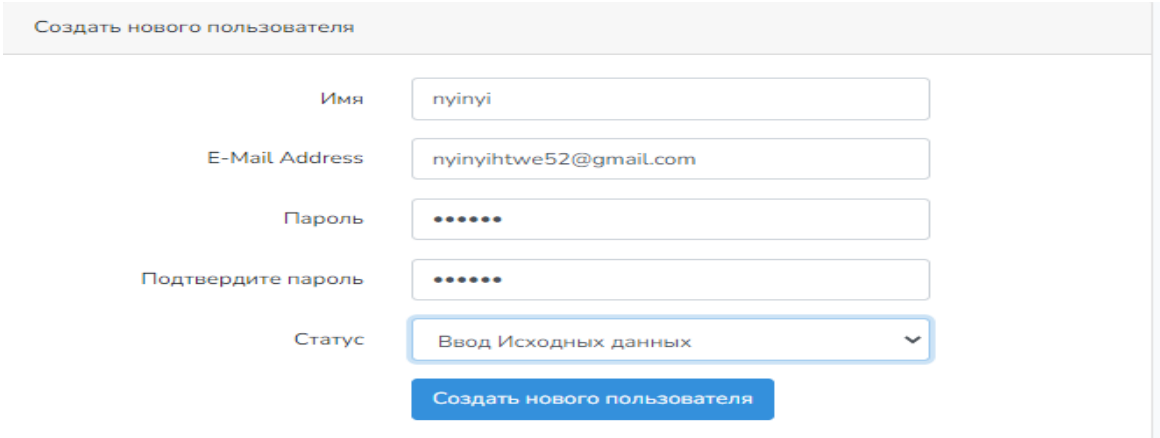


Рис. 4.12. Фрагмент экрана при вводе данных нового пользователя

В случае необходимости внесения каких-либо изменений в данные пользователя можно воспользоваться соответствующим разделом формы управления (рис.4.13). Выберите пользователя для обработки его данных из выпадающего списка пользователей в системе. После этого текстовые поля

заполняются текущей информацией о выбранном пользователе, а затем нажмите кнопку " **Change Info** ", чтобы изменить информацию о пользователе.

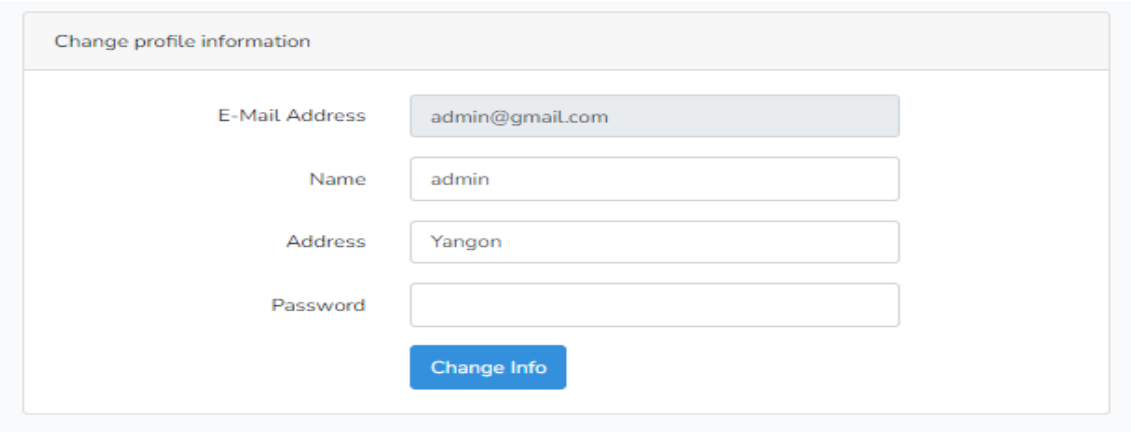


Рис.4.13. Фрагмент экрана для редактирования данных пользователя

Каждая предметная задача классифицируется по наименованиям. При вводе данных по новому заданию есть специальное поле формы, которое нужно заполнить, введя данные по новому заданию. Чтобы сохранить данные о новом задании необходимо воспользоваться кнопкой **Save** (рис.4.14)**.**

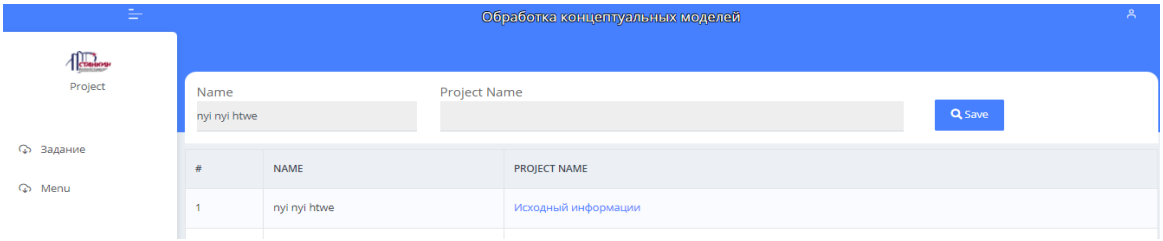


Рис.4.14. Фрагмент экрана для ввода данных нового задания и назначение пользователя на выполнение задания

По аналогии с данными пользователя можно изменить информацию о задании. В списке уже существующих заданий для этого выбираем нам необходимое и по кнопке **Edit** корректируем наименование. Для сохранения обновленных данных нажать кнопку **Save**.

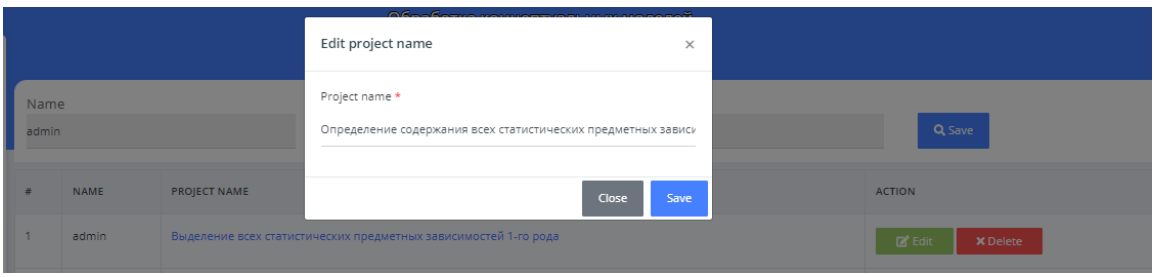


Рис. 4.15. Фрагмент экрана для редактирования данных задания

Программа подразумевает свободное назначение пользователей на концептуальное моделирование задачи. Чтобы назначить пользователя на выполнение конкретной задачи нужно выполнить ряд простых действий: выбрать пользователя и задание в раскрывающемся списке «пользователь и задание» (рис.4.16). Для сохранения информации по данному назначению необходимо нажать кнопку "**Project Name** (Наименование задания**).**

4.4.2 Функции пользователя - аналитика

В этой части инструкции по эксплуатации относится к описанию функций пользователя-аналитика при выполнении процедур формирования исходной информации. Аналитик выполняет действия, связанные с началом работы в системе и формирует исходную информацию.

При этом функции, связанные с началом и окончанием работы пользователя в системе, считаются управляющими и будут рассмотрены в данном разделе.

Каждый новый пользователь должен ввести имя пользователя и пароль в соответствующую форму **Signin**: Имя пользователя, Пароль (на рис. 4.11). В случае же если Имя пользователя и пароль неправильно, программа выдаст сообщение об ошибке.

Формирование концептуальной модели предметной задачи начинается с выбора своих данных, то есть имени пользователя из специальной формы с выпадающим списком. После выбора данной информации, пользователь увидит свое задание в соответствующем поле, данные заполняются на основе базы данных автоматически. Чтобы закончить работу с программой, воспользуйтесь кнопкой «Logout» (рис. 4.16).

В этом случае, все изменения, связанные с работой пользователя, сохраняются автоматически в базе данных, поэтому нет необходимости сохранять результаты другим способом.

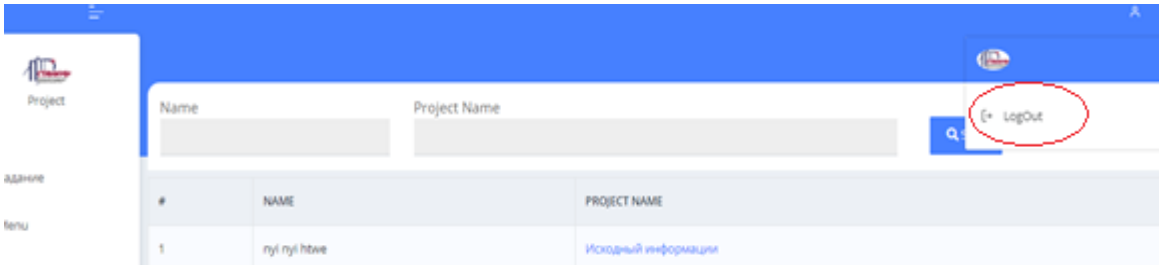


Рис.4.16. Фрагмент экрана для процедуры окончания работы

В вкладке «Исходная информация» представлены:

-формирование концептуальной структуры (КС),

-формирование системы предметных зависимостей (СПЗ -1),

-увязка КС и СПЗ-1 (Матричная диаграмма).

Это позволяет заполнять спецификации и просматривать табличное описание основных концептуальных конструкций по формам [2]: Форма F1- Элемент стат. структуры , Форма F2- Бин. связи стат. структуры, Форма F3- Элемент дин структуры, Форма F4- Бин. связи дин. структуры, Форма F6- Структура увязки.

Определяемые пользователем функции включают:

1) Формирование системы предметных зависимостей. Процесс формирования системы предметных зависимостей представляет табличном (спецификации F3). Функции пользователя по формированию системы предметных зависимостей включают ввод, редактирование и удаление данных.

Ввод данных - вкладка Формирование системы предметных зависимостей (СПЗ-1) содержит две вкладки для ввода и редактирования данных.

Для описания предметных зависимостей 1-го рода выбранной функциональной конструкции пользователю необходимо заполнить соответствующие строки виртуального аналога таблицы F3, причем часть данных, которая известна заранее и определена типом конструкции, уже автоматически сформирована.

По окончании ввода данных для занесения их в спецификации необходимо нажать кнопку «**Добавление данных**» новые ПЗ-1 в таблицу F3. При этом

автоматически осуществляется контроль введенной информации, переходом на вкладки спецификаций Просмотр таблицы F3 (на рис.4.17 и 4.18).

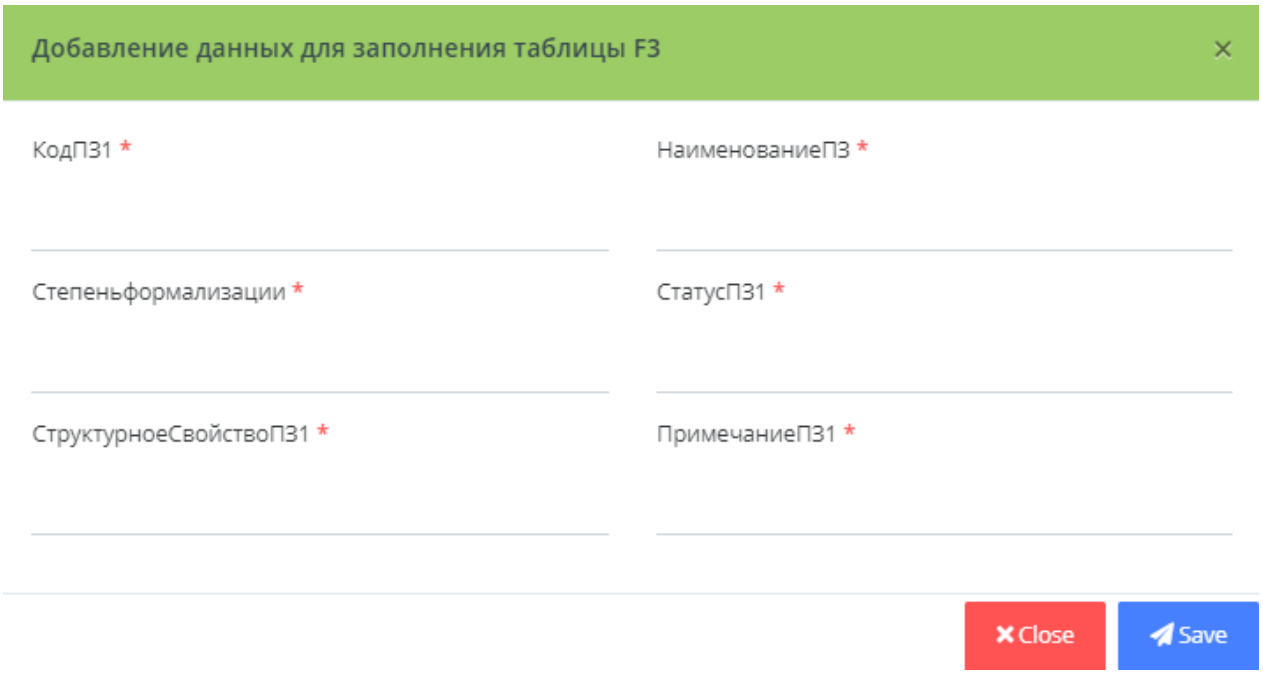


Рис. 4.17. Фрагмент экрана для добавления данных в таблице F3

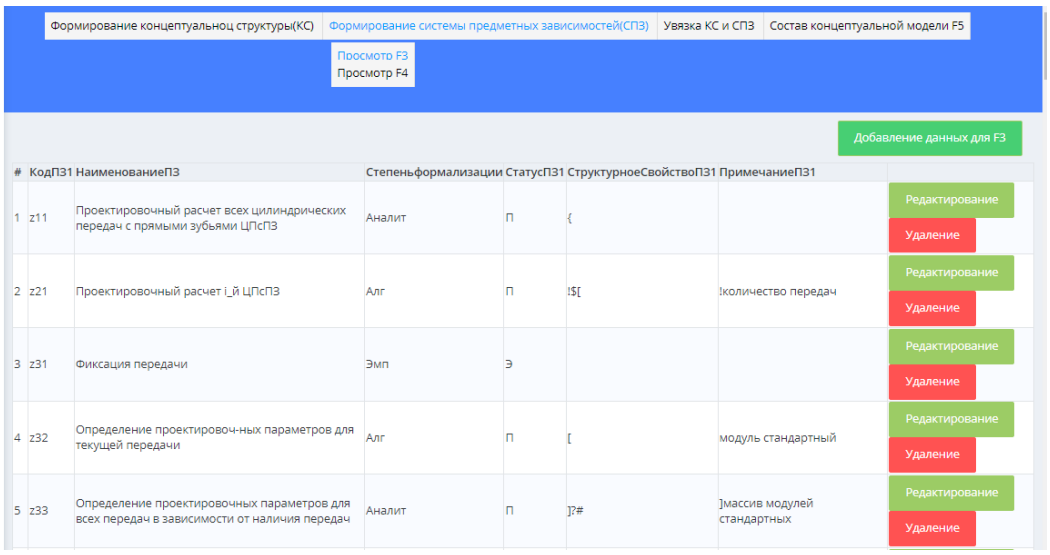


Рис.4.18. Фрагмент экрана для заполненной таблицы F3

Редактирование/удаление данных- подсистема редактирования и удаления данных СПЗ-1 позволяет изменять как свойства отдельной предметной зависимости – наименование, степень формализации, статус; так и вносить коррективы в структуру декомпозиции задачи – удалять выбранную ПЗ-1, удалять все предметные зависимости низших уровней, для которых данная ПЗ-1 является предком, добавлять (вставлять пропущенную) предметную зависимость 1-го рода.

Для редактирования предметной зависимости необходимо выбрать кнопку «**Редактирование**», и для удаления предметных зависимостей порожденных указанной (без удаления самой редактируемой ПЗ-1) в списке «Редактируется» предусмотрено удаление всех порожденных ПЗ-1 (на рис.4.19).

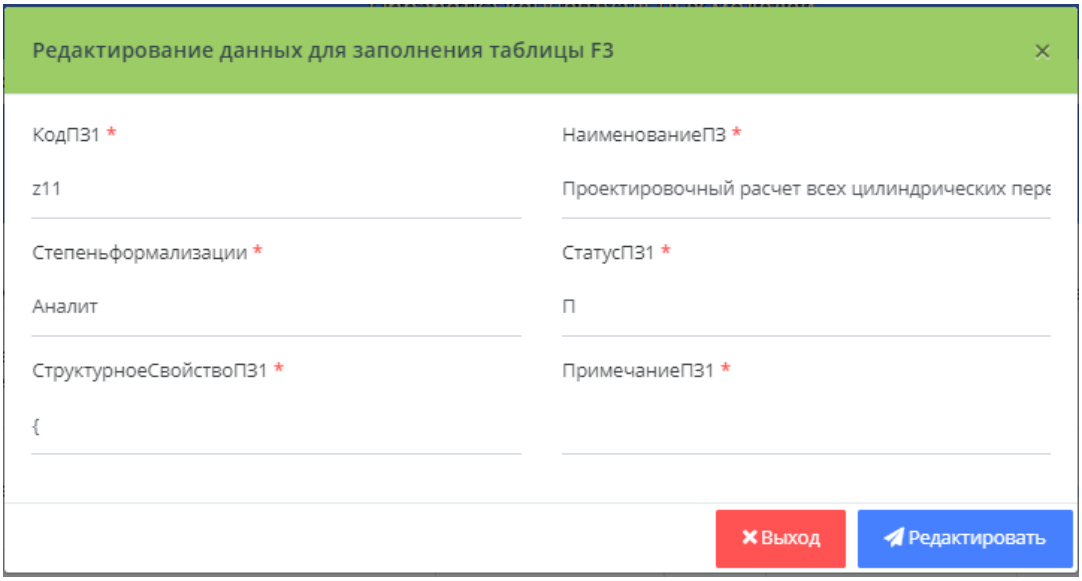


Рис.4.19. Фрагмент экрана для данных редактируемой предметной зависимости

Удаление данных- для удаления данных необходимо выбрать Удаление данных на вкладке, после этого в соответствующих списках будут отражены сообщения системы (на рис.4.20).

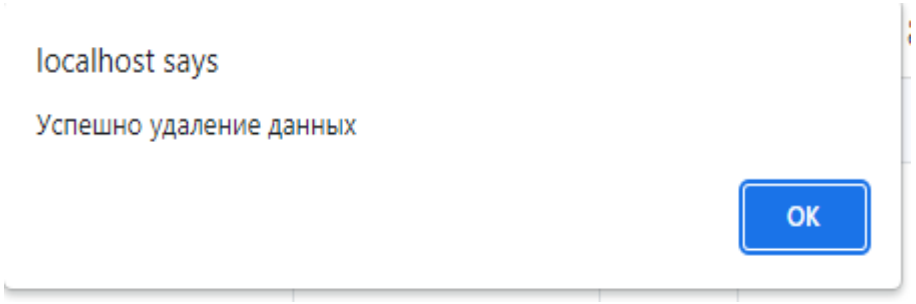
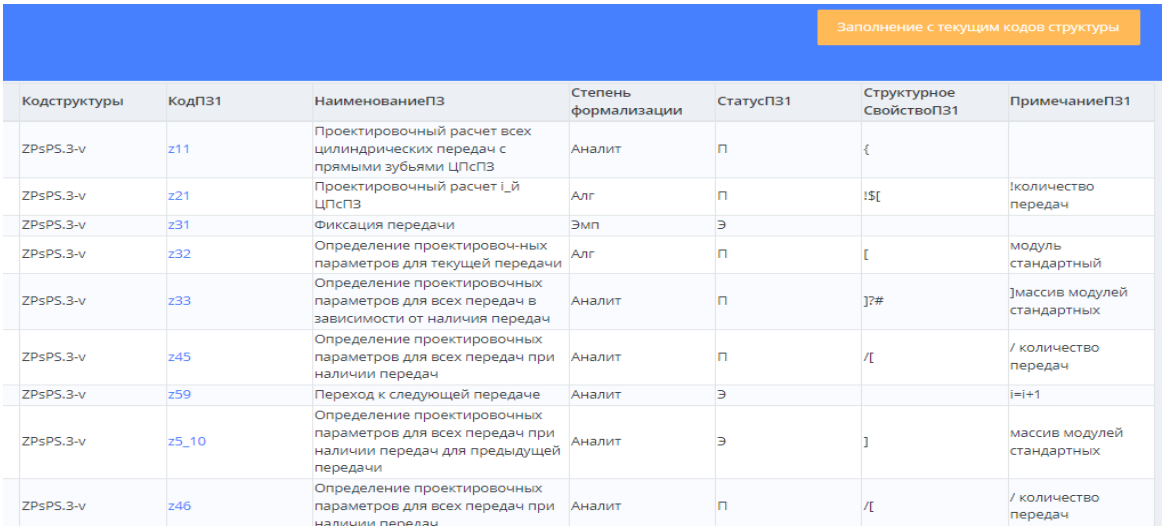


Рис.4.20. Фрагмент экрана для сообщения системы о невозможности удаления указанной ПЗ-1

Заполнение спецификации с текушими кодами структуры таблицы. (на рис. 4.21).



Кодструктуры	КодПЗ1	НаименованиеПЗ	Степень формализации	СтатусПЗ1	Структурное СвойствоПЗ1	ПримечаниеПЗ1
ZPsPS.3-v	z11	Проектировочный расчет всех цилиндрических передач с прямыми зубьями ЦПсПЗ	Аналит	П	{	
ZPsPS.3-v	z21	Проектировочный расчет i_й ЦПсПЗ	Алг	П	!$[	!количество передач
ZPsPS.3-v	z31	Фиксация передачи	Эмп	Э		
ZPsPS.3-v	z32	Определение проектировоч-ных параметров для текущей передачи	Алг	П	[	модуль стандартный
ZPsPS.3-v	z33	Определение проектировочных параметров для всех передач в зависимости от наличия передач	Аналит	П	]?#	]массив модулей стандартных
ZPsPS.3-v	z45	Определение проектировочных параметров для всех передач при наличии передач	Аналит	П	/[	/ количество передач
ZPsPS.3-v	z59	Переход к следующей передаче	Аналит	Э		i=i+1
ZPsPS.3-v	z5_10	Определение проектировочных параметров для всех передач при наличии передач для предыдущей передачи	Аналит	Э	]	массив модулей стандартных
ZPsPS.3-v	z46	Определение проектировочных параметров для всех передач при наличии передач	Аналит	П	/[	/ количество передач

Рис.4.21. Фрагмент экрана для заполнения спецификации с текушими кодами структуры таблицы F3

2) Функции пользователя по формированию концептуальной структуры включают ввод, редактирование и удаление данных.

Ввод данных - вкладка «Формирование концептуальной структуры (КС)» содержит три вкладки для ввода (на рис. 4.22-4.24), редактирования и удаления данных и для просмотра и визуального контроля результатов ввода – две спецификации (F1 и F2), построенные по данным указанных таблиц.

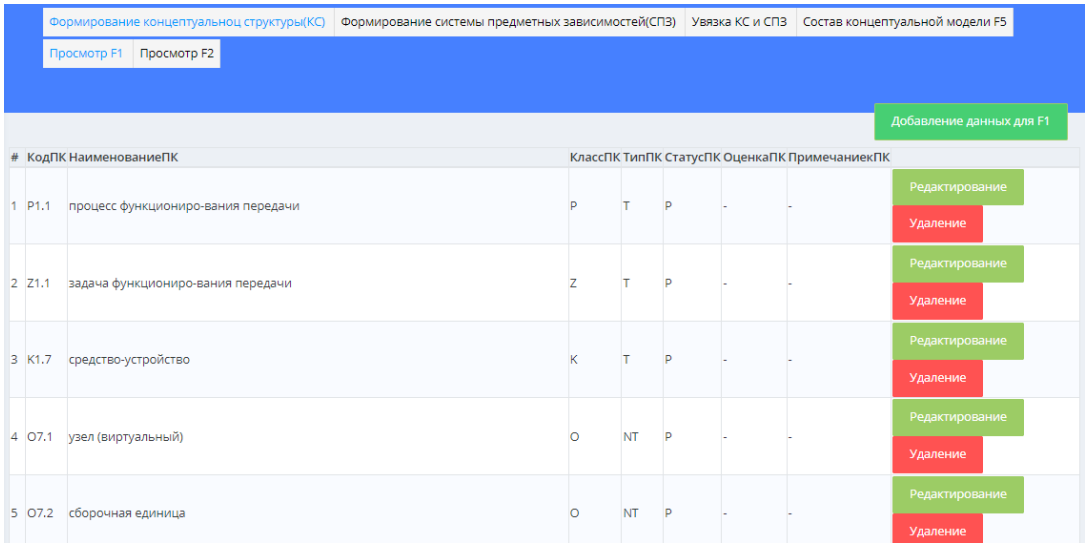


Рис. 4.22. Фрагмент экрана при вводе данных новой предметной категории в таблице F1

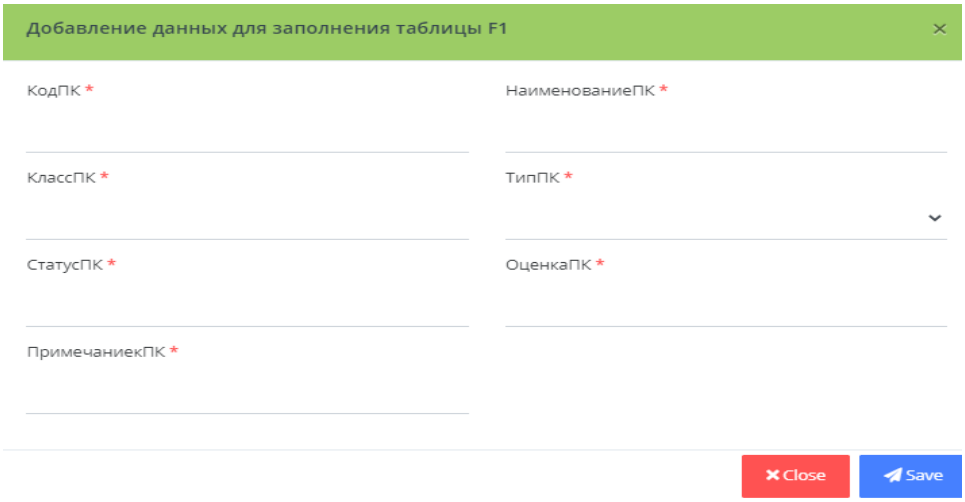


Рис. 4.23. Фрагмент экрана для отображения добавляемых записей в таблице F1

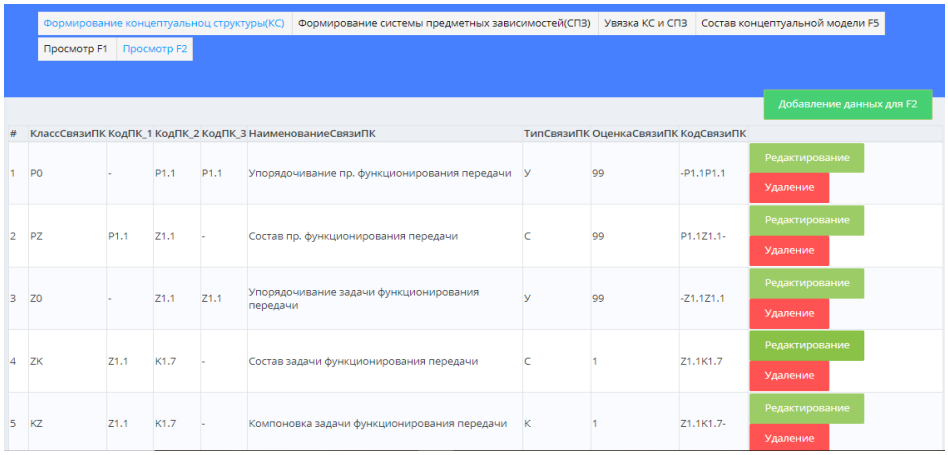
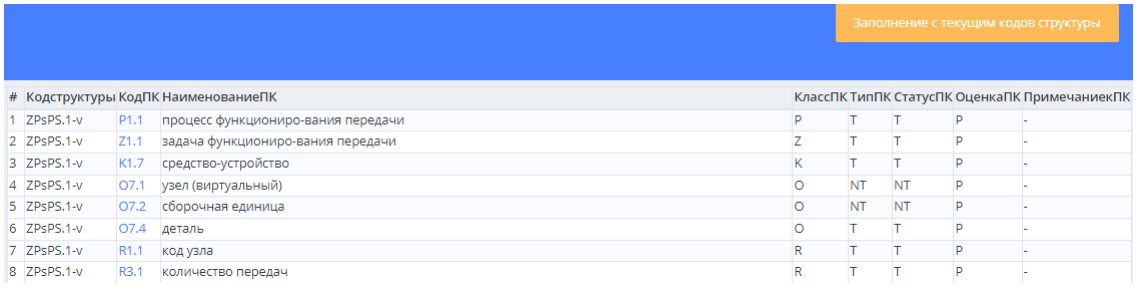


Рис. 4.24. Фрагмент экрана при вводе данных новой предметной категории в таблице F2

Данные, которые будут внесены в соответствующие спецификации F1 и F2, отображаются во второй зоне вкладки Добавление данных (на рис. 4.25), поэтому перед непосредственным вводом данных в таблицы полезно убедиться, что сохраняемая информация полностью устраивает пользователя.

Рис.4.25. Фрагмент экрана для отображения добавляемых записей в таблице F2

Для сохранения введенной информации в базе данных необходимо нажать кнопку «**Добавить новые данные в табл. F1 и F2**». Контроль введенной информации можно осуществить, перейдя на вкладки спецификаций Просмотр таблицы F1 (на рис.4.26) и Просмотр таблицы F2 (на рис. 4.27). Текущий фрагмент концептуальной структуры, основанный на данных таблиц F1 и F2.



#	Кодструктуры	КодПК	НаименованиеПК	КлассПК	ТипПК	СтатусПК	ОценкаПК	ПримечаниекПК
1	ZPsPS.1-v	P1.1	процесс функциониро-вания передачи	P	T	T	P	-
2	ZPsPS.1-v	Z1.1	задача функциониро-вания передачи	Z	T	T	P	-
3	ZPsPS.1-v	K1.7	средство-устройство	K	T	T	P	-
4	ZPsPS.1-v	O7.1	узел (виртуальный)	O	NT	NT	P	-
5	ZPsPS.1-v	O7.2	сборочная единица	O	NT	NT	P	-
6	ZPsPS.1-v	O7.4	деталь	O	T	T	P	-
7	ZPsPS.1-v	R1.1	код узла	R	T	T	P	-
8	ZPsPS.1-v	R3.1	количество передач	R	T	T	P	-

Рис. 4.26. Фрагмент экрана для заполнения спецификации с текушими кодами структуры таблицы F1

#	Кодструктуры	КлассСвязиПК	КодПК_1	КодПК_2	КодПК_3	НаименованиеСвязиПК	ТипСвязиПК	ОценкаСвязиПК	КодСвязиПК
1	ZPsPS.1-v	PZ	P1.1	Z1.1	-	Состав пр. функционирования передачи	C	99	P1.1Z1.1-
2	ZPsPS.1-v	Z0	-	Z1.1	Z1.1	Упорядочивание задачи функционирования передачи	У	99	-Z1.1Z1.1
3	ZPsPS.1-v	ZK	Z1.1	K1.7	-	Состав задачи функционирования передачи	C	1	Z1.1K1.7
4	ZPsPS.1-v	KZ	Z1.1	K1.7	-	Компоновка задачи функционирования передачи	K	1	Z1.1K1.7-

Рис. 4.27. Фрагмент экрана для заполнения спецификации с текушими кодами структуры таблицы F2

Редактирование данных - подсистема редактирования данных концептуальной структуры позволяет изменять как свойства отдельной предметной категории – код, наименование, тип, оценку, примечание (на рис. 4.28-4.29). Для редактирования кода предметной категории необходимо выбрать изменяемые данные.

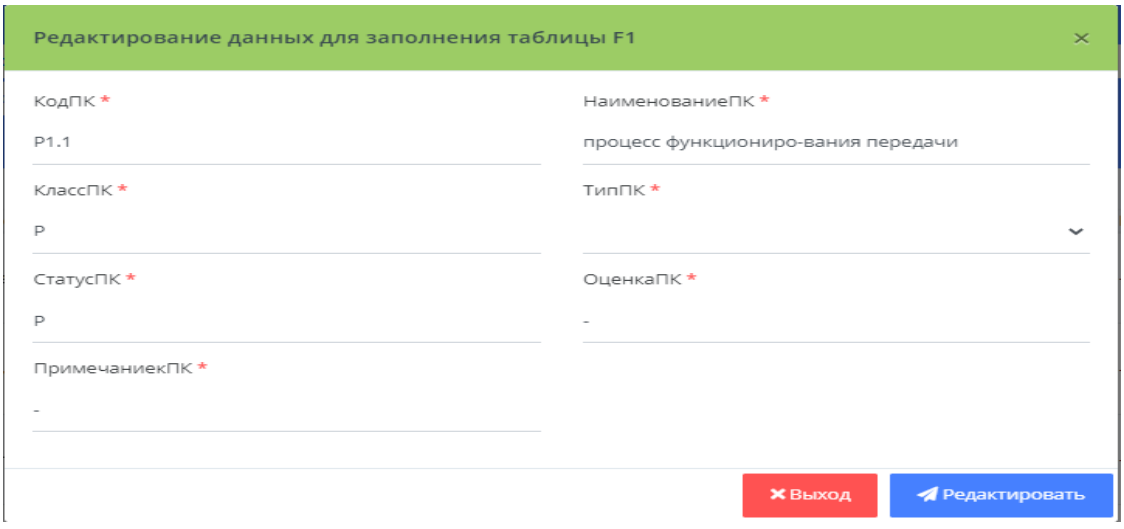


Рис. 4.28. Фрагмент экрана для редактирования данных, определяющих код ПК

Рис.4.29. Фрагмент экрана для выбора предметной категории для редактирования данных в таблице F2

Кроме того, в соответствующем списке отображаются записи таблицы F2, которые будут изменены в результате данного типа редактирования.

Удаление данных- для удаления данных необходимо выбрать Удаление данных на вкладке, после этого в соответствующих списках будут отражены удаляемые записи таблиц F1 и F2 (на рис.4.30). Помимо выбранной ПК удаляются все предметные категории нижних классов, связанные с текущей.

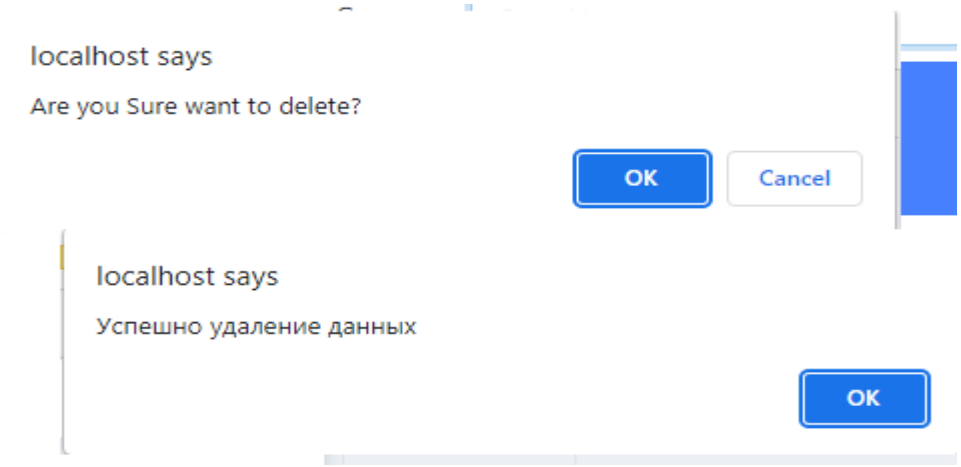


Рис.4.30. Фрагмент экрана для удаления данных из таблиц F1 и F2

3) Увязка КС и СПЗ. Процесс формирования концептуальной модели в целом заключается в увязке полученных на предыдущих этапах структур – системы предметных зависимостей 1-го рода и концептуальной структуры. Поэтому

переход к этой стадии проектирования возможен только после полного завершения работ, связанных с формированием первых двух компонентов модели. Функции пользователя по формированию данного компонента модели включают ввод, редактирование и удаление данных.

Ввод данных- вкладка «Увязка КС и СПЗ-1» содержит вкладку для ввода редактирования данных и вкладки для просмотра контроля результатов ввода – спецификация F6, построенной по данным указанной таблицы. Вкладка **Добавление данных** условно разделена на две взаимосвязанные зоны: первая предназначена для ввода информации пользователем о содержании текущей предметной зависимости и входящих в нее предметных категориях (на рис. 4.31), вторая предназначена для наглядного отображения вводимых данных в таблицу F6, а также уточнения ролевых и объемных свойств ПК (на рис. 4.32 и 4.33).

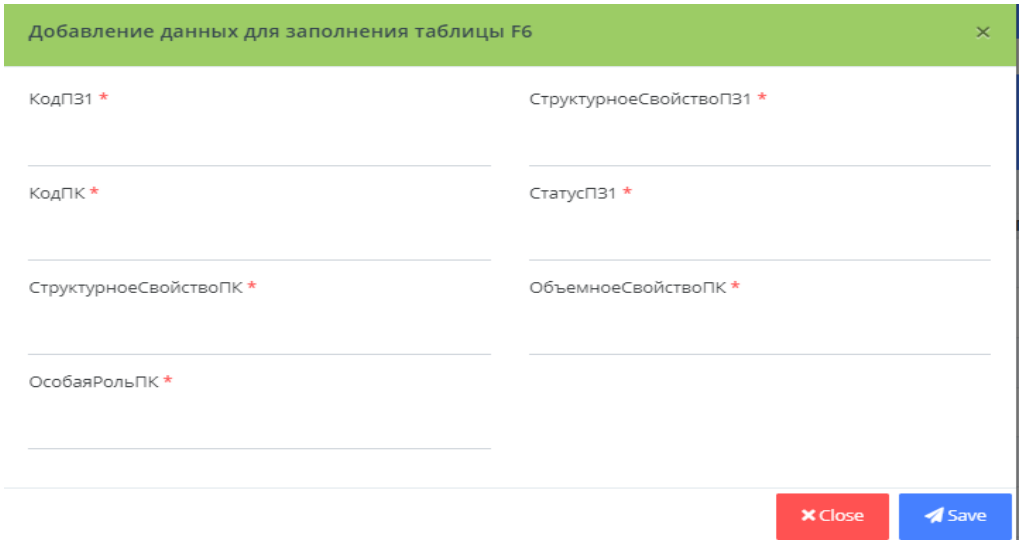


Рис.4.31. Фрагмент экрана при вводе данных по содержанию текущей ПЗ-1

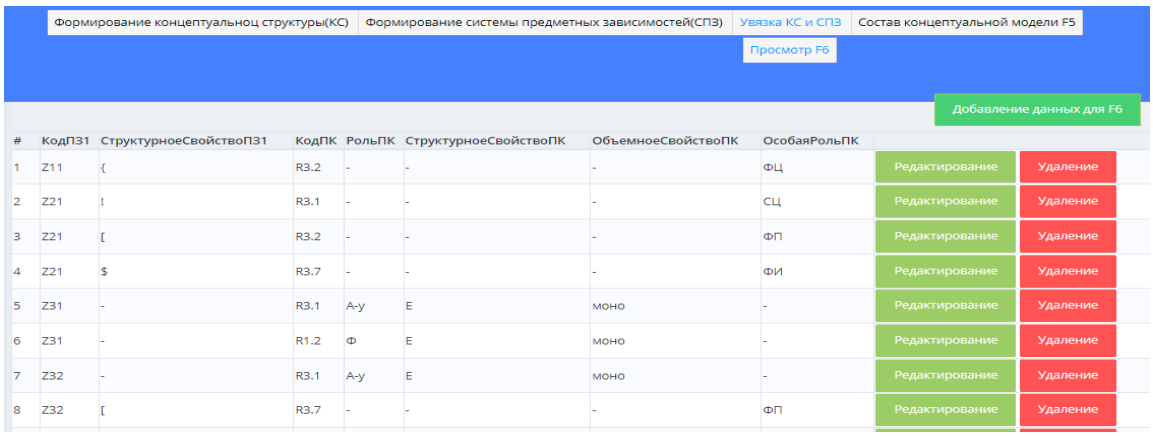
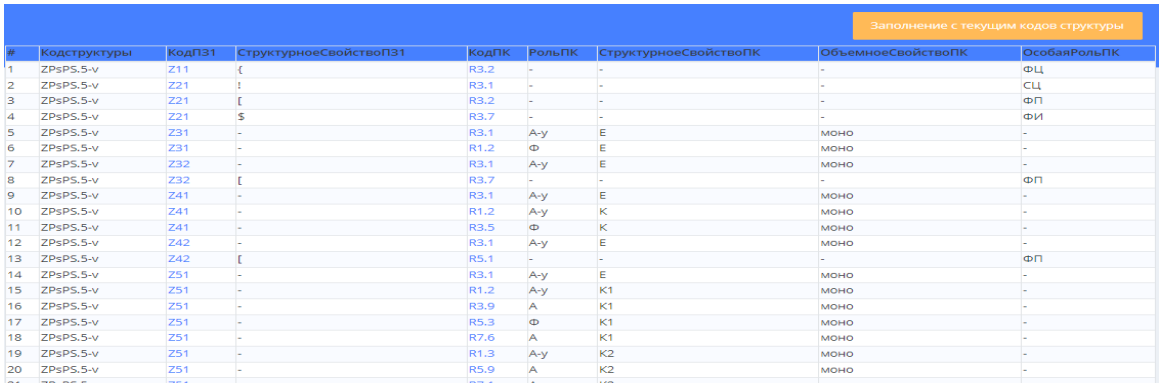


Рис.4.32. Фрагмент экрана при вводе информации в таблицу F6 о содержании текущей ПЗ-1



Заполнение с текущим кодов структуры

#	Кодструктуры	КодПЗ1	СтруктурноеСвойствоПЗ1	КодПК	РольПК	СтруктурноеСвойствоПК	ОбъемноеСвойствоПК	ОсобаяРольПК
1	ZPsPS.5-v	Z11	{	R3.2	-	-	-	ФЦ
2	ZPsPS.5-v	Z21	!	R3.1	-	-	-	СЦ
3	ZPsPS.5-v	Z21	[	R3.2	-	-	-	ФП
4	ZPsPS.5-v	Z21	$	R3.7	-	-	-	ФИ
5	ZPsPS.5-v	Z31	-	R3.1	A-y	E	моно	-
6	ZPsPS.5-v	Z31	-	R1.2	Ф	E	моно	-
7	ZPsPS.5-v	Z32	-	R3.1	A-y	E	моно	-
8	ZPsPS.5-v	Z32	[	R3.7	-	-	-	ФП
9	ZPsPS.5-v	Z41	-	R3.1	A-y	E	моно	-
10	ZPsPS.5-v	Z41	-	R1.2	A-y	K	моно	-
11	ZPsPS.5-v	Z41	-	R3.5	Ф	K	моно	-
12	ZPsPS.5-v	Z42	-	R3.1	A-y	E	моно	-
13	ZPsPS.5-v	Z42	[	R5.1	-	-	-	ФП
14	ZPsPS.5-v	Z51	-	R3.1	A-y	E	моно	-
15	ZPsPS.5-v	Z51	-	R1.2	A-y	K1	моно	-
16	ZPsPS.5-v	Z51	-	R3.9	A	K1	моно	-
17	ZPsPS.5-v	Z51	-	R5.3	Ф	K1	моно	-
18	ZPsPS.5-v	Z51	-	R7.6	A	K1	моно	-
19	ZPsPS.5-v	Z51	-	R1.3	A-y	K2	моно	-
20	ZPsPS.5-v	Z51	-	R5.9	A	K2	моно	-

Рис.4.33. Фрагмент экрана для заполнения спецификации с текушими кодами структуры таблицы F6

Редактирование/удаление данных - подсистемы редактирования и удаления данных концептуальной модели в целом позволяет вносить построчные изменения в содержимое таблицы F6. Для редактирования информации необходимо выбрать в поле со списком тип изменений значение «**Редактирование»**, а в списке «редактирование данных таблицы F6» соответствующую строку для корректировки. После выполнения этих действий данные текущей строки станут доступными для изменения (на рисунке 4.34). Пользователь может вручную изменить значения полей Структурное свойство ПЗ-1, Роль ПК, Структурное свойство ПК, Объемное свойство ПК, Особая роль ПК.

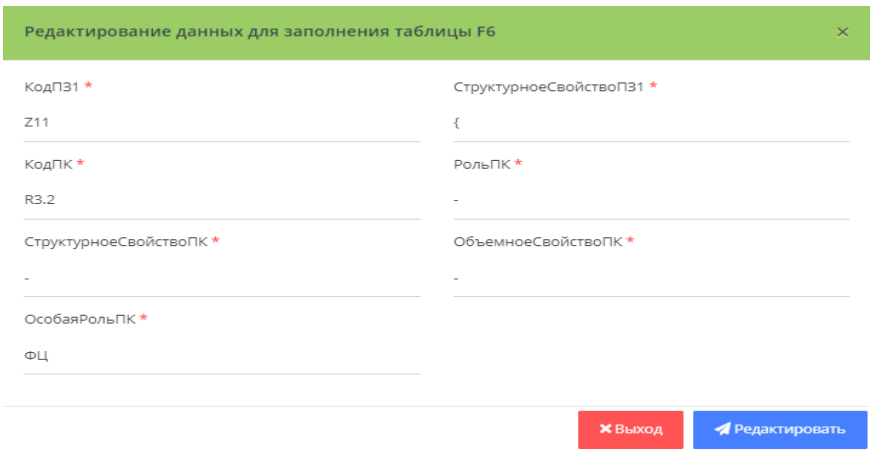


Рис.4.34. Фрагмент экрана для редактирования данных таблицы F6

При необходимости удалить строку выбранную строку из таблицы F6 необходимо выбрать кнопку «**Удаление**».

Частично табличное описание пользователя-аналитика для исходных данных приведено в Приложении А.

4.4.3 Функции пользователя - эксперта

В числе функций данного пользователя-эксперта также находится начало работы в системе процессы и операции, связанные с дополнительной обработкой концептуальной модели в ходе описания которой формируются базовые семантические конструкции.

И на вкладе «Обработка КМ» определены: выявление всех статистических ПЗ-1-го рода и их предварительной обработки, определение содержания всех статистических ПЗ-1-го рода, формирование концептуальных моделей для всех статистических ПЗ-1-го рода, определение типологии для всех статистических ПЗ-1-го рода и документирование моделей для них (на рис.4.1).

Процесс выделения статистических предметных ограничений и их предварительной обработки выполняется путем фиксации предметных зависимостей 1-го рода, которые имеет степень формализации как «статистическую». Далее следует выполнить упорядочение или ранжирование всех выделенных зависимостей.

Процесс выделения всех статистических предметных ограничений 1-го рода позволяет получить следующие результаты (рис.4.35).

Процесс определения содержания всех статистических предметных ограничений 1-го рода выполняется путем фиксации множества предметных категорий и их ролевых свойств в рамках этих ограничений 1-го рода и формируется во взаимосвязи таблиц **F3** и **F6**, служащих основой для заполнения спецификации **F6** и позволяет получить следующие результаты (рис.4.36).

Процесс формирование КМ для всех статистических ПЗ-1-го рода заключается в построении «родословных» для всех ПК, связанных со статистической ПЗ 1-го рода, определяющих соответствующую КМ 1-го рода.

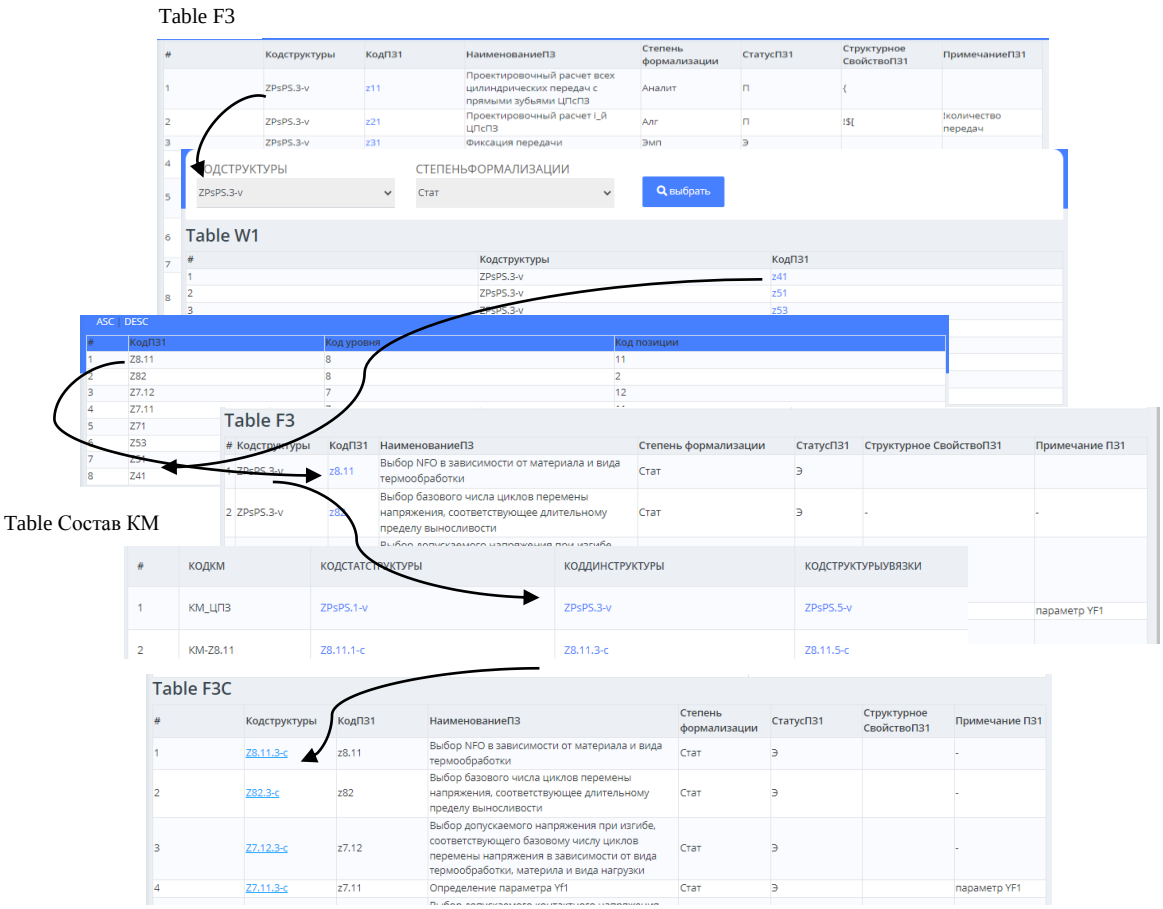


Рис.4.35. Фрагмент экрана для получения элементов дин. структуры [F3^c]

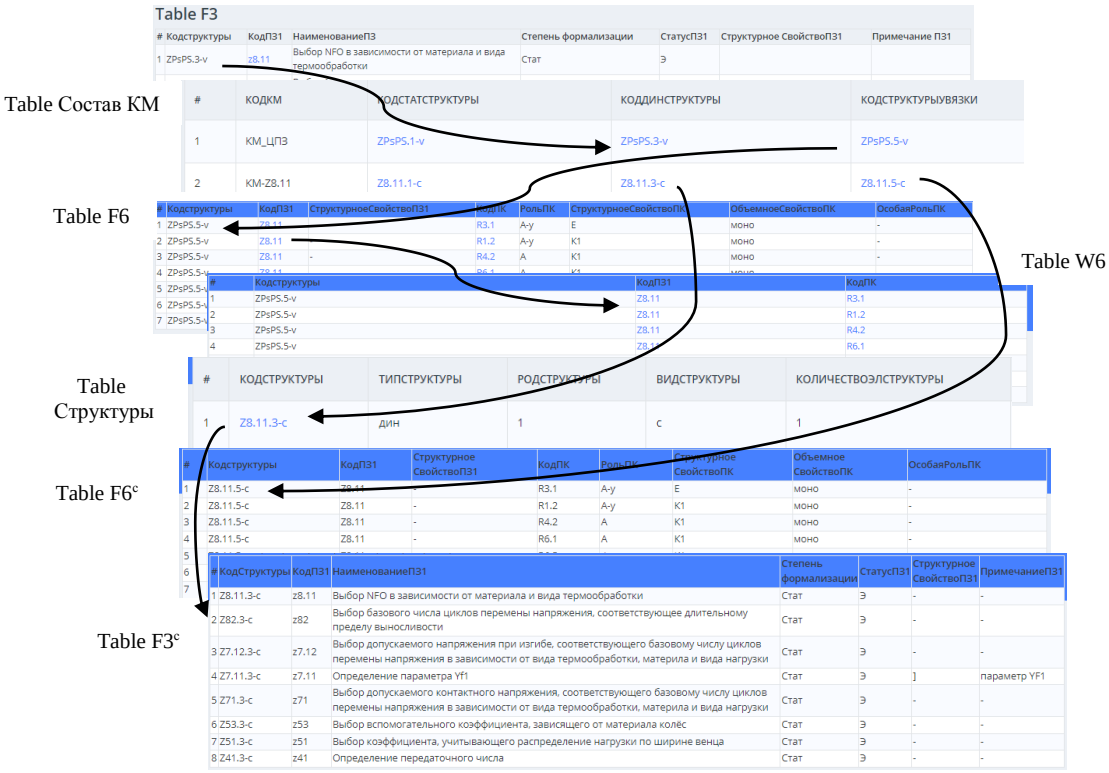


Рис.4.36. Фрагмент экрана для определения содержания всех статистических предметных зависимостей 1-го рода

Фактически выявленная концептуальная структура отражает систему знаний, которой обладает предметный специалист. Данные, которые будут внесены в соответствующие спецификации F1 и F2, отображаются во второй зоне вкладки Добавление данных. **Добавление данных** условно разделена на две взаимосвязанные зоны: первая предназначена для ввода информации пользователем о текущей предметной категории и ее связях с другими ПК, вторая предназначена для наглядного отображения вводимых данных в таблицы F1 и F2. Контроль введенной информации можно осуществить, перейдя на вкладки спецификаций Просмотр таблицы F1 и F2 (на рис.4.37).

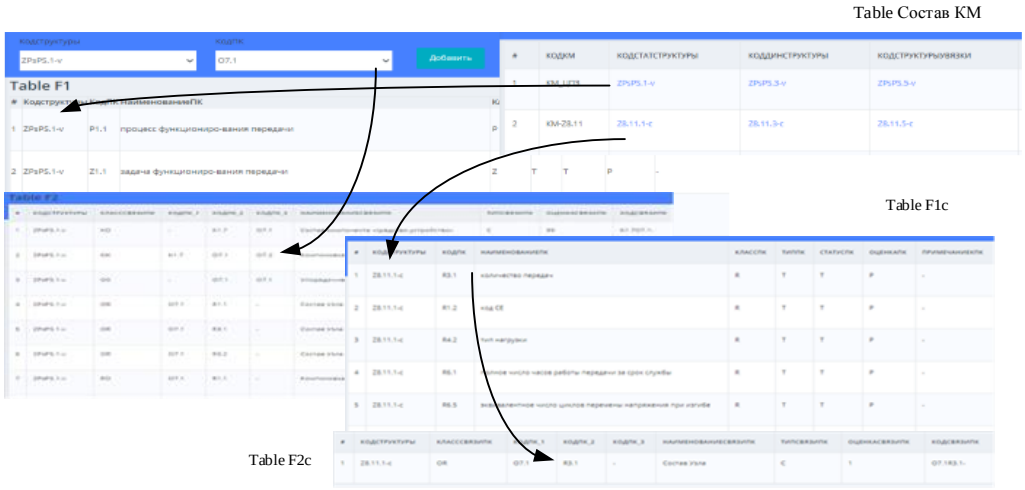


Рис.4.37. Фрагмент экрана для формирования “родословных” структур для всех статистических ПЗ-1-го рода

Процесс формирования концептуальных структур для всех статистических предметных зависимостей 1-го рода позволяет получить следующие результаты $F1^c$ и $F2^c$ (на рис. 4.38):

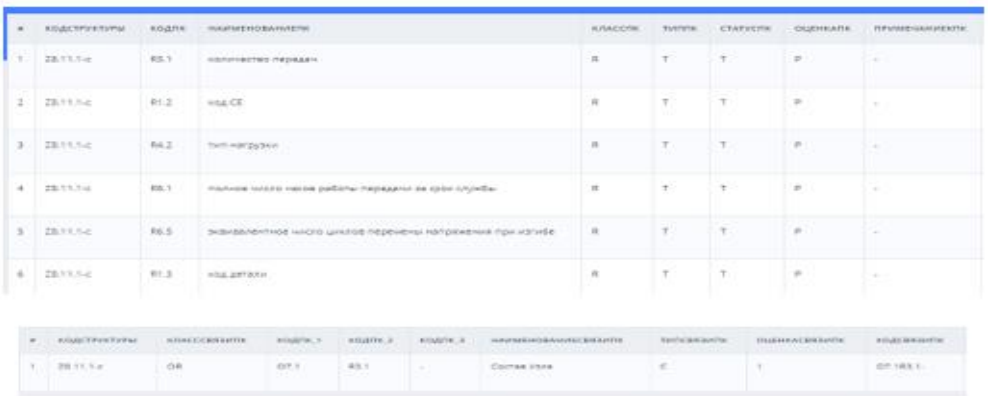


Рис.4.38. Фрагмент экрана для отображения добавляемых записей в таблицы $F1^c$ и $F2^c$

Процесс определения типологии для всех статистических ПЗ-1-го рода обусловлен ограничениями, накладываемыми универсальным концептуальным представлением, и фиксирует «корневую» или замыкающую ПК определенного класса для конкретной предметной зависимости 1-го рода (на рис.4.39-4.40).

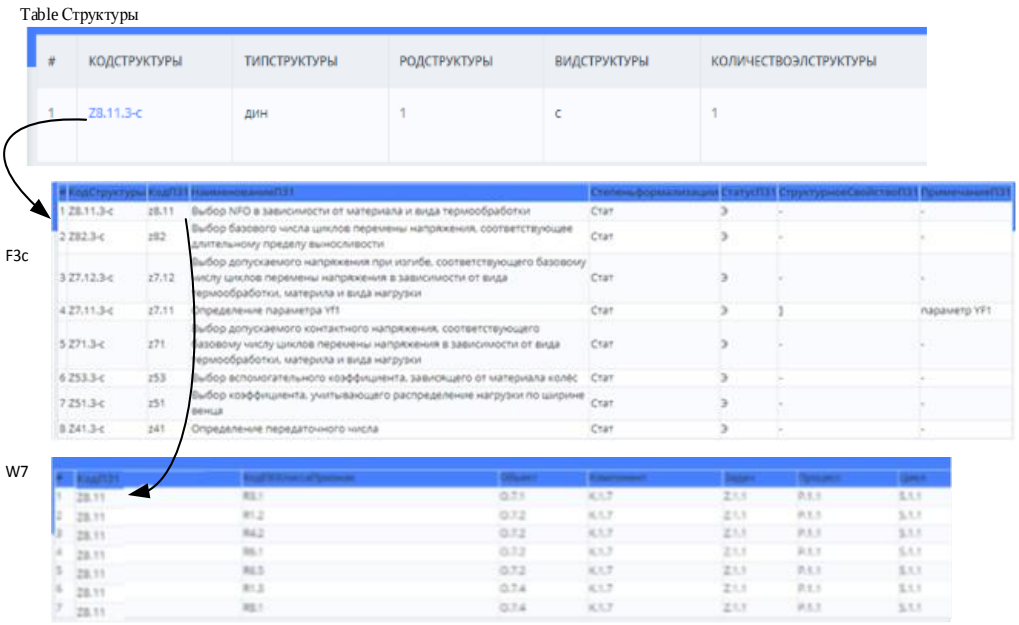


Рис. 4.39. Фрагмент экрана для определения типологии для всех статистических ПЗ-1-го рода

Процесс определения типологии для всех статистических ПЗ-1-го рода позволяет получить следующие результаты:

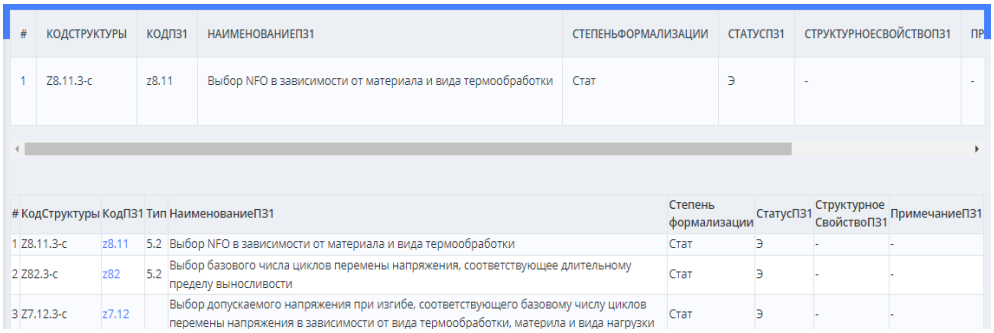


Рис.4.40. Фрагмент экрана для определения типологии для сформированных концептуальных моделей всех статистических ПЗ-1-го рода

На этапе документирование моделей для всех статистических ПЗ 1 -го рода выводятся хранящиеся в спецификациях компоненты модели на рис.4.41a-4.41d. Полностью табличное описание пользователя-эксперта для обработки концептуальной модели приведено в Приложении Б.

F1 | F1c | F2 | F2c | F3c | F3 | F4 | F4c | F6 | F6c

#	КодСтруктуры	КодПЗ1	Тип	НаименованиеПЗ1	Степень формализации	СтатусПЗ1	Структурное СвойствоПЗ1	ПримечаниеПЗ1
1	Z8.11.3-c	z8.11	5.2	Выбор NFO в зависимости от материала и вида термообработки	Стат	Э	-	-
2	Z82.3-c	z82	5.2	Выбор базового числа циклов перемены напряжения, соответствующее длительному пределу выносливости	Стат	Э	-	-
3	Z7.12.3-c	z7.12	5.2	Выбор допускаемого напряжения при изгибе, соответствующего базовому числу циклов перемены напряжения в зависимости от вида термообработки, материла и вида нагрузки	Стат	Э	-	-
4	Z7.11.3-c	z7.11	5.2	Определение параметра Yf1	Стат	Э	]	параметр Yf1
5	Z71.3-c	z71	5.2	Выбор допускаемого контактного напряжения, соответствующего базовому числу циклов перемены напряжения в зависимости от вида термообработки, материла и вида нагрузки	Стат	Э	-	-
6	Z53.3-c	z53	5.2	Выбор вспомогательного коэффициента, зависящего от материала колёс	Стат	Э	-	-
7	Z51.3-c	z51	5.2	Выбор коэффициента, учитывающего распределение нагрузки по ширине венца	Стат	Э	-	-
8	Z41.3-c	z41	5.2	Определение передаточного числа	Стат	Э	-	-

Рис.4.41a. Фрагмент экрана при просмотре таблицы F3^c

F1 | F1c | F2 | F2c | F3c | F3 | F4 | F4c | F6 | F6c

#	Кодструктуры	КодПЗ1	Структурное СвойствоПЗ1	КодПК	РольПК	Структурное СвойствоПК	Объемное СвойствоПК	ОсобаяРольПК
1	Z8.11.5-c	Z8.11	-	R3.1	A-y	E	моно	-
2	Z8.11.5-c	Z8.11	-	R1.2	A-y	K1	моно	-
3	Z8.11.5-c	Z8.11	-	R4.2	A	K1	моно	-
4	Z8.11.5-c	Z8.11	-	R6.1	A	K1	моно	-
5	Z8.11.5-c	Z8.11	-	R6.5	Ф	K1	моно	-
6	Z8.11.5-c	Z8.11	-	R1.3	A-y	K2	моно	-
7	Z8.11.5-c	Z8.11	-	R8.1	A	K2	моно	-
8	Z82.5-c	Z82	-	R3.1	A-y	E	моно	
9	Z82.5-c	Z82	-	R1.2	A-y	K1	моно	
10	Z82.5-c	Z82	-	R6.2	Ф	K1	моно	
11	Z82.5-c	Z82	-	R1.3	A-y	K2	моно	
12	Z82.5-c	Z82	-	R7.1	A	K2	моно	
13	Z82.5-c	Z82	-	R7.2	A	K2	моно	
14	Z82.5-c	Z82	-	R7.3	A	K2	моно	
15	Z82.5-c	Z82	-	R7.4	A	K2	моно	

Рис.4.41b. Фрагмент экрана при просмотре таблицы F6^c

F1 | F1c | F2 | F2c | F3c | F3 | F4 | F4c | F6 | F6c

#	Кодструктуры	КодПК	НаименованиеПК	КлассПК	ТипПК	СтатусПК	ОценкаПК	ПримечаниекПК
1	Z8.11.1-c	R3.1	количество передач	R	T	T	P	-
2	Z8.11.1-c	R1.2	код CE	R	T	T	P	-
3	Z8.11.1-c	R4.2	тип нагрузки	R	T	T	P	-
4	Z8.11.1-c	R6.1	полное число часов работы передачи за срок службы	R	T	T	P	-
5	Z8.11.1-c	R6.5	эквивалентное число циклов перемены напряжения при изгибе	R	T	T	P	-
6	Z8.11.1-c	R1.3	код детали	R	T	T	P	-
7	Z8.11.1-c	R8.1	частота вращения	R	T	T	P	-
8	Z82.1-c	R3.1	количество передач	R	T	T	P	-
9	Z82.1-c	R1.2	код CE	R	T	T	P	-
10	Z82.1-c	R6.2	базовое число циклов перемены напряжения при контакте	R	T	T	P	-
11	Z82.1-c	R1.3	код детали	R	T	T	P	-
12	Z82.1-c	R7.1	твердость поверхности зубьев	R	T	T	P	-
13	Z82.1-c	R7.2	материал	R	T	T	P	-
14	Z82.1-c	R7.3	марка материала	R	T	T	P	-
15	Z82.1-c	R7.4	вид термообработки	R	T	T	P	-
16	Z7.12.1-c	R3.1	количество передач	R	T	T	P	-
17	Z7.12.1-c	R1.3	код детали	R	T	T	P	-
18	Z7.12.1-c	R9.3	допускаемое напряже-ние при расчете на изгибную выносливость, соответствующее базовому числу циклов перемены напряжения	R	T	T	P	-

Рис.4.41c. Фрагмент экрана при просмотре таблицы F1^c

#	Кодструктуры	КлассСвязиПК	КодПК_1	КодПК_2	КодПК_3	Наименование СвязиПК	ТипСвязиПК	ОценкаСвязиПК	КодСвязиПК
1	Z8.11.1-c	OR	O7.1	R3.1	-	Состав Узла	C	1	O7.1R3.1-
2	Z8.11.1-c	OR	O7.2	R1.2	-	Состав CE	C	1	O7.2R1.2-
3	Z8.11.1-c	OR	O7.2	R4.2	-	Состав CE	C	1	O7.2R4.2-
4	Z8.11.1-c	OR	O7.2	R6.1	-	Состав CE	C	1	O7.2R6.1-
5	Z8.11.1-c	OR	O7.2	R6.5	-	Состав CE	C	1	O7.2R6.5-
6	Z8.11.1-c	OR	O7.4	R1.3	-	Состав Детали	C	1	O7.4R1.3-
7	Z8.11.1-c	OR	O7.4	R8.1	-	Состав Детали	C	1	O7.4R8.1-
8	Z82.1-c	OR	O7.1	R3.1	-	Состав Узла	C	1	O7.1R3.1-
9	Z82.1-c	OR	O7.2	R1.2	-	Состав CE	C	1	O7.2R1.2-
10	Z82.1-c	OR	O7.2	R4.2	-	Состав CE	C	1	O7.2R4.2-
11	Z82.1-c	OR	O7.2	R6.2	-	Состав CE	C	1	O7.2R6.2-
12	Z82.1-c	OR	O7.4	R1.3	-	Состав Детали	C	1	O7.4R1.3-
13	Z82.1-c	OR	O7.4	R7.1	-	Состав Детали	C	1	O7.4R7.1-
14	Z82.1-c	OR	O7.4	R7.2	-	Состав Детали	C	1	O7.4R7.2-
15	Z82.1-c	OR	O7.4	R7.3	-	Состав Детали	C	1	O7.4R7.3-
16	Z82.1-c	OR	O7.4	R7.4	-	Состав Детали	C	1	O7.4R7.4-
17	Z7.12.1-c	OR	O7.1	R3.1	-	Состав Узла	C	1	O7.1R3.1-
18	Z7.12.1-c	OR	O7.4	R1.3	-	Состав Детали	C	1	O7.4R1.3-
19	Z7.12.1-c	OR	O7.4	R9.3	-	Состав Детали	C	1	O7.4R9.3-
20	Z7.11.1-c	OR	O7.1	R3.1	-	Состав Узла	C	1	O7.1R3.1-
21	Z7.11.1-c	OR	O7.2	R1.2	-	Состав CE	C	1	O7.2R1.2-

Рис.4.41d. Фрагмент экрана при просмотре таблицы F2^c

4.5 Выводы по главе 4

1. Разработанный программный комплекс обеспечивает поддержку обработки статистических предметных ограничений, т.е. выполняется расслоение концептуальной модели под постоянную информацию и ее систематизация, при концептуальном моделировании прикладных задач.

2. Комплекс программного обеспечения может использоваться специальлистами-разработчиками автоматизированных систем на этапе моделирования предметных знаний задач, подлежащих автоматизации.

ЗАКЛЮЧЕНИЕ

Выполненные исследования и практическая работа позволили получить следующие выводы и результаты:

1. Диссертация представляет научно-квалификационную работу, в которой содержится решение задачи, заключающейся в повышении эффективности процесса создание прикладных автоматизированных систем и комплексов за счет разработки метода обработки концептуальных моделей предметных задач в рамках методологии автоматизации интеллектуального труда.

2. Анализ существующих методов, подходов и инструментальных средств их поддержки позволил установить, что по сравнению с методами других подходов, методология автоматизации интеллектуального труда имеет ряд преимуществ: полный набор компонентов модели и их увязка в модельном представлении; наличие закономерностей организации семантических модельных представлений в УКП; формирование КППЗ, как основы для единой интерпретации последующих формально-языковых представлений при создании прикладных автоматизированных систем.

3. Установлены связи между характеристиками моделей таблично оформленных предметных знаний и их аналитической интерпретацией.

4. Выполнено уточнение формального описания концептуального представления предметных задач, позволяющего разделить концептуальное представление на модели 1-го и 2-го родов.

5. Разработан метод обработки статистических предметных ограничений 1-го рода при концептуальном моделировании прикладных задач, включающий формальное описание концептуальных моделей 1-го рода на объектном и конкретном уровнях абстрагирования и их взаимосвязи; формальное описание расслоения концептуальных моделей 1-го рода под переменную и постоянную информацию; формальное описание типологии статистических предметных зависимостей 1-го рода и их геометрическую интерпретацию.

6. Разработанный метод позволяет систематизировать и классифицировать таблично оформленные знания, зафиксированные в технических нормативно-справочных документах машиностроительных организаций.

7. На основе формального описания концептуальных моделей 1-го рода и их конструкций была разработано методическое обеспечение в виде совокупности начальных моделей, обеспечивающее детальную проработку алгоритмов выполнения процедур.

8. На основе методического обеспечения были разработаны программные модули для процедур обработки концептуальных моделей при создании прикладных автоматизированных систем, которые расширяют возможности инструментальных средств поддержки методологии автоматизации интеллектуального труда.

9. Методическое обеспечение и программные средства были использованы при подготовке специалистов по направлению «Информатика и вычислительная техника».

СПИСОК ЛИТЕРАТУРЫ

1. Волкова, Г. Д. Методология автоматизации интеллектуального труда / Г. Д. Волкова // М.: Янус-К, 2013. - 104 с.
2. Волкова, Г.Д. Методология автоматизации проектно-конструкторской деятельности в машиностроении / Г.Д. Волкова // Учебное пособие. - М.: МГТУ «Станкин», 2000, 81с.
3. Волкова, Г. Д. Концептуальное моделирование проектных задач: учеб. Пособие / Г. Д. Волкова // М.: ФГБОУ ВО «МГТУ «СТАНКИН», 2016. -117 с.: цл.
4. Волкова, Г. Д. Методология автоматизации интеллектуального труда: предпосылки, основания, результаты, проблемы. Часть 1/ Г.Д. Волкова // Межотраслевая информационная служба. – 2009. – № 2. – С. 9-20.
5. Волкова, Г. Д. Методология автоматизации интеллектуального труда: предпосылки, основания, результаты, проблемы. Часть 3. (Части 1 и 2 см. № 2 и 3 за 2009 г.) / Г. Д. Волкова // Межотраслевая информационная служба. – 2009. – № 4. – С. 15-31.
6. Волкова, Г.Д. Методология автоматизации интеллектуального труда: Предпосылки, основания, результаты, проблемы. Часть 4/ Г.Д. Волкова // Межотраслевая информационная служба. 2009. № 4(149). С. 15–30.
7. Meder, N. Artifical Intelligence as a Tool of Classification, or: The Network of Language Games as Cognitive Paradigm /Inf.Classif.12(1985),N3,p. 128-132.
8. Лукашевич, Н.В. Тезаурусы в задачах информационного поиска/ Н.В. Лукашевич/ / М.: Издво МГУ, 2011. – 495 с.
9. Кара-Ушанов, В. Ю. Модель «Сущность—Связь»/ В. Ю. Кара-Ушанов: учеб. пособие //Екб.: Электронное текстовое издание, 2017. 64 с.
10. Пинягина, О. В. Лекции «Базы Данных»: ER-модель (entity–relationship) / О. В. Пинягина // [Электронный ресурс]: – Режим доступа: http://kek.ksu.ru/EOS/BD/ER_model.html/ (дата обращения 12.10.2019).

11. Волкова, Г. Д. Сравнительный анализ методов семантического моделирования: ERA-модель Чена и метод концептуального моделирования в методологии автоматизации интеллектуального труда (МАИТ) / Г.Д. Волкова, Н. Х. Ньи // Современная наука: актуальные проблемы теории и практики. Серия: Естественные и Технические Науки. -2020. -№04. -С. 52-54.
12. Методология IDEF1X [Электронный ресурс]: – Режим доступа: https://en.wikipedia.org/wiki/IDEF1X (дата обращения 1.03.2019).
13. Основы методологии IDEF1X [Электронный ресурс]: – Режим доступа: https://www.cfin.ru/vernikov/idef/idef1x.shtml (дата обращения 12.09.2022).
14. Марка, Д. Методология структурного анализа и проектирования SADT/ Д. Марка , К. МакГоуэн // М.: Метатехнология, 1993, 240с
15. Черемных, С.В. Структурный анализ систем. IDEF-технологии / С.В. Черемных, И.О. Семенов, В.С. Ручкин // М. Финансы и статистика, 2001, 208 с.
16. Черемных, С.В. Моделирование и анализ систем. IDEF-технологии: практикум / С.В. Черемных, И.О. Семенов, В.С. Ручкин. -М.: Финансы и статистика, 2006. - 192 с.
17. Нотация IDEF0 [Электронный ресурс]: – Режим доступа: https://www.businessstudio.ru/wiki/docs/current/doku.php/ru/csdesign/bpmodeling/idef0 (дата обращения 12.04.2022).
18. Универсальная функциональная модель торгового предприятия в нотации IDEF0 [Электронный ресурс]: – Режим доступа: https://trinion.org/blog/universalnaya-funkcionalnaya-model-torgovogo-predpriyatiya-v-notacii-idef0 (дата обращения 12.04.2022).
19. Data Flow Diagrams [Электронный ресурс]: – Режим доступа: https:// www.lucidchart.com/pages/data-flowdiagram#:~:text=A%20data%20flow %20diagram%20(DFD,the%20routes%20between%20each%20destination (дата обращения 12.09.2020).

20. Понятие и сущность бизнес – процессов в организации [Электронный ресурс]: – Режим доступа: https://www.evkova.org/kursovye-raboty/ponyatie-i-suschnost-biznes---protsessov-v-organizatsii (дата обращения 12.09.2020).
21. IDEF3 [Электронный ресурс]: – Режим доступа: https://www.conceptdraw.com/How-To-Guide/idef3 (дата обращения 12.04.2023).
22. Разработка САПР. Подходы и методы, используемые при автоматизации интеллектуальных и информационных процессов. Основы представления информации и знании о производственных процессах в технике. [Электронный ресурс]: – Режим доступа: https://vunivere.ru/work90038/page6 (дата обращения 12.04.2023).
23. Корнеев, В.В. Базы данных: интеллектуальная обработка информации / В.В. Корнеев, А.Ф. Гареев, С. В. Васютин, В.В. Райх // М.: Нолидж, 2000. 352 с.
24. Норман, Д. Семантические сети. / Д. Норман // Психология памяти. - М.: Че Ро. 2000, С.350-356.
25. Соломатин, Н.М. Информационно-семантические системы/ Н.М. Соломатин // М.: Высшая школа, 1989, 127с.
26. Минский, М. Фреймы для представления знаний / М. Минский // М.: Энергия, 1979, 152с.
27. Prolog [Электронный ресурс]: – Режим доступа: https://en.wikipedia.org/wiki/Prolog (дата обращения 01.04.2021).
28. Гаскаров, Д.В. Интеллектуальные информационные системы / Д.В. Гаскаров // М.: «Высшая школа», 2003.
29. Гладун, А.Я. Онтологии в корпоративных системах / А.Я. Гладун // Корпоративные системы: журн. М.: Комиздат, 2006. - С.13-26.
30. Описание метод Онтология. [Электронный ресурс]: – Режим доступа: https://ru.wikipedia.org/wiki/Онтология_(информатика) (дата обращения 18.03.2021).
31. Кузнецов, О.П., Суховеров В.С. Онтологический подход к оценке тематики научного текста // Онтология проектирования.- 2016.- Т.6, №1(19). – С.55-66.

32. Сидорова, Е.А. Подход к моделированию процесса извлечения информации из текста на основе онтологий // Онтология проектирования.- 2018.- Т.8, №1(27). – С.134-151.
33. Бабанов, А.М. Два современных подхода к семантическому моделированию – ORM и ERMM // Вестник Томского Государственного университета. – 2014.- №3(28).- С.46-56.
34. Object Oriented Approach [Электронный ресурс]: – Режим доступа: https://www.tutorialspoint.com/system_analysis_and_design/system_analysis_and_design_object_oriented_approach.htm (дата обращения 18.03.2021).
35. Фаулер, М. UML. Основы. Краткое руководство по стандартному языку объектного моделирования / М. Фаулер. - М.: Символ-Плюс, 2011. - 192 с.
36. Унифицированный язык моделирования (UML) [Электронный ресурс]: – Режим доступа: https://www.cybermedian.com/ru/unified-modeling-language-uml-introduction/ (дата обращения 6.03.2023).
37. Грэхем, И. Объектно-ориентированные методы. Принципы и практика / Иан Грэхем. -М.: Вильяме, 2004. - 880 с.
38. Фридман, А.Л. Основы объектно-ориентированной разработки программных систем / А. Л. Фридман. - М.: Финансы и статистика, 2000. - 192 с.
39. Симаков, К.В. Модели и методы извлечения знаний из текстов на естественном языке/ Диссертации на соискание ученой степени кандидата технических наук. Специальность 05.13.17. Москва. 2008.
40. Кузин, Е.С. Информационные технологии и проектирование прикладных программных систем // Информационные технологии и вычислительные системы РАН № 3, 1996 г.
41. Кузин, Е.С. Концепции информационной технологии функционально-ориентированного проектирования прикладных информационных систем. // Информационные технологии № 1, 2000 г.
42. Вагин, В.Н., Михайлов И.С. Обеспечение интероперабельности информационных систем на основе подхода метамоделирования и онтологии предметной области // Труды Второй Международной конференции

«Системный анализ и информационные технологии» -САИТ-2007, Обнинск, Россия. С.149-152.

43. Грэхем, И. Объектно-ориентированные методы. Принципы и практика. М.: Вильяме, 2004. - 880 с.
44. Леоненков, А. В. Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose. М.: Бином. Лаборатория знаний, 2006. - 320 с.
45. Волкова, Г. Д. Философские аспекты моделирования конструкторско-технологических знаний при создании систем автоматизации проектирования в машиностроении // Вестник МГТУ Станкин. – 2012. – № 1 (19). – С. 141-144.
46. Volkova, G.D. Cognitive Technologies for the Creation of CAD- System/ G.D. Volkova// Proceedings of the 13th IFAC Symposium on Information Control Problems in Manufacturing. –M.: june 3-5,2009, pp. 1881-1886.
47. Новоселова, О.В. Исследование технологий и инструментальных средств создания автоматизированных систем различного назначения/ О.В. Новоселова, Г.Д. Волкова, О.Г. Григорьев // журнал Технические науки - от теории к практике. 2015. №53: С. 52-62.
48. Сирота, И. М. Разработка методов и средств поддержки визуального концептуального моделирования проектно-конструкторских задач при создании сапр машиностроительного назначения: дис…канд. техн. наук: 05.13.12 / Сирота Илья Михайлович. – М.:1999. – 200 с.
49. Соколов, В.П. Инструментальные системы создания средств обеспечения САПР-конструкторского и технологического назначения/ В.П. Соколов // Межотраслевой научно-технический сборник «Техника. Экономика», серия автоматизация проектирования. - М.: ВИМИ, 1994, вып.4, с.39-48.
50. Бычкова, Н. А. Разработка метода и инструментальных средств визуального моделирования и документирования системы знаний предметных задач нри проектировании санр машиностроительного назначения: дис…канд. техн. наук: 05.13.12 / Бычкова Наталья Александровна. – М.:2006. – 274 с.

51. Калянов, Г.Н. Российский рьшок CASE-средств / Г.Н. Калянов // PCWeek/Russian Edition, 1998, июнь.
52. Макаров, Н. С. UML: поддержка проектирования и инструментальные среды / Н. С. Макаров // Прикладная информатика. – 2007. – № 2(8). – С. 66-82.
53. Вендров, А.М. CASE-технологии. Современные методы и средства проектирования информационных систем / А.М. Вендров // 2008, ноября.
54. Маклаков, С. В . BPwin и ERwin. CASE-средства разработки информационных систем / С. В. Маклаков // 2-е изд. - М.: Диалог-МИФИ, 2002, 304с.
55. Описание продукта Silverrun (Grandite) [Электронный ресурс]: – Режим доступа: http://www.silverrun.com (дата обращения 18.03.2021).
56. Панащук, С.А. Разработка информационных систем с использованием CASE-системы Silverrun. / С.А. Панащук // «СУБД», 1995, №3, С.41-48.
57. Вендров, А.М. CASE-технологии. Современные методы и средства проектирования информационных систем. Глава 5.1.1. Silverrun. // Обзор. - [Электронный ресурс]: – Режим доступа: http://citforum.ru/database/case/glava5_1_1.shtml (дата обращения 23.04.2021).
58. Маклаков, С. В. Моделирование бизнес-процессов с AllFusion Process Modeler / С. В. Маклаков // М.: Диалог-МИФИ, 2003, 236с.
59. Кватрани, Т. Rational Rose 2000 и U M L. Визуальное моделирование. / Т. Кватрани// М.: ДМК, 2009, 176с.
60. Трофимов, С.А. CASE-технологии: практическая работа в Rational Rose / С.А. Трофимов // М.: Бином, 2002, 288с.
61. Леоненков, А. В. Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose / А.В. Леоненков. — М.: Бином. Лаборатория знаний, 2006. - 320 с.
62. Описание продукта CASE Аналитик // Сайт stu24. - [Электронный ресурс]: – Режим доступа: https://www.stud24.ru/mathematic-modelling/naznachenie-i-prijomy-raboty-s/207214-606173-page1.html (дата обращения 20.04.2023).
63. Sybase Power Designer. [Электронный ресурс]: – Режим доступа: https://habr.com/ru/articles/15761/ (дата обращения 18.03.2022).

64. Благодаров, А.В., Тишкин Р.В. Автоматизация проектирования баз данных в среде Sybase PowerDesigner / А.В. Благодаров, Р.В. Тишкин // Методические указания к лабораторным работам. Рязан. гос. радиотехн. Универ.;Сост., Рязань, 2010. 35 с.
65. Колетски, П. Oracle Desginer.Настольная книга пользователя/ П. Дорси, П. Колетски // М.: Лори, 1999, 592с.
66. Браун, Б. Oracle 8i. Создание Web-приложений/ Б. Браун // М.: Лори, 2001, 768с.
67. Oracle Corporation. [Электронный ресурс]: – Режим доступа: http://www.oracle.com/ru/ (дата обращения 18.03.2022).
68. Горчинская, О.Ю. Designer/2000 - новое поколение CASE-продуктов фирмы Oracle/ О.Ю. Горчинская // «СУБД», 1995, №3, С.9-25.
69. Описание Microsoft Visio [Электронный ресурс]: – Режим доступа: https://en.wikipedia.org/wiki/Microsoft_Visio (дата обращения 18.03.2023).
70. Описание Microsoft Visio [Электронный ресурс]: – Режим доступа: https://studfile.net/preview/358494/page:4/ (дата обращения 18.03.2023).
71. Волкова, Г.Д. Анализ автоматизации интеллектуального труда / Г.Д. Волкова // «Информатика-машиностроение». - М.: НТП «Вираж-центр», 1999, №5-6, С.52-55.
72. Тюрбеева, Т. Б. Разработка метода моделирования процессов жизненного цикла прикладных автоматизированных систем, обеспечивающего формирование нормативно-методической среды их поддержки: дис…канд. техн. наук: 05.13.01 / Тюрбеева, Татьяна Борисовна. – М.:2015. – 230 с.
73. Тюрбеева, Т.Б. моделирование процессов создания, функционирования и развития прикладных автоматизированных систем на базе методологии автоматизации интеллектуального труда / Т.Б. Тюрбеева, Г.Д. Волкова, О.Г. Григорьев // Известия высших учебных заведений. Поволжский регион. Технические науки. 2013. № 4 (28). С. 189-198.
74. Хтве, Н. Н. Разработка формального описания типологии статистических предметных ограничений при концептуальном моделировании прикладных

задач / Н. Н. Хтве, Г. Д. Волкова, // Вестник МГТУ «Станкин», 2021. - №2(57) – С. 13-19.

75. Хтве, Н. Н. Формальное описание классификации и обработки концептуальной модели объектного уровня для предметной задачи / Н. Н. Хтве, Г. Д. Волкова, Т. Б. Тюрбеева // Цифровая экономика: технологии, управление, человеческий капитал: Материалы III всероссийской научно-практической конференции, Вологда, 25 сентября 2020 года. – Вологда: Общество с ограниченной ответственностью "Маркер", 2020. – С. 73-78.
76. Ньи, Н. Х. Формальное описание процедуры обработки концептуальной модели / Н. Х. Ньи, Г. Д. Волкова // Машиностроение: традиции и инновации (МТИ - 2020): Материалы XIII всероссийской конференции с международным участием, Москва, 26 октября – 06 2020 года. – Москва: Московский государственный технологический университет "СТАНКИН", 2020. – С. 298-305.
77. Волкова, Г. Д. Формальное описание классификации и обработки концептуальной модели конкретного уровня для предметной задачи / Г. Д. Волкова, Н. Н. Хтве // XXVIII конференция «Математика. Компьютер. Образование – 2021» МКО-2021. – 212 С. 49.
78. Htwe, N. N. Processing of Conceptual Models of Subject Problems for Extracting Knowledge from Technical Documentation / N. N. Htwe, G. D. Volkova, K. K. Zan., T. B. Tyurbeeva // Proceedings of the 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering, ElConRus 2021. – p. 2185-2189.
79. Ньи Ньи Хтве. Разработка формального аппарата для описания и обработки концептуальных моделей / Ньи Ньи Хтве, Г. Д. Волкова, О.Г. Григорьев // Труды Института системного анализа Российской академии наук. – 2023. – Т. 73, № 3. – С. 40-51.
80. Новоселова, О.В. Моделирование предметных задач на начальных этапах автоматизации проектной деятельности / О.В. Новоселова // учебное пособие. - М.: ФГБОУ ВО «МГТУ «СТАНКИН». 2016 г. 100 с.

81. Щукин, М. В. Разработка метода и средств поддержки аналитической обработки, визуализации и документирования концептуальных представлений при проектировании САПР машиностроительного назначения: дис…канд. техн. наук: 05.13.12 / Щукин максим владимирович.- М.:2003.-348с.
82. Тюрбеева, Т. Б. Концептуальное моделирование процессов конструирования сложных машиностроительных изделий / Т. Б. Тюрбеева, Г.Д. Волкова, Е.Г. Семячкова // Автоматизация и управление в машиностроении. – 2018. – № 2(31). – С. 11-22.
83. Качала В.В. Основы теории систем и системного анализа / В.В. Качала // Учебное пособие для вузов. – 2-е изд., испр. – М.: Горячая линия –Телеком, 2012. – 210 с.
84. Волкова В.Н. Системный анализ и принятие решений: Словарь-справочник / Под ред. В.Н. Волкова, В.Н. Козлова // Учебное пособие для вузов – М.: Высш.шк., 2004 – 616 с.
85. MySQL [Электронный ресурс]: – Режим доступа: https://www.w3schools.com/mySQl/default.asp (дата обращения 12.09.2019).
86. HTML [Электронный ресурс]: – Режим доступа: https://www.w3schools.com/html/default.asp (дата обращения 14.01.2020).
87. CSS [Электронный ресурс]: – Режим доступа: https://www.w3schools.com/ css/default.asp (дата обращения 07.07.2020).
88. JAVASCRIPT [Электронный ресурс]: – Режим доступа: https://www.w3schools.com/ js/default.asp (дата обращения 18.01.2021).
89. Macintyre, P.B. PHP: The Good Parts: Delivering the Best of PHP/ P.B. Macintyre // USA.: «First Edition», 2010, pp176.
90. PHP [Электронный ресурс]: – Режим доступа: https://www.w3schools.com/php/ (дата обращения 12.07.2021).
91. Laravel Framework [Электронный ресурс]: – Режим доступа: https://docs.google.com/viewer?url=https://www.computerpdf.com/pdf/0915-learning-laravel.pdf (дата обращения 18.9.2021).

ПРИЛОЖЕНИЕ А

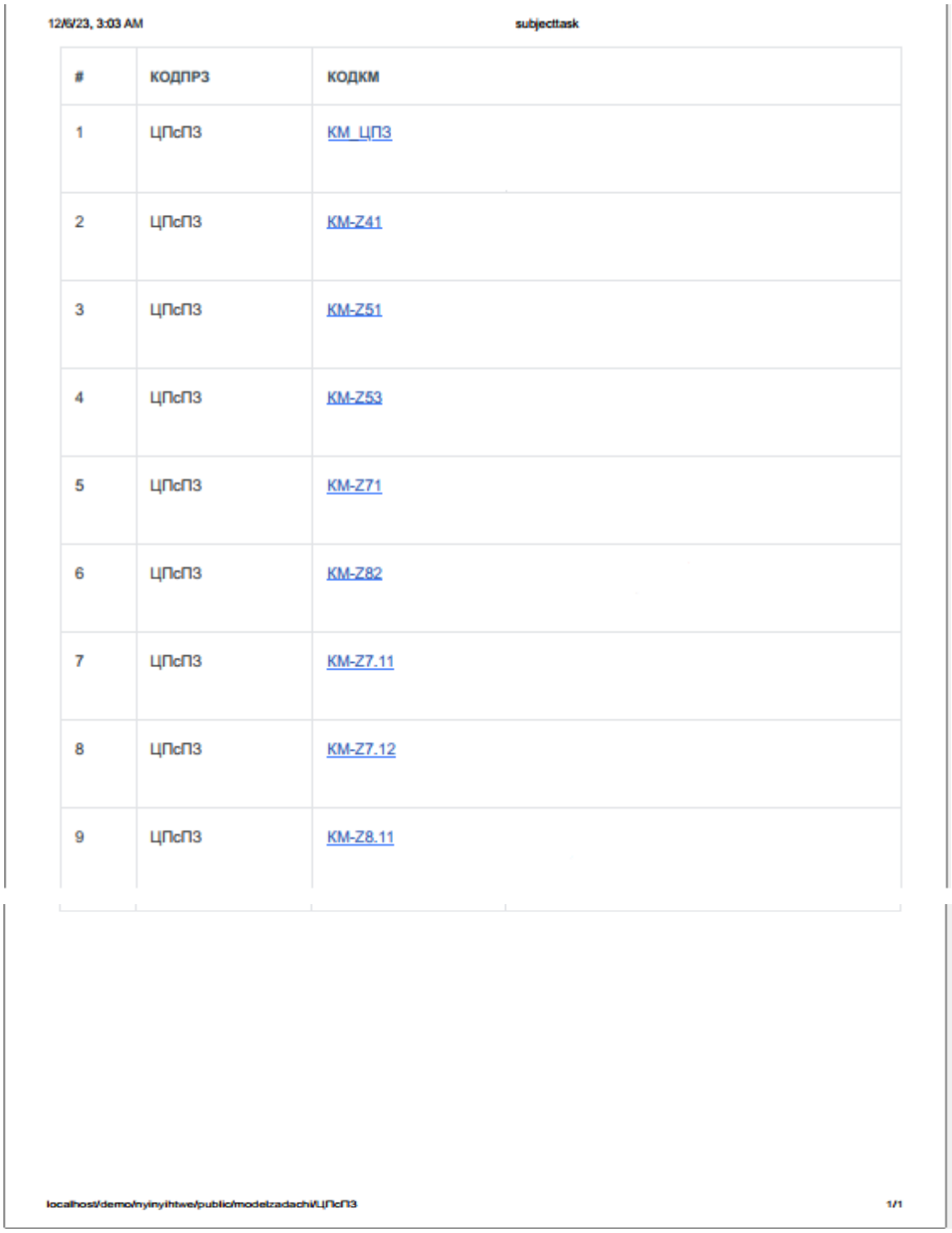
В данном приложении приводятся программные формы инструментального средства поддержки дополнительной обработки концептуальных моделей для тестового примера «исходных данных»:

1. Фрагмент экрана для формы «Предметная задача»…… 230
2. Фрагмент экрана для формы «Модели задачи»…… 230
3. Фрагмент экрана для формы «Концептуальные модели»…… 231
4. Фрагмент экрана для формы «Состав концептуальных моделей»…… 231
5. Фрагмент экрана для формы «Структуры концептуальных моделей»……. 232
6. Фрагмент экрана для формы «Элемент статистических структуры F1»…….. 233
7. Фрагмент экрана для формы «Бинарные связи статистических структуры F2»…..... 234
8. Фрагмент экрана для формы «Элемент динамических структуры F3»…….. …… 235
9. Фрагмент экрана для формы «Бинарные связи динамических структуры F4»…..... 236
10. Фрагмент экрана для формы «Структуры увязки F6»……... 237

11/29/23, 9:10 PM subjecttask

#	КОДПРЗ	НАИМЕНОВАНИЕПРЗ	КОЛИЧЕСТВОСТАТКМВПРЗ
1	ЦПсПЗ	Проектировочный расчет цилиндрических передач с прямыми зубьями	8

Рис А1. Фрагмент экрана для формы «Предметная задача»



12/6/23, 3:03 AM subjecttask

#	КОДПРЗ	КОДКМ
1	ЦПсПЗ	КМ_ЦПЗ
2	ЦПсПЗ	КМ-Z41
3	ЦПсПЗ	КМ-Z51
4	ЦПсПЗ	КМ-Z53
5	ЦПсПЗ	КМ-Z71
6	ЦПсПЗ	КМ-Z82
7	ЦПсПЗ	КМ-Z7.11
8	ЦПсПЗ	КМ-Z7.12
9	ЦПсПЗ	КМ-Z8.11

localhost/demo/myinyihtwe/public/modelzadachi/ЦПсПЗ 1/1

Рис А2. Фрагмент экрана для формы «Модели задачи»

12/4/23, 9:49 PM КМ

#	КОДКМ	РОДКМ	ВИДКМС
1	КМ_ЦПЗ	1	v
2	КМ-Z8.11	1	c
3	КМ-Z82	1	c
4	КМ-Z7.12	1	c
5	КМ-Z7.11	1	c
6	КМ-Z71	1	c
7	КМ-Z53	1	c
8	КМ-Z51	1	c
9	КМ-Z41	1	c

Рис А3. Фрагмент экрана для формы «Концептуальные модели»

12/4/23, 9:50 PM compositionkmst

#	КОДКМ	КОДСТАТСТРУКТУРЫ	КОДДИНСТРУКТУРЫ	КОДСТРУКТУРЫУВЯЗКИ
1	КМ_ЦПЗ	ZPsPS.1-v	ZPsPS.3-v	ZPsPS.5-v
2	КМ-Z8.11	Z8.11.1-c	Z8.11.3-c	Z8.11.5-c
3	КМ-Z82	Z82.1-c	Z82.3-c	Z82.5-c
4	КМ-Z7.12	Z7.12.1-c	Z7.12.3-c	Z7.12.5-c
5	КМ-Z7.11	Z7.11.1-c	Z7.11.3-c	Z7.11.5-c
6	КМ-Z71	Z71.1-c	Z71.3-c	Z71.5-c
7	КМ-Z53	Z53.1-c	Z53.3-c	Z53.5-c
8	КМ-Z51	Z51.1-c	Z51.3-c	Z51.5-c
9	КМ-Z41	Z41.1-c	Z41.3-c	Z41.5-c

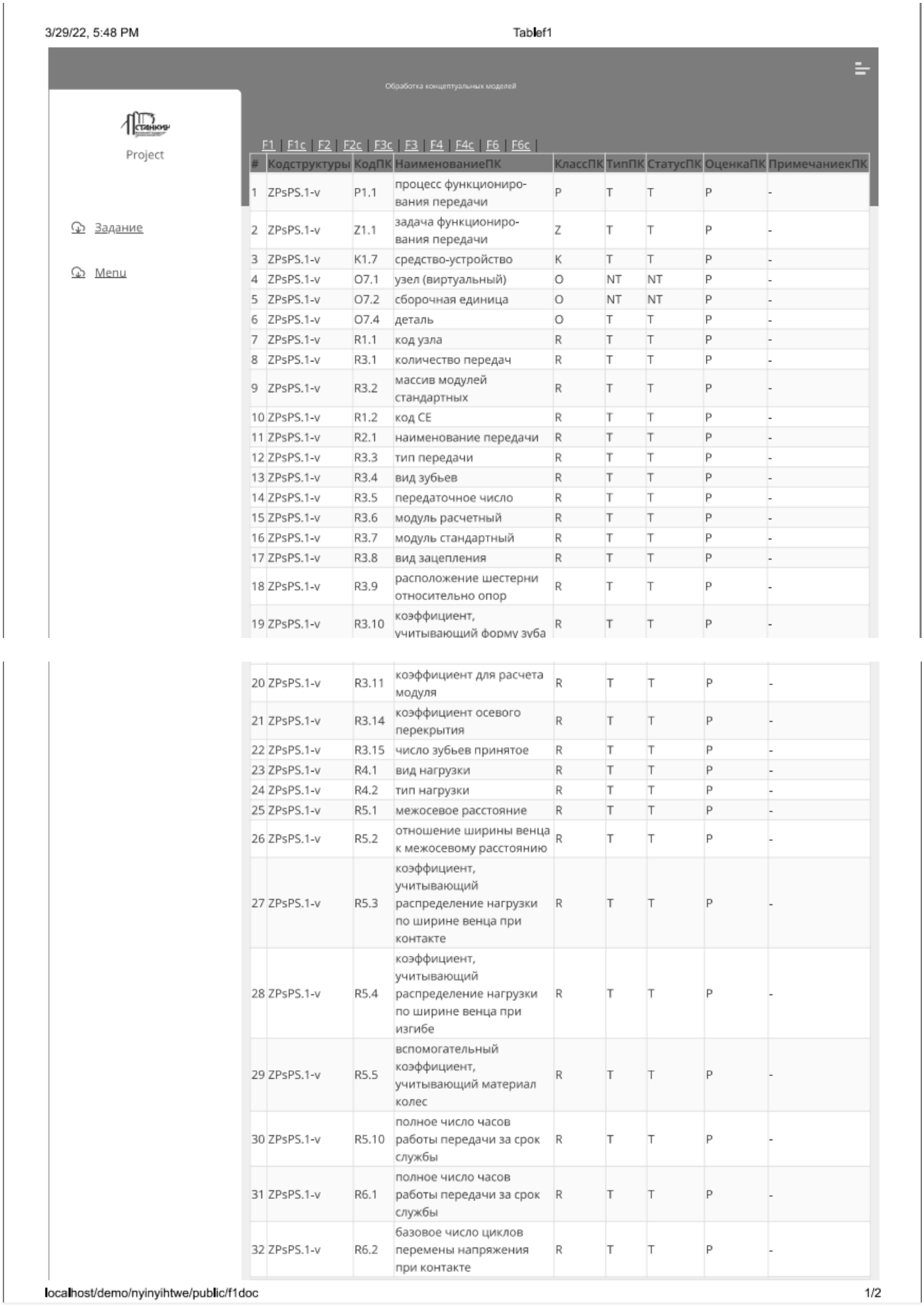
Рис А4. Фрагмент экрана для формы «Состав концептуальных моделей»

12/4/23, 9:51 PM structure

#	КОДСТРУКТУРЫ	ТИПСТРУКТУРЫ	РОДСТРУКТУРЫ	ВИДСТРУКТУРЫ	КОЛИЧЕСТВОЭЛСТРУКТУРЫ
1	ZPsPS.1-v	стат	1	v	60
2	ZPsPS.3-v	дин	1	v	59
3	ZPsPS.5-v	увязки	1	v	274
4	Z8.11.1-c	стат	1	c	7
5	Z8.11.3-c	дин	1	c	1
6	Z8.11.5-c	увязки	1	c	7
7	Z82.1-c	стат	1	c	8
8	Z82.3-c	дин	1	c	1
9	Z82.5-c	увязки	1	c	8
10	Z7.12.1-c	стат	1	c	3
11	Z7.12.3-c	дин	1	c	1
12	Z7.12.5-c	увязки	1	c	3
13	Z7.11.1-c	стат	1	c	6
14	Z7.11.3-c	дин	1	c	1
15	Z7.11.5-c	увязки	1	c	6
16	Z71.1-c	стат	1	c	9
17	Z71.3-c	дин	1	c	1
18	Z71.5-c	увязки	1	c	9
19	Z53.1-c	стат	1	c	5
20	Z53.3-c	дин	1	c	1
21	Z53.5-c	увязки	1	c	5

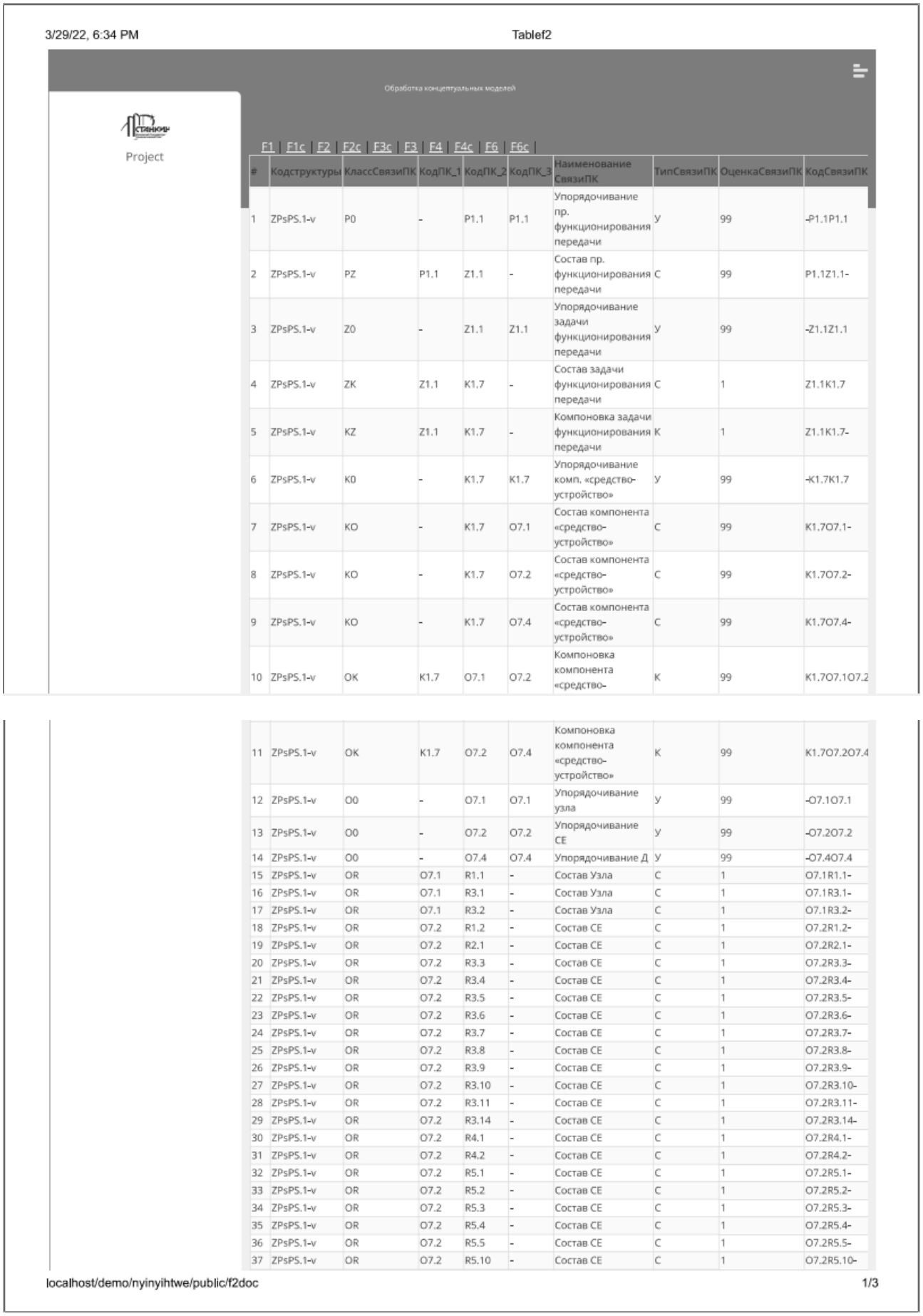
localhost/demo/nyinyihtwe/public/structure/ZPsPS.1-v 1/2

Рис А5. Фрагмент экрана для формы «Структуры концептуальных моделей»



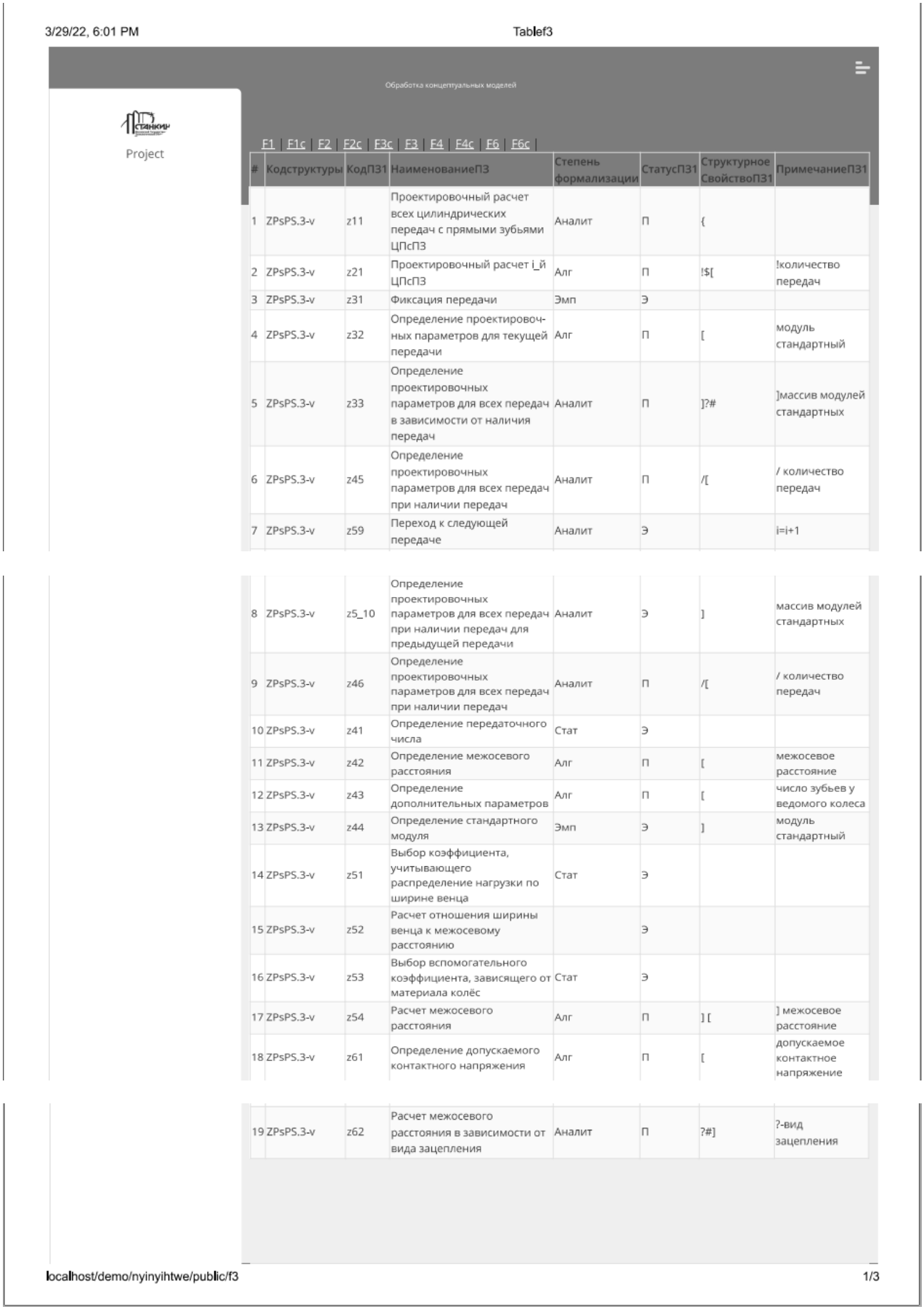
#	Кодструктуры	КодПК	НаименованиеПК	КлассПК	ТипПК	СтатусПК	ОценкаПК	ПримечаниекПК
1	ZPsPS.1-v	P1.1	процесс функциониро-вания передачи	P	T	T	P	-
2	ZPsPS.1-v	Z1.1	задача функциониро-вания передачи	Z	T	T	P	-
3	ZPsPS.1-v	K1.7	средство-устройство	K	T	T	P	-
4	ZPsPS.1-v	O7.1	узел (виртуальный)	O	NT	NT	P	-
5	ZPsPS.1-v	O7.2	сборочная единица	O	NT	NT	P	-
6	ZPsPS.1-v	O7.4	деталь	O	T	T	P	-
7	ZPsPS.1-v	R1.1	код узла	R	T	T	P	-
8	ZPsPS.1-v	R3.1	количество передач	R	T	T	P	-
9	ZPsPS.1-v	R3.2	массив модулей стандартных	R	T	T	P	-
10	ZPsPS.1-v	R1.2	код CE	R	T	T	P	-
11	ZPsPS.1-v	R2.1	наименование передачи	R	T	T	P	-
12	ZPsPS.1-v	R3.3	тип передачи	R	T	T	P	-
13	ZPsPS.1-v	R3.4	вид зубьев	R	T	T	P	-
14	ZPsPS.1-v	R3.5	передаточное число	R	T	T	P	-
15	ZPsPS.1-v	R3.6	модуль расчетный	R	T	T	P	-
16	ZPsPS.1-v	R3.7	модуль стандартный	R	T	T	P	-
17	ZPsPS.1-v	R3.8	вид зацепления	R	T	T	P	-
18	ZPsPS.1-v	R3.9	расположение шестерни относительно опор	R	T	T	P	-
19	ZPsPS.1-v	R3.10	коэффициент, учитывающий форму зуба	R	T	T	P	-
20	ZPsPS.1-v	R3.11	коэффициент для расчета модуля	R	T	T	P	-
21	ZPsPS.1-v	R3.14	коэффициент осевого перекрытия	R	T	T	P	-
22	ZPsPS.1-v	R3.15	число зубьев принятое	R	T	T	P	-
23	ZPsPS.1-v	R4.1	вид нагрузки	R	T	T	P	-
24	ZPsPS.1-v	R4.2	тип нагрузки	R	T	T	P	-
25	ZPsPS.1-v	R5.1	межосевое расстояние	R	T	T	P	-
26	ZPsPS.1-v	R5.2	отношение ширины венца к межосевому расстоянию	R	T	T	P	-
27	ZPsPS.1-v	R5.3	коэффициент, учитывающий распределение нагрузки по ширине венца при контакте	R	T	T	P	-
28	ZPsPS.1-v	R5.4	коэффициент, учитывающий распределение нагрузки по ширине венца при изгибе	R	T	T	P	-
29	ZPsPS.1-v	R5.5	вспомогательный коэффициент, учитывающий материал колес	R	T	T	P	-
30	ZPsPS.1-v	R5.10	полное число часов работы передачи за срок службы	R	T	T	P	-
31	ZPsPS.1-v	R6.1	полное число часов работы передачи за срок службы	R	T	T	P	-
32	ZPsPS.1-v	R6.2	базовое число циклов перемены напряжения при контакте	R	T	T	P	-

Рис А6. Фрагмент экрана для формы «Элемент статистических структуры F1»



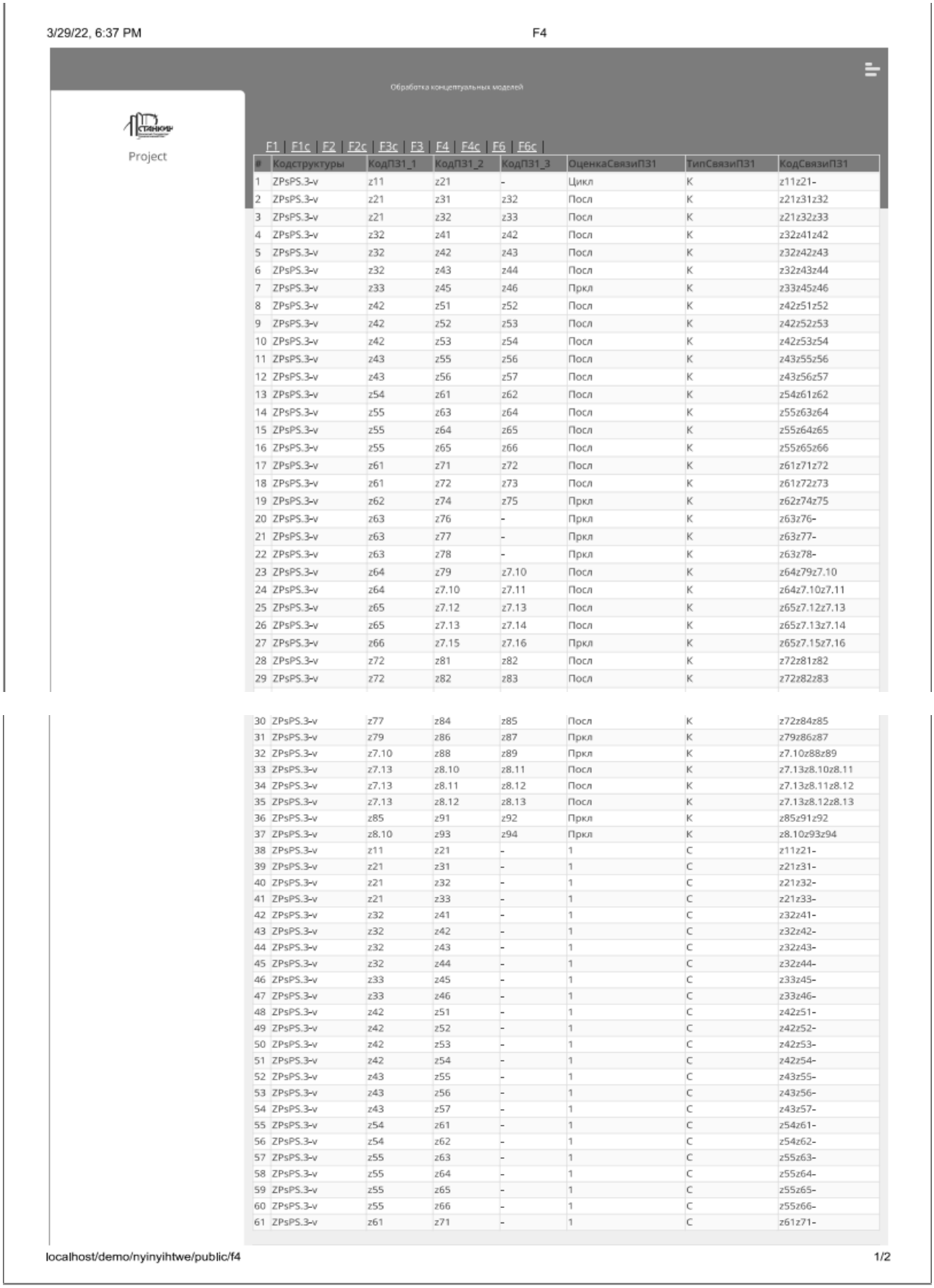
#	Кодструктуры	КлассСвязиПК	КодПК_1	КодПК_2	КодПК_3	Наименование СвязиПК	ТипСвязиПК	ОценкаСвязиПК	КодСвязиПК
1	ZPsPS.1-v	P0	-	P1.1	P1.1	Упорядочивание пр. функционирования передачи	У	99	-P1.1P1.1
2	ZPsPS.1-v	PZ	P1.1	Z1.1	-	Состав пр. функционирования передачи	С	99	P1.1Z1.1-
3	ZPsPS.1-v	Z0	-	Z1.1	Z1.1	Упорядочивание задачи функционирования передачи	У	99	-Z1.1Z1.1
4	ZPsPS.1-v	ZK	Z1.1	K1.7	-	Состав задачи функционирования передачи	С	1	Z1.1K1.7
5	ZPsPS.1-v	KZ	Z1.1	K1.7	-	Компоновка задачи функционирования передачи	К	1	Z1.1K1.7-
6	ZPsPS.1-v	K0	-	K1.7	K1.7	Упорядочивание комп. «средство-устройство»	У	99	-K1.7K1.7
7	ZPsPS.1-v	KO	-	K1.7	O7.1	Состав компонента «средство-устройство»	С	99	K1.7O7.1-
8	ZPsPS.1-v	KO	-	K1.7	O7.2	Состав компонента «средство-устройство»	С	99	K1.7O7.2-
9	ZPsPS.1-v	KO	-	K1.7	O7.4	Состав компонента «средство-устройство»	С	99	K1.7O7.4-
10	ZPsPS.1-v	OK	K1.7	O7.1	O7.2	Компоновка компонента «средство-	К	99	K1.7O7.1O7.2
11	ZPsPS.1-v	OK	K1.7	O7.2	O7.4	Компоновка компонента «средство-устройство»	К	99	K1.7O7.2O7.4
12	ZPsPS.1-v	O0	-	O7.1	O7.1	Упорядочивание узла	У	99	-O7.1O7.1
13	ZPsPS.1-v	O0	-	O7.2	O7.2	Упорядочивание СЕ	У	99	-O7.2O7.2
14	ZPsPS.1-v	O0	-	O7.4	O7.4	Упорядочивание Д	У	99	-O7.4O7.4
15	ZPsPS.1-v	OR	O7.1	R1.1	-	Состав Узла	С	1	O7.1R1.1-
16	ZPsPS.1-v	OR	O7.1	R3.1	-	Состав Узла	С	1	O7.1R3.1-
17	ZPsPS.1-v	OR	O7.1	R3.2	-	Состав Узла	С	1	O7.1R3.2-
18	ZPsPS.1-v	OR	O7.2	R1.2	-	Состав СЕ	С	1	O7.2R1.2-
19	ZPsPS.1-v	OR	O7.2	R2.1	-	Состав СЕ	С	1	O7.2R2.1-
20	ZPsPS.1-v	OR	O7.2	R3.3	-	Состав СЕ	С	1	O7.2R3.3-
21	ZPsPS.1-v	OR	O7.2	R3.4	-	Состав СЕ	С	1	O7.2R3.4-
22	ZPsPS.1-v	OR	O7.2	R3.5	-	Состав СЕ	С	1	O7.2R3.5-
23	ZPsPS.1-v	OR	O7.2	R3.6	-	Состав СЕ	С	1	O7.2R3.6-
24	ZPsPS.1-v	OR	O7.2	R3.7	-	Состав СЕ	С	1	O7.2R3.7-
25	ZPsPS.1-v	OR	O7.2	R3.8	-	Состав СЕ	С	1	O7.2R3.8-
26	ZPsPS.1-v	OR	O7.2	R3.9	-	Состав СЕ	С	1	O7.2R3.9-
27	ZPsPS.1-v	OR	O7.2	R3.10	-	Состав СЕ	С	1	O7.2R3.10-
28	ZPsPS.1-v	OR	O7.2	R3.11	-	Состав СЕ	С	1	O7.2R3.11-
29	ZPsPS.1-v	OR	O7.2	R3.14	-	Состав СЕ	С	1	O7.2R3.14-
30	ZPsPS.1-v	OR	O7.2	R4.1	-	Состав СЕ	С	1	O7.2R4.1-
31	ZPsPS.1-v	OR	O7.2	R4.2	-	Состав СЕ	С	1	O7.2R4.2-
32	ZPsPS.1-v	OR	O7.2	R5.1	-	Состав СЕ	С	1	O7.2R5.1-
33	ZPsPS.1-v	OR	O7.2	R5.2	-	Состав СЕ	С	1	O7.2R5.2-
34	ZPsPS.1-v	OR	O7.2	R5.3	-	Состав СЕ	С	1	O7.2R5.3-
35	ZPsPS.1-v	OR	O7.2	R5.4	-	Состав СЕ	С	1	O7.2R5.4-
36	ZPsPS.1-v	OR	O7.2	R5.5	-	Состав СЕ	С	1	O7.2R5.5-
37	ZPsPS.1-v	OR	O7.2	R5.10	-	Состав СЕ	С	1	O7.2R5.10-

Рис. А7. Фрагмент экрана для формы «Бинарные связи статистических структуры F2»



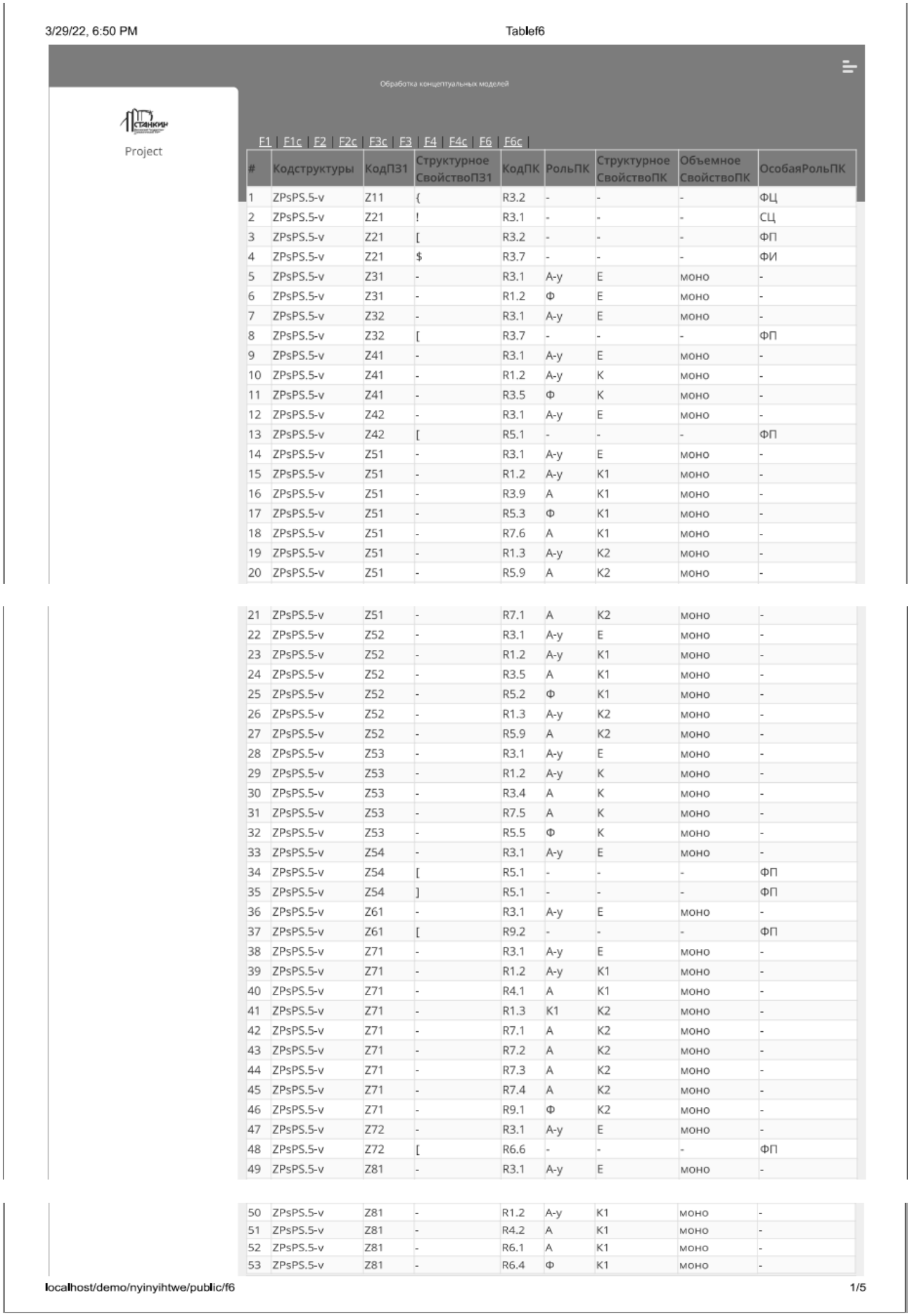
#	Кодструктуры	КодПЗ1	НаименованиеПЗ	Степень формализации	СтатусПЗ1	СтруктурноеСвойствоПЗ1	ПримечаниеПЗ1
1	ZPsPS.3-v	z11	Проектировочный расчет всех цилиндрических передач с прямыми зубьями ЦПсПЗ	Аналит	П	{	
2	ZPsPS.3-v	z21	Проектировочный расчет i_й ЦПсПЗ	Алг	П	!$[	!количество передач
3	ZPsPS.3-v	z31	Фиксация передачи	Эмп	Э		
4	ZPsPS.3-v	z32	Определение проектировочных параметров для текущей передачи	Алг	П	[	модуль стандартный
5	ZPsPS.3-v	z33	Определение проектировочных параметров для всех передач в зависимости от наличия передач	Аналит	П	]?#	]массив модулей стандартных
6	ZPsPS.3-v	z45	Определение проектировочных параметров для всех передач при наличии передач	Аналит	П	/[	/ количество передач
7	ZPsPS.3-v	z59	Переход к следующей передаче	Аналит	Э		i=i+1
8	ZPsPS.3-v	z5_10	Определение проектировочных параметров для всех передач при наличии передач для предыдущей передачи	Аналит	Э	]	массив модулей стандартных
9	ZPsPS.3-v	z46	Определение проектировочных параметров для всех передач при наличии передач	Аналит	П	/[	/ количество передач
10	ZPsPS.3-v	z41	Определение передаточного числа	Стат	Э		
11	ZPsPS.3-v	z42	Определение межосевого расстояния	Алг	П	[	межосевое расстояние
12	ZPsPS.3-v	z43	Определение дополнительных параметров	Алг	П	[	число зубьев у ведомого колеса
13	ZPsPS.3-v	z44	Определение стандартного модуля	Эмп	Э	]	модуль стандартный
14	ZPsPS.3-v	z51	Выбор коэффициента, учитывающего распределение нагрузки по ширине венца	Стат	Э		
15	ZPsPS.3-v	z52	Расчет отношения ширины венца к межосевому расстоянию		Э		
16	ZPsPS.3-v	z53	Выбор вспомогательного коэффициента, зависящего от материала колёс	Стат	Э		
17	ZPsPS.3-v	z54	Расчет межосевого расстояния	Алг	П	] [	] межосевое расстояние
18	ZPsPS.3-v	z61	Определение допускаемого контактного напряжения	Алг	П	[	допускаемое контактное напряжение
19	ZPsPS.3-v	z62	Расчет межосевого расстояния в зависимости от вида зацепления	Аналит	П	?#]	?-вид зацепления

Рис. А8. Фрагмент экрана для формы «Элемент динамических структуры F3»



#	Кодструктуры	КодПЗ1_1	КодПЗ1_2	КодПЗ1_3	ОценкаСвязиПЗ1	ТипСвязиПЗ1	КодСвязиПЗ1
1	ZPsPS.3-v	z11	z21	-	Цикл	K	z11z21-
2	ZPsPS.3-v	z21	z31	z32	Посл	K	z21z31z32
3	ZPsPS.3-v	z21	z32	z33	Посл	K	z21z32z33
4	ZPsPS.3-v	z32	z41	z42	Посл	K	z32z41z42
5	ZPsPS.3-v	z32	z42	z43	Посл	K	z32z42z43
6	ZPsPS.3-v	z32	z43	z44	Посл	K	z32z43z44
7	ZPsPS.3-v	z33	z45	z46	Пркл	K	z33z45z46
8	ZPsPS.3-v	z42	z51	z52	Посл	K	z42z51z52
9	ZPsPS.3-v	z42	z52	z53	Посл	K	z42z52z53
10	ZPsPS.3-v	z42	z53	z54	Посл	K	z42z53z54
11	ZPsPS.3-v	z43	z55	z56	Посл	K	z43z55z56
12	ZPsPS.3-v	z43	z56	z57	Посл	K	z43z56z57
13	ZPsPS.3-v	z54	z61	z62	Посл	K	z54z61z62
14	ZPsPS.3-v	z55	z63	z64	Посл	K	z55z63z64
15	ZPsPS.3-v	z55	z64	z65	Посл	K	z55z64z65
16	ZPsPS.3-v	z55	z65	z66	Посл	K	z55z65z66
17	ZPsPS.3-v	z61	z71	z72	Посл	K	z61z71z72
18	ZPsPS.3-v	z61	z72	z73	Посл	K	z61z72z73
19	ZPsPS.3-v	z62	z74	z75	Пркл	K	z62z74z75
20	ZPsPS.3-v	z63	z76	-	Пркл	K	z63z76-
21	ZPsPS.3-v	z63	z77	-	Пркл	K	z63z77-
22	ZPsPS.3-v	z63	z78	-	Пркл	K	z63z78-
23	ZPsPS.3-v	z64	z79	z7.10	Посл	K	z64z79z7.10
24	ZPsPS.3-v	z64	z7.10	z7.11	Посл	K	z64z7.10z7.11
25	ZPsPS.3-v	z65	z7.12	z7.13	Посл	K	z65z7.12z7.13
26	ZPsPS.3-v	z65	z7.13	z7.14	Посл	K	z65z7.13z7.14
27	ZPsPS.3-v	z66	z7.15	z7.16	Пркл	K	z65z7.15z7.16
28	ZPsPS.3-v	z72	z81	z82	Посл	K	z72z81z82
29	ZPsPS.3-v	z72	z82	z83	Посл	K	z72z82z83
30	ZPsPS.3-v	z77	z84	z85	Посл	K	z72z84z85
31	ZPsPS.3-v	z79	z86	z87	Пркл	K	z79z86z87
32	ZPsPS.3-v	z7.10	z88	z89	Пркл	K	z7.10z88z89
33	ZPsPS.3-v	z7.13	z8.10	z8.11	Посл	K	z7.13z8.10z8.11
34	ZPsPS.3-v	z7.13	z8.11	z8.12	Посл	K	z7.13z8.11z8.12
35	ZPsPS.3-v	z7.13	z8.12	z8.13	Посл	K	z7.13z8.12z8.13
36	ZPsPS.3-v	z85	z91	z92	Пркл	K	z85z91z92
37	ZPsPS.3-v	z8.10	z93	z94	Пркл	K	z8.10z93z94
38	ZPsPS.3-v	z11	z21	-	1	C	z11z21-
39	ZPsPS.3-v	z21	z31	-	1	C	z21z31-
40	ZPsPS.3-v	z21	z32	-	1	C	z21z32-
41	ZPsPS.3-v	z21	z33	-	1	C	z21z33-
42	ZPsPS.3-v	z32	z41	-	1	C	z32z41-
43	ZPsPS.3-v	z32	z42	-	1	C	z32z42-
44	ZPsPS.3-v	z32	z43	-	1	C	z32z43-
45	ZPsPS.3-v	z32	z44	-	1	C	z32z44-
46	ZPsPS.3-v	z33	z45	-	1	C	z33z45-
47	ZPsPS.3-v	z33	z46	-	1	C	z33z46-
48	ZPsPS.3-v	z42	z51	-	1	C	z42z51-
49	ZPsPS.3-v	z42	z52	-	1	C	z42z52-
50	ZPsPS.3-v	z42	z53	-	1	C	z42z53-
51	ZPsPS.3-v	z42	z54	-	1	C	z42z54-
52	ZPsPS.3-v	z43	z55	-	1	C	z43z55-
53	ZPsPS.3-v	z43	z56	-	1	C	z43z56-
54	ZPsPS.3-v	z43	z57	-	1	C	z43z57-
55	ZPsPS.3-v	z54	z61	-	1	C	z54z61-
56	ZPsPS.3-v	z54	z62	-	1	C	z54z62-
57	ZPsPS.3-v	z55	z63	-	1	C	z55z63-
58	ZPsPS.3-v	z55	z64	-	1	C	z55z64-
59	ZPsPS.3-v	z55	z65	-	1	C	z55z65-
60	ZPsPS.3-v	z55	z66	-	1	C	z55z66-
61	ZPsPS.3-v	z61	z71	-	1	C	z61z71-

Рис. А9. Фрагмент экрана для формы «Бинарные связи динамических структуры F4»



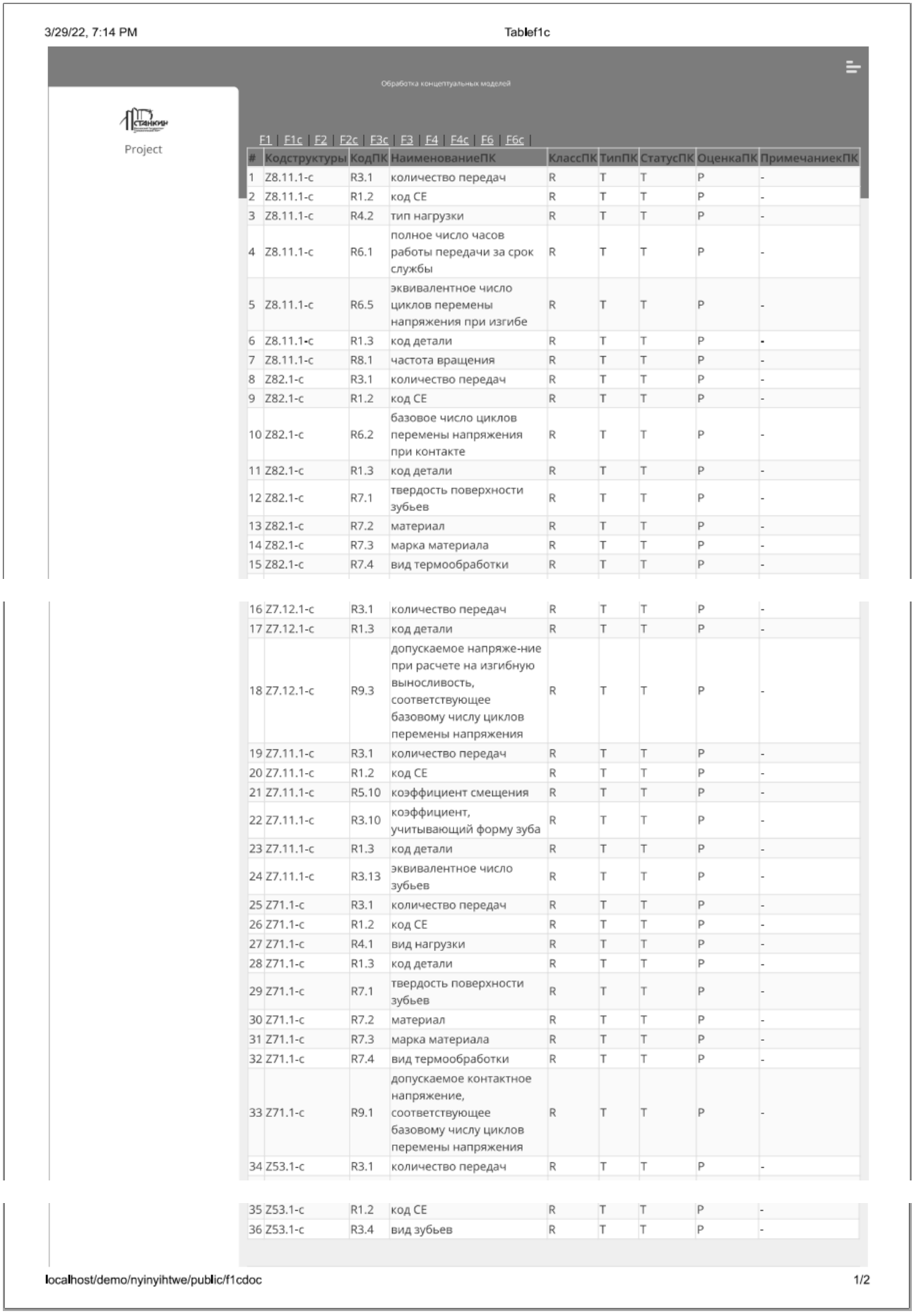
#	Кодструктуры	КодПЗ1	СтруктурноеСвойствоПЗ1	КодПК	РольПК	СтруктурноеСвойствоПК	ОбъемноеСвойствоПК	ОсобаяРольПК
1	ZPsPS.5-v	Z11	{	R3.2	-	-	-	ФЦ
2	ZPsPS.5-v	Z21	!	R3.1	-	-	-	СЦ
3	ZPsPS.5-v	Z21	[	R3.2	-	-	-	ФП
4	ZPsPS.5-v	Z21	$	R3.7	-	-	-	ФИ
5	ZPsPS.5-v	Z31	-	R3.1	A-y	E	моно	-
6	ZPsPS.5-v	Z31	-	R1.2	Ф	E	моно	-
7	ZPsPS.5-v	Z32	-	R3.1	A-y	E	моно	-
8	ZPsPS.5-v	Z32	[	R3.7	-	-	-	ФП
9	ZPsPS.5-v	Z41	-	R3.1	A-y	E	моно	-
10	ZPsPS.5-v	Z41	-	R1.2	A-y	K	моно	-
11	ZPsPS.5-v	Z41	-	R3.5	Ф	K	моно	-
12	ZPsPS.5-v	Z42	-	R3.1	A-y	E	моно	-
13	ZPsPS.5-v	Z42	[	R5.1	-	-	-	ФП
14	ZPsPS.5-v	Z51	-	R3.1	A-y	E	моно	-
15	ZPsPS.5-v	Z51	-	R1.2	A-y	K1	моно	-
16	ZPsPS.5-v	Z51	-	R3.9	A	K1	моно	-
17	ZPsPS.5-v	Z51	-	R5.3	Ф	K1	моно	-
18	ZPsPS.5-v	Z51	-	R7.6	A	K1	моно	-
19	ZPsPS.5-v	Z51	-	R1.3	A-y	K2	моно	-
20	ZPsPS.5-v	Z51	-	R5.9	A	K2	моно	-
21	ZPsPS.5-v	Z51	-	R7.1	A	K2	моно	-
22	ZPsPS.5-v	Z52	-	R3.1	A-y	E	моно	-
23	ZPsPS.5-v	Z52	-	R1.2	A-y	K1	моно	-
24	ZPsPS.5-v	Z52	-	R3.5	A	K1	моно	-
25	ZPsPS.5-v	Z52	-	R5.2	Ф	K1	моно	-
26	ZPsPS.5-v	Z52	-	R1.3	A-y	K2	моно	-
27	ZPsPS.5-v	Z52	-	R5.9	A	K2	моно	-
28	ZPsPS.5-v	Z53	-	R3.1	A-y	E	моно	-
29	ZPsPS.5-v	Z53	-	R1.2	A-y	K	моно	-
30	ZPsPS.5-v	Z53	-	R3.4	A	K	моно	-
31	ZPsPS.5-v	Z53	-	R7.5	A	K	моно	-
32	ZPsPS.5-v	Z53	-	R5.5	Ф	K	моно	-
33	ZPsPS.5-v	Z54	-	R3.1	A-y	E	моно	-
34	ZPsPS.5-v	Z54	[	R5.1	-	-	-	ФП
35	ZPsPS.5-v	Z54	]	R5.1	-	-	-	ФП
36	ZPsPS.5-v	Z61	-	R3.1	A-y	E	моно	-
37	ZPsPS.5-v	Z61	[	R9.2	-	-	-	ФП
38	ZPsPS.5-v	Z71	-	R3.1	A-y	E	моно	-
39	ZPsPS.5-v	Z71	-	R1.2	A-y	K1	моно	-
40	ZPsPS.5-v	Z71	-	R4.1	A	K1	моно	-
41	ZPsPS.5-v	Z71	-	R1.3	K1	K2	моно	-
42	ZPsPS.5-v	Z71	-	R7.1	A	K2	моно	-
43	ZPsPS.5-v	Z71	-	R7.2	A	K2	моно	-
44	ZPsPS.5-v	Z71	-	R7.3	A	K2	моно	-
45	ZPsPS.5-v	Z71	-	R7.4	A	K2	моно	-
46	ZPsPS.5-v	Z71	-	R9.1	Ф	K2	моно	-
47	ZPsPS.5-v	Z72	-	R3.1	A-y	E	моно	-
48	ZPsPS.5-v	Z72	[	R6.6	-	-	-	ФП
49	ZPsPS.5-v	Z81	-	R3.1	A-y	E	моно	-
50	ZPsPS.5-v	Z81	-	R1.2	A-y	K1	моно	-
51	ZPsPS.5-v	Z81	-	R4.2	A	K1	моно	-
52	ZPsPS.5-v	Z81	-	R6.1	A	K1	моно	-
53	ZPsPS.5-v	Z81	-	R6.4	Ф	K1	моно	-

Рис. А10. Фрагмент экрана для формы «Структуры увязки F6»

ПРИЛОЖЕНИЕ Б

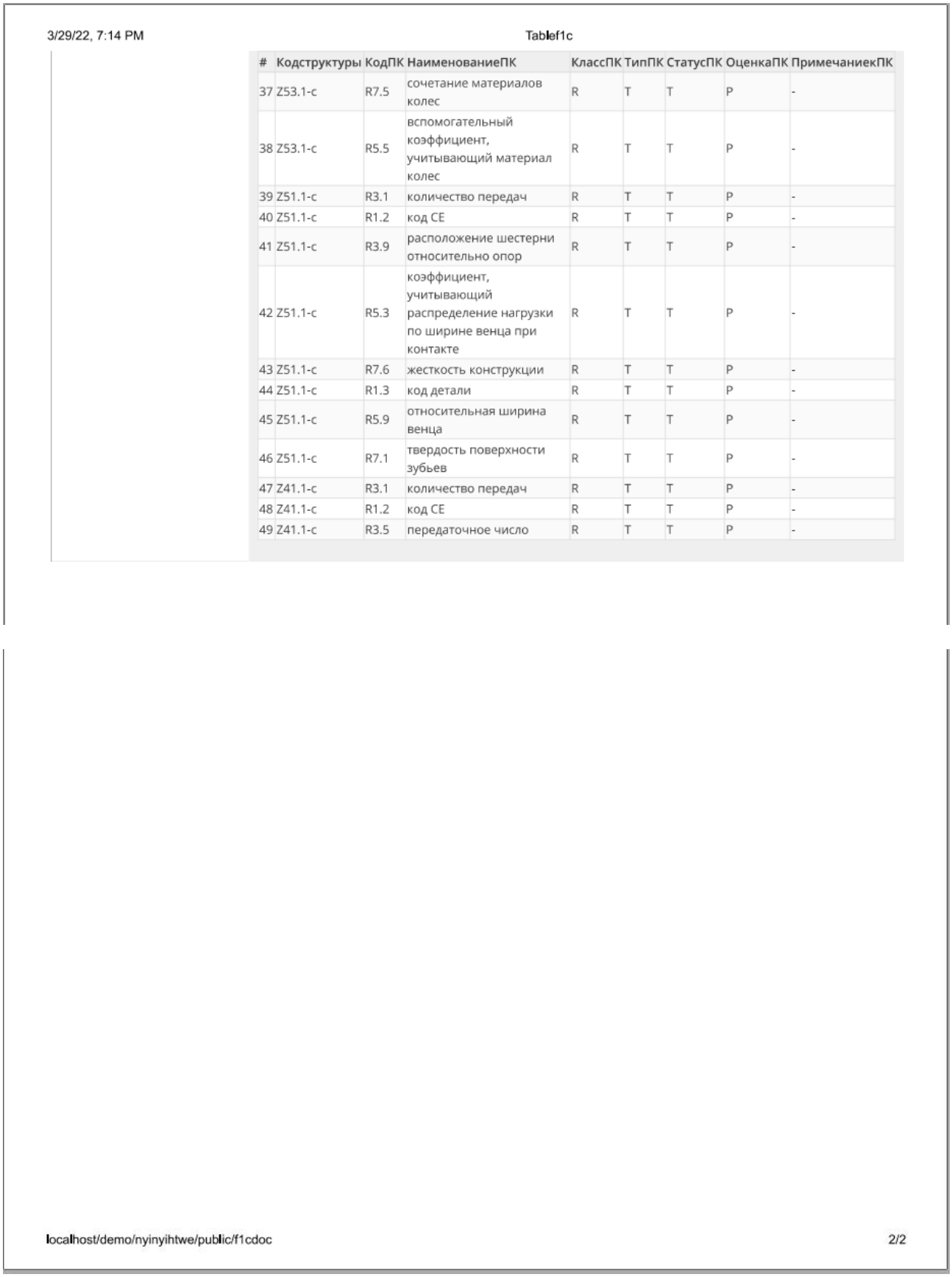
В данном приложении приводятся программные формы инструментального средства поддержки дополнительной обработки концептуальных моделей для тестового примера «обработки КМ»:

1. Фрагмент экрана для формы «Элемент статистических структуры $F1^c$»…….. 239
2. Фрагмент экрана для формы «Бинарных связи статистических структуры $F2^c$»…….. 241
3. Фрагмент экрана для формы «Элемент динамических структуры $F3^c$» …… 244
4. Фрагмент экрана для формы «Структура увязки $F6^c$» …….. 245



#	Кодструктуры	КодПК	НаименованиеПК	КлассПК	ТипПК	СтатусПК	ОценкаПК	ПримечаниекПК
1	Z8.11.1-c	R3.1	количество передач	R	T	T	P	-
2	Z8.11.1-c	R1.2	код СЕ	R	T	T	P	-
3	Z8.11.1-c	R4.2	тип нагрузки	R	T	T	P	-
4	Z8.11.1-c	R6.1	полное число часов работы передачи за срок службы	R	T	T	P	-
5	Z8.11.1-c	R6.5	эквивалентное число циклов перемены напряжения при изгибе	R	T	T	P	-
6	Z8.11.1-c	R1.3	код детали	R	T	T	P	-
7	Z8.11.1-c	R8.1	частота вращения	R	T	T	P	-
8	Z82.1-c	R3.1	количество передач	R	T	T	P	-
9	Z82.1-c	R1.2	код СЕ	R	T	T	P	-
10	Z82.1-c	R6.2	базовое число циклов перемены напряжения при контакте	R	T	T	P	-
11	Z82.1-c	R1.3	код детали	R	T	T	P	-
12	Z82.1-c	R7.1	твердость поверхности зубьев	R	T	T	P	-
13	Z82.1-c	R7.2	материал	R	T	T	P	-
14	Z82.1-c	R7.3	марка материала	R	T	T	P	-
15	Z82.1-c	R7.4	вид термообработки	R	T	T	P	-
16	Z7.12.1-c	R3.1	количество передач	R	T	T	P	-
17	Z7.12.1-c	R1.3	код детали	R	T	T	P	-
18	Z7.12.1-c	R9.3	допускаемое напряже-ние при расчете на изгибную выносливость, соответствующее базовому числу циклов перемены напряжения	R	T	T	P	-
19	Z7.11.1-c	R3.1	количество передач	R	T	T	P	-
20	Z7.11.1-c	R1.2	код СЕ	R	T	T	P	-
21	Z7.11.1-c	R5.10	коэффициент смещения	R	T	T	P	-
22	Z7.11.1-c	R3.10	коэффициент, учитывающий форму зуба	R	T	T	P	-
23	Z7.11.1-c	R1.3	код детали	R	T	T	P	-
24	Z7.11.1-c	R3.13	эквивалентное число зубьев	R	T	T	P	-
25	Z71.1-c	R3.1	количество передач	R	T	T	P	-
26	Z71.1-c	R1.2	код СЕ	R	T	T	P	-
27	Z71.1-c	R4.1	вид нагрузки	R	T	T	P	-
28	Z71.1-c	R1.3	код детали	R	T	T	P	-
29	Z71.1-c	R7.1	твердость поверхности зубьев	R	T	T	P	-
30	Z71.1-c	R7.2	материал	R	T	T	P	-
31	Z71.1-c	R7.3	марка материала	R	T	T	P	-
32	Z71.1-c	R7.4	вид термообработки	R	T	T	P	-
33	Z71.1-c	R9.1	допускаемое контактное напряжение, соответствующее базовому числу циклов перемены напряжения	R	T	T	P	-
34	Z53.1-c	R3.1	количество передач	R	T	T	P	-
35	Z53.1-c	R1.2	код СЕ	R	T	T	P	-
36	Z53.1-c	R3.4	вид зубьев	R	T	T	P	-

Рис. Б1. Фрагмент экрана для формы «Элемент статистических структуры F1^{c}»

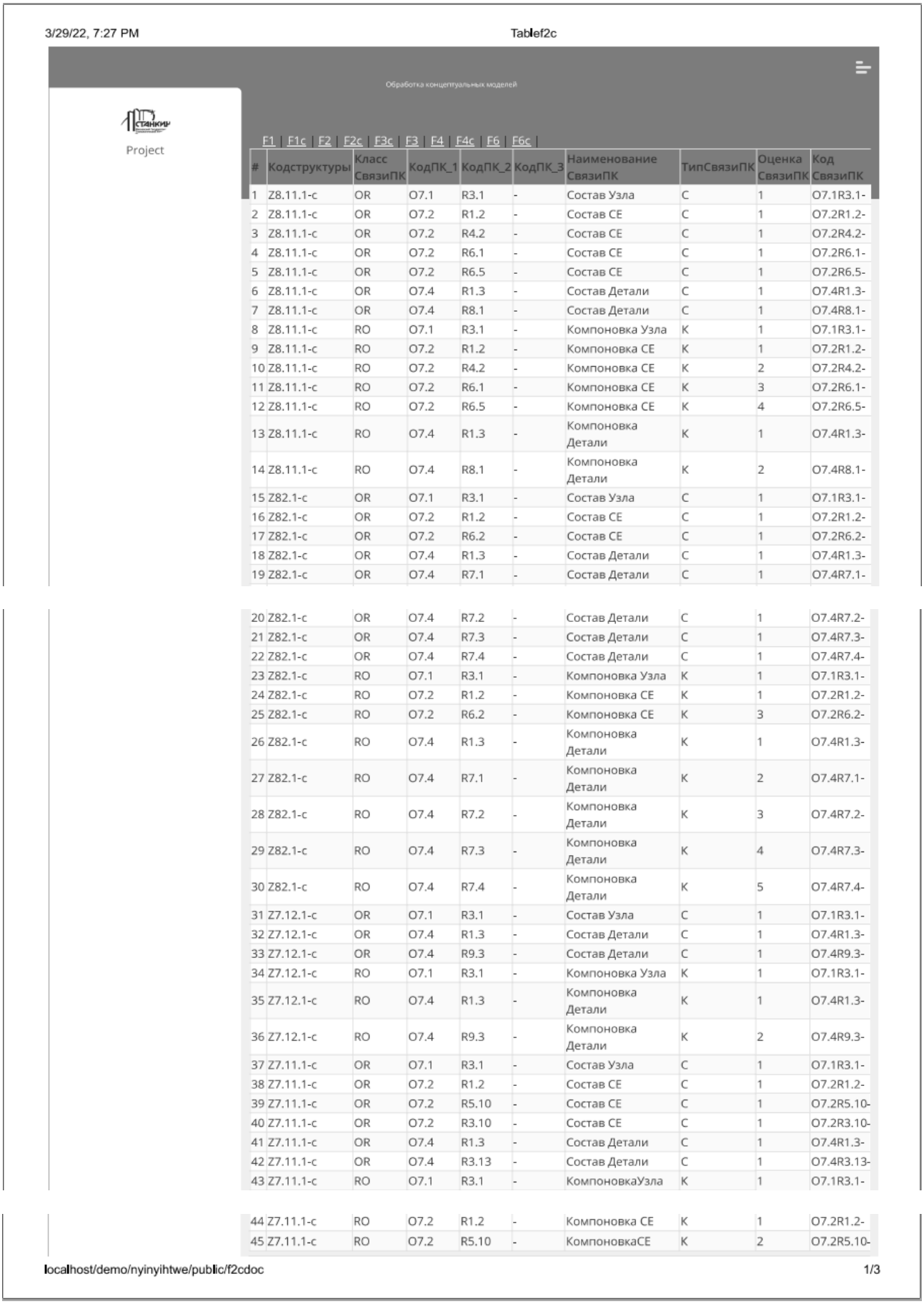


3/29/22, 7:14 PM Tablef1c

#	Кодструктуры	КодПК	НаименованиеПК	КлассПК	ТипПК	СтатусПК	ОценкаПК	ПримечаниекПК
37	Z53.1-c	R7.5	сочетание материалов колес	R	T	T	P	-
38	Z53.1-c	R5.5	вспомогательный коэффициент, учитывающий материал колес	R	T	T	P	-
39	Z51.1-c	R3.1	количество передач	R	T	T	P	-
40	Z51.1-c	R1.2	код CE	R	T	T	P	-
41	Z51.1-c	R3.9	расположение шестерни относительно опор	R	T	T	P	-
42	Z51.1-c	R5.3	коэффициент, учитывающий распределение нагрузки по ширине венца при контакте	R	T	T	P	-
43	Z51.1-c	R7.6	жесткость конструкции	R	T	T	P	-
44	Z51.1-c	R1.3	код детали	R	T	T	P	-
45	Z51.1-c	R5.9	относительная ширина венца	R	T	T	P	-
46	Z51.1-c	R7.1	твердость поверхности зубьев	R	T	T	P	-
47	Z41.1-c	R3.1	количество передач	R	T	T	P	-
48	Z41.1-c	R1.2	код CE	R	T	T	P	-
49	Z41.1-c	R3.5	передаточное число	R	T	T	P	-

localhost/demo/nyinyihtwe/public/f1cdoc 2/2

Рис. Б1. Фрагмент экрана для формы «Элемент статистических структуры F1^{c}» (окончание)



3/29/22, 7:27 PM Tablef2c

Обработка концептуальных моделей

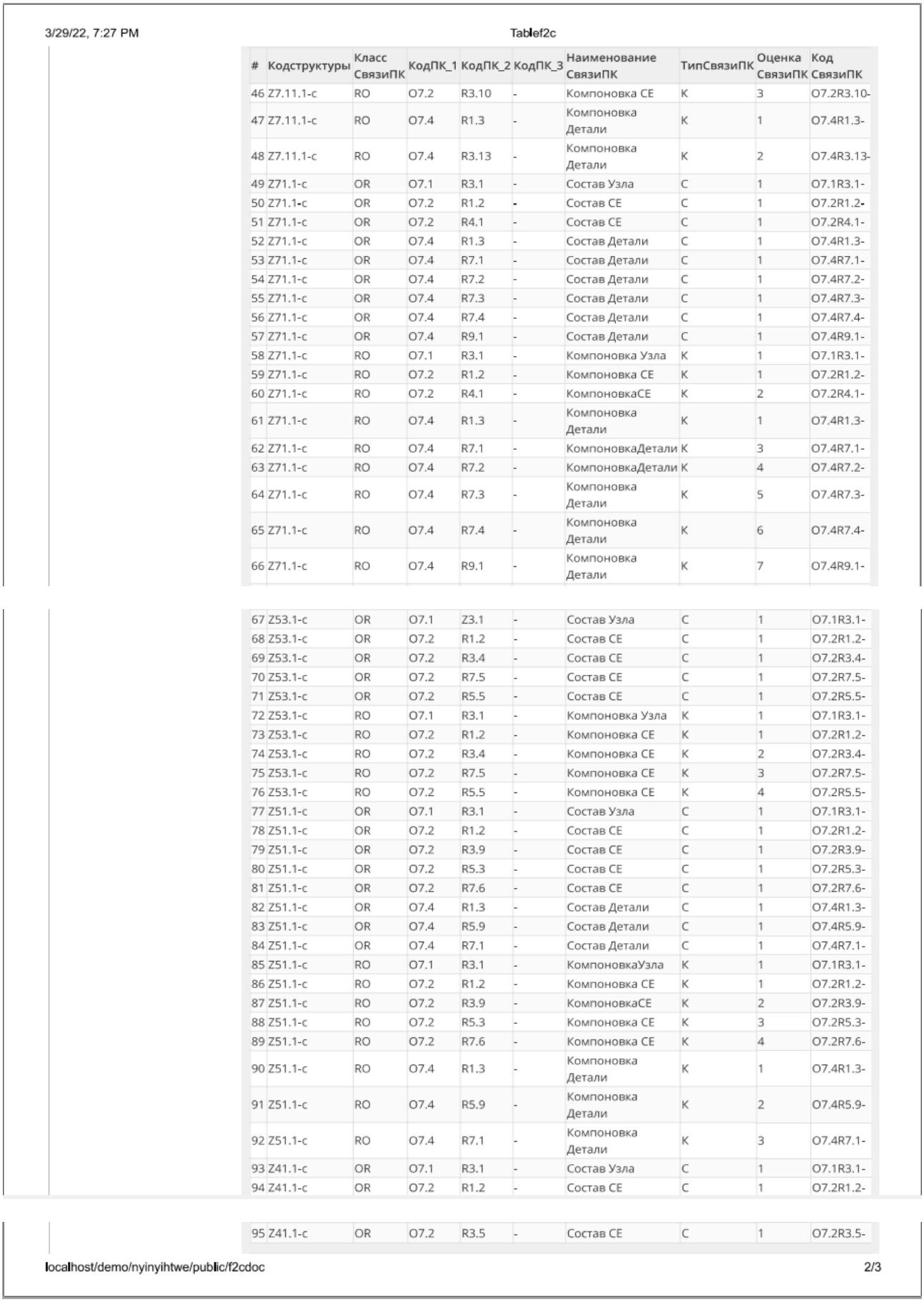
Project

F1 | F1c | F2 | F2c | F3c | F3 | F4 | F4c | F6 | F6c |

#	Кодструктуры	Класс СвязиПК	КодПК_1	КодПК_2	КодПК_3	Наименование СвязиПК	ТипСвязиПК	Оценка СвязиПК	Код СвязиПК
1	Z8.11.1-c	OR	O7.1	R3.1	-	Состав Узла	C	1	O7.1R3.1-
2	Z8.11.1-c	OR	O7.2	R1.2	-	Состав CE	C	1	O7.2R1.2-
3	Z8.11.1-c	OR	O7.2	R4.2	-	Состав CE	C	1	O7.2R4.2-
4	Z8.11.1-c	OR	O7.2	R6.1	-	Состав CE	C	1	O7.2R6.1-
5	Z8.11.1-c	OR	O7.2	R6.5	-	Состав CE	C	1	O7.2R6.5-
6	Z8.11.1-c	OR	O7.4	R1.3	-	Состав Детали	C	1	O7.4R1.3-
7	Z8.11.1-c	OR	O7.4	R8.1	-	Состав Детали	C	1	O7.4R8.1-
8	Z8.11.1-c	RO	O7.1	R3.1	-	Компоновка Узла	K	1	O7.1R3.1-
9	Z8.11.1-c	RO	O7.2	R1.2	-	Компоновка CE	K	1	O7.2R1.2-
10	Z8.11.1-c	RO	O7.2	R4.2	-	Компоновка CE	K	2	O7.2R4.2-
11	Z8.11.1-c	RO	O7.2	R6.1	-	Компоновка CE	K	3	O7.2R6.1-
12	Z8.11.1-c	RO	O7.2	R6.5	-	Компоновка CE	K	4	O7.2R6.5-
13	Z8.11.1-c	RO	O7.4	R1.3	-	Компоновка Детали	K	1	O7.4R1.3-
14	Z8.11.1-c	RO	O7.4	R8.1	-	Компоновка Детали	K	2	O7.4R8.1-
15	Z82.1-c	OR	O7.1	R3.1	-	Состав Узла	C	1	O7.1R3.1-
16	Z82.1-c	OR	O7.2	R1.2	-	Состав CE	C	1	O7.2R1.2-
17	Z82.1-c	OR	O7.2	R6.2	-	Состав CE	C	1	O7.2R6.2-
18	Z82.1-c	OR	O7.4	R1.3	-	Состав Детали	C	1	O7.4R1.3-
19	Z82.1-c	OR	O7.4	R7.1	-	Состав Детали	C	1	O7.4R7.1-
20	Z82.1-c	OR	O7.4	R7.2	-	Состав Детали	C	1	O7.4R7.2-
21	Z82.1-c	OR	O7.4	R7.3	-	Состав Детали	C	1	O7.4R7.3-
22	Z82.1-c	OR	O7.4	R7.4	-	Состав Детали	C	1	O7.4R7.4-
23	Z82.1-c	RO	O7.1	R3.1	-	Компоновка Узла	K	1	O7.1R3.1-
24	Z82.1-c	RO	O7.2	R1.2	-	Компоновка CE	K	1	O7.2R1.2-
25	Z82.1-c	RO	O7.2	R6.2	-	Компоновка CE	K	3	O7.2R6.2-
26	Z82.1-c	RO	O7.4	R1.3	-	Компоновка Детали	K	1	O7.4R1.3-
27	Z82.1-c	RO	O7.4	R7.1	-	Компоновка Детали	K	2	O7.4R7.1-
28	Z82.1-c	RO	O7.4	R7.2	-	Компоновка Детали	K	3	O7.4R7.2-
29	Z82.1-c	RO	O7.4	R7.3	-	Компоновка Детали	K	4	O7.4R7.3-
30	Z82.1-c	RO	O7.4	R7.4	-	Компоновка Детали	K	5	O7.4R7.4-
31	Z7.12.1-c	OR	O7.1	R3.1	-	Состав Узла	C	1	O7.1R3.1-
32	Z7.12.1-c	OR	O7.4	R1.3	-	Состав Детали	C	1	O7.4R1.3-
33	Z7.12.1-c	OR	O7.4	R9.3	-	Состав Детали	C	1	O7.4R9.3-
34	Z7.12.1-c	RO	O7.1	R3.1	-	Компоновка Узла	K	1	O7.1R3.1-
35	Z7.12.1-c	RO	O7.4	R1.3	-	Компоновка Детали	K	1	O7.4R1.3-
36	Z7.12.1-c	RO	O7.4	R9.3	-	Компоновка Детали	K	2	O7.4R9.3-
37	Z7.11.1-c	OR	O7.1	R3.1	-	Состав Узла	C	1	O7.1R3.1-
38	Z7.11.1-c	OR	O7.2	R1.2	-	Состав CE	C	1	O7.2R1.2-
39	Z7.11.1-c	OR	O7.2	R5.10	-	Состав CE	C	1	O7.2R5.10-
40	Z7.11.1-c	OR	O7.2	R3.10	-	Состав CE	C	1	O7.2R3.10-
41	Z7.11.1-c	OR	O7.4	R1.3	-	Состав Детали	C	1	O7.4R1.3-
42	Z7.11.1-c	OR	O7.4	R3.13	-	Состав Детали	C	1	O7.4R3.13-
43	Z7.11.1-c	RO	O7.1	R3.1	-	КомпоновкаУзла	K	1	O7.1R3.1-
44	Z7.11.1-c	RO	O7.2	R1.2	-	Компоновка CE	K	1	O7.2R1.2-
45	Z7.11.1-c	RO	O7.2	R5.10	-	КомпоновкаCE	K	2	O7.2R5.10-

localhost/demo/nyinyihtwe/public/f2cdoc 1/3

Рис. Б2. Фрагмент экрана для формы «Бинарных связи статистических структуры F2^{c}»



3/29/22, 7:27 PM Tablef2c

#	Кодструктуры	Класс СвязиПК	КодПК_1	КодПК_2	КодПК_3	Наименование СвязиПК	ТипСвязиПК	Оценка СвязиПК	Код СвязиПК
46	Z7.11.1-c	RO	O7.2	R3.10	-	Компоновка CE	K	3	O7.2R3.10-
47	Z7.11.1-c	RO	O7.4	R1.3	-	Компоновка Детали	K	1	O7.4R1.3-
48	Z7.11.1-c	RO	O7.4	R3.13	-	Компоновка Детали	K	2	O7.4R3.13-
49	Z71.1-c	OR	O7.1	R3.1	-	Состав Узла	C	1	O7.1R3.1-
50	Z71.1-c	OR	O7.2	R1.2	-	Состав CE	C	1	O7.2R1.2-
51	Z71.1-c	OR	O7.2	R4.1	-	Состав CE	C	1	O7.2R4.1-
52	Z71.1-c	OR	O7.4	R1.3	-	Состав Детали	C	1	O7.4R1.3-
53	Z71.1-c	OR	O7.4	R7.1	-	Состав Детали	C	1	O7.4R7.1-
54	Z71.1-c	OR	O7.4	R7.2	-	Состав Детали	C	1	O7.4R7.2-
55	Z71.1-c	OR	O7.4	R7.3	-	Состав Детали	C	1	O7.4R7.3-
56	Z71.1-c	OR	O7.4	R7.4	-	Состав Детали	C	1	O7.4R7.4-
57	Z71.1-c	OR	O7.4	R9.1	-	Состав Детали	C	1	O7.4R9.1-
58	Z71.1-c	RO	O7.1	R3.1	-	Компоновка Узла	K	1	O7.1R3.1-
59	Z71.1-c	RO	O7.2	R1.2	-	Компоновка CE	K	1	O7.2R1.2-
60	Z71.1-c	RO	O7.2	R4.1	-	КомпоновкаCE	K	2	O7.2R4.1-
61	Z71.1-c	RO	O7.4	R1.3	-	Компоновка Детали	K	1	O7.4R1.3-
62	Z71.1-c	RO	O7.4	R7.1	-	КомпоновкаДетали	K	3	O7.4R7.1-
63	Z71.1-c	RO	O7.4	R7.2	-	КомпоновкаДетали	K	4	O7.4R7.2-
64	Z71.1-c	RO	O7.4	R7.3	-	Компоновка Детали	K	5	O7.4R7.3-
65	Z71.1-c	RO	O7.4	R7.4	-	Компоновка Детали	K	6	O7.4R7.4-
66	Z71.1-c	RO	O7.4	R9.1	-	Компоновка Детали	K	7	O7.4R9.1-
67	Z53.1-c	OR	O7.1	Z3.1	-	Состав Узла	C	1	O7.1R3.1-
68	Z53.1-c	OR	O7.2	R1.2	-	Состав CE	C	1	O7.2R1.2-
69	Z53.1-c	OR	O7.2	R3.4	-	Состав CE	C	1	O7.2R3.4-
70	Z53.1-c	OR	O7.2	R7.5	-	Состав CE	C	1	O7.2R7.5-
71	Z53.1-c	OR	O7.2	R5.5	-	Состав CE	C	1	O7.2R5.5-
72	Z53.1-c	RO	O7.1	R3.1	-	Компоновка Узла	K	1	O7.1R3.1-
73	Z53.1-c	RO	O7.2	R1.2	-	Компоновка CE	K	1	O7.2R1.2-
74	Z53.1-c	RO	O7.2	R3.4	-	Компоновка CE	K	2	O7.2R3.4-
75	Z53.1-c	RO	O7.2	R7.5	-	Компоновка CE	K	3	O7.2R7.5-
76	Z53.1-c	RO	O7.2	R5.5	-	Компоновка CE	K	4	O7.2R5.5-
77	Z51.1-c	OR	O7.1	R3.1	-	Состав Узла	C	1	O7.1R3.1-
78	Z51.1-c	OR	O7.2	R1.2	-	Состав CE	C	1	O7.2R1.2-
79	Z51.1-c	OR	O7.2	R3.9	-	Состав CE	C	1	O7.2R3.9-
80	Z51.1-c	OR	O7.2	R5.3	-	Состав CE	C	1	O7.2R5.3-
81	Z51.1-c	OR	O7.2	R7.6	-	Состав CE	C	1	O7.2R7.6-
82	Z51.1-c	OR	O7.4	R1.3	-	Состав Детали	C	1	O7.4R1.3-
83	Z51.1-c	OR	O7.4	R5.9	-	Состав Детали	C	1	O7.4R5.9-
84	Z51.1-c	OR	O7.4	R7.1	-	Состав Детали	C	1	O7.4R7.1-
85	Z51.1-c	RO	O7.1	R3.1	-	КомпоновкаУзла	K	1	O7.1R3.1-
86	Z51.1-c	RO	O7.2	R1.2	-	Компоновка CE	K	1	O7.2R1.2-
87	Z51.1-c	RO	O7.2	R3.9	-	КомпоновкаCE	K	2	O7.2R3.9-
88	Z51.1-c	RO	O7.2	R5.3	-	Компоновка CE	K	3	O7.2R5.3-
89	Z51.1-c	RO	O7.2	R7.6	-	Компоновка CE	K	4	O7.2R7.6-
90	Z51.1-c	RO	O7.4	R1.3	-	Компоновка Детали	K	1	O7.4R1.3-
91	Z51.1-c	RO	O7.4	R5.9	-	Компоновка Детали	K	2	O7.4R5.9-
92	Z51.1-c	RO	O7.4	R7.1	-	Компоновка Детали	K	3	O7.4R7.1-
93	Z41.1-c	OR	O7.1	R3.1	-	Состав Узла	C	1	O7.1R3.1-
94	Z41.1-c	OR	O7.2	R1.2	-	Состав CE	C	1	O7.2R1.2-
95	Z41.1-c	OR	O7.2	R3.5	-	Состав CE	C	1	O7.2R3.5-

localhost/demo/nyinyihtwe/public/f2cdoc 2/3

Рис. Б2. Фрагмент экрана для формы «Бинарных связи статистических структуры $F2^c$» (прододжение)

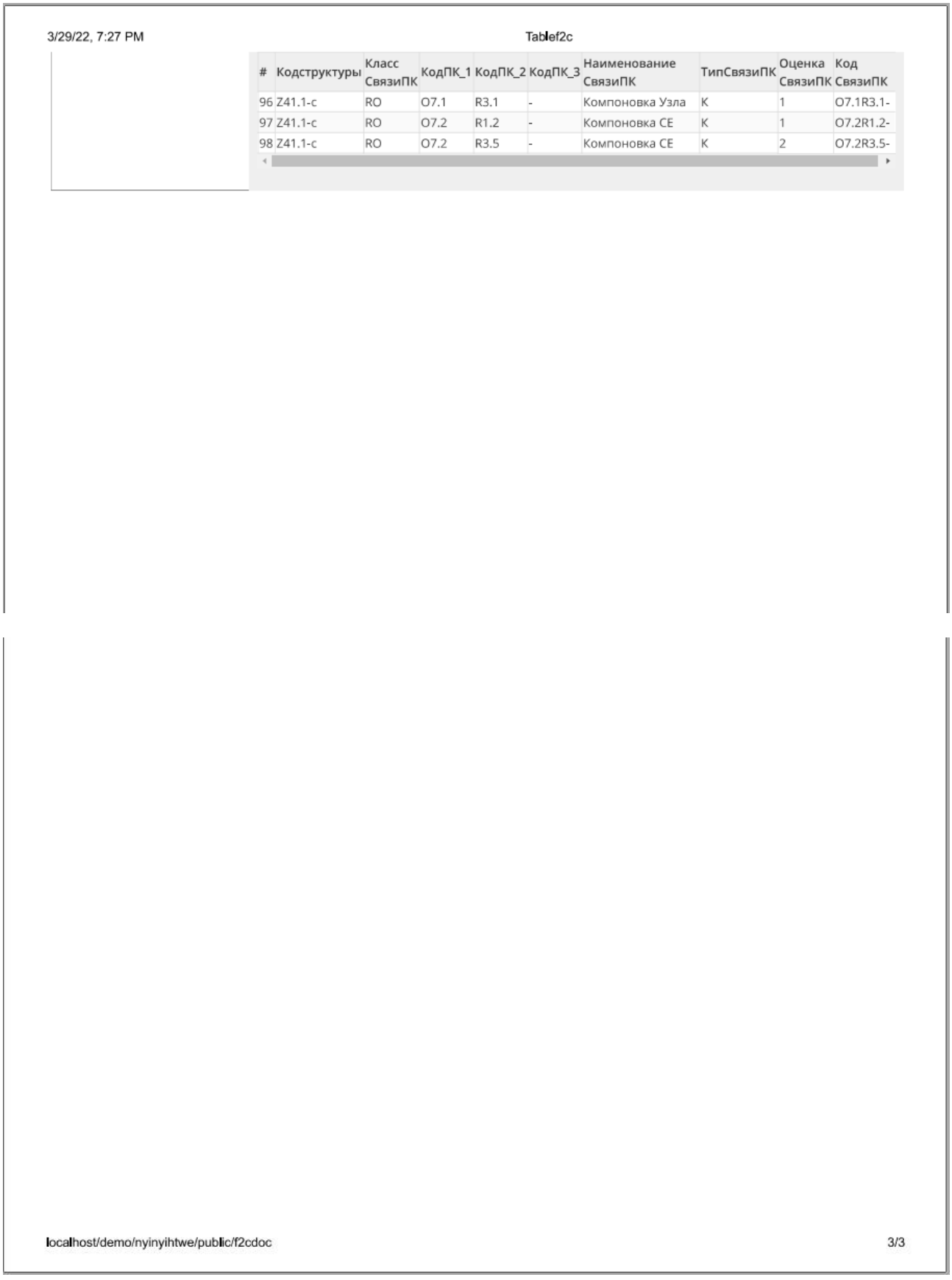
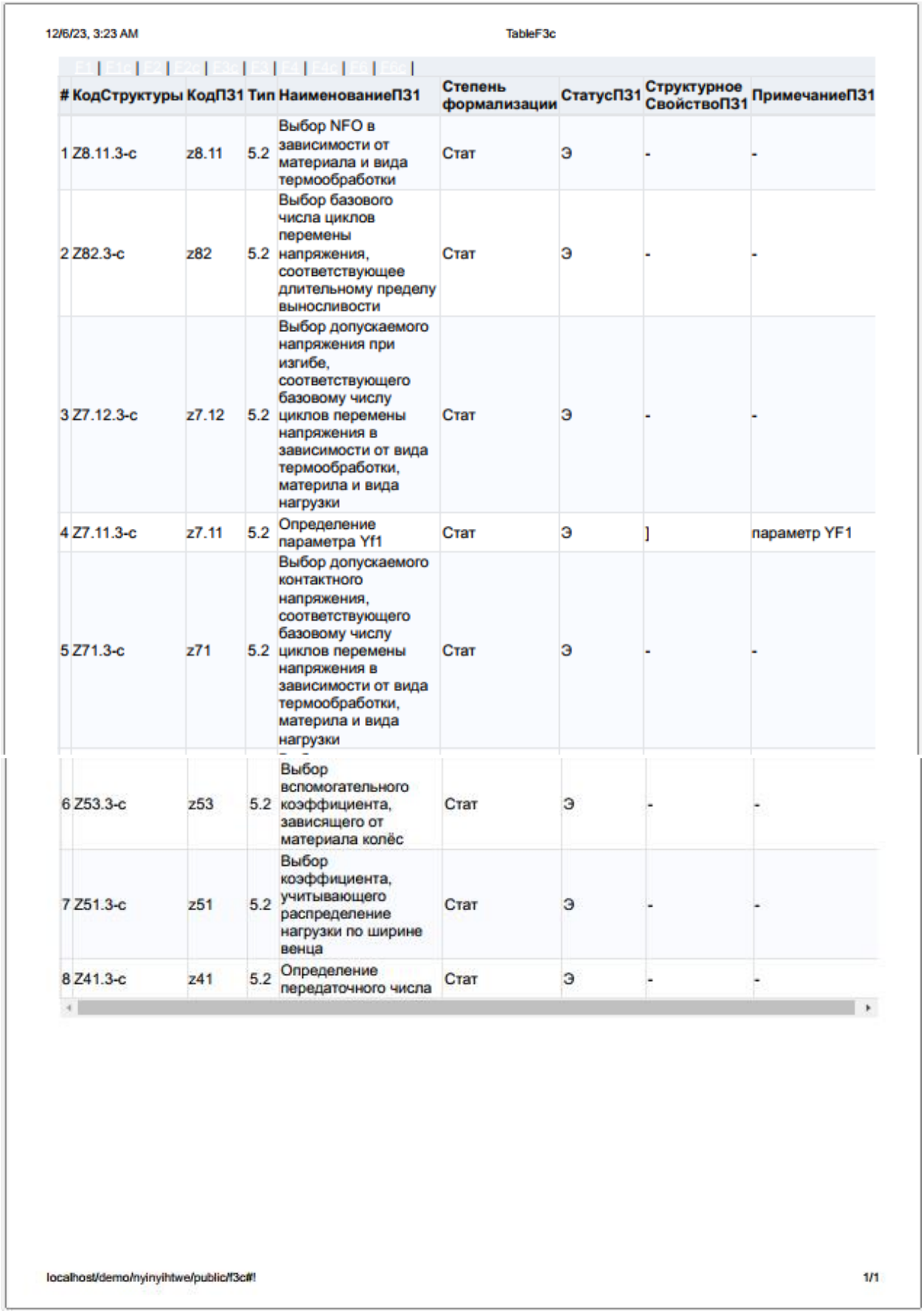


Рис. Б2. Фрагмент экрана для формы «Бинарных связи статистических структуры $F2^c$» (окончание)

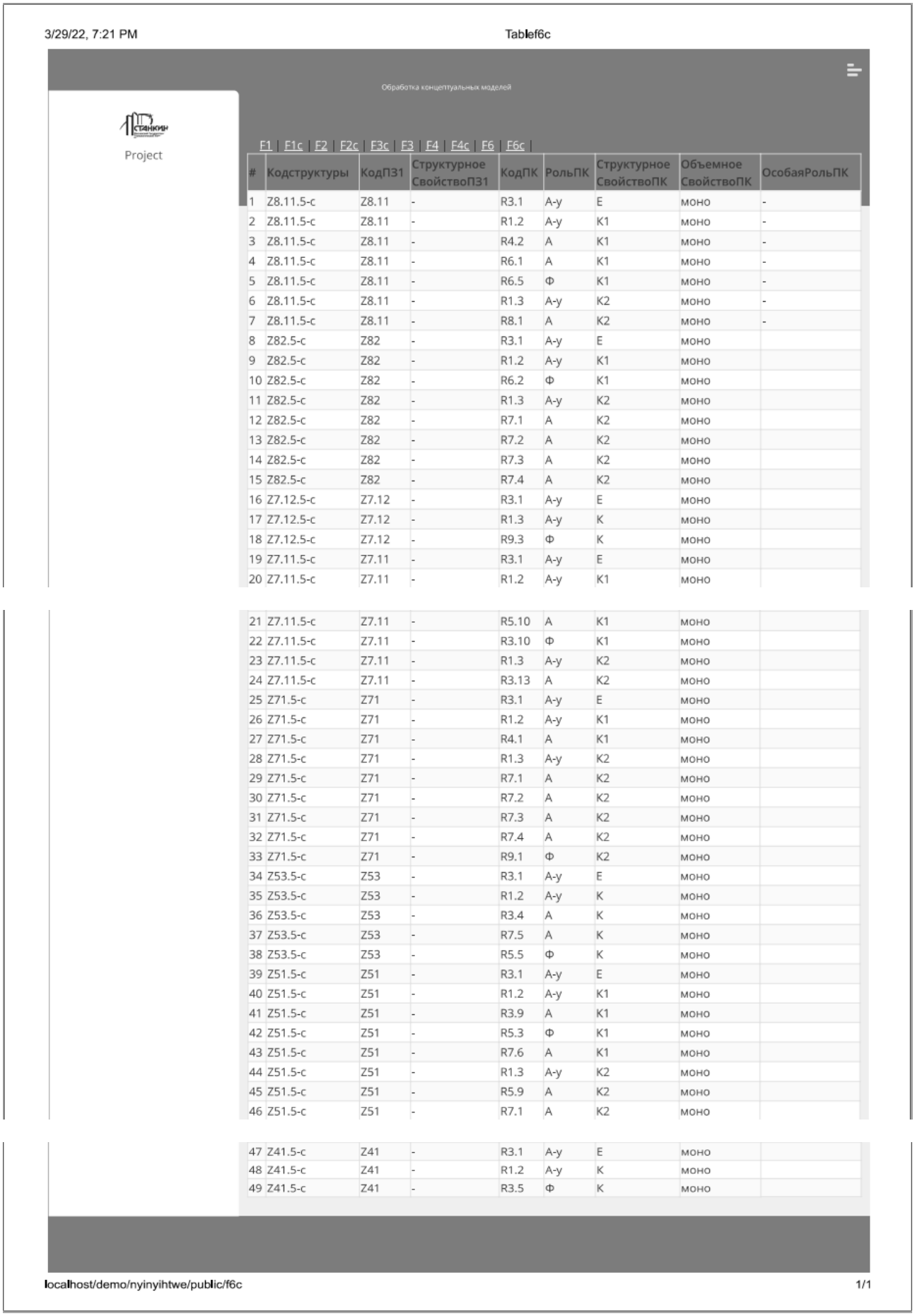


12/6/23, 3:23 AM TableF3c

#	КодСтруктуры	КодП31	Тип	НаименованиеП31	Степень формализации	СтатусП31	СтруктурноеСвойствоП31	ПримечаниеП31
1	Z8.11.3-c	z8.11	5.2	Выбор NFO в зависимости от материала и вида термообработки	Стат	Э	-	-
2	Z82.3-c	z82	5.2	Выбор базового числа циклов перемены напряжения, соответствующее длительному пределу выносливости	Стат	Э	-	-
3	Z7.12.3-c	z7.12	5.2	Выбор допускаемого напряжения при изгибе, соответствующего базовому числу циклов перемены напряжения в зависимости от вида термообработки, материла и вида нагрузки	Стат	Э	-	-
4	Z7.11.3-c	z7.11	5.2	Определение параметра Yf1	Стат	Э	]	параметр YF1
5	Z71.3-c	z71	5.2	Выбор допускаемого контактного напряжения, соответствующего базовому числу циклов перемены напряжения в зависимости от вида термообработки, материла и вида нагрузки	Стат	Э	-	-
6	Z53.3-c	z53	5.2	Выбор вспомогательного коэффициента, зависящего от материала колёс	Стат	Э	-	-
7	Z51.3-c	z51	5.2	Выбор коэффициента, учитывающего распределение нагрузки по ширине венца	Стат	Э	-	-
8	Z41.3-c	z41	5.2	Определение передаточного числа	Стат	Э	-	-

localhost/demo/nyinyihtwe/public/f3c#! 1/1

Рис. Б3. Фрагмент экрана для формы «Элемент динамических структуры F3^{c}»



#	Кодструктуры	КодПЗ1	Структурное СвойствоПЗ1	КодПК	РольПК	Структурное СвойствоПК	Объемное СвойствоПК	ОсобаяРольПК
1	Z8.11.5-c	Z8.11	-	R3.1	A-y	E	моно	-
2	Z8.11.5-c	Z8.11	-	R1.2	A-y	K1	моно	-
3	Z8.11.5-c	Z8.11	-	R4.2	A	K1	моно	-
4	Z8.11.5-c	Z8.11	-	R6.1	A	K1	моно	-
5	Z8.11.5-c	Z8.11	-	R6.5	Ф	K1	моно	-
6	Z8.11.5-c	Z8.11	-	R1.3	A-y	K2	моно	-
7	Z8.11.5-c	Z8.11	-	R8.1	A	K2	моно	-
8	Z82.5-c	Z82	-	R3.1	A-y	E	моно	
9	Z82.5-c	Z82	-	R1.2	A-y	K1	моно	
10	Z82.5-c	Z82	-	R6.2	Ф	K1	моно	
11	Z82.5-c	Z82	-	R1.3	A-y	K2	моно	
12	Z82.5-c	Z82	-	R7.1	A	K2	моно	
13	Z82.5-c	Z82	-	R7.2	A	K2	моно	
14	Z82.5-c	Z82	-	R7.3	A	K2	моно	
15	Z82.5-c	Z82	-	R7.4	A	K2	моно	
16	Z7.12.5-c	Z7.12	-	R3.1	A-y	E	моно	
17	Z7.12.5-c	Z7.12	-	R1.3	A-y	K	моно	
18	Z7.12.5-c	Z7.12	-	R9.3	Ф	K	моно	
19	Z7.11.5-c	Z7.11	-	R3.1	A-y	E	моно	
20	Z7.11.5-c	Z7.11	-	R1.2	A-y	K1	моно	
21	Z7.11.5-c	Z7.11	-	R5.10	A	K1	моно	
22	Z7.11.5-c	Z7.11	-	R3.10	Ф	K1	моно	
23	Z7.11.5-c	Z7.11	-	R1.3	A-y	K2	моно	
24	Z7.11.5-c	Z7.11	-	R3.13	A	K2	моно	
25	Z71.5-c	Z71	-	R3.1	A-y	E	моно	
26	Z71.5-c	Z71	-	R1.2	A-y	K1	моно	
27	Z71.5-c	Z71	-	R4.1	A	K1	моно	
28	Z71.5-c	Z71	-	R1.3	A-y	K2	моно	
29	Z71.5-c	Z71	-	R7.1	A	K2	моно	
30	Z71.5-c	Z71	-	R7.2	A	K2	моно	
31	Z71.5-c	Z71	-	R7.3	A	K2	моно	
32	Z71.5-c	Z71	-	R7.4	A	K2	моно	
33	Z71.5-c	Z71	-	R9.1	Ф	K2	моно	
34	Z53.5-c	Z53	-	R3.1	A-y	E	моно	
35	Z53.5-c	Z53	-	R1.2	A-y	K	моно	
36	Z53.5-c	Z53	-	R3.4	A	K	моно	
37	Z53.5-c	Z53	-	R7.5	A	K	моно	
38	Z53.5-c	Z53	-	R5.5	Ф	K	моно	
39	Z51.5-c	Z51	-	R3.1	A-y	E	моно	
40	Z51.5-c	Z51	-	R1.2	A-y	K1	моно	
41	Z51.5-c	Z51	-	R3.9	A	K1	моно	
42	Z51.5-c	Z51	-	R5.3	Ф	K1	моно	
43	Z51.5-c	Z51	-	R7.6	A	K1	моно	
44	Z51.5-c	Z51	-	R1.3	A-y	K2	моно	
45	Z51.5-c	Z51	-	R5.9	A	K2	моно	
46	Z51.5-c	Z51	-	R7.1	A	K2	моно	
47	Z41.5-c	Z41	-	R3.1	A-y	E	моно	
48	Z41.5-c	Z41	-	R1.2	A-y	K	моно	
49	Z41.5-c	Z41	-	R3.5	Ф	K	моно	

Рис. Б4. Фрагмент экрана для формы «Структура увязки F6^{c}»

ПРИЛОЖЕНИЕ В. Справка об использовании результатов диссертационного исследования

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технологический университет «СТАНКИН»
(ФГБОУ ВО «МГТУ «СТАНКИН»)

СПРАВКА
об использовании результатов диссертационного исследования

Научно-практические результаты, изложенные в диссертационной работе Ньи Ньи Хтве на тему «Разработка метода и средств поддержки процессов обработки концептуальных представлений предметных задач» использованы при разработке программных средств и методического обеспечения поддержки обработки статистических предметных ограничений при концептуальном моделировании прикладных задач.

Программные средства были апробированы и использованы в образовательной и научной деятельности Федерального государственного бюджетного образовательного учреждения высшего образования «Московский государственный технологический университет «СТАНКИН».

В частности, при подготовке обучающихся, в качестве учебно-проектного средства поддержки дисциплины «Концептуальное моделирование проектных задач» при подготовке бакалавров по направлению 09.03.01 «Информатика и вычислительная техника», профили подготовки «Модели, методы и программное обеспечение анализа проектных решений».

Обработка и документирование концептуальных представлений, формируемых при создании автоматизированных систем, позволяет контролировать описания компонентов моделей и систематизировать ограничения, зафиксированные в нормативно-справочной документации в предметных областях. Это приводит к сокращению количества возникаемых ошибок и уменьшению числа итераций, необходимых для формирования и доработки модельных представлений.

Полагаем, что данную методику и инструментальные средства целесообразно использовать и в дальнейшем для модификации и развития средств поддержки методологии интеллектуального труда, как основы учебно-проектной деятельности студентов.

Проректора по образовательной деятельности и молодежной политике ФГБОУ ВО «МГТУ «СТАНКИН», канд. техн. наук, доцент /Бильчук М. В./

«21» 12 2023 г.

Подпись руки Бильчук М.В. удостоверяю
УД ФГБОУ ВО «МГТУ «СТАНКИН»
Главный специалист
Корнилова М.Р.