

Lotsia PDM PLUS

Администрирование и настройка системы

Краткое учебное пособие

Москва, 2018

© 1997-2018, Лоция Софтвэа

Содержание

1 ВВЕДЕНИЕ.....	11
2 НАЗНАЧЕНИЕ И СОСТАВ LOTSIA PDM PLUS	12
2.1 Резюме	15
3 УСТАНОВКА ПРОГРАММЫ, ПОДГОТОВКА К РАБОТЕ И НАЧАЛО РАБОТЫ	16
3.1 Установка программы.....	16
3.2 Установка аппаратной защиты.....	18
3.3 Создание пустой базы данных Sybase SQL Anywhere с использованием средств, входящих в стандартную поставку Lotsia PDM PLUS	18
3.4 Настройка ODBC при работе с Sybase SQL Anywhere	19
3.5 Инициализация базы данных.....	21
3.6 Запуск программы.....	22
3.7 Обновление версий Lotsia PDM PLUS	24
3.7.1 Обновление клиентской части программы	24
3.7.2 Настройка автоматического обновления клиентской части программы	26
3.7.3 Обновление серверной части программы	26
3.8 Резюме	27
3.9 Установка программы. Вопросы для самопроверки	27
4 БАЗОВЫЕ ПОНЯТИЯ ПРОГРАММЫ LOTSIA PDM. СТРУКТУРА БАЗЫ ДАННЫХ	28
4.1 Информационный объект, проекты.....	28
4.2 Элементы структуры БД.....	29
4.2.1 Связи между объектами в системе Lotsia PDM	30
4.2.2 Тип объекта	31
4.2.2.1 Входямость объектов.....	32
4.2.3 Атрибуты	33
4.2.3.1 Тип данных атрибута	34
4.2.3.2 Способы редактирования атрибута	34
Особенности значений строковых атрибутов	35
4.2.3.3 Множественность атрибута	36
4.2.3.4 Связь атрибута с объектом (проектом)	36
4.2.3.5 Соответствие «тип объекта» – «атрибут»	36
4.2.3.6 Группа атрибутов.....	36
4.3 Создание элементов структуры БД	36

4.3.1	Создание типов связей	38
4.3.2	Создание типов объектов.....	39
4.3.3	Определение входимости для типов объектов.....	39
4.3.4	Создание атрибутов	41
4.3.5	Настройка соответствия тип объекта – атрибут	42
4.4	Резюме	43
4.5	Создание элементов структуры БД. Учебное задание	43
4.6	Элементы структуры БД. Вопросы для самопроверки.....	45
5	СОЗДАНИЕ ОБЪЕКТОВ И ДЕРЕВА ПРОЕКТА	46
5.1	Создание новых объектов, добавление создаваемых объектов в проект ...	46
5.2	Использование различных типов связей.....	50
5.2.1	Альтернативные способы связывания (установки ссылок) различных сущностей	51
5.3	Резюме	52
5.4	Создание проекта. Учебное задание	52
5.4.1	Создание проекта. Присвоение атрибутов. Заимствование	52
5.4.2	Справочник предприятия.....	53
5.4.3	Использование горизонтальных связей.....	55
5.4.4	Использование дополнительной подчиненной связи	55
5.5	Создание объектов и проектов. Вопросы для самопроверки	57
6	ВАРИАНТЫ, ИСПОЛНЕНИЯ И КОПИИ ОБЪЕКТОВ.....	58
6.1	Вариантное проектирование.....	58
6.2	Создание вариантов	58
6.3	Создание исполнения объекта.....	59
6.4	Создание копии объекта	59
6.5	Резюме	60
6.6	Варианты, исполнения, копии. Учебное задание	60
6.7	Варианты, копии, исполнения. Вопросы для самопроверки	61
7	ИМПОРТ СТРУКТУРЫ БД.....	62
7.1	Резюме	62
7.2	Импорт структуры БД. Учебное задание	63
7.3	Вопрос для самопроверки.....	63
8	РАБОТА С ПРОЕКТОМ.....	64
8.1	Поиск объектов	64

8.2	Навигация по дереву проекта	66
8.3	Подборка объектов.....	68
8.4	Удаление объекта из проекта (дерева связей) и БД.....	71
8.5	Сравнение состава проектов и атрибутов	73
8.5.1	Сравнение состава проектов.....	73
8.5.2	Сравнение атрибутов двух объектов.....	73
8.6	Резюме.....	74
8.7	Работа с проектом. Учебное задание.....	74
8.8	Работа с проектом. Вопросы для самопроверки.....	75
9 КЛАССИФИКАТОРЫ. ВОЗМОЖНОСТИ И ОСОБЕННОСТИ РЕАЛИЗАЦИИ.....		76
9.1	Формирование значение атрибута из списка возможных значений и ручным вводом.....	77
9.2	Организация структурированного списка возможных значений атрибута.....	78
9.3	Последовательное формирование значений атрибута.....	78
9.4	Формирование значения атрибута из нескольких параллельных, заранее определенных ветвей классификатора, в заданной последовательности	78
9.5	Одновременное присвоение значения нескольким атрибутам.	79
9.6	Использование классификатора в действиях над объектами.....	79
9.7	Преимущества и ограничения использования классификатора.....	79
9.8	Резюме	80
9.9	Настройка классификатора. Учебное задание	80
9.10	Классификатор. Вопросы для самопроверки.....	83
10 НАСТРОЙКА ИНТЕРФЕЙСА		84
10.1	Настройка Главного меню и соответствующей инструментальной панели.....	84
10.2	Формы для объектов, поисковые формы и формы для подборок.....	87
10.2.1	Настройка форм. Создание форм типа «Атрибуты»	88
10.2.1.1	Использование вложенных форм	96
10.2.2	Особенности настройки форм типов «Дочерние объекты», «Входимость», «Связанные объекты», «Подборка объектов»	98
10.2.3	Привязка форм к типам связей.....	101
10.2.4	Формы типов «Документы архива» и «Документы потомков».....	102
10.2.5	Начальные сведения по составлению выражений.....	103
10.2.6	Всплывающие подсказки	108
10.2.7	Создание названий колонок и вычисляемых полей.	109
10.3	Профили и настройки пользователей.....	110
10.3.1	Подключение Главных меню к профилю	111

10.3.2	Формирование названия объекта в дереве.....	112
10.3.3	Настройка отображаемой пиктограммы	114
10.3.4	Настройка сортировки объектов, находящихся в дереве связей на одном уровне	115
10.3.5	Настройка соответствия форм типам объектов.....	117
10.3.6	Настройки окна проекта	119
10.3.7	Настройка контекстных меню и инструментальной панели окна	120
10.3.8	Настройка вкладок и закрепленной панели окна программы.....	121
10.3.9	Параметры импорта документов.....	122
10.3.10	Настройки стандартных окон и вкладок.....	125
10.3.11	Настройки и параметры, используемые при работе с сообщениями.	125
10.4	Настройка стандартных вкладок окна дерева проектов	126
10.5	Представления проекта и фильтры	127
10.6	Резюме	128
10.7	Выполнение настроек интерфейса пользователей. Учебное задание	129
10.8	Настройки интерфейса. Вопросы для самопроверки	131
11	СОЗДАНИЕ ПОЛЬЗОВАТЕЛЕЙ, КОНТАКТОВ, АДРЕСНЫХ КНИГ ...	132
11.1	Резюме	138
11.2	Создание пользователей и выполнение необходимых настроек. Учебное задание	138
11.3	Создание пользователей и выполнение необходимых настроек. Вопросы для самопроверки.....	138
16	ПРАВА ДОСТУПА	139
16.1	Права доступа. Общая информация	139
16.2	Доступ к информационным объектам, проектам, атрибутам.....	141
16.2.1	Установка прав на уровне структуры БД	141
16.2.2	Права на уровне экземпляров объектов.....	144
16.2.3	Изменение унаследованных прав доступа.....	147
16.2.4	Регенерация прав доступа	148
16.3	Шаблоны прав.....	149
16.4	Права доступа к документам архива (файлам)	152
17	ДЕЙСТВИЯ НАД ОБЪЕКТАМИ LOTSIA PDM.....	155
17.1	Общий порядок создания действий над объектами.....	155
17.2	Различные способы запуска действий	159
17.3	Настройка и использование форм шагов действий	160
17.3.1	Настройка редактирования колонок средствами редактора форм	160

17.3.1.1	Настройка редактирования переменных с типом значения «Число», «Строка» и «Дата/время» с типом редактирования «Свободный».....	161
17.3.1.2	Настройка редактирования переменных с типом значения «Строка» и с типом редактирования «Выбор файла».....	164
17.3.1.3	Использование окон выбора объектов в формах действий над объектами	165
17.3.2	Настройка редактирования заданием свойств переменной: способы редактирования, условия отбора	168
17.3.2.1	Формирования списка возможных значений переменной: использование имеющихся значений атрибута	169
17.3.2.2	Формирования списка возможных значений переменной: использование выражений	170
17.3.2.3	Условия отбора	172
17.3.2.4	Выбор файла или папки	175
17.3.3	Автоматическое завершение форм	175
17.3.4	Использование классификаторов в формах действий.....	176
17.3.5	Использование кнопок в формах действий	177
17.4	Использование шагов типов «Окно выбора объектов из дерева» и «Окно выбора объектов из списка»	178
17.5	Использование и получение идентификаторов (кодов ID) в действиях. Присвоение значений сущностям, имеющим ID	183
17.6	Использование меток шагов и функции GoTo. Циклическая обработка в действиях над объектами.....	184
17.7	Работа с NULL – значениями	185
17.8	Вложенные действия	186
17.9	Некоторые свойства и особенности действий над объектами	187
17.10	Резюме	189
17.11	Примеры (упражнения по написанию) действий	190
17.11.1	Простейшее действие по открытию объекта	190
17.11.2	Выбор объекта переключателем	193
17.11.3	Действие по добавлению объекта служебной записки	195
17.11.4	Добавление объекта документа.....	198
17.11.5	Присвоение регистрационного номера с помощью действий	201
17.11.6	Создание объекта документа с его автоматической регистрацией. Пример использования вложенного действия	203
17.12	Создание действий над объектами. Учебное задание	204
17.13	Действия над объектами. Вопросы для самопроверки	204
18 ИМПОРТ ОБЪЕКТОВ. РЕЗЮМЕ		206

19	АРХИВ ДОКУМЕНТОВ	207
19.1	Создание защищенного хранилища, подключение его к библиотеке	208
19.2	Создание библиотеки.....	210
19.3	Регистрация типов документов	211
19.4	Регистрация приложения и метода (-ов) его запуска	212
19.5	Другие настройки, относящиеся к работе с архивом	214
19.6	Импорт документов в объект	216
19.7	Импорт структуры папок из файловой системы.....	217
19.8	Резюме	218
19.9	Архив. Учебное задание	219
19.10	Архив документов. Вопросы для самопроверки.....	220
20	СОЗДАНИЕ И НАСТРОЙКА ОТЧЕТОВ	222
20.1	Общий порядок создания отчета.....	222
20.2	Перемещение и копирование отчетов.....	230
20.3	Использование отчета в качестве рабочего (интерфейсного) окна конечного пользователя.....	230
20.4	Использование отчета в качестве формы (вкладки) для объекта в дереве связей.....	231
20.5	Примеры создания отчетов.....	231
20.5.1	Последовательность создания отчета	232
20.5.2	Сортировка строк отчета	237
20.5.3	Нумерация строк отчета:	238
20.5.4	Группировка строк отчета	238
20.5.5	Обработка повторяющихся строк отчетов.....	241
20.5.6	Использование аргументов отчета.....	247
20.5.6.1	Особенности настройки форм аргументов отчетов.....	248
20.5.7	Добавление в отчет информации из связанных с потомками объектов.....	249
20.5.8	Формирование строк отчета с использованием информации из различных объектов БД. Создание отчета с источником данных.....	250
20.6	Резюме	255
20.7	Отчеты. Учебное задание	256
20.8	Отчеты. Вопросы для самопроверки.....	256
21	БИЗНЕС – ПРАВИЛА.....	257
21.1	Бизнес-правило типа «Действие».....	258
21.2	Бизнес-правило типа «Соответствие».....	259
21.3	Бизнес-правило типа «Значение»	261

21.4	Резюме	263
22 ДОКУМЕНТООБОРОТ. ОБЩАЯ ИНФОРМАЦИЯ.....		264
22.1	Свободное сообщение.....	264
22.2	Добавление и настройка почтовых папок	267
22.3	Временная фильтрация сообщений.....	269
22.4	Предопределенный процесс	270
22.5	Работа с сообщениями. Учебное задание	270
22.6	Резюме	271
22.7	Работа с сообщениями. Вопросы для самопроверки.....	271
23 СОЗДАНИЕ ПРЕДОПРЕДЕЛЕННЫХ ПРОЦЕССОВ.....		273
23.1	Создание элементов шаблона.....	273
23.2	Определение возможных исполнителей этапов	274
23.2.1	Дополнительная информация по назначению исполнителей	274
23.3	Отключение вкладок «Сообщение» и «Документы». Работа без открытия окна сообщения. Хранение сообщения начала работы	276
23.4	Создание документов шаблона. Привяки документов шаблона к этапам работы	276
23.5	Формы этапов работы и переходов.....	277
23.5.1	Формы переходов	279
23.6	Просмотр формы задачи без открытия входящего сообщения в отдельном окне	281
23.7	Автоматическое завершение работы.....	282
23.8	Проверка шаблона.....	282
23.9	Различные способы запуска работы.....	282
23.10	Выполнение работы	283
23.11	Отказ от выполнения задачи.....	284
23.12	Сервер автоматических этапов	285
23.13	Проверка условий выполнения переходов.....	288
23.14	Автоматические переходы.....	289
23.15	Переменные Lotsia Workflow. Общая информация	290
23.15.1	Определение исполнителей с использованием переменных	292
23.15.2	Дополнительные (пользовательские) кнопки в формах задач Workflow	293
23.16	Действия в шаблонах работ. Общий порядок использования.....	294
23.16.1	Запоминание текущего исполнителя задачи. Использование действий для назначения исполнителей.....	297
23.16.2	Формирование сообщения задачи.....	301

23.16.3	Управление стандартными кнопками и вкладками в форме задачи..	303
23.16.4	Скрытие задач	306
23.16.5	Использование действий над объектами в Workflow	307
23.16.5.1	Регистрация согласуемого документа	308
23.17	Использование действий при работе с документами шаблона	309
23.18	Добавление прочих элементов шаблона	310
23.18.1	Добавление информационной рассылки.....	310
23.18.2	Добавление вложенных и внешних работ	311
23.18.2.1	Добавление внешних работ	312
23.18.2.2	Добавление вложенных работ	313
23.18.2.2.1	Начало и завершение вложенных работ.....	314
23.18.3	Параллельное выполнение этапов работ (разветвление)	314
23.18.3.1	Настройки разветвления	315
23.18.3.2	Особенности настройки объединителя	317
23.19	Замечания по использованию локальных и глобальных переменных ..	318
23.20	Массивы. Использование и некоторые приемы обработки.....	319
23.20.1	Использование массивов для хранения списков значений	320
23.20.2	Создание форм массивов.....	321
23.20.3	Разделение и объединение массивов. Пример настройки шаблона ..	325
23.20.4	Преобразование массивов в одиночное значение	328
23.20.5	Преобразование строки с разделителями в массив	329
23.21	Резюме	329
23.22	Шаблоны работ. Учебное задание.....	331
23.23	Шаблоны работ. Вопросы для самопроверки	331
24ПРИСВОЕНИЕ ЗНАЧЕНИЙ РАЗЛИЧНЫМ ПАРАМЕТРАМ И ПЕРЕМЕННЫМ – ИМЕЮЩИЕСЯ ВОЗМОЖНОСТИ		333
24.1	Значения по умолчанию	333
24.2	Ручной ввод значений. Формирование и использование списков значений	333
24.2.1	Динамическое формирование списка возможных значений переменной для формы Workflow.....	334
24.3	Способы автоматической обработки значений	335
24.3.1	Использование действий над объектами	335
24.3.2	Использование действий над переменными.....	336
24.3.3	Использования SQL-запросов	337
24.3.3.1	Применение SQL-запросов в шаблонах работ.....	337
24.3.3.2	Использование SQL-запросов в выражениях	337
24.3.3.3	Использование SQL-запросов в действиях над объектами	338

24.3.3.4 Другие случаи использования SQL-запросов	338
---	-----

25СОЗДАНИЕ РАБОЧИХ ОКОН ДЛЯ КОНЕЧНОГО ПОЛЬЗОВАТЕЛЯ. ТАБЛИЧНЫЙ ИНТЕРФЕЙС	339
--	------------

26ПРИЛОЖЕНИЯ.....	340
26.1 Файл импорта типов объектов.....	340
26.2 Файл импорта атрибутов	340
26.3 Файл импорта списка возможных значений атрибутов	341
26.4 Файл импорта типов связей.....	341
26.5 Файл импорта входимости типов объектов по типам связей.....	342
26.6 Файл импорта соотношения тип объекта – атрибут	342
26.7 Примерная последовательность создания и развертывания прикладной системы на основе Lotsia PDM PLUS.....	343

1 Введение

В данном пособии описаны основные функциональные возможности системы **Lotsia PDM PLUS версии 5.70, сборка 180312** и приемы её настройки; рассмотрены базовые понятия системы, кратко описана процедура инсталляции системы.

Освоение описанных приемов работы с программой и их использование потребуются от администратора в любом случае, независимо от того, в какой предметной области будет применяться настраиваемая система. Возможные учебные информационные модели рассматриваются по ходу изложения материала.

Пособие предназначено администраторам Lotsia PDM PLUS для использования в качестве дополнительного учебного материала, в том числе, при самостоятельном освоении и администрировании системы Lotsia PDM PLUS, а также для применения в качестве учебного пособия при прохождении обучения по курсу «Администрирование и настройка системы Lotsia PDM PLUS». Подразумевается, что читателю доступна документация на систему Lotsia PDM PLUS, и он обращается к ней по мере необходимости.

Предварительные требования, предъявляемые к администратору Lotsia PDM PLUS следующие:

- знакомство с операционными системами (далее – ОС) семейства Microsoft Windows, знакомство с сетевой операционной системой вашей организации (рекомендуется иметь опыт администрирования сетевых операционных систем).
- знакомство с реляционными системами управления базами данных SQL Database (навыков составления SQL-запросов не требуется, но для выполнения некоторых настроек системы указанные навыки могут быть полезны)
- наличие всех необходимых прав доступа для установки и сопровождения программы.

В ходе работы с данным пособием предусматривается выполнение практических задач по установке и настройке Lotsia PDM PLUS.

Предполагается, что при изучении данного пособия имеется и используется дистрибутивный диск программы Lotsia PDM PLUS, ключ аппаратной защиты и необходимое аппаратное обеспечение, отвечающее требованиям программы Lotsia PDM PLUS к аппаратному обеспечению, изложенными в документации на программу.

2 Назначение и состав Lotsia PDM PLUS

Lotsia PDM PLUS является инструментом реализации различных моделей данных (информационных моделей). Информационная модель должна быть, хотя бы в общих чертах, разработана до начала настройки и использования программы для решения конкретных производственных задач.

С другой стороны, функциональные возможности Lotsia PDM PLUS (в первую очередь возможность гибкой настройки структуры базы данных (далее – БД), графический интерфейс представления данных и процессов) позволяют разрабатывать, визуализировать и тестировать различные модели организации данных и выполнения процессов, выбирая и развивая наилучший вариант, пригодный для дальнейшего промышленного использования.

С примерами реализации некоторых информационных моделей можно ознакомиться, инициализировав БД с различными предопределенными настройками, описания которых находятся в каталоге /Manuals вашего дистрибутивного диска Lotsia PDM PLUS.

С общетехнической точки зрения, Lotsia PDM PLUS предназначена для решения следующих основных задач:

- управления данными – хранения, ввода, изменения, поиска, обработки, обмена и организации данных, а также разграничения доступа к данным. В зависимости от используемой информационной модели, под данными могут пониматься различные сущности. Общими являются принципы организации данных в системе: использование информационных объектов с атрибутивным наполнением, возможность связывания объектов между собой различным образом (подробнее рассмотрено далее, см. раздел «Базовые понятия программы Lotsia PDM. Структура базы данных»);
- управления документами. Под документами понимаются файлы, созданные в различных приложениях. Документы хранятся связанными с объектами, доступ к документам возможен из Lotsia PDM, с открытием документов в соответствующих приложениях (подробнее рассмотрено далее, см. раздел «Архив»);
- отправки, получения и маршрутизации сообщений. Используется свободный обмен сообщениями между пользователями системы и/или запуск настраиваемых процессов – последовательностей обработки данных и обмена сообщениями. Та часть системы, которая решает данную задачу, называется Lotsia Workflow (подробнее рассмотрено далее, см. раздел «Документооборот. Общая информация»).

С точки зрения прикладного использования, в зависимости от области применения программы и её настройки, выполняемой средствами администратора системы, Lotsia PDM PLUS может, соответственно, обеспечивать выполнение примерно следующего набора функций:

Для конструктора и технолога – ведение информации о составе и конфигурациях изделий на протяжении их жизненных циклов. Организацию хранения и доступа к документации по изделиям. Создание, хранение и выбор различных вариантов конструкции составных частей изделий и технологических процессов их изготовления. Реализацию процессов согласования различной технической документации, внесения изменений с выпуском и согласованием извещений об изменении и т.п.

Для проектировщика – ведение информации о составе проектов на протяжении их жизненных циклов, обеспечение упорядоченного хранения и доступа к проектной документации. Реализацию процессов выдачи заданий и согласования технической документации, обработку замечаний экспертизы и т.п.

Для руководителя – получение сводной информации в требуемом виде, с возможностью её детализации и получения, при необходимости, доступа непосредственно к документу. Получение, рассмотрение и отправку на исполнение документов, постановку на контроль и управление исполнением документов.

Для сотрудника канцелярии – регистрацию и упорядоченное хранение информации о корреспонденции и внутренней документации предприятия. Обработку, контроль исполнения, рассылку, согласование различных документов и т.п.

Благодаря имеющимся возможностям гибкой настройки, на практике Lotsia PDM PLUS успешно применяется в совершенной различных предметных областях, таких как:

- Проектные организации
- Нефтегазовая отрасль
- Горнодобыча и переработка
- Строительство
- Машиностроение
- Приборостроение
- Транспортное машиностроение и автомобилестроение
- Авиакосмос
- Судостроение
- Приборостроение
- Швейная промышленность
- Деревообработка
- Электротехника и электроника
- Энергетика
- Холдинги
- Биотехнологии
- Химическая
- Масс-медиа

Возможная область применения Lotsia PDM PLUS никоим образом не ограничивается данным списком, который постоянно расширяется.

Система построена в архитектуре «клиент-сервер». Lotsia PDM PLUS включает в себя:

- клиентскую часть – программу, запускаемую на рабочих местах пользователей;
- серверную часть – базу данных (БД), расположенную на сервере;

- хранилище документов – файловый сервер (в БД хранятся ссылки на документы);
- утилиту CreateDB, используемую администратором для первоначальной инициализации БД и выполнения её последующих обновлений. Данная утилита также может использоваться для выгрузки базы данных и загрузки выгруженных таблиц в пустую базу данных (применяется, например, для миграции с одной СУБД на другую);
- сервер автоматических этапов – программа для выполнения автоматических элементов в процессах Lotsia Workflow (см. раздел Сервер автоматических этапов). Для одной БД запускается один экземпляр сервера автоматических этапов;
- программу BeInFlow, которая может запускаться на рабочих местах пользователей и информировать их о наличии не прочтенных сообщений. BeInFlow не расходует лицензий Lotsia PDM PLUS, позволяя таким образом экономить указанные лицензии;
- утилиту автоматического обновления. При условии выполнения необходимых настроек, обеспечивает обновление клиентской части программы Lotsia PDM PLUS в автоматическом режиме, а также дает возможность проверять и синхронизировать файлы в рабочем каталоге Lotsia PDM PLUS и его подкаталогах на рабочих местах (группах рабочих мест) пользователей с файлами, находящимися на общем сетевом ресурсе (ресурсах).

Функционирование БД и подключение к ней клиентских программ обеспечивает СУБД (в состав Lotsia PDM PLUS не входит^{*}). Lotsia PDM PLUS может работать с различными СУБД, их перечень и особенности использования с Lotsia PDM PLUS приведены в документации по Lotsia PDM PLUS. На рабочих местах пользователей, при подключении к БД, также запускается клиентская часть соответствующей СУБД.

Имеются дополнительные средства, не входящие в стандартную поставку Lotsia PDM PLUS, расширяющие базовые возможности программы. К ним могут быть отнесены:

- модули интеграции с различными САПР и другими внешними приложениями;
- модуль Twain-сканер, обеспечивающий сканирование документов с использованием Twain-совместимых сканеров непосредственно в архив Lotsia PDM PLUS;
- конвертеры данных;
- модуль репликации (используется, например, при распределенной (филиальной) модели хранения и обработки данных);
- другие средства, в том числе – продукт «Lotsia WEB», обеспечивающий работу с базой данных Lotsia PDM PLUS через стандартный WEB – браузер, без установки на рабочем месте пользователя клиентской программы.

^{*} Для создания и использования однопользовательской БД под управлением Sybase SQL Anywhere достаточно содержимого дистрибутивного диска Lotsia PDM PLUS (см. Создание пустой базы данных Sybase SQL Anywhere).

Дополнительные средства для Lotsia PDM PLUS в данном пособии не рассматриваются, информацию о них можно получить на сайте www.lotsia.com либо обратившись в ООО «Лотсия Софтвэа».

2.1 Резюме

Lotsia PDM PLUS используется для реализации и разработки различных информационных моделей, предназначена для управления данными, документами, процессами, а также обмена информацией

Программа построена в архитектуре «клиент-сервер».

Имеются дополнительные средства, не входящие в стандартную поставку Lotsia PDM PLUS, расширяющие возможности программы

3 Установка программы, подготовка к работе и начало работы

Последовательность установки программы следующая:

- Установка Lotsia PDM – описано далее – см. раздел «Установка программы».
- Установка драйвера устройства аппаратной защиты и самого устройства аппаратной защиты – описано далее – см. раздел «Установка аппаратной защиты». Данный пункт может быть выполнен заранее, до п.1
- Создание пустой базы данных средствами Вашей СУБД (для Sybase SQL Anywhere описано далее – см. раздел «Создание пустой базы данных Sybase SQL Anywhere», настройка ODBC – описано далее – см. раздел «Настройка ODBC при работе с Sybase SQL Anywhere »). Данный пункт может быть выполнен заранее, до пп.1, 2
- Инициализация базы данных – описано далее – см. раздел «Инициализация базы данных».
- Запуск программы – описано далее – см. раздел «Запуск программы» и начало работы с ней.
- Создание защищенного хранилища документов – будет описано далее, см. раздел «Архив документов». Запуск сервера автоматических этапов – будет описано далее, см. раздел «Сервер автоматических этапов». Данный пункт можно осуществить позднее, по мере необходимости.

3.1 Установка программы

Для установки Lotsia PDM PLUS зарегистрируйтесь в операционной системе с полными административными правами.

Для установки системы на рабочем месте Администратора:

Запустите файл «LPDMXXX.msi» (расположен в корневом каталоге Вашего дистрибутивного диска).

Введите Ваше имя, название организации, серийный номер, каталог для установки программы. Серийный номер находится на регистрационной карточке, входящей в комплект поставки.

Выберите вид установки. Вид установки «Для пользователя» отличается от «Для администратора» отсутствием утилиты CreateDB.

Для того чтобы установить компоненты Sybase SQL Anywhere следует использовать вид установки «Выборочная», включить установку Sybase SQL Anywhere (Рисунок 2).

Таким образом, для того, чтобы установить на рабочее место компоненты Sybase SQL Anywhere, CreateDB и примеры настроек, используйте вид установки «Выборочная» и выберите все компоненты.

В случае установки многопользовательской Lotsia PDM PLUS Вам дополнительно потребуется установить на рабочих местах клиентскую часть Вашей СУБД и настроить сервер СУБД. В этом случае Sybase SQL Anywhere из поставки Lotsia PDM PLUS можно использовать как дополнительный инструмент, например, для создания локальных БД в учебно-ознакомительных целях.

Установка программы на рабочих местах пользователей может выполняться централизованно, при помощи групповой политики Microsoft Active Directory. При автоматической установке приложений используются параметры по умолчанию: устанавливается только приложение Lotsia PDM PLUS (то есть, клиентская часть программы) в папку по умолчанию.

Независимо от способа установки (централизованно или вручную), для установки программы на рабочих местах пользователей удобно использовать административный образ дистрибутива программы, который, в отличие от полного дистрибутива, содержит только те компоненты Lotsia PDM PLUS, которые требуются конечному пользователю. Кроме того, при создании административного образа программы создается папка с распакованными файлами программы, которые можно использовать для автоматического обновления клиентской части встроенными средствами Lotsia PDM PLUS (будет рассмотрено далее, см. п. 3.7.2)

- Для создания административного образа необходимо выполнить следующее:
- Создать папку, куда будет помещен создаваемый административный образ;
- Выполнить команду `msiexec /a LPDMXXX.msi`, при этом для `LPDMXXX.msi` необходимо указать полный путь;
- При выполнении создания административного образа, указать папку, созданную на первом шаге (Рисунок 1).

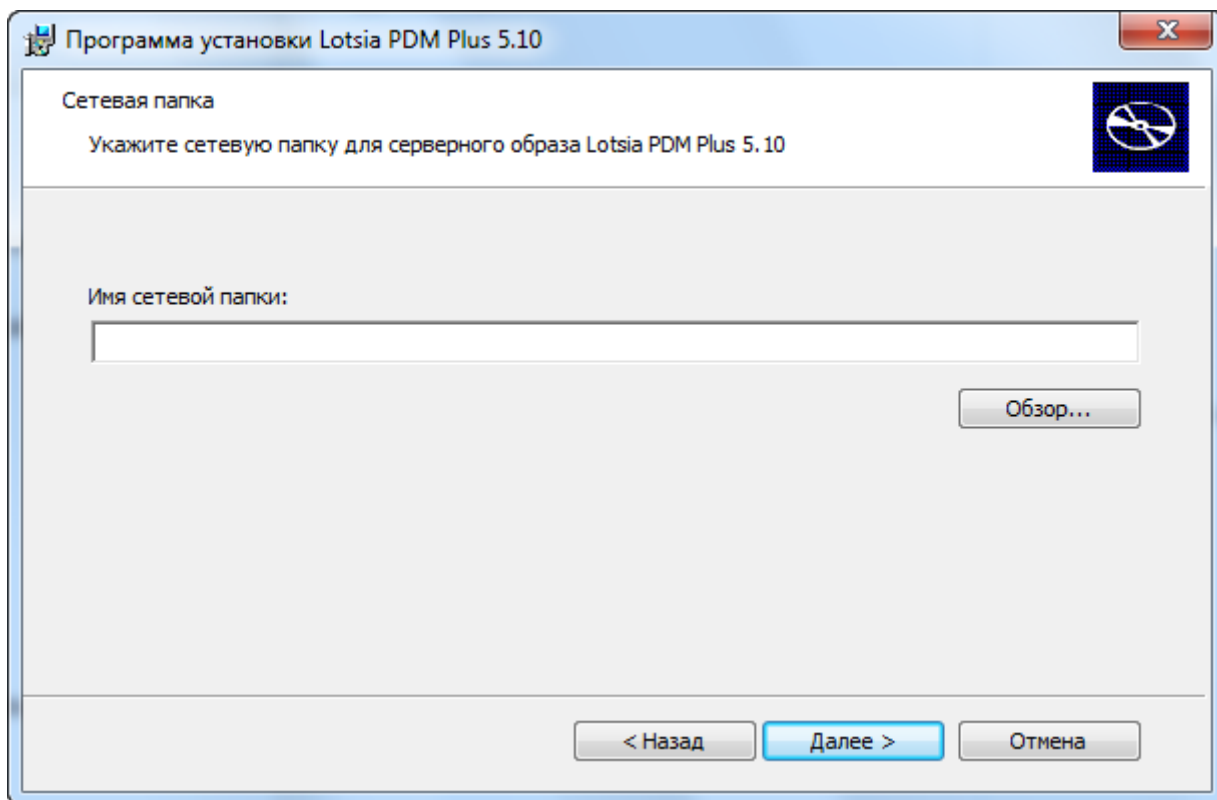


Рисунок 1 Указание папки для создания административного образа

Например, если вы создали папку `C:\1\A` для административного образа, а файл `LPDMXXX.msi` (вместо XXX в именах файлов указывается номер версии) расположен в папке `C:\1`, команда для создания административного образа будет следующей:

```
msiexec /a c:\1\LPDMXXX.msi
```

По её выполнения, в папке «А» будет создан файл LPDMXXX.msi – административный образ, он будет меньшего размера, чем исходный файл LPDMXXX.msi, и папка \LotsiaSoft\PartY, содержащая распакованные файлы.

3.2 Установка аппаратной защиты

Для учета количества используемых лицензий, в комплект поставки программы Lotsia PDM PLUS включается устройство аппаратной защиты – электронный ключ Guardant компании «Актив», подключаемый к USB-порту.

В зависимости от версии (однопользовательская или сетевая), программа может использовать разные типы ключей аппаратной защиты.

До установки ключа аппаратной защиты в USB-порт следует установить драйвер Guardant. Драйвер необходимо устанавливать только на тот компьютер, к которому непосредственно будет подключен электронный ключ. Драйвера Guardant расположены в папке \Guardant\Drivers0020дистрибутивного диска Lotsia PDM PLUS.

Для установки драйвера запустите файл setup.exe и следуйте указаниям программы установки. По окончании установки драйвера установите ключ в USB-порт.

Для работы с сетевым ключом необходимо установить сервер Guardant Net на том же компьютере, на котором установлен ключ.

Более подробно установка аппаратной защиты описана в разделе «Установка аппаратной защиты» документа «Общая часть руководств», расположенного в каталоге /Manuals Вашего дистрибутивного диска.

3.3 Создание пустой базы данных Sybase SQL Anywhere с использованием средств, входящих в стандартную поставку Lotsia PDM PLUS

Для создания пустой БД запустите файл Crsawdb.cmd, расположенный в каталоге с программой (обычно это \Program Files\LotsiaSoft\PartY\) с параметром – именем БД. Например:

C:\Program Files\LotsiaSoft\PartY\Crsawdb.cmd BD_file_name
BD_file_name – имя создаваемой БД.

Созданную БД можно перенести в отдельный каталог, либо сразу, при создании указать полный путь до создаваемого файла.

Созданная БД может использоваться как локальная БД. При работе с локальной БД никаких предварительных действий по её запуску выполнять не требуется, но возможно подключение к ней только с одного рабочего места одновременно. Для создания и использования локальной БД достаточно установить на Вашем рабочем месте версию режима исполнения (run-time) SQL сервера Sybase SQL Anywhere из стандартной поставки Lotsia PDM PLUS – см. рисунок 2.

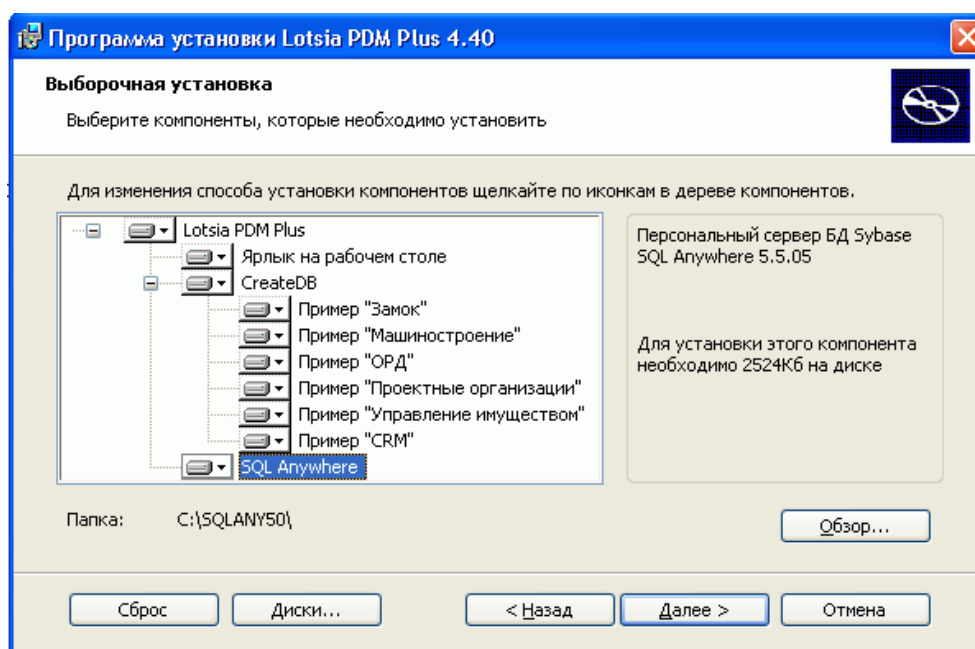


Рисунок 2 Выбор компонента Sybase SQL Anywhere при установке Lotsia PDM PLUS

Та же самая БД может быть использована как серверная, для одновременной работы некоторого количества пользователей. При этом требуется установка и запуск компонентов СУБД, не входящих в комплект поставки Lotsia PDM PLUS (то есть, клиентской и серверной частей СУБД). Интерфейс Lotsia PDM и порядок работы при подключении к локальной или к серверной БД не различаются.

3.4 Настройка ODBC при работе с Sybase SQL Anywhere

Источник ODBC используется для подключения к БД Sybase SQL Anywhere (при использовании других СУБД создание источника может не потребоваться). Источник ODBC создается на рабочем месте, с которого будет осуществляться подключение к данной БД с помощью ODBC.

Отметим, что подключение клиентской программы Lotsia PDM PLUS к БД Sybase SQL Anywhere возможно как с использованием ODBC – источника, так и без него, путем прямого указания имени сервера и имени БД (Рисунок 9). Однако при использовании локальной БД Sybase SQL Anywhere удобнее выполнять с использованием ODBC – источника, (в противном случае потребуется отдельно осуществить запуск БД).

Источник ODBC для данного рабочего места и данной БД можно создать непосредственно при инициализации БД или заранее. Источник создается при помощи приложения odbcad32, которое запускается из панели управления (Control Panel) или **командной строки**.

Для локальной БД Sybase SQL Anywhere создание ODBC-источника выполняется следующим образом:

После запуска приложения odbcad32 открывается окно администрирования ODBC. (см. рисунок 3).

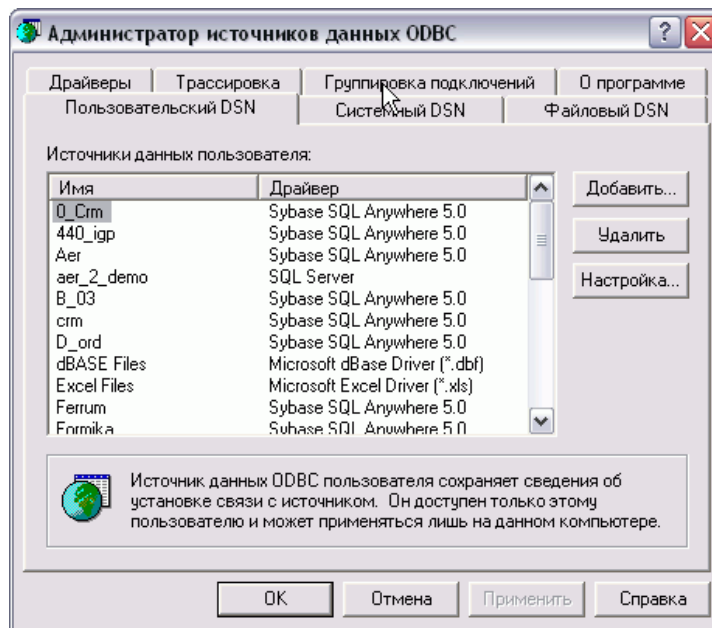


Рисунок 3. Окно администрирования ODBC

Выберите вкладку User DSN или System DSN.

Для создания нового источника ODBC нажмите кнопку «Add» («Создать»). Появится окно со списком драйверов. Выделите в списке «Sybase SQL Anywhere» и нажмите «Готово» («Finish»). Появится окно настройки ODBC (в зависимости от используемой версии Sybase SQL Anywhere вид данного окна может отличаться). (См. рисунок 4).

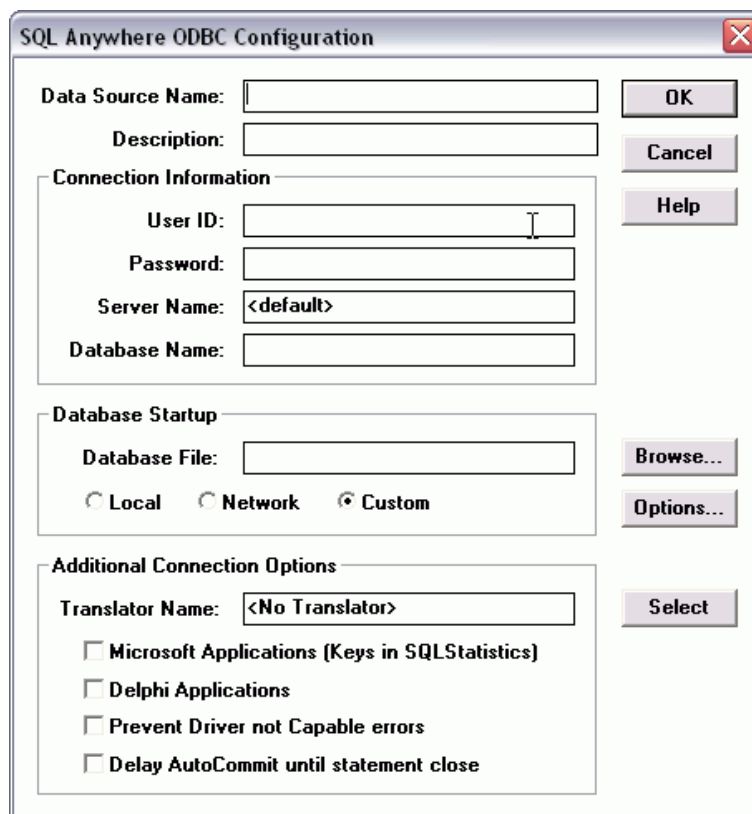


Рисунок 4. Настройка локального источника ODBC

Установите переключатель в положение «Local» (или «Custom») и задайте в поле «Database File» путь и имя файла базы данных (нажмите «Browse...» и выберите или введите), например, «C:\Database\ BD_file_name».

В поле «Data Source Name» появится имя источника ODBC («BD_file_name»). Его можно изменить ручным вводом, если это требуется.

Настройте параметры запуска, для этого нажмите «Options». Откроется окно «Startup Options» (см. рисунок 5).

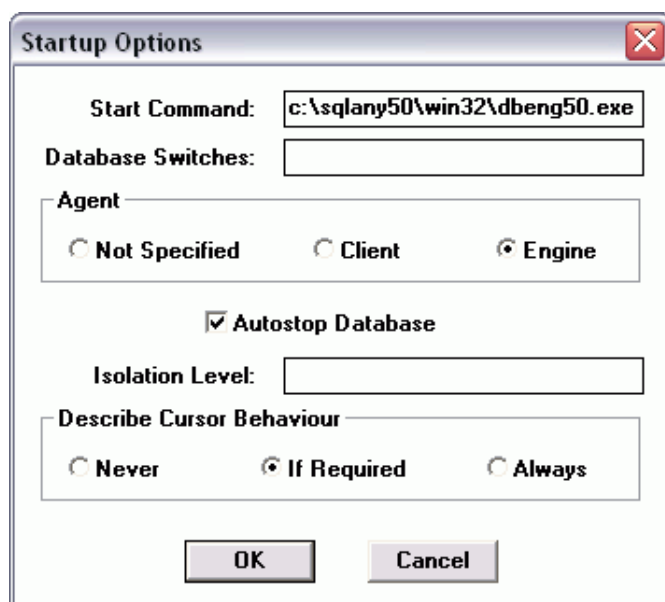


Рисунок 5 Настройка параметров запуска для локальной БД

Установите переключатели как на рисунке, сохраните.

Последовательность создания источников ODBC для различных случаев описана в разделе «Настройка ODBC-источников» документа «Общая часть руководств», расположенного в каталоге /Manuals Вашего дистрибутивного диска.

Создайте ODBC-источник для подключения к созданной БД.

3.5 Инициализация базы данных

Запустите программу CreateDB.

Выберите опцию «Создание или пересоздание БД». Далее последовательно следуйте инструкциям программы.

При использовании Sybase SQL Anywhere: ведите тип СУБД – Sybase SQL Anywhere, имя созданного ранее ODBC из списка, имя и пароль администратора – dba, sql.

Перед запуском процедуры инициализации БД вам будет предложено загрузить в инициализированную БД одну из predefined настроек (см. рисунок 6).

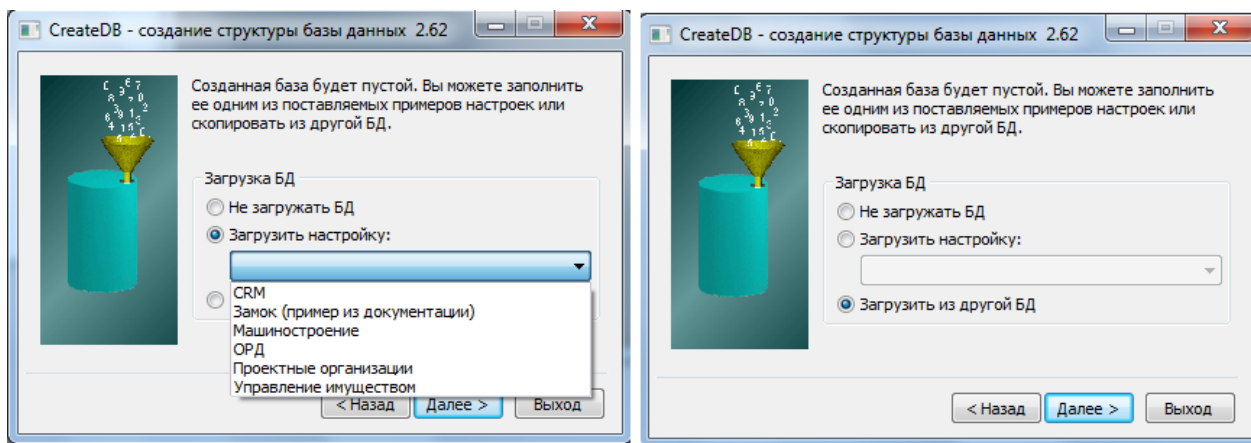


Рисунок 6. Выбор predeterminedной настройки для загрузки в инициализированную БД (слева), загрузка данных из другой БД (справа)

Для дальнейшей работы с настоящим пособием вам понадобится пустая БД, поэтому выберите «<Не загружать БД>».

Ознакомиться с имеющимися predeterminedными настройками рекомендуется самостоятельно, проинициализировав соответствующие БД и используя описания указанных настроек, имеющиеся в каталоге \manuals Вашего дистрибутивного диска.

3.6 Запуск программы

В группе «Программы» меню «Пуск» выберите Lotsia PDM PLUS – Lotsia PDM.

В появившемся окне с параметрами подключения нажмите кнопку «Параметры».

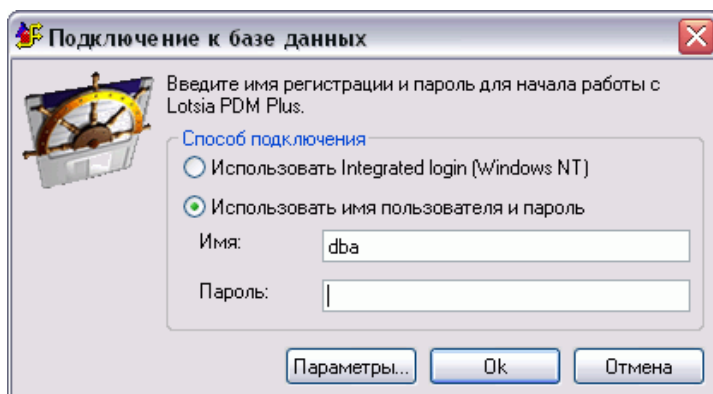


Рисунок 7 Окно подключения к БД

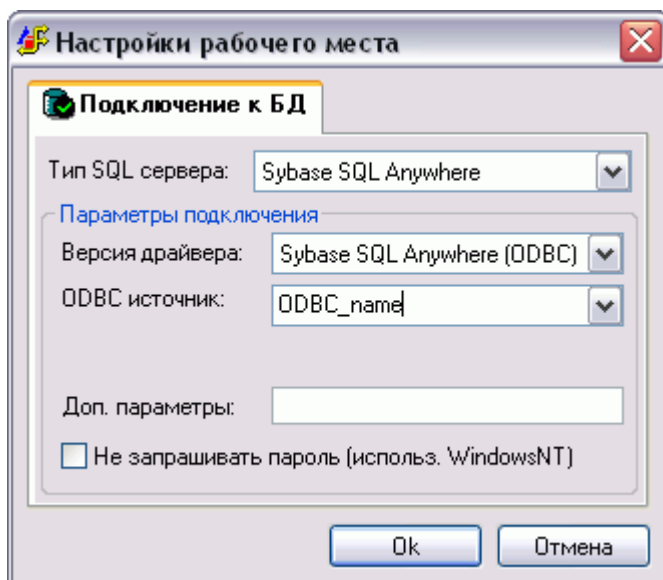


Рисунок 8 Окно параметров подключения при использовании ODBC-источника

Укажите тип SQL сервера «Sybase SQL Anywhere» и ваш ODBC-источник (вместо «ODBC_name»). Введите имя (dba) и пароль (sql) Администратора (Рисунок 7).

Если ODBC источник на данном рабочем месте не создавался, можно выбрать требуемую версию драйвера (например, при использовании run-time версии SQL сервера Sybase SQL Anywhere из стандартной поставки: Sybase SQL Anywhere 5.0) и указать параметры подключения к БД (Рисунок 9).

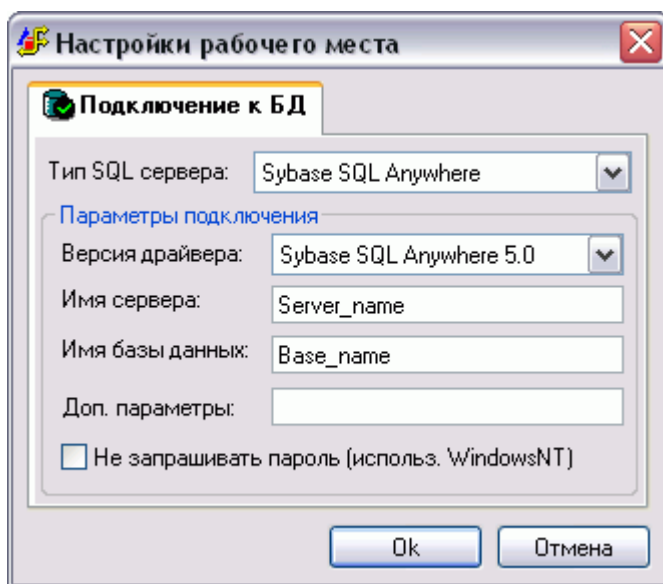


Рисунок 9 Окно параметров подключения без использования ODBC – источника

Вместо «Server_name» и «Base_name» укажите, соответственно, имя Вашего сервера и базы данных. При использовании локальной БД Sybase SQL Anywhere и в качестве имени сервера, и в качестве имени базы можно указать имя файла БД без расширения.

Перед подключением к любой БД она должна предварительно быть запущена. При использовании локальной БД Sybase SQL Anywhere и подключении к ней через ODBC-источник, запуск БД происходит автоматически, то есть, никаких предварительных действий для запуска такой БД производить не требуется. Если подключение к той же БД осуществляется без ODBC-источника, необходимо предварительно запустить БД,

например, выполнив команду dbeng50 Base_name.db (вместо Base_name.db укажите полное имя файла Вашей БД с расширением).

В случае получения сообщения о несоответствии версий компонентов БД (Рисунок 10) выполните обновление БД.

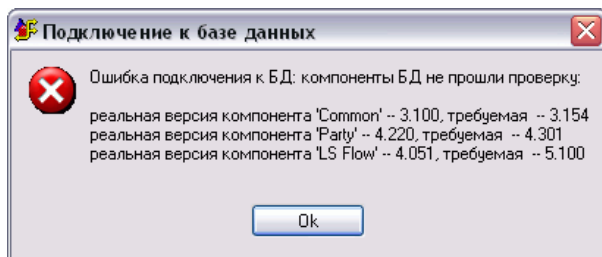


Рисунок 10 Сообщение о несоответствии версий компонентов БД. Пример

3.7 Обновление версий Lotsia PDM PLUS

Lotsia PDM PLUS постоянно развивается, поэтому обновление программы можно рассматривать как штатную процедуру её администрирования. Файлы обновлений доступны для скачивания с сервера www.lotsia.com.

В общем случае обновление может включать в себя следующие файлы (вместо XXX в именах файлов указывается номер версии, вместо nnnn.nn – сборка (build) соответствующего программного обеспечения):

1. Readme.txt – содержит необходимую информацию о назначении и использовании данного обновления;
2. pr_XXXdoc.exe – содержит документ pr_XXX.doc – краткое описание отличий данной версии и сборки программного обеспечения. Рекомендуется изучить содержание данного документа перед принятием решения об обновлении программного обеспечения;
3. LPDMXXX.nnnn.nn.msp – обновление Lotsia PDM PLUS (см. п. 3.7.1);

В пакет обновлений системы могут входить и другие файлы, назначение и порядок использования которых описаны в соответствующих им текстовых файлах. В последнем случае следует поступать согласно указаниям, содержащимся в текстовых файлах описаний.

Очередные обновления, как правило, содержат в себе и предыдущие обновления, могут устанавливаться на предыдущие версии Lotsia PDM PLUS, согласно соответствующим файлам описаний. Установка непрерывной последовательности всех обновлений не требуется. Решение об установке обновления принимается после изучения соответствующего документа pr_XXX.doc.

В отдельных случаях возможен выпуск обновлений, предназначенных для установки только на указанную сборку Lotsia PDM PLUS, необходимая информация о таких обновлениях содержится в соответствующих текстовых файлах, поставляемых вместе с обновлениями.

В большинстве случаев требуется выполнить обновление клиентской и серверной частей программы. Рассмотрим данную процедуру подробнее.

3.7.1 Обновление клиентской части программы

Обновление (как и установка) клиентской части программы на рабочих местах может выполняться централизованно, при помощи групповой политики Microsoft Active Directory (данный метод здесь не рассматриваем). Можно осуществить обновление

программы и «вручную», путем запуска соответствующего файла LPDMXXX.nnnn.nn.msp.

Кроме того, Lotsia PDM PLUS имеет собственные средства для автоматического обновления клиентской программы.

Рекомендуется следующий порядок обновления:

- Программа на рабочем месте администратора обновляется «вручную», путем запуска LPDMXXX.nnnn.nn.msp. Отметим, что при обновлении рабочего места администратора, в отличие от рабочих мест конечных пользователей, обычно требуется и обновление утилиты CreateDB;
- Администратор выполняет обновление административного образа (о создании административного образа см. п. 3.1) путем выполнения команды:

```
msiexec /a LPDMXXX.msi /p LPDMXXX.nnnn.nn.msp
```

при этом: LPDMXXX.msi – полное имя необновленного административного образа, LPDMXXX.nnnn.nn.msp – полное имя пакета обновлений.

Например, если административный образ находится в папке c:\1\A, а пакет обновлений – в папке c:\1\, команда для обновления административного образа будет такая:

```
msiexec /a c:\1\A\LPDMXXX.msi /p c:\1\ LPDMXXX.nnnn.nn.msp
```

При выполнении данной команды обновляется файл административного образа и распакованные файлы программы, находящиеся в папке c:\1\A \LotsiaSoft\PartY (если в ходе выполнения команды указана папка для административного образа – c:\1\A\).

Отметим, что для выполнения указанной команды необходимо, чтобы указанный файл LPDMXXX.msi был именно административным образом, а не дистрибутивом программы, а также, чтобы данный административный образ не был ранее уже обновлен до той же или более поздней сборки. В отдельных случаях (если это оговорено в сопровождающих данный пакет обновлений текстовых файлах) для данного пакета обновлений может потребоваться использование административного образа конкретной сборки программы.

Далее, в зависимости от принятого на данном предприятии порядка, обновление программного обеспечения конечных пользователей может осуществляться либо с использованием групповой политики Microsoft Active Directory (здесь не рассматриваем), либо с использованием собственных средств Lotsia PDM PLUS.

Отметим, что если обновленная клиентская часть программы не соответствует используемой версии серверной части (то есть, требует более поздних версий компонентов БД, к которой вы подключаетесь), то работа с данной версией серверной части с помощью данной клиентской программы невозможна, о чем будет выдано сообщение (см. рисунок 10). В этом случае следует выполнить обновление серверной части программы (будет рассмотрено далее, см. п. 3.7.3). Если же, по каким-либо причинам, обновление серверной части программы проводить не представляется возможным, следует установить соответствующую (более старую) сборку клиентской программы (например, путем использования распакованных файлов соответствующей сборки программы, либо через полное удаление новой сборки и последующую установку предыдущей сборки).

Итак, при использовании собственных средств Lotsia PDM PLUS для автоматического обновления клиентской программы на рабочих местах пользователей, необходимо:

- обновить клиентскую часть программы на рабочем месте администратора в «ручном» режиме, путем запуска LPDMXXX.nnnn.nn.msp;
- при необходимости обновить серверную часть программы (БД) (будет рассмотрено далее, см. п.3.7.3)
- с использованием обновленной клиентской программы подключиться к обновленной БД и выполнить необходимые настройки автоматического обновления (рассмотрено далее, см. п.3.7.2).

После выполнения данной процедуры, по мере подключения пользователей к БД, на их рабочих местах пользователей обновление клиентской части программы будет происходить в автоматическом режиме, в соответствии со сделанными настройками.

3.7.2 Настройка автоматического обновления клиентской части программы

Для настройки параметров автоматического обновления выберите в Главном меню пункт «Администрирование» → «Общие параметры БД». В открывшемся одноименном Окне перейдите на вкладку «Обновление».

На данной вкладке осуществляется настройка параметров автоматического обновления, в том числе:

Указывается общедоступная папка с обновлениями, куда распаковываются файлы обновлений, устанавливается номер версии, с которой следует сравнить версию клиентской части программы, установленной на рабочем месте пользователя.

При подключении очередного пользователя к БД сравнивается номер текущей версии программы, установленной на данном рабочем месте, и номер версии, записанной в БД. При несовпадении указанных номеров версий производится копирование файлов программы из папки с обновлениями на данное рабочее место, обновляя, таким образом, версию клиентской программы, установленной на данном рабочем месте.

Возможно и рекомендуется обновление с использованием Windows Installer. Для такого обновления файлы дистрибутива (*.msi, *.msp) или административного образа должны быть помещены в общедоступные папки, указанные в соответствующих полях в области «Сеть (msi/msp)».

С целью оптимизации сетевого трафика можно использовать различные общедоступные папки с обновлениями для различных групп рабочих мест.

Ознакомиться с порядком настройки автоматического обновления и назначением устанавливаемых параметров подробнее можно, воспользовавшись контекстной справкой, либо обратившись к документации по системе Lotsia PDM PLUS.

3.7.3 Обновление серверной части программы

Под обновлением серверной части программы в данном случае понимается обновление базы данных.

Перед обновлением БД необходимо выполнить её резервное копирование. Пренебрежение данным требованием чревато безвозвратной потерей всех имеющихся данных, кроме того, в случае необходимости возврата к старой версии серверной части программы, единственная возможность для этого – использование резервной копии БД.

Обновление базы данных осуществляется с помощью утилиты CreateDB.

Чтобы обеспечить обновление БД до требуемых значений номеров версий её компонент, используемая для обновления БД утилита CreateDB, должна быть предварительно обновлена до требуемой версии.

Обновление CreateDB осуществляется при обновлении программы на рабочем месте администратора, при запуске LPDMXXX.nnnn.nn.msp.

Для обновления БД необходимо запустить обновленную утилиту CreateDB, в открывшемся окне выбрать «Обновление существующей БД», нажимать «Далее>» и следовать инструкциям программы. Имя пользователя, под которым выполняется обновление БД: LSDBO, пароль по умолчанию: sql.

Подробнее порядок обновления базы данных описан в документации по системе Lotsia PDM PLUS.

3.8 Резюме

Для установки программы следует запустить «LPDMXXX.msi» (вместо XXX в именах файлов указывается номер версии), расположенный в корневом каталоге Вашего дистрибутивного диска. Для установки и обновления программы на рабочих местах пользователей можно использовать административный образ дистрибутива.

При установке программы можно установить и компоненты Sybase SQL Anywhere, достаточные для создания и использования локальной БД.

Для работы с программой необходим ключ аппаратной защиты. Драйвера ключа устанавливаются до подсоединения ключа к порту компьютера.

Необходимо создать пустую БД, для чего использовать средства СУБД. Возможно создание пустой БД Sybase SQL Anywhere средствами, входящими в комплект поставки. В любом случае, до начала работы, для созданной пустой БД необходимо выполнить процедуру инициализации БД.

При использовании Sybase SQL Anywhere необходимо выполнить настройку ODBC-источника.

При подключении к БД возможно появление сообщения о несоответствии версий ПО и БД. В этом случае следует обновить утилиту CreateDB, а затем с её помощью обновить БД.

3.9 Установка программы. Вопросы для самопроверки

- Чем отличается установка рабочего места Lotsia PDM для администратора и пользователя?
- Требуется ли установка драйвера электронного ключа на каждом рабочем месте Lotsia PDM PLUS, почему?
- Как создать пустую БД Sybase и подключиться к ней, располагая только дистрибутивным диском Lotsia PDM PLUS?
- Как создать БД с предопределенной настройкой «Замок»?
- Что следует сделать, получив при попытке подключения к БД сообщение: «Компоненты БД не прошли проверку: Реальная версия компонента 'Party--..., требуемая....'...»?
- Как обновить клиентскую и серверную часть Lotsia PDM PLUS?

4 Базовые понятия программы Lotsia PDM. Структура базы данных

Здесь кратко рассматриваются основные понятия и термины, используемые в дальнейшем. После создания в учебной базе данных минимально необходимых элементов структуры, можно будет перейти к созданию экземпляров объектов и построению проектов.

4.1 Информационный объект, проекты

Информационный объект (далее «**Объект**», или «Экземпляр объекта») в Lotsia PDM – это структурированная единица хранения информации. Объект Lotsia PDM может описывать предметы, действия или явления окружающего мира (или соответствовать им).

Например, объектом Lotsia PDM может быть: Сборочная единица XXXX.XXXXXX, Деталь YYYYY.YYYYYY, Сборочный чертеж двигателя XXX, Письмо №367 от 21.03.02, сотрудник Иванов А.А., Отдел Главного Технолога, отпуск сотрудника, наработка транспортного средства, ремонт единицы оборудования и т.п.

Объекты могут связываться друг с другом связями различных типов. Одной из наиболее востребованных возможностей является возможность построения и отображения на экране т.н. «деревьев» (Рисунок 11).

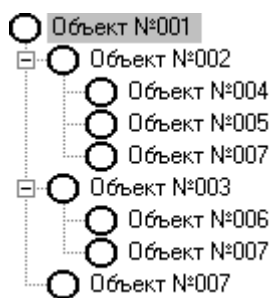


Рисунок 11. Дерево

«Дерево» или «Проект» представляет собой произвольное количество объектов, связанных таким образом, что имеется т.н. «**корневой**» объект (Объект №001, см. рисунок 11), с ним связано некоторое количество **объектов первого уровня** (в данном примере – Объект №002, Объект №003, Объект №007). С объектами первого уровня могут быть аналогичным образом связаны другие объекты, количество уровней не ограничивается.

При построении деревьев говорят, что объекты последующего уровня – «**потомки**», «**входят**» в объекты предыдущего уровня – в «**родителей**». Например, Объект №004 входит в Объект №002, то есть, используется понятие «**входимость**». Объекты могут входить (то есть, быть связаны т.н. «подчиненной» связью) в один или в **несколько** других объектов (проектов).

Объект №001 («**корневой**») в данном контексте не входит в объект более высокого уровня (то есть, родители у данного объекта отсутствуют вообще, или в данном открытом окне не отображаются).

Если объект входит в несколько объектов, то есть, имеет более одного родителя (например, Объект №007, см. рисунок 11), говорят о «**множественной**» **входимости** данного объекта. Если объект имеет только одного родителя (Объект №002, Объект №003, Объект №004, Объект №005), то **входимость** данного объекта «**единичная**».

В зависимости от реализуемой информационной модели, для БД может быть запрещена (так по умолчанию) либо разрешена **многократная** входимость объектов. Многократная входимость означает возможность включать один и тот же объект в одного родителя более одного раза.

Создаваемые экземпляры объектов имеют набор обязательных или возможных характеристик (свойств). Такой набор свойств называется «**Тип объекта**». Тип объекта обязательно и однозначно определяется при создании каждого экземпляра объекта, то есть, сразу создается объект конкретного типа. Таким образом, еще до начала создания экземпляров объектов необходимо определить в БД возможные типы объектов. При создании (настройке) типов объектов определяются и параметры, описывающие данный тип объекта. Иначе говоря, создаются (настраиваются) **элементы структуры БД**.

Каждый экземпляр объекта имеет **описание** – текстовую строку, длиной до 255 символов. Данное свойство изначально заложено в программе, настройки не требует. Объекты и экземпляры подчиненных связей могут содержать дополнительные свойства – **атрибуты**. Атрибуты представляют собой поля, в которых хранятся значения отдельных параметров объекта или экземпляра связи.

Возможность (и обязательность) использования различных атрибутов определяется:

- для данного экземпляра объекта – типом данного объекта
- для данного экземпляра подчиненной связи – типом объекта – потомка

Указанная возможность настраивается администратором.

Атрибут, относящийся к объекту, называется «Атрибут объекта», а атрибут, относящийся к связи – «Атрибут проекта».

К экземпляру объекта может прикрепляться документ. Сам документ (файл) объектом Lotsia PDM не является.

Экземпляры объектов, экземпляры связей между объектами, документы архива, элементы структуры БД и другие сущности системы имеют уникальные идентификаторы – ID.

Полные определения и другая информация содержится в документации по Lotsia PDM PLUS, обращаться к которой рекомендуется по мере необходимости.

4.2 Элементы структуры БД

К элементам структуры БД относятся:

- Типы объектов
- Атрибуты
- Группы атрибутов
- Типы связей объектов

Элементы структуры БД создаются и настраиваются Администратором* в зависимости от имеющихся потребностей, определяемых используемой информационной моделью. Кроме того, в инициализированной БД имеется некоторое количество

* Здесь и далее под «Администратором» понимается как сам Администратор Lotsia PDM PLUS, так и другие пользователи, которым даны необходимые административные привилегии и обеспечена возможность использования требуемых пунктов меню.

предопределенных элементов структуры БД, использование которых не является обязательным, но для значительного множества предметных областей вполне оправдано.

Не рекомендуется изменять названия предопределенных элементов структуры БД, так как это в дальнейшем может затруднить обмен данными с другими БД, если в этом возникнет потребность.

4.2.1 Связи между объектами в системе Lotsia PDM

При связывании каждой пары экземпляров объектов создается экземпляр связи между двумя объектами. При создании каждого экземпляра связи однозначно определяется основной параметр данной связи – её тип, то есть, сразу создается связь определенного типа (аналогично созданию экземпляра объекта). Типы связей создаются при создании (настройке) элементов структуры БД. Имеется несколько предопределенных типов связей. При создании типа связи, помимо её названия (описания) однозначно определяется её параметр, также называемый «Тип связи»^{**}, который характеризует основные свойства данного типа связи. Этот параметр после сохранения в БД созданного типа связи изменяться не может.

В Lotsia PDM возможны следующие связи (то есть, типы для типов связей) между объектами:

- подчиненные;
- горизонтальные равнозначные;
- горизонтальные направленные;
- специальные предопределенные: календарь, исполнения, старые варианты

Подчиненные связи позволяют объединять объекты в дерево, с визуализацией дерева в окне проекта. Изначально имеется предопределенный тип подчиненной связи «Дерево проектов», дополнительные подчиненные типы связей создаются Администратором.

Горизонтальная равнозначная связь указывает на то, что между данными объектами имеется связь, но относительно друг друга положение связанных объектов равнозначное. Предопределенных равнозначных горизонтальных типов связей изначально в БД нет, все горизонтальные типы связей создаются Администратором. Примером горизонтальной связи может служить связь «Похожие», которая может использоваться для явного связывания объектов, сходных между собой по определенным пользователем критериям.

Горизонтальные направленные связи применяются для того, чтобы указать направление связи между объектами. При создании экземпляра связи такого типа между объектами, указывается направление: «исходящая» или «входящая». Объектом – родителем в данном случае является тот объект, для которого направленная связь является исходящей. Предопределенных направленных горизонтальных типов связей изначально в БД нет, все горизонтальные типы связей создаются Администратором. Примером подобной связи может служить связь «Замена», используемая для отображения возможности замены данного объекта на другой, но не наоборот (см. рисунок 12)

^{**} В таблицах БД хранится соответствующий данному параметру буквенный код, именуемый «Вид связи». Его возможные значения: «Т» – подчиненная, «Е» – горизонтальная равнозначная, «D» – горизонтальная направленная, «Р» – исполнение, «V» – старые варианты, «К» – календарь. Данный код может использоваться, например, при настройке форм для объектов (см. п. 10.2.1.1)

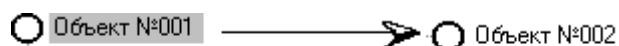


Рисунок 12 Иллюстрация понятия «Горизонтальная направленная связь»

Возможность установления подчиненных связей («Входимость») для пары экземпляров объектов определяется при настройке типов объектов отдельно для каждого типа подчиненных связей.

Экземпляры подчиненных связей могут снабжаться дополнительной атрибутивной информацией – атрибутами проекта. Возможность (обязательность) установки таких атрибутов на созданный экземпляр связи определяется в зависимости от типа дочернего объекта данной связи, при настройке данного типа объекта.

Любые горизонтальные связи можно устанавливать для любых объектов любых типов, дополнительных настроек для этого не требуется.

Типы связей: «календарь», «исполнения», «старые варианты» только предопределенные, Администратором такие типы связей не создаются. Имеется 2 типа связи типа «Календарь» – «Групповые события» и «События календаря», и по одному типу – «Исполнения» и «Старые варианты». Указанные типы связей используются системой для соответствующих целей, их применение описано в документации по Lotsia PDM PLUS. О создании исполнений и вариантов см. также разделы 6.1, 6.3.

4.2.2 Тип объекта

Тип объекта – элемент структуры БД, определяющий совокупность параметров, которые смогут (или должны иметь) создаваемые экземпляры объектов данного типа (Таблица 1).

Таблица 1 Некоторые параметры, определяемые при настройке типов объектов

№	Параметр	Возможность изменения для созданного элемента структуры БД	Возможные значения	Назначение
1.	Вид объекта	Нет	<ul style="list-style-type: none"> • Изделие • ОКД • НКД • Прч.Док 	Определяет положение объекта в дереве относительно объектов <u>других</u> видов (также см. раздел 10.3.4)
2.	Ключ сортировки	Да	Целое число	Определяет положение объекта в дереве относительно других объектов <u>данного</u> вида (также см. раздел 10.3.4)
3.	Пиктограмма	Да	Файл формата *.ICO, *.GIF или *.BMP, либо системные пиктограммы	Визуализация экземпляра объекта в окне дерева

№	Параметр	Возможность изменения для созданного элемента структуры БД	Возможные значения	Назначение
4.	Атрибуты	Да	Список, состоящий из имеющихся в БД атрибутов, с возможностью задания обязательных и возможных элементов	Хранение свойств экземпляра объекта или свойств экземпляра входящей подчиненной связи
5.	Типы объектов	Да	Список из имеющихся в БД типов объектов, с возможностью задания обязательных и возможных элементов	Определяется возможность установления исходящих подчиненных связей заданного типа между текущим объектом данного типа и экземплярами объектов различных типов
6.	Входимость	Да		Определяется возможность установления входящих подчиненных связей заданного типа между текущим объектом данного типа и экземплярами объектов различных типов

В системе имеется некоторое количество предопределенных типов, возможно создание новых типов объектов.

Далее рассмотрим подробнее некоторые свойства, определяемые при настройке типов объектов (см. п. 4.2.2.1):

4.2.2.1 Входимость объектов

Далее приведем несколько примеров задания входимости.

В объект типа «Сборочная единица» могут входить объекты следующих типов:

- «Сборочная единица»;
- «Деталь»;
- «Спецификация»;
- «Сборочный чертеж»;
- «Паспорт» и др.

Объект типа «Сборочная единица» может входить в объекты следующих типов:

- «Комплекс»;
- «Комплект»;
- «Сборочная единица» (но не в «Деталь»!).

В объект типа «Спецификация» может входить объект типа «Извещение об изменении» и т.д.

В объект типа «Офисное помещение» могут входить объекты следующих типов:

- «Мебель»;
- «Оргтехника»;
- «Сотрудник» и т. д.

Объект типа «Офисное помещение» может входить в объекты следующих типов:

- «Этаж здания»

См. также пример в разделе 4.5

Входимость определяется для типов объектов по каждому типу подчиненной связи.

Каждая связь в системе Lotsia PDM не зависима от других связей. В этом контексте входимость объектов является единичной, если конкретный объект связан данной связью только с одним родительским объектом. Если же объект связан данной связью с несколькими родительскими объектами, то входимость является множественной. Так, например, с точки зрения территориальной связи, рабочий стол сотрудника фирмы входит только в проект «Комната № 1» (единичная входимость).

Если же задать связь с названием «Использование столов», то этот стол может входить в проект «Рабочие места сотрудников в комнате № 1» и проект «Подставки под компьютеры в комнате № 1».

Множественная или единичная входимость данного типа объекта может задаваться отдельно для каждого типа связанного объекта.

Обязательная входимость типа объекта «А» в тип объекта «В» предполагает автоматическое включение объекта типа «А» в создаваемый объект типа «В». Например, различные виды изделий имеют различные основные (обязательные) конструкторские документы. Для сборочной единицы это спецификация, для детали – чертеж детали. Таким образом, для спецификации можно задать обязательную входимость в сборочную единицу (а также в комплект и комплекс).

Итак, в зависимости от конкретной предметной области и Вашего желания можно задать входимость различных типов объектов по различным типам подчиненных связей. Для горизонтальных связей входимость задавать не требуется.

4.2.3 Атрибуты

Атрибут – элемент данных программы Lotsia PDM, который выражает определенное свойство или характеристику объекта или экземпляра подчиненной связи (то есть, данной входимости объекта) (цвет, наименование, количество и т. д.). Каждый атрибут имеет определенное значение (или значения). Например, объект «Вертолетная площадка» имеет атрибуты: Наименование, Обозначение, Дата завершения объекта, Статус объекта, Шифр проекта – для определенности значения атрибутов:

Наименование – «Вертолетная площадка»;

Обозначение – «1921.2-B-2»;

Шифр проекта – «1921».

В системе имеется некоторое количество предопределенных атрибутов, Администратор может добавить произвольное количество новых атрибутов.

Атрибуты характеризуются перечисленными ниже свойствами, определяемыми при их создании (то есть, при создании или настройке атрибута как элемента структуры

БД). Указанные свойства проявляются при установлении данного атрибута конкретному экземпляру объекта (добавлении атрибута к объекту или изменении значения атрибута).

4.2.3.1 Тип данных атрибута

Значения атрибута соответствуют одному из трех типов данных:

- Дата/Время;
- Число (не более 15 знаков);
- Строка (до 2000 знаков)
- У созданного атрибута тип данных не может быть изменен.

4.2.3.2 Способы редактирования атрибута

В Lotsia PDM существует четыре типа ввода (редактирования) значений атрибутов, определяемые на уровне структуры БД (свойств данного атрибута):

- Свободный. При работе с проектом новое значение вводится пользователем с клавиатуры.
- Из списка. При работе с проектом ввод нового значения осуществляется путем выбора из списка значений. При задании списка значений атрибута показ в списке выбора некоторых значений атрибута можно отключить (см. рисунок 13).

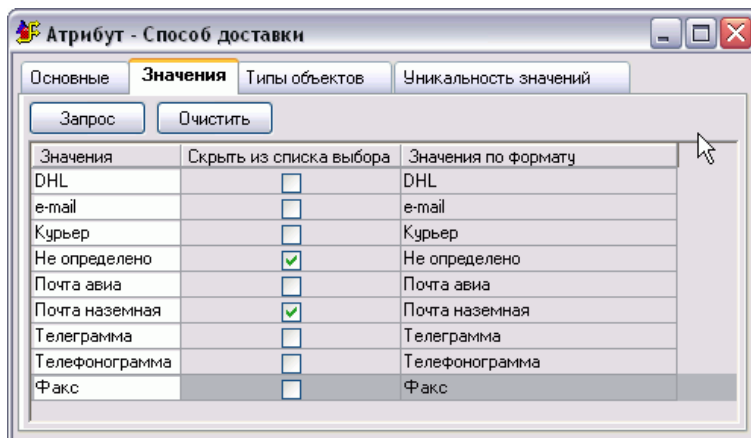


Рисунок 13 Пример настройки списка значений атрибута. Показ некоторых значений отключен

- Из классификатора. Выбор производится путем последовательного формирования значения с использованием специального иерархического списка значений (см. п. 9).
- Внешнее. Изменение атрибута возможно только с использованием внешнего программного модуля или с помощью действия (макрокоманды Lotsia PDM, их настройка и применение будут рассмотрены позднее).

Использование действий для присвоения значений атрибутам возможно независимо от типа редактирования данного атрибута.

Отметим, что с использованием действий, а также форм (то есть, на уровне настраиваемого интерфейса конечного пользователя) могут быть реализованы и другие способы присвоения значений атрибутам (окно выбора, переключатель, флажок, динамически формируемые списки значений и другое, в том числе, возможно использование различных способов редактирования атрибута, в зависимости от конкретной ситуации). Однако если в качестве способа редактирования атрибута в его

свойствах указано «Список», присвоить атрибуту значение, отсутствующее в списке значений будет невозможно и при помощи действия.

У созданного атрибута способ редактирования может быть изменен.

Особенности значений строковых атрибутов

Имеется следующая неочевидная особенность, которую следует учитывать при присвоении значений строковым атрибутам. Несколько упрощенно и без технических подробностей, значения хранятся и присваиваются следующим образом:

- ранее присвоенные значения атрибутов хранятся в БД, для каждого атрибута – собственный (отдельный) список значений (независимо от способа редактирования установленного для данного атрибута). Каждое значение атрибута имеет уникальный идентификатор (ID).
- при попытке присвоить (добавить или изменить) данному атрибуту данного объекта (проекта) значение, производится поиск в БД, в списке уже имеющихся значений данного атрибута. Если присваиваемое значение в БД найдено – атрибуту присваивается найденное в БД значение. Присвоение значения осуществляется путем сопоставления ID значения, ID атрибута, ID объекта (и ID связи – для атрибутов проекта). Если присваиваемое значение в БД отсутствует – в БД дописывается новое значение, ему присваивается идентификатор и данное значение атрибута аналогично сопоставляется (присваивается) данному атрибуту.
- поиск значений атрибутов в БД регистронезависимый – то есть, значение ищется без учета того, какие символы используются – строчные или прописные. Но присваиваемое значение состоит из конкретного набора как строчных, так и прописных символов, в том виде, как хранится в БД (для найденного значения), либо как введено (для нового значения).

Таким образом, при присвоении «нового» (то есть, отсутствующего в БД) значения атрибуту может быть использован произвольный набор строчных и прописных символов, введенное значение будет записано в БД (и присвоено атрибуту) именно в том виде, как было введено. Но, если введенное значение уже имеется в списке значений, но строчные и прописные символы в имеющемся значении расположены иначе – будет использовано именно «старое» значение. Например: имеющееся в БД значение – с прописной буквы, при попытке присвоить такое же, но со строчной буквы – будет присвоено значение с прописной буквы (имеющееся).

Учитывая вышесказанное, следует помнить, что если требуется изменить значение атрибута без изменения набора символов, а только изменив регистр отдельных символов, то выполнять такую операцию следует только путем прямого редактирования списка значений данного атрибута, на вкладке «Значения» окна свойств данного атрибута (см. рисунок 13). Попытки добавить или редактировать значение конкретного атрибута для объекта (проекта) в данном случае не приведут к требуемому результату.

При прямом редактировании значения в списке значений атрибута (см. рисунок 13), изменятся значения у всех данных атрибутов всех объектов (проектов), где данное значение было использовано (последнее весьма удобно и востребовано, например, для исправления грамматических или других допущенных при вводе значения ошибок).

Прямое редактирование списка значений атрибута доступно только Администратору (или пользователю, наделенному Администратором соответствующими функциями).

4.2.3.3 Множественность атрибута

Параметр множественности указывает, может ли одновременно данный атрибут быть присвоен одному и тому же объекту несколько раз. Например, множественным может быть атрибут «Формат» – в случае выполнения документа на нескольких листах различного формата.

- У созданного атрибута данное свойство не может быть изменено.
- Использование множественных атрибутов следует ограничивать, учитывая, что использовать с ними такие инструменты Lotsia PDM, как импорт, настраиваемые отчеты, пользовательские формы можно с ограничениями или невозможно. Для практических целей обычно вместо множественных используются единичные строковые атрибуты, содержащие несколько значений, отделенных друг от друга символом-разделителем. Удобство применения таких атрибутов для пользователя обеспечивается использованием действий (макрокоманд). Использование множественного атрибута в ряде случаев удобно заменить созданием экземпляров дочерних объектов, содержащих набор атрибутов.

4.2.3.4 Связь атрибута с объектом (проектом)

Атрибут объекта характеризует свойства объекта, не зависящие от контекста вхождения данного объекта в различные проекты. Если данный объект входит в несколько проектов, (имеет множественную входимость, то есть, связан подчиненными связями с несколькими объектами – родителями), то при изменении у него атрибута объекта он будет входить во все проекты с измененным значением данного атрибута. Например, атрибуты: «Наименование», «Обозначение» и «Номер инвентарный» для объекта типа «Чертеж».

Атрибут проекта выражает характеристику объекта в контексте его вхождения в другой объект (родитель), иными словами, такой атрибут соответствует конкретной подчиненной связи объекта с другим объектом. Например, атрибуты: «Позиция» (номер позиции на чертеже и в спецификации) и «Количество» для объекта типа «Деталь» или «Сборочная единица». У созданного атрибута данное свойство не может быть изменено.

4.2.3.5 Соответствие «тип объекта» – «атрибут»

Каждый тип объекта наделяется определенным набором атрибутов. Например, для здания – это «Наименование», «Количество этажей», «Дата завершения объекта» и др.

Атрибуты могут быть возможными или обязательными для данного типа объекта. Данное свойство относится как к типу объекта, так и к атрибуту, может быть изменено при настройке любого из указанных элементов структуры БД. Данное соответствие атрибутов проекта настраивается для типа объекта – потомка.

4.2.3.6 Группа атрибутов

Каждый атрибут обязательно включается в одну (только в одну) из групп атрибутов. Группы атрибутов создаются при настройке структуры БД, служат для быстрого поиска атрибута в окне выбора атрибута и для обеспечения некоторых других функций системы. Группы атрибутов могут быть изменены, в отличие от других элементов структуры, пустые группы можно удалить из БД.

4.3 Создание элементов структуры БД

В ходе изучения системы, рекомендуется создать типы объектов и атрибуты, далее, на их основе реализовать учебную настройку системы. При наличии достаточной информации об объекте автоматизации и требований к реализуемой информационной

модели, такая настройка может рассматриваться как прототип последующей реальной настройки, применимой для промышленного использования на Вашем предприятии. Однако, построение адекватной информационной модели для реального применения – само по себе нетривиальная и трудоемкая задача – зачастую значительно более трудозатратная, чем выполнение технической настройки системы.

Исходя из сказанного, можно рекомендовать, при создании учебной настройки не ставить в качестве первоочередной задачу максимально полного соответствия данной настройке конкретному предприятию, где планируется развертывание системы.

Основная цель создания учебной настройки – получение практических навыков работы с Lotsia PDM PLUS и изучение возможностей программы. Чтобы достигнуть указанной цели в сжатые сроки, рекомендуется не делать учебную настройку сложной.

Следующие общие замечания можно рассматривать в качестве рекомендаций при создании настроек Lotsia PDM PLUS, независимо от конкретной предметной области, информационной модели и прикладных задач, для решения которых используется создаваемая настройка:

- Администрировать систему удобнее, если все содержащиеся в ней проекты Lotsia PDM объединены в один общий проект. В идеале, объектов не входящих в проекты, кроме корневого объекта, быть не должно.
- Первоначальное структурирование проектов можно вести сверху вниз – так нагляднее и удобнее с точки зрения администрирования системы (хотя, для реальной настройки данный вариант – и не единственно возможный).

Исходя из сказанного, ввод данных на начальном этапе создания настройки может начинаться с создания корневого объекта «Информационные ресурсы», куда входят, например, следующие основные проекты:

- «Проекты в разработке»
- «Архив проектов»
- «Справочники»
- «Служебные».

Приведенный перечень и названия условны, но, как правило, похожие проекты содержатся в том или ином виде и в реальных настройках.

Для создания корневого объекта и указанных основных проектов удобно использовать отдельный тип объекта, для определенности назовем его «Папка» (в реальной настройке подобных типов объектов, скорее всего, потребуется больше).

Состав и структура проектов целиком зависят от предметной области. В учебных целях можно создать настройку, приближенную к Вашей области деятельности, например:

- для машиностроительного предприятия, структура рабочих и архивных проектов будет, вероятнее всего, аналогична структуре разрабатываемых изделий;
- для проектного предприятия структура проекта создается аналогично структуре выпускаемых проектов;
- если решаемая настройкой задача (или одна из задач) – учет корреспонденции предприятия – папки проектов могут содержать соответствующие объекты входящих и исходящих документов,

структурированные требуемым образом – предметно, хронологически или иначе;

- например, локальная задача: учета имущества, находящегося в здании предприятия. Данная задача может решаться созданием проектов, структурирующих указанное имущество каким-либо образом, например – по его размещению в здании, на этажах, в помещениях и т. д.

Потребуется, соответственно, типы объектов, отражающих структуру и состав Вашей переметной области, с необходимой степенью детализации.

Как пример – для машиностроения это могут быть объекты изделий: комплекс, комплект сборочная единица, деталь, материал... и объекты документов: спецификация, чертеж детали и т.д.

Для проектного предприятия – проект, раздел проекта, проектный документ, сметный документ и т.д.

Для случая учета корреспонденции – папка года, месяца, тематическая..., документ входящий, документ исходящий...

Аналогично для других применений.

При изучении Lotsia PDM PLUS будет вполне достаточно и правильно, если для начала ограничиться простейшей учебной настройкой, перечень требуемых элементов структуры БД для которой приведен в разделе 4.5. Создание в учебных целях настройки, отражающей, в необходимой степени, особенности вашей предметной области приветствуется, но не является основной целью в данном контексте и может быть выполнено позднее.

Используемые в создаваемой настройке справочники также в большинстве случаев определяются областью применения настройки.

Для большинства случаев в числе справочников характерно использование справочника данного предприятия, имеющего структуру, аналогичную организационной структуре данного предприятия. Подобный справочник может включать в себя подразделения различного (объекты одного или различных типов) уровня с входящими в них сотрудниками (также объекты Lotsia PDM соответствующего типа или нескольких типов). В учебной настройке предполагается создание простейшего варианта такого справочника.

Другой характерный пример справочника – справочник контрагентов. Соответственно – тип объекта – «Контрагент».

В проект «Служебные» могут включаться объекты, обеспечивающие функционирование системы, и не используемые непосредственно пользователями. К таким объектам могут относиться, например, т.н. «Счетчики» – объекты, используемые для хранения последних значений присваиваемых номеров, и т.п. Специальных типов объектов для учебной настройки для данного раздела можно и не создавать, использовать тип «Папка».

4.3.1 Создание типов связей

Связи создаются с использованием пункта Главного меню «Типы связей» («Администрирование»). Для создания нового типа связи в контекстном меню открытого окна «Типы связей объектов» выберите «Добавить», введите название связи, измените её тип, если это необходимо. Возможен выбор среди следующих типов: «Подчиненная», «Равнозначная», «Направленная». Код типа связи присваивается программой автоматически при сохранении созданной связи.

4.3.2 Создание типов объектов

Для добавления нового типа объекта, в Главном меню выберите «Администрирование» → «Структура данных» → «Типы объектов». В открывшемся окне «Выбор типа объекта» виден список имеющихся типов объектов – во вновь созданной БД это только predefined типы объектов. Для создания нового типа объекта, нажмите кнопку «Создать» данного окна. В открывшемся окне (Рисунок 14) в поле «Описание» введите название создаваемого типа объекта, в поле «Мнемо» – произвольный короткий набор символов, который можно сопоставить данному типу объекта и использовать в дальнейшем, например для поиска.

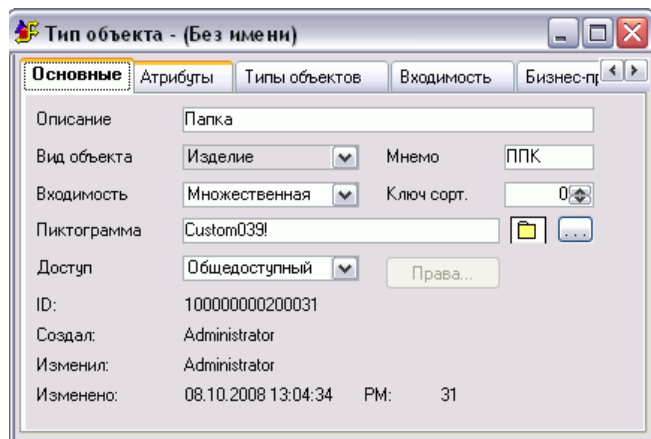





Рисунок 14. Создание нового типа объекта

Вид объекта используется для сортировки объектов в дереве связей, может быть изменен только при создании объекта – сделайте это, если собираетесь использовать данный параметр. При необходимости укажите ключ сортировки, который позволяет изменить положение экземпляра объекта в дереве связей относительно других объектов данного вида, находящихся с данным объектом на одном уровне. Подробнее о сортировке объектов в дереве связей см. раздел 10.3.4.


Нажмите кнопку , расположенную справа от поля «Пиктограмма», выберите «Стандартные...» или «Из файла...», выберите пиктограмму для данного типа объекта (любую, какая вам покажется подходящей из имеющихся). Для сохранения нажмите  из верхней панели инструментов. Отметим, что для отображения в дереве экземпляров объектов данного типа можно использовать как пиктограмму, указанную для данного типа объекта (отображается по умолчанию), так и другие пиктограммы – после выполнения настроек, которые рассмотрены в разделе 10.3.3.


Для создания нового типа объекта, не закрывая окна с созданным типом объекта, нажмите  из верхней панели инструментов, откроется новое окно создания типа объекта. Повторите ту же последовательность действий для остальных создаваемых типов объектов (конкретные типы создаются в зависимости от используемой модели данных, см. также раздел 4.5).

4.3.3 Определение входимости для типов объектов

Зададим входимость созданных типов объектов. Для этого откроем окно созданного типа объекта, например – «Папка» (тип «Папка» следует предварительно создать). («Главное меню» → «Администрирование» → «Структура данных» → «Типы объектов» → выбрать из списка «Папка»). Входимость, то есть, типы объектов, в которые может входить объект данного типа (типы возможных родителей для данного типа объекта) по данной подчиненной связи, задается на вкладке «Входимость». Например, папка в учебной настройке может входить в другие объекты типа «Папка».

На вкладке «Типы объектов» можно указать типы объектов, которые могут входить в объект данного типа. Для «Папки», например, это могут быть все типы объектов, используемые в данной учебной настройке (конкретные типы укажите в зависимости от используемой модели данных, см. также раздел 4.5).

Для добавления типов объектов, которые могут входить в объект данного типа по данной подчиненной связи, откройте вкладку «Типы объектов». В контекстном меню выберите «Добавить». В открывшемся окне «Выбор типов объектов» выберите требуемые типы. Нажмите «Ок», сохраните, нажав  из верхней панели инструментов.

Для задания входимости для следующего типа объектов, не закрывая окна с предыдущим типом объекта, нажмите  из верхней панели инструментов, откроется окно выбора типа объекта. Выберите очередной используемый в Вашей настройке тип объекта, убедитесь, что на вкладке «Входимость» для него уже указано «Папка». Откройте вкладку «Типы объектов», укажите возможные типы потомков, сохраните.

Повторите ту же последовательность действий для остальных созданных Вами типов объектов.

Отметим, что при добавлении входимости «самого в себя» для типа объекта (например, возможность добавления папки в папку), можно выполнять данную настройку на любой из вкладок («Объекты» или «Входимость»), но делать это следует только на одной из них. После сохранения указанной настройки, на вторую вкладку данная настройка будет добавлена автоматически. Если добавить указанную входимость сразу на обе указанные вкладки, при сохранении настройки будет выдано сообщение об ошибке.

В учебной настройке допускается заимствование элементов одних проектов в другие. На практике данное положение не всегда соответствует действительности, поэтому при необходимости следует настраивать возможность заимствования для пар типов объектов, устанавливая для таких пар множественную или единичную входимость (см. рисунок 15).

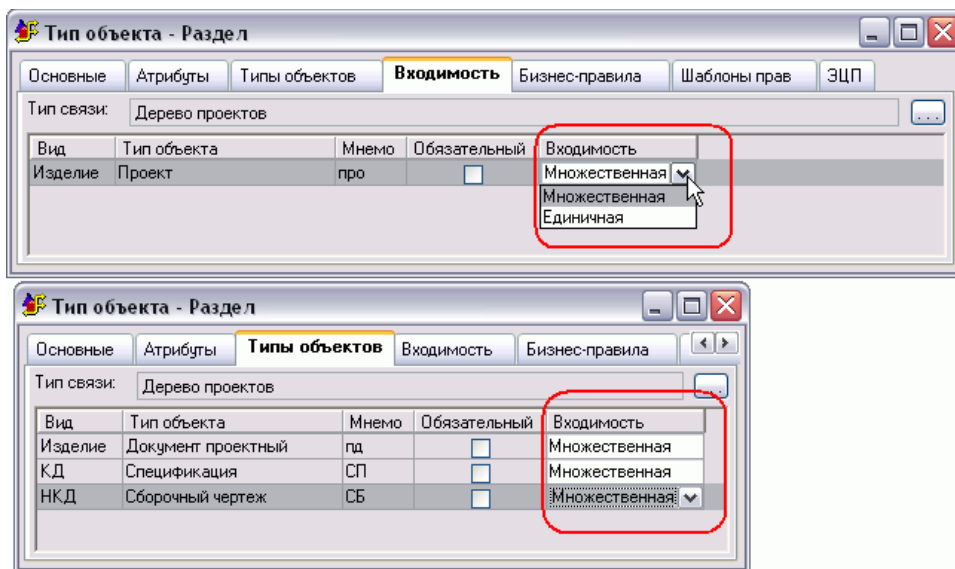


Рисунок 15 Установка входимости для пар типов объектов

Множественная входимость применяется в настройках, связанных с машиностроением, приборостроением и т.п., ввиду того, что для данных отраслей особенно развита унификация – то есть, применение в неизменном виде ранее разработанных или типовых изделий и их элементов во вновь разрабатываемых изделиях. Во многих предметных областях подобная практика гораздо менее распространена (например, в проектировании зданий). Применение единичной входимости позволяет,

например, при учете имущества, исключить возможность нахождения одного и того же учитываемого объекта в двух различных местах одновременной (например, один и тот же стул не может находиться одновременно в двух помещениях или на двух складах...).

Тем не менее, в большинстве случаев на практике используется множественная входимость – таким образом, не ограничивается возможность структурирования одних и тех же объектов различным образом.

При включенном флажке обязательной входимости, при создании экземпляра объекта родительского типа к нему автоматически добавляется экземпляр объекта дочернего типа. Обязательная входимость типа объекта «самого в себя» может быть задана, но, при создании объектов дочерние объекты того же типа, что и созданный объект, автоматически создаваться не будут (Рисунок 16).

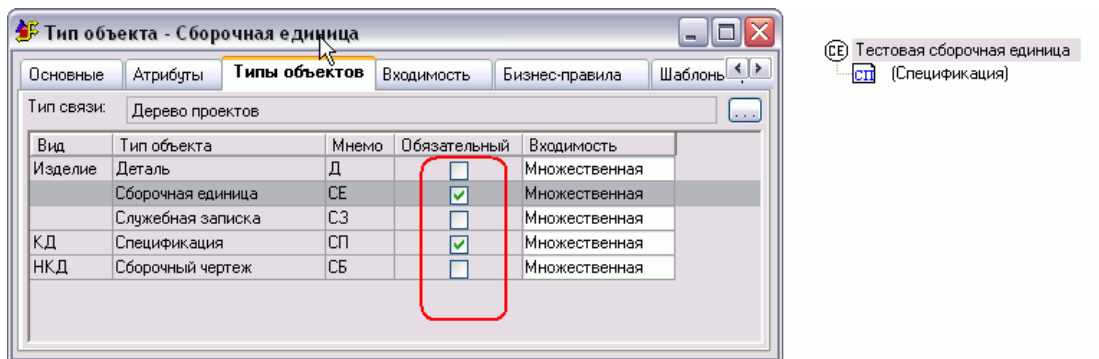


Рисунок 16 Пример задания обязательной входимости и результата её применения

Отметим, что определять входимость можно одновременно с созданием типов объектов или позднее, по мере возникновения необходимости, в связи с развитием информационной модели.

Далее необходимо создать атрибуты и сопоставить их типам объектов.

4.3.4 Создание атрибутов

Для создания атрибутов выберите «Главное меню» → «Администрирование» → «Структура данных» → «Атрибуты». Откроется окно «Выбор атрибута», содержащее список всех имеющихся в системе атрибутов (в случае, если работа ведется в созданной пустой БД, это будут только предопределенные атрибуты). Видно, что в системе изначально имеются атрибуты «Наименование», «Обозначение» и «Примечание». Создавать указанные атрибуты не потребуется, будем использовать имеющиеся.

Для создания остальных атрибутов в окне «Выбор атрибута» нажмите «Создать».

В открывшемся окне укажите параметры для атрибута «Номер регистрационный» (Рисунок 17).

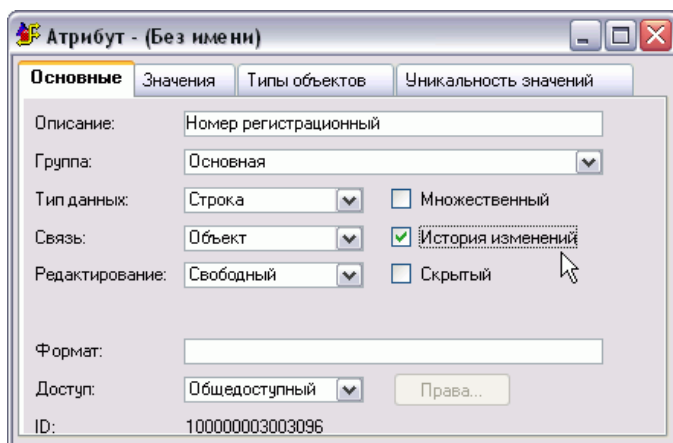




Рисунок 17. Начало создания нового атрибута

Введите описание атрибута, выберите группу (если не создано групп атрибутов – можно указать «Основная»). Выберите тип данных. Для номеров рекомендуется использовать тип данных «Строка», чтобы обеспечить возможность указывать номера, содержащие не только цифры, например «768а».

Остальные параметры можно пока оставить без изменения.

Для сохранения нажмите  из верхней панели инструментов.

Для создания очередного атрибута, не закрывая окна с созданным атрибутом, нажмите  из верхней панели инструментов, откроется новое окно создания атрибута. Повторите ту же последовательность действий для остальных создаваемых атрибутов.

4.3.5 Настройка соответствия тип объекта – атрибут

Для установления соответствия между одним типом объекта и одним или несколькими атрибутами откройте окно свойств данного типа объекта: «Главное меню» → «Администрирование» → «Структура данных» → «Типы объектов» → выберите из списка требуемый тип объекта. Перейдите на вкладку «Атрибуты», в контекстном меню выберите «Добавить». Выделите в открывшемся окне требуемые атрибуты, нажмите «Ok», сохраните изменения.

Если требуется установить указанное соответствие между одним атрибутом и одним или несколькими типами объектов, откройте окно свойств данного атрибута: «Главное меню» → «Администрирование» → «Структура данных» → «Атрибуты» → выберите из списка требуемый атрибут. Перейдите на вкладку «Типы объектов», в контекстном меню выберите «Добавить». Выделите в открывшемся окне требуемые типы объектов, нажмите «Ok», сохраните изменения.

Возможно указание обязательных атрибутов для данного типа объекта, (либо установка обязательности данного атрибута для указанных типов объектов) – см. рисунок 18. При создании объектов данного типа, экземпляру объекта будет автоматически добавлен набор обязательных атрибутов со значениями, заданными по умолчанию.

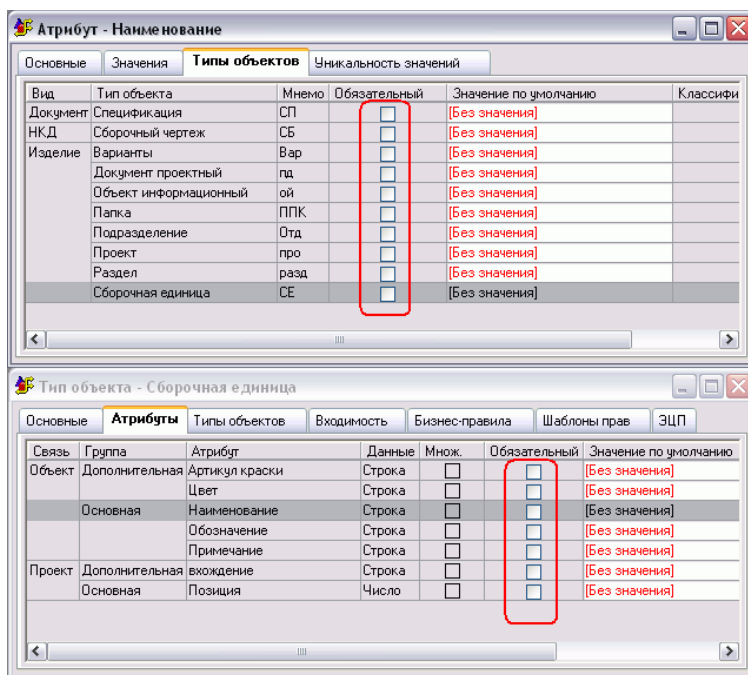


Рисунок 18 Указание обязательных атрибутов

4.4 Резюме

Базовая единица хранения данных в Lotsia PDM – объект. Объекты могут соединяться между собой различными связями. Объекты и связи могут содержать информационные поля – атрибуты.

Возможно построение и отображение на экране структурированных проектов. Используемые для этого связи – подчиненные.

Объекты создаются различных типов, для созданного экземпляра объекта тип изменить нельзя.

При создании типа объекта описывается возможная конфигурация экземпляров объектов данного типа – возможность использования различных атрибутов и подчиненных связей (правила входимости) для пар объектов (различных или одинаковых типов).

Горизонтальными связями могут связываться пары любых объектов, выполнение настройки типов объектов для этого не требуется.

Возможные типы данных атрибутов – число, строка (до 200 символов), дата/время.

4.5 Создание элементов структуры БД. Учебное задание

Предлагаемая для создания учебная настройка – упрощенный и сокращенный вариант настройки Lotsia PDM для управления проектами. Понятие «Проект» в данном случае может трактоваться достаточно широко – от проекта здания или сооружения, выполняемого проектной организацией, до проекта проведения мероприятия, строительства, организации производства и т.д. Данная настройка не претендует на практическое применение, однако, её реализация позволит вам познакомиться с основными возможностями системы, и впоследствии самостоятельно выполнять настройки, пригодные для практического применения в Вашей предметной области.

В данной настройке **проект** состоит из **разделов**, в которые могут входить **проектные документы, спецификации и сборочные чертежи**. Для организации

проектов используются **папки**. Отдельно будет настроен простейший справочник предприятия, состоящий из объектов **подразделений** и **сотрудников**

Создайте типы связей (можно выполнить позднее, по мере рассмотрения примеров использования дополнительных связей): «Хранение бумажных документов» – подчиненная, «Похожие» – равнозначная; «Прототип» – направленная.

Создайте типы объектов: «Папка», «Проект», «Раздел проекта», «Документ проектный», «Подразделение», «Сотрудник», «Полка», «Стеллаж». Вид объекта для всех создаваемых типов, кроме «документа проектного», оставьте установленный по умолчанию – «Изделие». Для документа проектного установите вид объекта «Прч.Док.» – прочий документ. Входимость оставьте установленную по умолчанию – «Множественная». Будут также использоваться предопределенные типы объектов – например, «Спецификация» и «Сборочный чертеж» – их создавать не надо, используйте имеющиеся.

Создайте атрибуты объектов: «Номер инвентарный», «Дата регистрации», «Шифр проекта», «Должность основная», «Статус» и атрибут проекта: «Должность в подразделении». Будут также использоваться предопределенные атрибуты «Наименование», «Обозначение», «Примечание» – их создавать не надо, используйте имеющиеся.

Указанные атрибуты строковые, единичные. Для атрибута «Статус» в поле «Редактирование» укажите «Список». На вкладке «Значения», используя контекстное меню, добавьте значения: «Разрабатывается», «Аннулирован», «Утвержден». Для остальных атрибутов в поле «Редактирование» укажите «Свободный».

Используя созданные и предопределенные типы объектов, задайте следующие возможные входимости для типа связи «Дерево проектов»:

Родитель	Потомок
Папка	Папка, Проект, Раздел проекта, Документ проектный, Подразделение, Сотрудник, Спецификация, Сборочный чертеж
Проект	Раздел проекта, Документ проектный, Спецификация, Сборочный чертеж
Раздел проекта	Документ проектный, Спецификация, Сборочный чертеж
Подразделение	Подразделение, Сотрудник

Используя созданные и предопределенные типы объектов, задайте следующие возможные входимости для типа связи «Хранение бумажных документов» (можно выполнить позднее, по мере рассмотрения примеров использования дополнительных связей):

Родитель	Потомок
Папка	Стеллаж
Стеллаж	Полка

Используя созданные и предопределенные типы объектов и атрибуты, сопоставьте атрибуты типам объектов:

Тип объекта	Возможные атрибуты
Папка	Наименование, Обозначение, Примечание
Проект, Раздел проекта	Наименование, Обозначение, Шифр проекта, Примечание
Документ проектный, Спецификация, Сборочный чертеж	Наименование, Обозначение, Шифр проекта, Примечание, Номер инвентарный, Дата регистрации
Подразделение	Наименование, Примечание
Сотрудник	Должность основная, Должность в подразделении, Примечание

По мере развития данной настройки список типов объектов, атрибутов и их соответствий будет дополняться.

4.6 Элементы структуры БД. Вопросы для самопроверки

1. Чем определяется взаимное расположение объектов, находящихся в дереве относительно друг друга на одном уровне?
2. Как настраивается соответствие между типами объектов и атрибутами?
3. Как настраивается возможность установки подчиненных связей между различными типами объектов?
4. Какие способы присвоения значений атрибутам могут использоваться в Lotsia PDM?
5. Как сделать, чтобы атрибут создавался автоматически при создании нового объекта данного типа?
6. Как сделать, чтобы при создании нового объекта данного типа, автоматически создавался еще один объект, входящий в данный объект? Могут ли такие объекты быть одного типа?

5 Создание объектов и дерева проекта

5.1 Создание новых объектов, добавление создаваемых объектов в проект

Для создания экземпляров объектов предварительно необходимо выполнить настройку типов объектов (см. п. 4.3).

Создание объектов в учебной настройке будем осуществлять, начиная с объектов верхнего уровня. Для создания корневого объекта выберите пункт «Создать объект» Главного меню. В открывшемся окне выбора типа объекта выберите тип создаваемого объекта («Папка»), введите описание объекта (например – «Информационные ресурсы предприятия»).

Если дополнительные типы подчиненных связей в БД не создавались (см. п. 4.3.1), либо для данного профиля пользователя определен тип связи, используемый по умолчанию в окнах проектов (будет рассмотрено далее, см. раздел 10.3), откроется окно дерева связей с созданным объектом. Соответственно, указанное окно будет открыто по типу связи «Дерево проектов», либо указанному по умолчанию в настройке данного профиля.

При наличии в БД нескольких типов подчиненных связей и отключенном флажке «Использовать тип связи по умолчанию в окнах проектов» (будет рассмотрено далее – см. рисунок 87), вам будет предложено выбрать тип связи, по которому следует открыть окно созданного объекта.

В результате будет открыто дерево связей с созданным объектом. Дерево связей в данном окне строится по тому типу связи, по которому данное окно открыто («Текущий тип связи»). Чтобы изменить текущий тип связи, необходимо выделить объект в дереве проектов, в верхнем меню «Вид» выбрать «Сменить тип связей...» и выбрать требуемый тип связи из открывшегося списка.

Для добавления создаваемого проекта в созданный выделите созданный объект в окне дерева связей, в контекстном меню выделенного объекта выберите «Добавить новый объект», выберите тип добавляемого объекта («Папка»), введите описание объекта (например, «Проекты в разработке»), продолжайте аналогичным образом строить дерево проекта. Созданное в результате выполнения учебного задания (см. раздел 5.4) дерево проектов может быть похожим на показанное на рисунке (Рисунок 19).

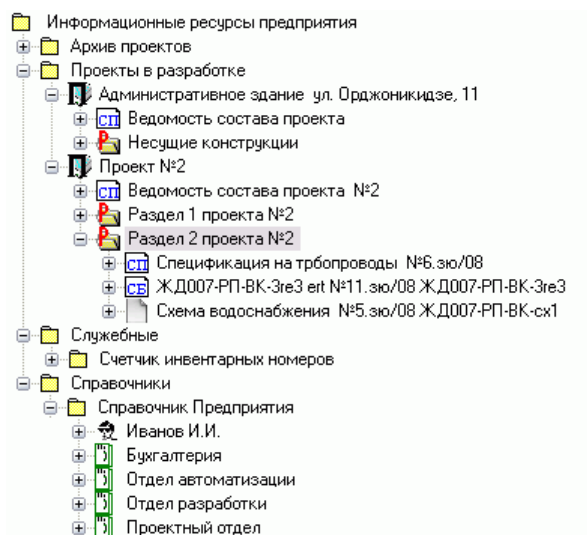


Рисунок 19 Дерево основных проектов учебной настройки. Один из возможных примеров.

Общее количество создаваемых объектов и их описания – произвольные.

Дерево связей можно строить и из имеющихся в БД объектов, путем установки между ними подчиненных связей.

Для построения дерева проектов из существующих в БД объектов, можно использовать метод drag-and-drop, помещая объекты на соответствующий объект в открытом окне дерева проектов. Перетаскиваемый объект может изначально находиться в отдельном открытом окне дерева связей (см. рисунок 20), или в том же окне, куда он перетаскивается.

Добавление объектов возможно также на вкладки «Объекты» и «Входимость», при этом для выделенного объекта добавляется связь, соответственно, с потомком или родителем. Для добавления объектов на указанные вкладки можно также использовать метод drag-and-drop («перенеси-и-положи»), либо контекстные меню вкладок (см. рисунок 20).

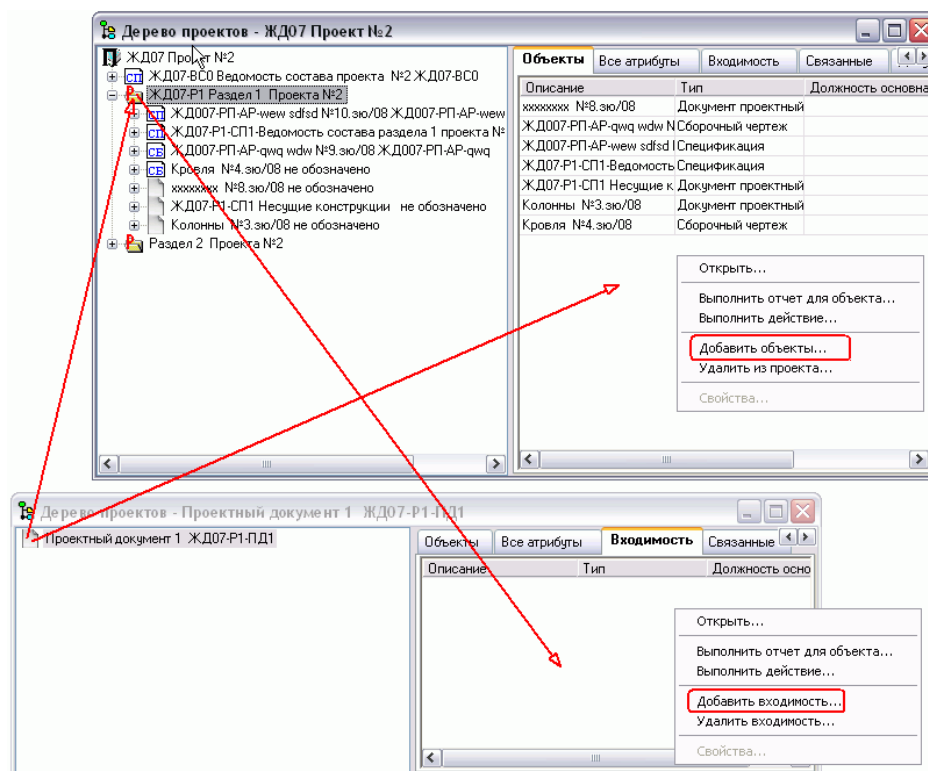


Рисунок 20 Возможные способы добавления объекта в проект: стрелками показаны варианты перетаскивания объектов методом drag-and-drop, обведены возможные варианты использования контекстных меню

В результате использования любого из вариантов указанных методов объект проектного документа будет добавлен в указанное место проекта – в данном примере – в раздел 1. При этом добавленный объект отобразится в дереве связей и на вкладке «Объекты» родительского объекта, а родительский объект отобразится на вкладке «Входимость» добавленного объекта (см. рисунок 21).

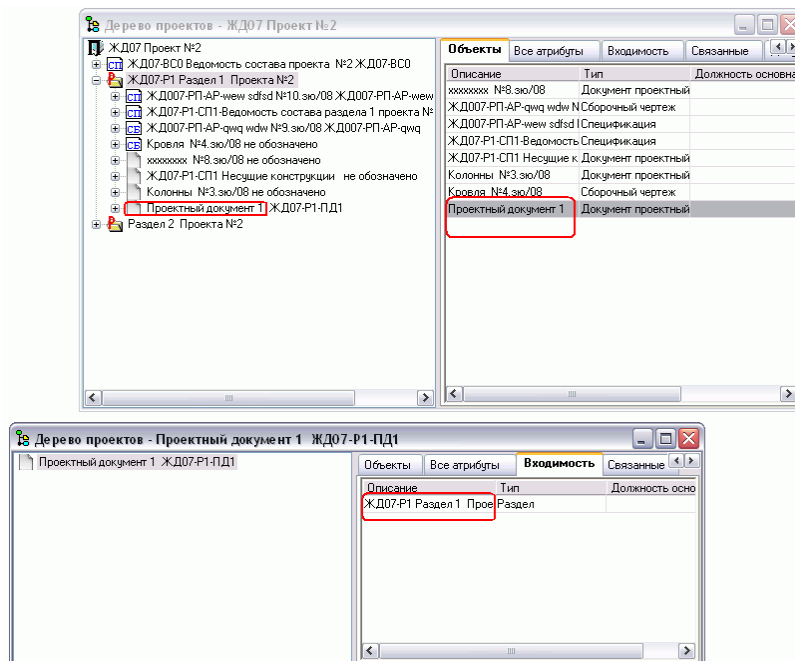


Рисунок 21 Объект добавлен в проект

Созданным объектам могут быть установлены атрибуты, сопоставленные ранее для соответствующих типов объектов. Установить атрибуты можно как во время создания

очередного объекта, так и позднее. В любом случае, для добавления атрибута перейдите на вкладку «Все атрибуты» выделенного в дереве, либо создаваемого объекта. Для выделенного в дереве объекта, для изменения атрибутивной информации следует перейти в режим редактирования, для чего необходимо выполнить двойной щелчок левой клавишей мыши на вкладке «Все атрибуты». Для редактирования откроется отдельное окно с вкладками данного объекта. Для добавления атрибута, в контекстном меню данной вкладки (находящейся в режиме редактирования) выберите «Добавить», выберите добавляемый атрибут из открывшегося списка, введите значение атрибута, сохраните изменения.

Несколько замечаний.

Рассмотренный выше порядок создания объектов и установки атрибутов не требует дополнительных (помимо создания элементов структуры БД – см. раздел 4.3) настроек и одинаков для различных типов объектов и атрибутов, поэтому создание учебных проектов мы начали осуществлять именно таким образом. Отметим, что конечными пользователями рассмотренный порядок работы применяется редко, ввиду трудоемкости и возможности ошибок (например, при неправильном выборе типа создаваемого объекта и т.п.). Для обеспечения удобства работы конечных пользователей применяются средства автоматизации – атрибутивные формы, действия и др. Настройка и использование указанных инструментов будет рассмотрена позднее. При использовании средств автоматизации работы, фактически выполняются те же операции, что и при рассмотренном порядке создания объектов и установки атрибутов «вручную».

При добавлении в дерево очередного объекта, вы можете столкнуться с тем, что в списке типов создаваемых объектов (Рисунок 22) отсутствует требуемый вам тип.

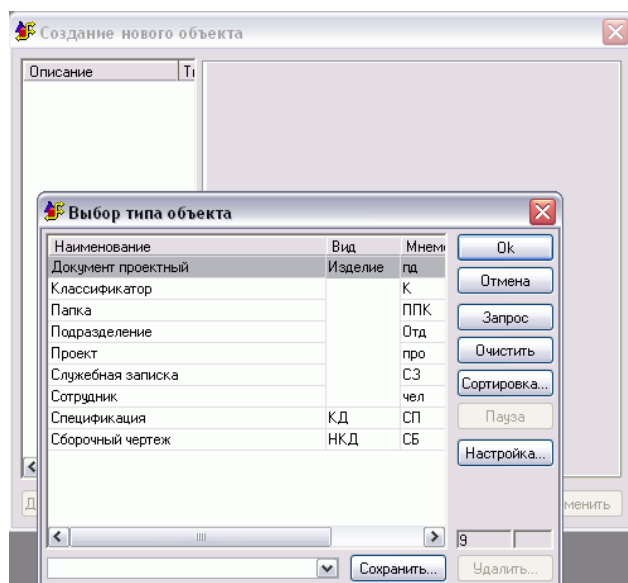


Рисунок 22 Создание нового объекта. Выбор типа создаваемого объекта

Данная ситуация означает, что в свойствах данного типа объекта не задана требуемая входимость для данных типов объектов. В указанном случае настройте требуемую входимость (см. п. 4.3.3) для типа объекта родителя, либо для типа добавляемого потомка. Если при добавлении в проект нового объекта не открывается окно выбора типа добавляемого объекта – значит, для типа объекта-родителя указан единственно возможный тип объекта-потомка.

При добавлении к объекту атрибута, требуемого атрибута может не оказаться в списке добавляемых атрибутов. Данная ситуация говорит о том, что атрибут объекту уже присвоен, либо отсутствует настройка соответствия данного атрибута данному типу

объекта. В последнем случае выполните требуемую настройку для атрибута, либо для типа объекта (см. п. 4.3.5).

5.2 Использование различных типов связей

В некоторых случаях имеет смысл использовать в настройке несколько различных подчиненных связей, что позволяет строить проекты с использованием одних и тех же объектов, но использовать различные правила входимости. Кроме того, использование различных типов связей позволяет разделить информацию по использованию информационных объектов в различных контекстах.

Например, структура изделия для конструктора и технолога может существенно различаться, при этом используются одни и те же детали и сборочные единицы, но в технологическом дереве присутствуют дополнительные элементы, такие как технологические операции, заготовки, промежуточные сборки и др., кроме того, документация по изделию для конструктора и технолога различна. В данном случае целесообразно для построения технологического дерева создать и использовать отдельный тип связи – «Технологическая»

При построении справочника предприятия, в ряде случаев требуется обеспечить возможность учета функциональной подчиненности сотрудников, которая может различаться для различных сфер деятельности и различных случаев. Например, проверяющие и исполнители для различных типов документов могут различаться и даже меняться местами. Один из способов решения данной задачи – создание и использование различных связей «Функциональная подчиненность».

Учет корреспонденции может вестись хронологически, отдельно входящей, исходящей и внутренней. Вместе с тем, в ряде случаев удобно дополнительно строить дерево переписки, состоящее из входящих, исходящих и внутренних документов. В этом случае может быть создана и использована подчиненная связь «История переписки».

В учебных целях рекомендуется создать проект с использованием созданного типа подчиненной связи, например – «Хранение бумажных документов» (см. пп. 4.5, 5.4).

Отметим, что после добавления типа подчиненной связи (см. раздел 4.3.1) правила входимости объектов необходимо задавать отдельно (см. раздел 4.3.3) для различных типов подчиненных связей. Способы установки подчиненных связей не отличаются для различных типов подчиненных связей, рассмотрены в разделе 5.1.

Соответствующий тип подчиненной связи может быть определен для различных пользователей по умолчанию (будет рассмотрено далее, см. раздел 10.3). В результате, дерево проектов у различных сотрудников может автоматически открываться по требуемому им типу связи.

Примеры возможных типов горизонтальных связей приведены в разделе 4.2.1, порядок их создания – см. раздел 4.3.1. Для горизонтальных связей указывается тип «Равнозначная» или «Направленная». Напомним, что для горизонтальных связей правила входимости не настраиваются.

Чтобы связать 2 объекта горизонтальной связью, можно выделить объект в окне дерева связей, и удерживая нажатой правую клавишу мыши, перетащить его на другой объект, с которым требуется установить горизонтальную связь (Рисунок 34). Далее отпустить клавишу мыши, из открывшегося выпадающего меню выбрать «Добавить связь...». В открывшемся окне «Выбор типа связи» выбрать требуемый тип горизонтальной связи.

Если выбрана равнозначная связь – она будет создана сразу.

Если выбрана направленная связь – откроется окно выбора направления связи; после указания направления связь будет создана.

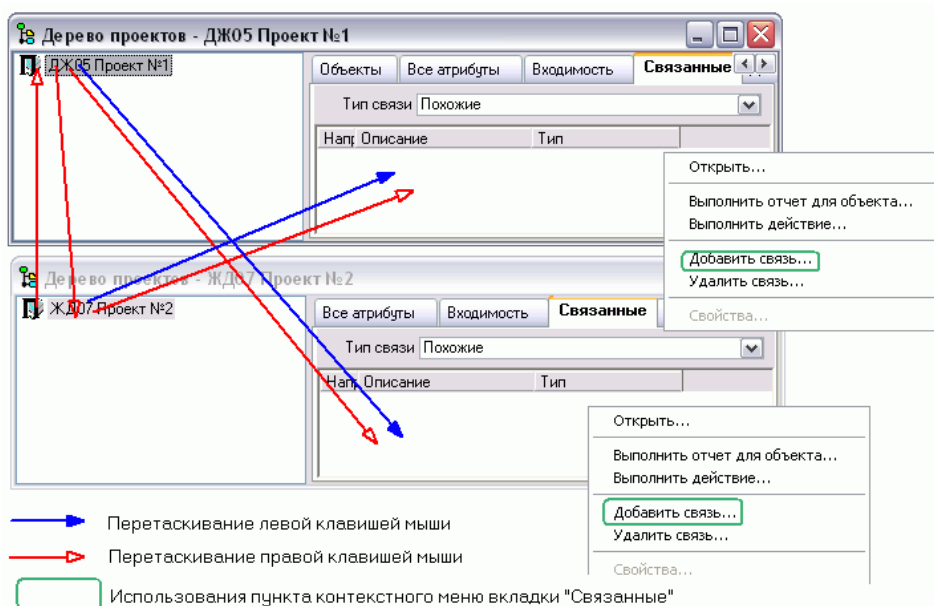


Рисунок 23 Возможные способы установки горизонтальных связей

Аналогично можно устанавливать горизонтальные связи, пользуясь вкладкой «Связанные» выделенного объекта. Для установки связей используется пункт «Добавить связь» контекстного меню данной вкладки, либо перетаскивать методом drag-and-drop на данную вкладку связываемые объекты (при этом можно пользоваться как левой, так и правой клавишей мыши) (Рисунок 34). Окно выбора типа связи не открывается, устанавливается связь текущего типа – указанного в поле «Тип связи» данной вкладки.

5.2.1 Альтернативные способы связывания (установки ссылок) различных сущностей

Часто оказывается удобным реализовать связь между объектами путем записи идентификатора (ID) связываемого объекта в атрибут. Подобным образом можно записать также идентификатор другой, связанной с данным объектом сущности, например, пользователя или запущенного на выполнение процесса и т. п. В атрибут может быть записан как единственный идентификатор (в этом случае можно использовать атрибут с типом данных «Число»), так и несколько идентификаторов в виде строки с разделителями (в этом случае тип данных атрибута – «Строка» и её длина не более 2000 символов).

Как для записи идентификаторов в атрибуты, так и для чтения записанных значений, осуществления доступа к соответствующим сущностям и выполнения требуемых обработок, как правило, используются действия над объектами (будет рассмотрено далее, см. п. 16). В любом случае, ручной ввод идентификаторов конечным пользователем должен быть исключен. Отображение в пользовательских формах значений атрибутов, содержащих идентификаторы, также должно быть исключено. В пользовательских формах, вместо значений указанных атрибутов, при необходимости могут применяться вычисляемые поля, отображающие понятную пользователю информацию (настройка пользовательских форм будет рассмотрена далее, см. п. 10.2).

Использование данного способа «связывания» сущностей, строго говоря, является «установкой ссылок», так как установка соответствующего атрибута осуществляется для данного конкретного объекта, а для соответствующей «связанной» сущности – это совершенно не обязательно. Несмотря на необходимость написания действий и выполнения дополнительных настроек пользовательских форм, запись идентификаторов в

атрибуты в ряде случаев оказывается удобнее, чем установка связей между объектами (а если ссылка устанавливается не на объект – это единственно возможный способ связывания сущностей).

При реализации различных информационных моделей, принимая решения об используемых способах связывания объектов, следует дополнительно учитывать требуемый для конкретного случая порядок распределения прав доступа к связанным объектам. В частности, при использовании наследуемых прав, автоматическое наследование происходит по всем подчиненным связям (всем типам подчиненных связей). Права доступа не наследуются по горизонтальным связям. Защита подчиненных связей реализована в рамках отдельной привилегии – «Доступ к проекту», а прав доступа для горизонтальных связей не устанавливается. Доступ к атрибутам объекта осуществляется в рамках отдельной привилегии «Доступ к атрибутам объекта». Полная информация о правах доступа имеется в документации по Lotsia PDM PLUS и в контекстной справке по Lotsia PDM PLUS

5.3 Резюме

При создании объекта указывается его тип.

Совокупность объектов, связанных подчиненной связью, называется «Проектом».

При добавлении объекта в проект проверяются правила входимости.

Один объект может входить в один (единичная входимость) или несколько объектов (множественная входимость, заимствование).

Объект не входящий в другие объекты, называется «Корневым», объекты, входящие в корневой объект называются потомками первого уровня, объекты, входящие в потомки первого уровня – «Потомки второго уровня», и т.д.

Правила входимости задаются отдельно для каждого типа подчиненной связи.

Объект, в который входит потомок, для данного потомка считается «Родителем».

На каждое вхождение объекта в проект могут устанавливаться собственные значения атрибутов проекта.

5.4 Создание проекта. Учебное задание

Создайте объекты основных проектов (объект «Информационные ресурсы предприятия» с потомками первого уровня, см. рисунок 19). Используйте предопределенный тип связи «Дерево проектов» во всех случаях, где иное не оговорено особо. Для учебной настройки при создании указанных объектов достаточно использования единственного типа объекта – «Папка».

5.4.1 Создание проекта. Присвоение атрибутов. Заимствование

В папке «Проекты в разработке» создайте 2 или более проекта (то есть, объекты типа «Проект»). Описания объектов – произвольные.

Создайте в каждом проекте 2 или более раздела (то есть, объекты типа «Раздел»). Создайте в проектах и разделах документы различных типов: спецификации, сборочные чертежи, документы проектные (то есть, объекты различных типов).

Присвойте созданным объектам атрибуты:

«Наименование», «Обозначение», «Шифр проекта».

Некоторые замечания по назначению и использованию данных атрибутов:

Наименование – данный предопределенный атрибут используется в подавляющем большинстве прикладных решений. В данной настройке используется для документов, проектов, разделов проектов. При создании прикладных решений для реального применения, часто требуются еще и дополнительные атрибуты сходного назначения, например: «Наименование полное», «Наименование сокращенное» и т.п. В учебной настройке достаточно использования имеющегося предопределенного атрибута.

Обозначение – краткое кодовое название документа, изделия, проекта и т.д. Используется в различных предметных областях, правила формирования могут быть самыми различными. Для учебной настройки будем считать, что обозначение должно формироваться в следующем виде:

Для проекта: ssss, где ssss – значение шифра проекта четыре любых символа;

Для раздела проекта: ssss-rr, где ssss – обозначение проекта, в который входит данный раздел, «-» – разделитель (тире), rr – обозначение собственно данного раздела;

Для документа обозначение будем формировать из обозначения родителя, разделителя и обозначения собственно документа («ddd»). Соответственно, обозначение для документа входящего в проект и в раздел будет строиться следующим образом:

ssss-ddd; ssss-rr-ddd

Аналогичные правила формирования обозначений часто встречаются на практике, в различных предметных областях, хотя и не являются единственно возможными.

Шифр проекта устанавливается одинаковый для данного проекта, его разделов и документов, входящих в данный проект и его разделы. Аналогичные атрибуты часто применяются в различных практических случаях, могут называться иначе (например, «Номер заказа» и т.п.). Подобные атрибуты удобно использовать для отбора требуемых элементов, относящихся к данному проекту. Применение единого шифра для проекта и, например, для договора на выполнение данного проекта облегчает прослеживаемость процесса реализации данного договора от его заключения до окончания действия.

Входимость для объектов документов учебной настройки – множественная (см. рисунок 15), можно **заимствовать** документы одного проекта в другой. На практике подобное применяется, например, для исходных данных, нормативных документов и др.

Используя метод drag-and-drop, перетаскивайте объект документа из одного проекта в другой (можно использовать объекты из одного или различных открытых окон; чтобы открыть выделенный объект в отдельном окне дерева связей, используйте верхнее меню «Объект» → «Открыть дерево проекта»). Убедитесь на различных типах объектов, что условие входимости работает, не позволяя добавлять объекты тех типов, которые не имеют входимости в данный тип объекта.

5.4.2 Справочник предприятия

Создайте учебный проект «Справочник предприятия»:

Добавьте в папку «Справочники» папку «Справочник предприятия». Предназначений у подобных справочников в реальных настройках, в зависимости от решаемого круга задач, может быть несколько:

- Использование в качестве электронного телефонного справочника предприятия – аналога бумажного внутреннего справочника;
- Поиск и выбор исполнителя, удовлетворяющего заданным условиям;

- Объекты сотрудников могут использоваться в качестве электронных «Личных кабинетов», то есть, быть связанными с выданными и полученными заданиями, рабочими документами и т.д.

В учебной настройке данный справочник позволит проиллюстрировать некоторые возможности системы.

Добавьте в папку справочника предприятия объекты подразделений (объекты типа «Подразделение»). Для учебной настройки достаточно для объектов подразделений использования только описаний объектов, соответствующих названиям подразделений, использовать атрибуты не обязательно.

Добавьте в объекты подразделений объекты сотрудников. На практике объекты, соответствующие в подобном справочнике сотрудникам, как правило, имеют немалый набор атрибутов. В числе таких атрибутов могут быть: «Фамилия», «Имя», «Отчество», «Дата рождения», «Пол», «Телефон местный», «Телефон городской» и т.д. В учебной настройке, на данном этапе её создания, сделаем набор атрибутов для объектов сотрудников минимальным. В учебных целях ограничимся описанием объекта, где укажем фамилию и инициалы сотрудника, и двумя атрибутами:

- «Должность основная» – атрибут объекта, соответствующий записанной в личном деле сотрудника должности;
- «Должность в данном подразделении» – атрибут проекта, соответствующий функциям, выполняемым данным сотрудником в данном подразделении.

Рассмотрим следующий учебный пример:

Пусть сотрудник Иванов работает в отделе автоматизации, является начальником данного отдела. В личном деле данного сотрудника, условно, указана должность «Инженер 1 категории». Одновременно данный сотрудник является заместителем генерального директора предприятия по автоматизации, куратором проектного отдела по применению средств автоматизации проектных работ, и еще руководителем народного театра.

Таким образом, объект сотрудника Иванова будет входить в несколько объектов подразделений, для каждого вхождения будет установлен свой атрибут проекта «Должность в данном подразделении».

Добавьте объект сотрудника Иванова в какое-либо из перечисленных подразделений (для учебной настройки высший менеджмент предприятия и народный театр будем рассматривать как подразделения). Установите атрибуты «Должность основная» и «Должность в данном подразделении». Добавьте объект сотрудника Иванова в остальные подразделения. Убедитесь, что значение атрибута объекта отображается одинаково для всех вхождений данного объекта сотрудника, а значения атрибута проекта для каждого вхождения установите для каждого вхождения необходимо устанавливать отдельно (см. рисунок 24).

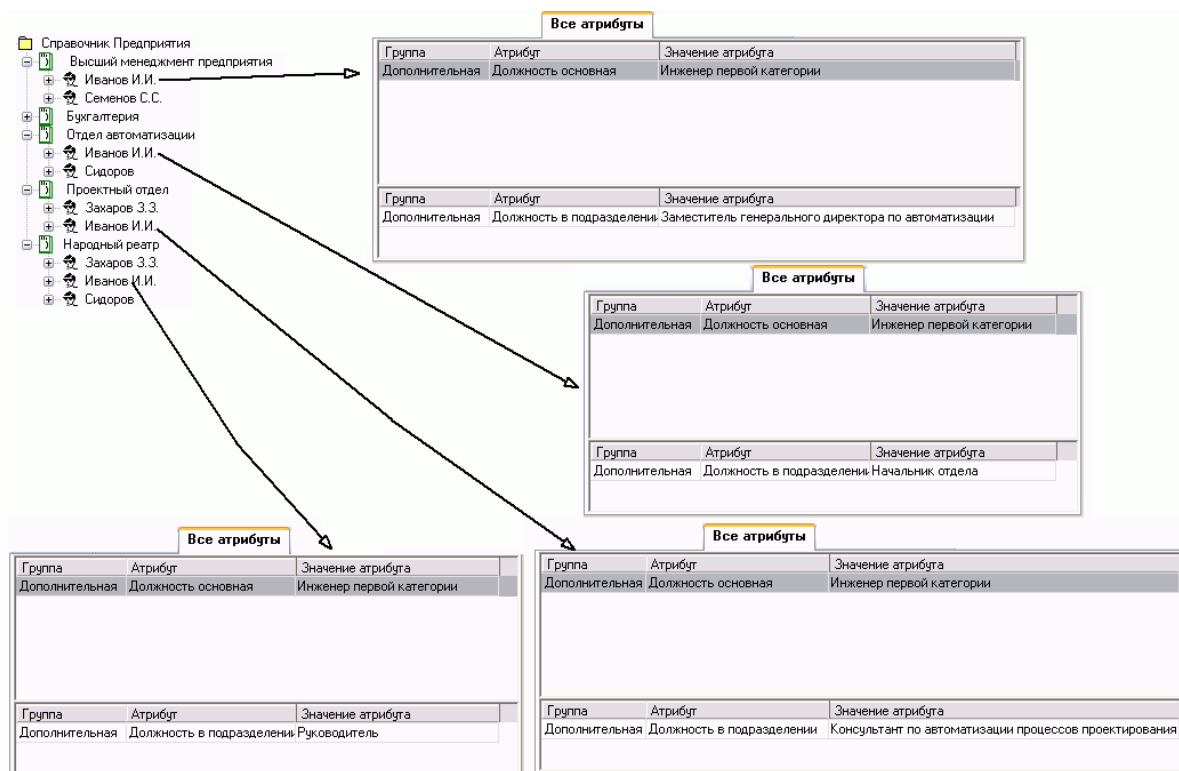


Рисунок 24 Настройка справочника предприятия – использование атрибутов проекта. Учебный пример

Измените у объекта значение атрибут объекта «Должность основная» на «Инженер высшей категории», убедитесь, что введенное значение изменилось для всех вхождений объекта.

Выделите объект сотрудника Иванова, входящий в отдел автоматизации. Измените значение атрибута «Должность в данном подразделении», убедитесь, что введенное значение изменилось только для данного вхождения объекта.

Откройте объект сотрудника Иванова в отдельном окне, (выделите объект в дереве, в верхнем меню «Объект» выберите «Открыть дерево проекта») перейдите на вкладку «Все атрибуты». Убедитесь, что для корневого объекта атрибуты проекта не отображаются.

5.4.3 Использование горизонтальных связей

Для ознакомления с использованием горизонтальных связей, установите равнозначную связь «Похожие» для созданных объектов проектных документов, и направленную связь «Прототип» для созданных объектов проектов. (Если данные типы связей в Вашей БД еще не созданы – создайте их – см. раздел 4.3.1).

Откройте вкладку «Связанные» для различных объектов, с которыми устанавливались горизонтальные связи, ознакомьтесь со списком связанных объектов для различных типов связи на данной вкладке. Двойным щелчком по строке в списке откройте связанный объект.

5.4.4 Использование дополнительной подчиненной связи

Постройте дерево проекта с использованием созданной подчиненной связи «Хранение бумажных документов» (Если данный тип связи в Вашей БД еще не создан – создайте его – см. раздел 4.3.1). Смысл использования данного типа связи в учебной настройке следующий: С точки зрения разработчика проекта, хранение документов проекта целесообразно организовать исходя из структуры данного проекта. С точки

зрения работника архива, хранение бумажных документов может быть удобно организовать исходя из физических размеров (формата) бумажного документа – крупноформатные документы требуют широких стеллажей и полок, документы меньшего размера хранятся на полках и стеллажах, соответственно, другого размера. На практике подобный способ организации архива бумажных документов используется достаточно часто, особенно, если один и тот же документ применяется в различных проектах или изделиях, то есть, заимствуется. Смоделируем данный способ хранения документов в учебной настройке.

Если это не было сделано ранее, создайте типы объектов «Стеллаж» и «Полка» (см. разделы 4.3.2, 4.3.3, 4.5).

В корневой объект «Информационные ресурсы предприятия» добавьте папку «Подлинники на бумажном носителе» (Рисунок 25).

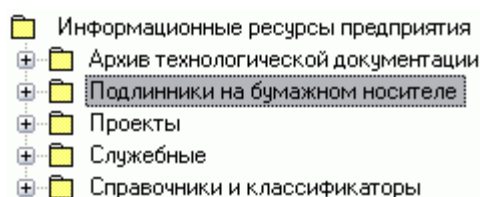


Рисунок 25 Добавлена папка «Подлинники на бумажном носителе»

Откройте добавленную папку в отдельном окне, измените тип связи данного окна на «Хранение бумажных документов». Окно общего дерева проектов оставьте открытым.

Добавьте в данную папку объекты стеллажей, в стеллажи добавьте полки (используйте тип связи «Хранение бумажных документов»).

Используя метод drag-and-drop, перетащите объекты документов из окна общего дерева проекта на соответствующие полки (Рисунок 26).

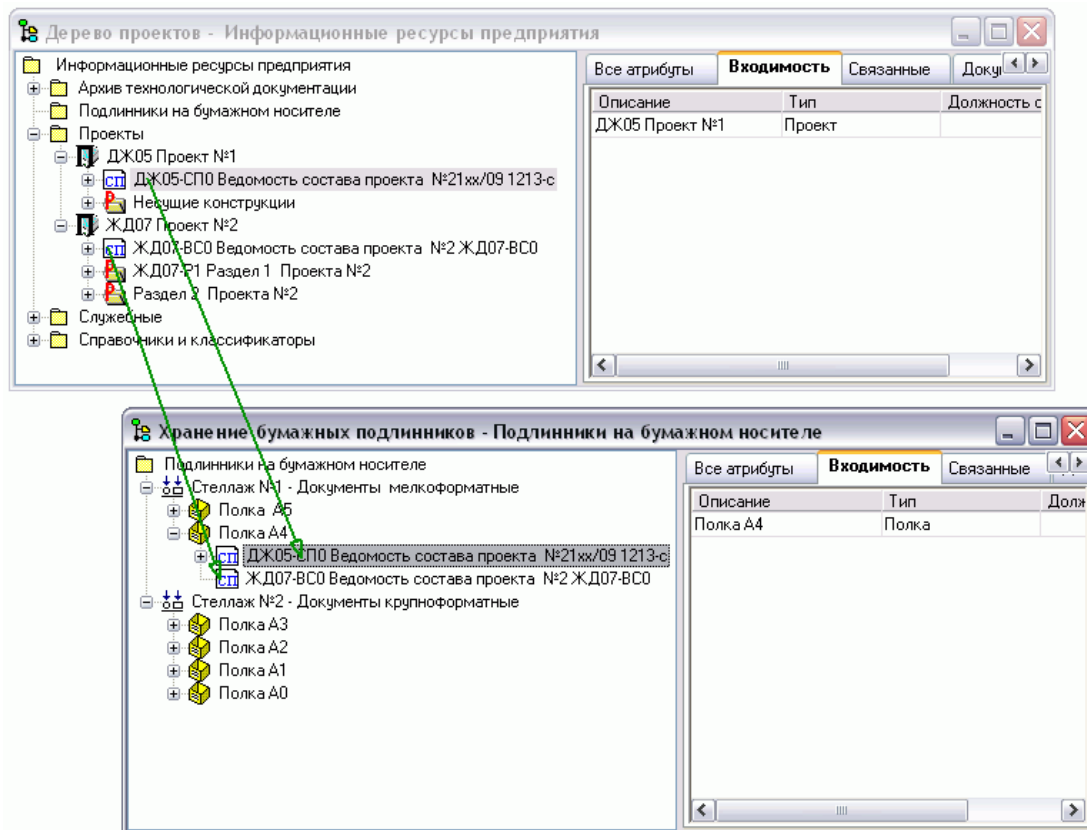


Рисунок 26 Создание проектов с использованием различных типов подчиненной связи

Таким образом, одни и те же объекты документов в учебной настройке структурируются различным образом – с точки зрения состава проекта и с точки зрения особенностей хранения бумажных документов. Скорее всего, информация о хранении бумажных документов интересна сотрудникам технического архива. Составом проекта будут пользоваться разработчики проектов и другие пользователи, которым, в свою очередь, информация о хранении не нужна. Использование общих объектов документов и различных типов связей для построения проектов позволяет обеспечить соответствующее представление информации для различных пользователей, избегая её дублирования.

Убедитесь, что входимость одних и тех же объектов по различным типам связей различная.

Отметим, что хотя рассмотренный пример создания и использования дополнительных типов связей можно считать условным (его польза заключается, главным образом, в простоте настроек и универсальности), после самостоятельной реализации данного учебного примера не должно возникать сложностей с созданием и применением дополнительных типов связей в настройках, предназначенных для практического применения.

5.5 Создание объектов и проектов. Вопросы для самопроверки

1. Имеют ли значение при создании нового объекта, не входящего ни в один проект, настройки входимости для данного типа объекта?
2. Объект не входит ни в один проект. Как установить для данного объекта атрибут проекта?
3. Каким образом (различными способами) можно добавить данный объект в несколько проектов?

6 Варианты, исполнения и копии объектов

6.1 Вариантное проектирование


На начальной стадии разработки проекта некоторые его составные части могут разрабатываться в нескольких вариантах. Впоследствии один из вариантов принимается за основной вариант. Остальные варианты, созданные в процессе проектирования, будут сохранены в базе данных, их можно будет просматривать. Аналогично, может потребоваться подготовить несколько вариантов одного и того же документа, а впоследствии один из вариантов принять в качестве основного.

Для отображения вариантности, в Lotsia PDM используется предопределенный тип объекта – «Варианты». Для реализации данной функции необходимо задать входимость для типа объекта «Варианты» – то есть, определить возможные типы родителей и потомков для объектов данного типа.

6.2 Создание вариантов

Для создания варианта следует:

- определить для типа объекта «Варианты» требуемые правила входимости («Главное меню»→ «Администрирование»→ «Структура данных»→ «Типы объектов»→ «Варианты»→ вкладки «Типы объектов» и «Входимость»...);
- выделить в дереве связей объект, для которого требуется создать вариант;
- в контекстном меню выберите пункт «Добавить вариант»;
- в открывшемся окне выбора объектов выберите нужный объект и щелкните на кнопке «Ок» (либо, для создания нового объекта нажмите «Создать», выберите тип создаваемого объекта и т.д.).

Объект, выделенный в дереве, будет заменен объектом типа «Варианты» , к которому будут добавлены объект из дерева и объект (объекты) из списка. Если у объекта имеются старые варианты, то все они будут также добавлены к объекту типа «Варианты».

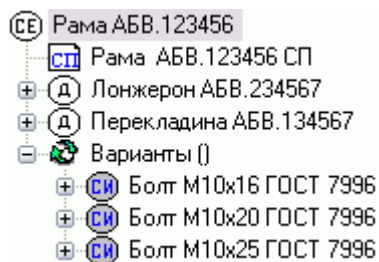


Рисунок 27. Вариант в дереве проекта. Пример

Для удаления варианта выделите в дереве связей объект, который входит в объект типа «Варианты» и удалите его из проекта.

Для принятия объекта как основного варианта:

1. выделите объект, которое находится на нижнем уровне объекта типа «Варианты»
2. в контекстном меню выберите «Принять как основной вариант».

Просмотр старых вариантов выполняется с помощью пункта «Список старых вариантов» из верхнего меню «Правка».

6.3 Создание исполнения объекта

Изделия, обладающие общими конструктивными признаками при некоторых различиях между собой, называются исполнениями. Подробнее об исполнениях – см. ГОСТ 2.113.

Создать исполнение можно только для объектов вида «Изделие». Создаваемое исполнение является копией исходного изделия. Различие – в значении атрибута «Обозначение», в котором появляется (изменяется на «1») суффикс (например, -01, -001). Число разрядов суффикса («Маска номера исполнения в обозначении») задается в «Общих параметрах БД» (Главное меню → Общие параметры БД → Вкладка «Параметры» → параметр «Маска номера исполнения в обозначении»). При создании исполнения оно попадает в список исполнений, который доступен через верхнее меню «Объект» → «Список исполнений...».

Создание исполнения выполняется в следующей последовательности:

- выделите объект в дереве связей;
- выберите пункт «Создать исполнение» из верхнего меню «Объект»;
- в открывшемся окне задайте параметры нового объекта, который будет исполнением, нажмите «Ok».

В списке исполнений (верхнее меню «Объект» → «Список исполнений...») базовое исполнение отмечено пиктограммой в первом столбце.

6.4 Создание копии объекта

Данная функция может оказаться полезной в случае, когда пользователь создает идентичные объекты (с одинаковыми параметрами и одинаковым набором атрибутов объекта).

При копировании объекта создается новый объект (с новым ID), подобный исходному объекту. Новый объект имеет такой же набор атрибутов объекта. Документы архива и права доступа к объекту не копируются.

Если копируемый объект имел дочерние объекты, копия объекта будет иметь те же самые входящие объекты (входящие объекты получают дополнительно к имеющимся еще одну входимость – в созданную копию). Таким образом, проект и его копия имеют общий состав объектов. (Но, если для каких – либо входящих в копируемый объект дочерних объектов была задана единичная входимость, то в составе копии этих объектов не будет).

Для создания копии объекта:

- выделите объект, копию которого вы хотите создать;
- выберите пункт «Создать копию объекта» верхнего меню «Объект»;
- в открывшемся окне введите описание создаваемого объекта-копии;
- если необходимо, отредактируйте созданный объект.

Для создания копий проектов используется копирование по шаблону. Вначале необходимо создать шаблон копирования (Главное меню → «Администрирование» → «Структура данных» → «Шаблоны копирования проектов»). Далее создается и выполняется действие над объектами, которое и осуществляет копирование по шаблону. Информация о настройке шаблонов копирования содержится в документации и в контекстной справке по системе Lotsia PDM PLUS.

6.5 Резюме

Поддерживается вариантное проектирование – объект или проект может включаться в добавляемый на его место в проекте объект типа «Варианты». Параллельно в тот же объект «Варианты» могут добавляться другие объекты. При завершении вариантного проектирования один из вариантов принимается за основной, остальные варианты и объект «Варианты» при этом из проекта убираются, но остаются доступными в списке старых вариантов. Для использования вариантов, для типа объекта «Варианты» следует настроить правила входимости.

Возможно создание копий объектов.

При копировании объекта с потомками, для исходного объекта создается копия, а для потомков – дополнительная входимость (связь с копией).

Возможного копирование проекта целиком, по различным правилам. В этом случае требуется настройка и использование шаблона копирования.

Поддерживается создание исполнений (для изделий). При создании исполнения в атрибут «Обозначение» очередного исполнения добавляется порядковый номер исполнения.

6.6 Варианты, исполнения, копии. Учебное задание

Создайте несколько вариантов проектного документа, входящего в состав раздела проекта, примите один из вариантов в качестве основного, просмотрите список старых вариантов. Верните старые варианты, для чего добавьте в качестве варианта текущий документ (принятый ранее в качестве основного).

Отметим, что предварительно для предопределенного типа объекта «Варианты» необходимо будет настроить правила входимости, обеспечивающие возможность включения данного типа объекта в соответствующее место в дереве проекта: для рассмотренного случая – см. рисунок 28.

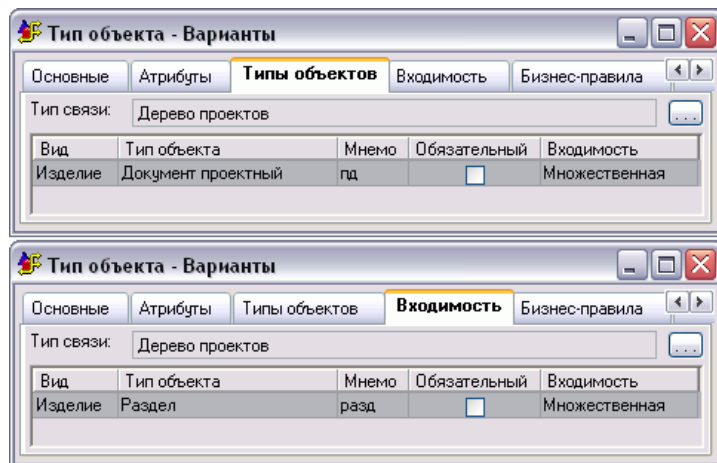


Рисунок 28 Настройка входимости для вариантов. Пример

Для создания исполнения используйте объект вида «Изделие», содержащий атрибут «Обозначение» и другие атрибуты. Рекомендуется* произвести настройку предопределенных типов объектов, например, детали и/или сборочной единицы. Создав исполнение, убедитесь, что значение атрибута «Обозначение» созданного исполнения отличается от обозначения базового исполнения наличием номера исполнения,

* Создание исполнения не является необходимым в случае, если в Вашей предметной области данное понятие не используется.

присвоенного в соответствии с настройкой маски номера исполнения в обозначении. Убедитесь, что значения остальных атрибутов объекта исполнения совпадают с исходными значениями соответствующих атрибутов. Убедитесь, что состав созданного объекта исполнения изначально совпадает с составом базового исполнения (различия могут быть только в случае использования единичной входимости для потомка первого уровня). Путем удаления/добавления составных частей созданного исполнения убедитесь, что состав (набор потомков первого уровня) исполнения не влияет на состав базового исполнения и может от него отличаться. Откройте список исполнений данного изделия, убедитесь в наличии в данном списке имеющихся исполнений изделия.

Создайте копию проектного документа, убедитесь, что при этом создан новый объект (например, в контекстном меню выделенного объекта выберите «Свойства...», на вкладке «Основные» проверьте значения ID объектов, убедившись, что они различны для исходного объекта и созданной копии).

Создайте копию раздела проекта, убедитесь, что входящие в раздел объекты документов получили дополнительную входимость – входят как в исходный объект раздела проекта, так и в созданную копию.

6.7 Варианты, копии, исполнения. Вопросы для самопроверки

- Какие настройки входимости следует выполнить для использования вариантов?
- Для каких видов объектов могут создаваться исполнения?
- Чем копия объекта изделия отличается от его исполнения?
- Как (где) настраивается маска номера исполнения в обозначении?
- Как просмотреть список исполнений; старых вариантов?
- Что происходит с потомками (первого уровня, последующих уровней) объекта при создании его варианта, исполнения, копии?

7 Импорт структуры БД

Программа позволяет импортировать элементы структуры БД из текстовых файлов – таблиц с разделителями – символами табуляции. В некоторых случаях данная возможность позволяет ускорить создание элементов структуры БД или сделать её более наглядной (обозримой в целом).

Для осуществления импорта необходимо подготовить файлы импорта – см. пп. 26.1, 26.2, 26.3, 26.4, 26.5, 26.6 (данные файлы не должны содержать заголовков).

Подготовить файлы импорта можно как с использованием ручного ввода, так и выгрузив информацию из таблиц имеющейся БД-прототипа, структуру которой вы хотите воспроизвести в той или иной мере. Описания таблиц БД находятся в файлах Таблицы БД.doc и db_structure.pdf, расположенных в каталоге \Manuals Вашего дистрибутивного диска. Для описания элементов структуры БД вам потребуются следующие таблицы:

- lsdbo.attrib – атрибуты;
- lsdbo.attrib_group – группы атрибутов;
- lsdbo.business_attrib2object – соответствие «Атрибут – тип объекта»;
- lsdbo.object_type – типы объектов;
- lsdbo.business_part_rule – соответствие «Тип объекта – тип объекта»;
- lsdbo.link_type – типы связей

Выгруженные данные можно отредактировать (подготовить к импорту), используя, например средства Microsoft Excel.

После окончания подготовки файлов импорта выберите «Главное меню» → «Администрирование» → «Структура данных» → «Импорт структуры данных».

Программа откроет окно импорта структуры данных (см. рисунок 29).

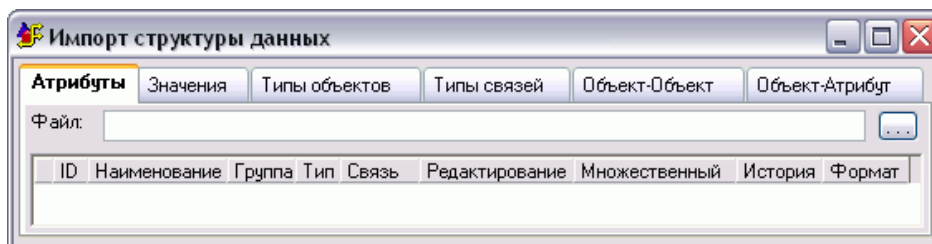


Рисунок 29 Импорт структуры данных

Данное окно состоит из 6 вкладок, соответствующих подготовленным файлам импорта элементов структуры. Далее необходимо:

- загрузить файл импорта (выбор файла с помощью кнопки справа от поля «Файл»);
- записать данные в БД (пункт «Записать данные в БД» верхнем меню «Импорт»)

Может быть пропущен любой этап импорта структуры базы данных, если имеющаяся структура достаточна для дальнейшего импорта.

7.1 Резюме

Имеется стандартный инструмент, позволяющий осуществлять, полностью или частично, импорт элементов структуры из текстовых файлов стандартного формата.

7.2 Импорт структуры БД. Учебное задание

Ввиду тривиальности данной операции, нет необходимости выполнения в учебных целях импорта структуры БД в полном объеме (тем более что подготовка файлов импорта вручную может потребовать значительного времени). Рекомендуется ограничиться, например, импортом нескольких атрибутов в учебную БД, предварительно сделав резервную копию БД, после чего восстановить БД из резервной копии и продолжить работу.

Для случая использования локальной БД Sybase, для создания резервной копии достаточно средствами ОС скопировать файл БД и лог-файл (из Lotsia PDM PLUS необходимо предварительно выйти, иначе указанные файлы будут заняты программой). Для восстановления БД из резервной копии, соответственно, скопируете файлы резервной копии на их исходное место. Для других случаев – для резервного копирования и восстановления БД – используйте средства Вашей СУБД.

Создайте, например, с использованием Microsoft Excel, таблицу, содержащую импортируемые атрибуты, сохраните её как текстовый файл с разделителями табуляции.

Формат колонок указанной таблицы – см. п. 26.2. Для учебных целей достаточно будет скопировать из электронной версии настоящего документа строки из таблицы (см. Таблица 4) без строки заголовка.

Выполните импорт атрибутов, используя указанный файл, убедитесь, что импортированные атрибуты добавлены в БД.

7.3 Вопрос для самопроверки

Каким образом с использованием процедуры импорта структуры передать часть атрибутов из одной БД в другую?

8 Работа с проектом

8.1 Поиск объектов

Для поиска и выбора информационных объектов в БД могут использоваться штатные средства Lotsia PDM – окно **поиска** и окно **выбора** объектов.

Работа пользователя с данными окнами во многом сходна, но есть некоторые отличия:

- окно поиска позволяет формировать и выполнять поисковые запросы, осуществлять открытие одного или нескольких найденных объектов. После открытия найденного объекта, окно поиска остается открытым и работа с ним может быть продолжена
- окно выбора позволяет из найденных объектов выбрать и открыть единственный требуемый объект, после чего указанное окно автоматически закрывается. При открытом окне выбора доступны только кнопки данного окна и недоступно верхнее меню.

Для открытия окон поиска и выбора используются пункты главного меню «Поиск объектов» и «Дерево проекта», либо соответствующие кнопки инструментальной панели.

Указанные окна содержат вкладки: «Выбор», «Последние объекты», «Атрибуты», «Дерево проектов», «Доп. условия». При открытии окна активна вкладка «Выбор» (Рисунок 30).

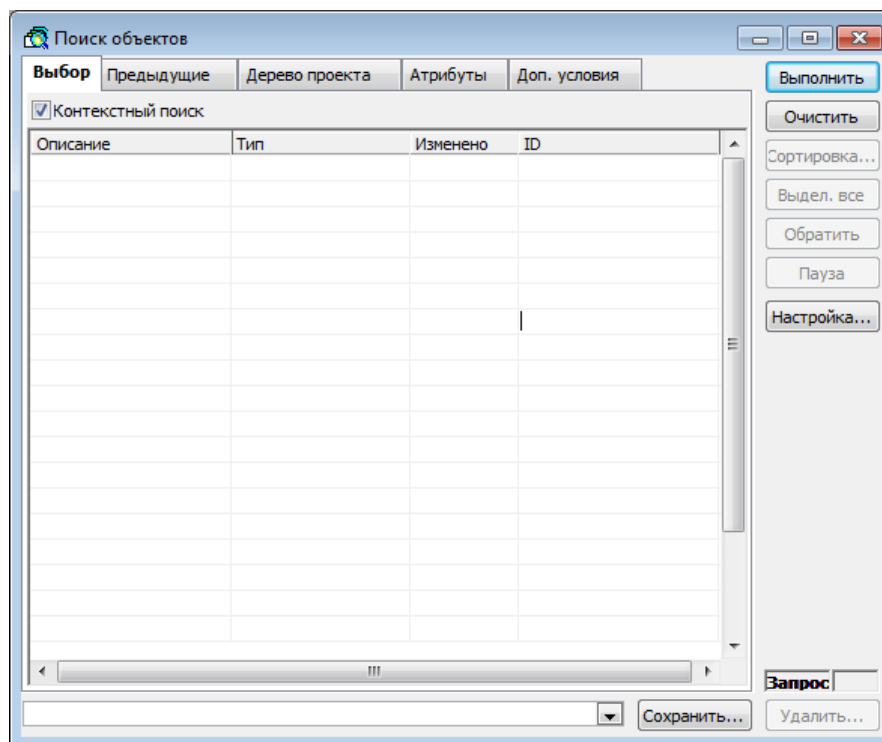


Рисунок 30 Окно "Поиск объектов". Вкладка "Выбор"

На вкладке «Дерево проектов» пользователь имеет возможность выбирать объект путем навигации по открытым деревьям проектов (Рисунок 31).

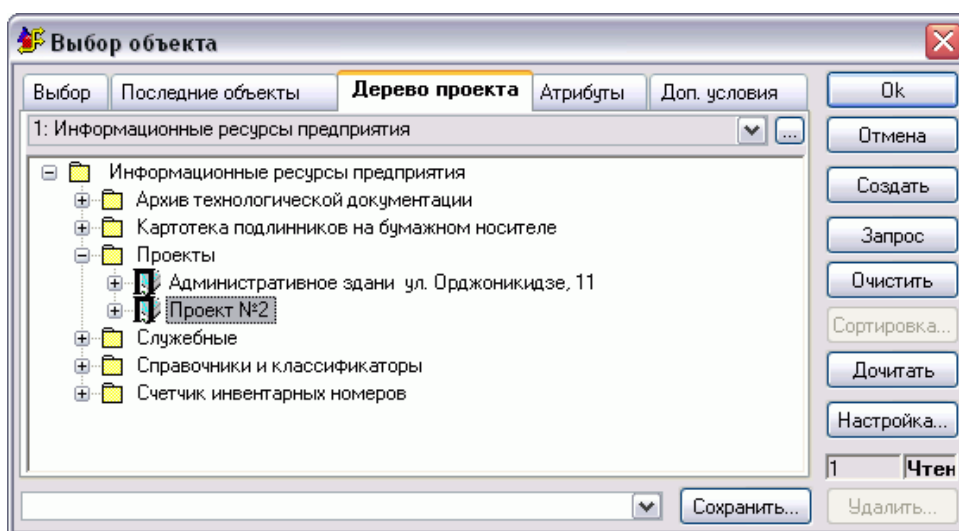


Рисунок 31 Вкладка «Дерево проекта» Окна выбора объекта. Пример

На вкладке «Последние объекты» хранится список 40 последних объектов, для которых открывалось окно проекта.

Для осуществления поиска объектов используются запросы. Запросы могут формироваться пользователем, либо, использоваться ранее созданные и сохраненные в системе запросы. После ввода запроса, по нажатию кнопки «Выполнить» происходит выполнение запроса. Результат выполнения запроса отображается в виде отдельных строк, соответствующих найденным объектам (Рисунок 32).

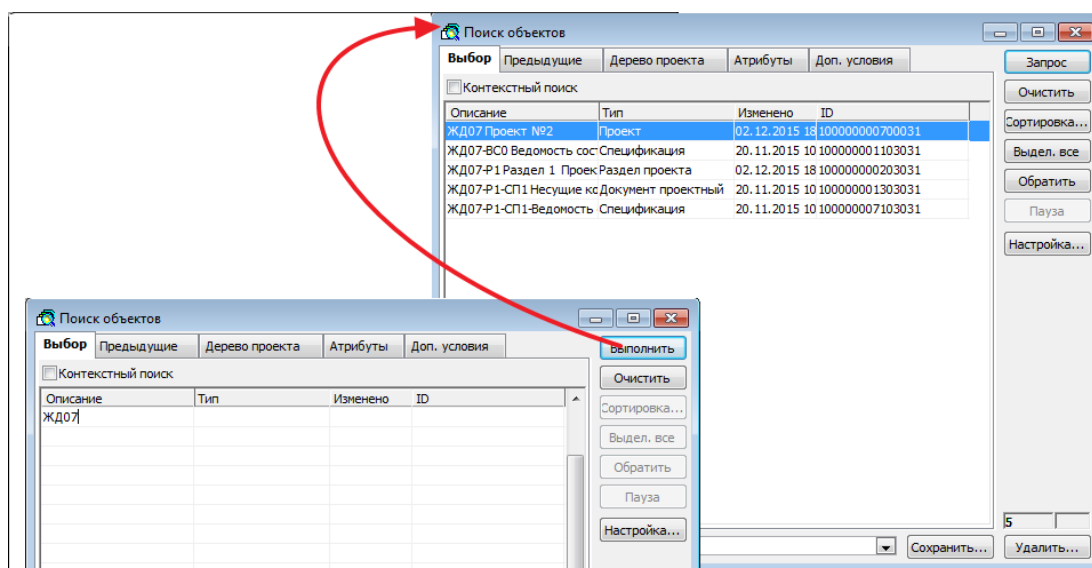


Рисунок 32. Запрос и результат его выполнения. Пример

Для ввода запросов могут использоваться вкладки «Выбор» и «Атрибуты». Отметим, что данные вкладки могут настраиваться – на вкладке «Выбор» можно задать отображаемые колонки, а для вкладки «Атрибуты» могут быть настроены поисковые формы. Указанные настройки будут рассмотрены позднее – см., соответственно, разделы 10.3.10 и 10.2.

На вкладке «Доп. условия» могут быть заданы ограничения по входимости искомым объектов.

Если условие запроса не составлено, то считается, что оно отсутствует.

Поисковые значения могут вводиться с клавиатуры, выбираться из списков или вставляться из буфера обмена.

На вкладке «Выбор» поиск осуществляется по колонкам «Описание» и «Тип», а также по любым колонкам, добавленным в результате настройки данной вкладки (настройка окон будет рассмотрена позднее – см. раздел 10.3.10). Для выполнения поиска следует ввести текст запроса в указанные колонки и нажать кнопку «Выполнить».

Для поиска по описанию объекта и строковым атрибутам во многих случаях достаточно использовать простейшие запросы – ввод в соответствующее поле начала или фрагмента описания или типа объекта. При поиске по фрагменту следует включить флажок «Контекстный поиск». Это равносильно указанию перед искомым значением символа «%» (этот служебный символ заменяет набор любых символов). Служебный символ «_» заменяет любой один символ.

Регистр символов в запросе не учитывается, то есть, строчные и прописные буквы – на результат поиска не влияет. Более подробно о составлении запросов см. раздел «Запросы» Руководства пользователя программы Lotsia PDM.

Для составления более сложных запросов можно использовать контекстное меню (Рисунок 33), всплывающее при нажатии правой кнопки мыши. Пункт «Значения» контекстного меню вызывает хранящийся в базе данных список значений, соответствующий текущей колонке. В этом списке можно выделить одно значение и вставить в текст запроса.

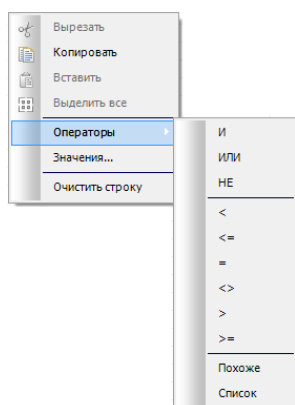


Рисунок 33 Контекстное меню Окна "Поиск объектов"

На вкладке «Атрибуты» пользователь может вводить условия запроса по значениям атрибутов объектов (а также же их типа, вида, мнемо, описания). Возможно использование настраиваемых поисковых форм. В качестве поисковых используются настраиваемые формы типа «Атрибуты». После настройки такой поисковой формы (см. п. 10.2) её следует подключить к профилю пользователя как поисковую форму.

8.2 Навигация по дереву проекта

В правой части окна дерева связей отображаются вкладки, содержимое которых соответствует выделенному объекту. На вкладке «Объекты», для выделенного в дереве объекта отображается список потомков первого уровня (Рисунок 34).

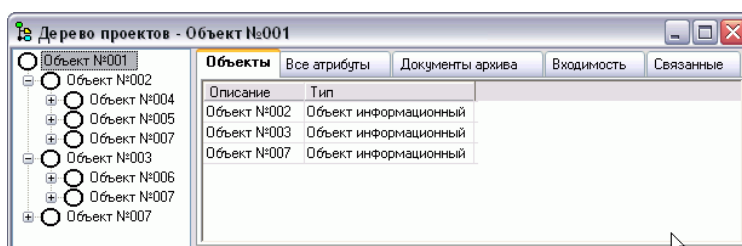



Рисунок 34 Окно проекта, вкладка «Объекты»

По двойному щелчку левой клавишей мыши по строке в списке на вкладке «Объекты», в окне дерева проектов текущий объект соответствующим образом меняется.

В верхней инструментальной панели (при текущем окне дерева связей) имеется блок кнопок , обеспечивающих, соответственно: «Назад» – возврат назад (на предыдущий выделенный объект), «Вперед» – перемещение вперед – выделение объекта, с которого вы переместились кнопкой «Назад», «Перейти на один уровень вверх» – обеспечивает перемещение вверх по дереву связей.

На вкладке «Входимость» отображается список родителей первого уровня для выделенного в дереве объекта (Рисунок 35).

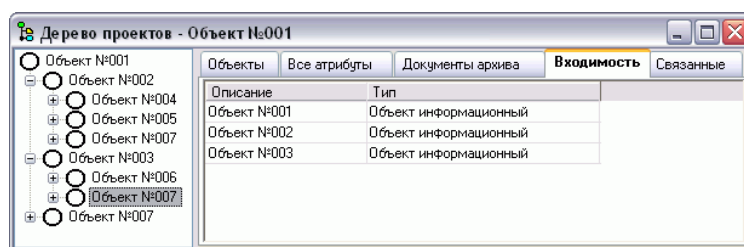


Рисунок 35 Окно проекта, вкладка «Входимость»

По двойному щелчку левой клавишей мыши по строке в списке на вкладке «Входимость», открывается новое окно дерева проектов, в котором соответствующий объект будет корневым.

Таким образом, пользуясь вкладками «Объекты» и «Входимость» можно, помимо получения информации о составе проекта и родителях данного объекта, перемешаться, соответственно, вниз и вверх от выделенного объекта по дереву.

Указанные вкладки могут также использоваться для изменения состава проекта: добавления в проект новых объектов или удаления объектов из проекта (будет рассмотрено далее, см. п. 8.4, Рисунок 42). Данные вкладки – настраиваемые. Настройки указанных вкладок будут рассмотрены в разделе 10.4

На вкладке «Связанные», либо в окне «Связанные объекты» отображается список объектов, связанных с выделенным объектом горизонтальной связью. Тип горизонтальной связи выбирается в поле «Тип связи» в верхней части вкладки или окна (Рисунок 36, Рисунок 37).

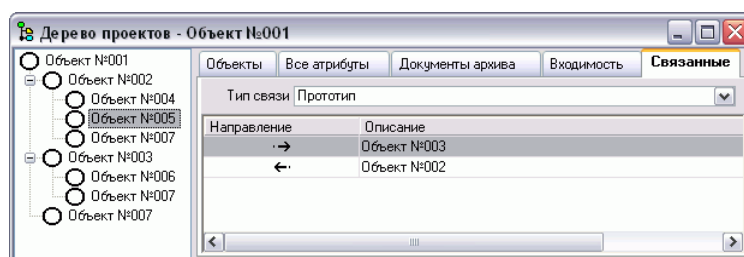


Рисунок 36 Окно проекта, вкладка «Связанные»

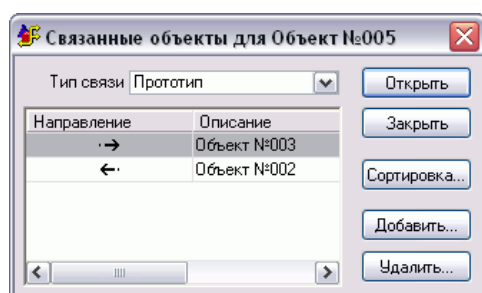


Рисунок 37 Окно «Связанные объекты»

Для открытия окна «Связанные объекты» следует: выделить объект в дереве → Верхнее меню «Объект» → «Связанные объекты...».

По двойному щелчку левой клавишей мыши по строке в списке на вкладке «Связанные» либо в окне «Связанные объекты», открывается новое окно дерева проектов, в котором соответствующий объект будет корневым.

Вкладка «Связанные» и окно «Связанные объекты» настраиваемые. Настройки указанного окна и вкладки будут рассмотрены в разделе 10.3.10

8.3 Подборка объектов

Подборка – произвольный набор объектов, организованный пользователем. Подборки могут быть персональными или общими. Персональная подборка видна только пользователю, создавшему данную подборку. На общую подборку назначаются права доступа, пользователи получают к подборке доступ согласно имеющимся у них правам.

Помещение объекта в одну или несколько подборок не изменяет входимости и других свойств объекта.

Подборка служит еще одним (кроме дерева связей) средством организации объектов.

Для открытия подборки используйте пункт Главного меню «Подборка объектов». В открывшемся окне выберите подборку из списка, либо для создания новой подборки нажмите «Создать».

Откроется окно новой подборки (см. рисунок 38)

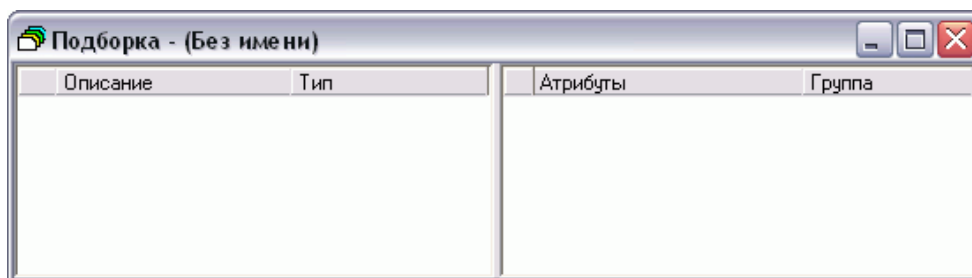


Рисунок 38 Созданная подборка

В зависимости от нажатия кнопок инструментальной панели, внешний вид подборки может изменяться (Рисунок 39).

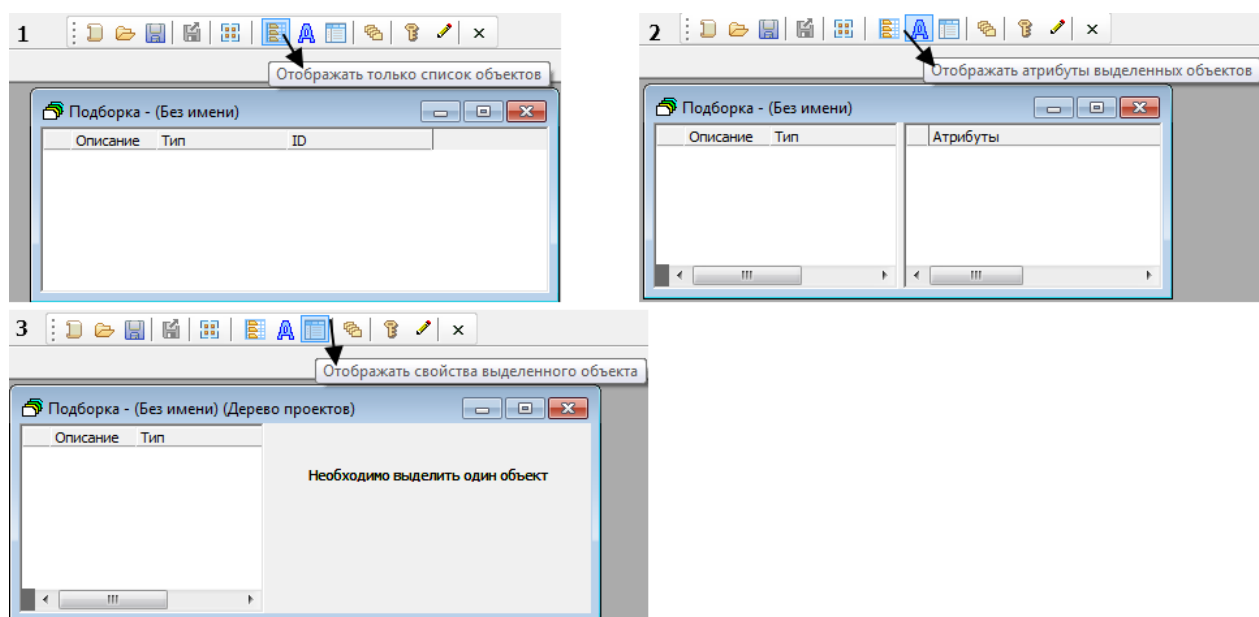


Рисунок 39 Внешний вид окна подборки в зависимости от нажатой кнопки инструментальной панели

При нажатой кнопке «Отображать только список объектов» (Рисунок 39) (1) отображается только левая половина окна подборки (там располагается список объектов, помещенных в данную подборку). Список отображаемых по умолчанию колонок настраивается через профили пользователей (либо настройки пользователя): «Объекты» → пункт «Списки объектов» → вкладка «Подборка». Могут быть добавлены колонки с атрибутами объекта (задумайтесь, почему при настройке не доступны атрибуты проекта?) и некоторой другой информацией об объекте. Сортировка объектов в подборке осуществляется щелчком левой клавишей мыши по заголовку столбца (в окне подборки), по которому вы хотите отсортировать объекты. Повторный щелчок приведет к обратной сортировке.

При нажатой кнопке «Отображать атрибуты выделенных объектов» (2) в правой части окна подборки отображается список атрибутов, имеющихся у помещенных в подборку объектов (на рисунке подборка пустая, список атрибутов также пуст).

При нажатой кнопке «Отображать свойства выделенного объекта» (3) в правой части окна подборки будут отображаться вкладки выделенного объекта, аналогично вкладкам в окне дерева связей. Объект выделяется в левой части окна подборки (для отображения вкладок должен быть выделен только 1 объект).

Помимо отображаемых по умолчанию в левой части окна подборки колонок, для каждой подборки может использоваться собственная настройка колонок. Настройка колонок для данной подборки задается в свойствах подборки: при выделенном окне подборки, Верхнее меню «Файл» → «Свойства» → вкладка «Форма» (Рисунок 40).

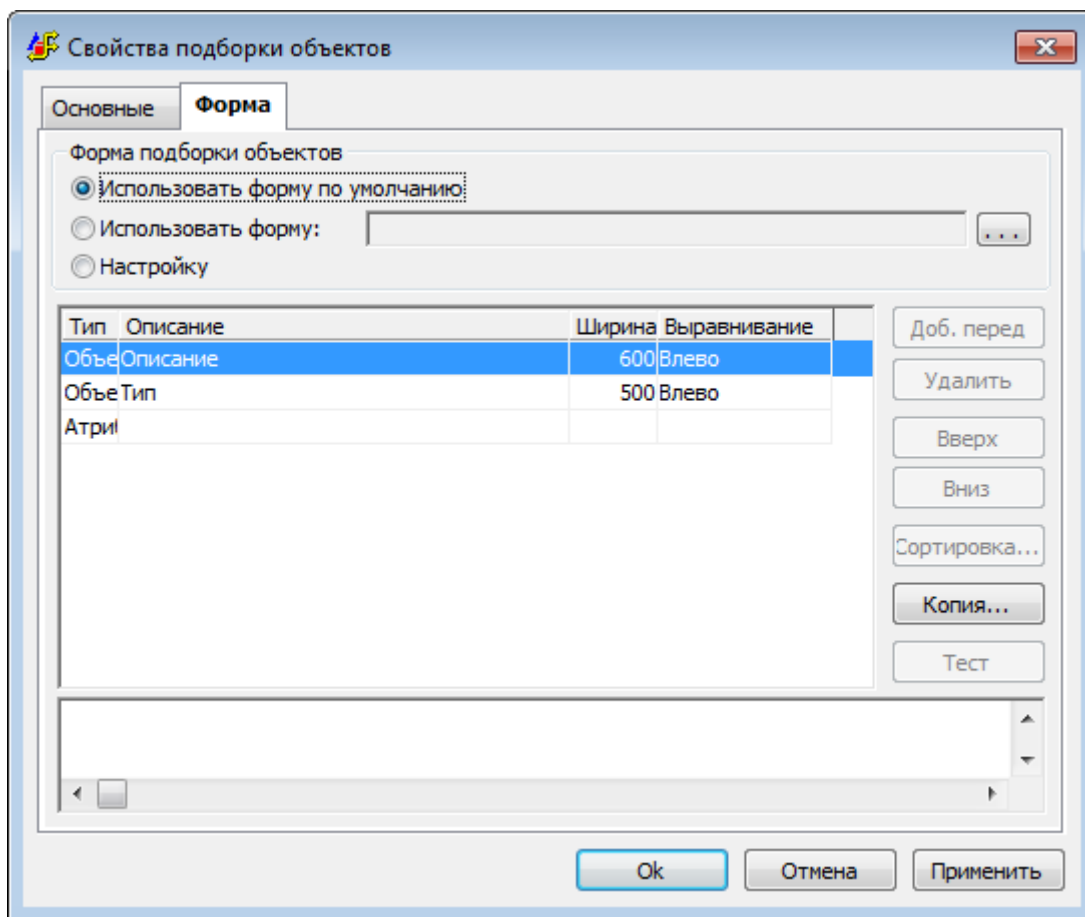


Рисунок 40 Настройка колонок левой части окна для данной подборки

При положении переключателя «Использовать форму по умолчанию» отображаются колонки «по умолчанию» согласно настройкам профиля пользователя (либо настройкам пользователя).

При положении переключателя «Использовать форму» для данной подборки будет использована указанная справа форма (указать можно только форму типа «подборка»). Данная форма должна быть создана предварительно, средствами редактора форм (см. п. 10.2).

При положении переключателя «Настройку», колонки данной подборки задаются непосредственно в данном окне (Рисунок 40).

Для помещения выделенного в окне дерева связей объекта в открытую подборку можно использовать метод drag-and-drop.

Можно поместить в подборку проект целиком, для чего следует выделить в открытом проекте объект, имеющий в своем составе другие объекты и перетащить выделенный объект в подборку, удерживая **правую** клавишу мыши. После отпущения правой клавиши мыши, в открывшемся меню выбрать пункт «Добавить все». В подборку будет добавлен выделенный объект и все объекты, в него входящие.

Для помещения объектов в подборку также используются пункты контекстного меню левой части окна подборки «Добавить объект...», «Добавить проект...».

Возможна работа с подборками с использованием действий над объектами и скриптов.

Подборка позволяет обрабатывать одновременно один или несколько выделенных в ней объектов. Можно, одновременно для выделенных объектов подборки:

- редактировать (добавлять, удалять, изменять) атрибут
- открывать в окнах проекта выделенные объекты (проекты)
- выполнить отчет или действие, запускать предопределенный процесс, отправить сообщение
- удалить из БД (только Администратору, только объекты, не входящие ни в один проект и не имеющие документов архива)

Для выделенных в подборке объектов можно устанавливать и удалять атрибуты, используя контекстное меню правой части окна подборки (см. рисунок 41).

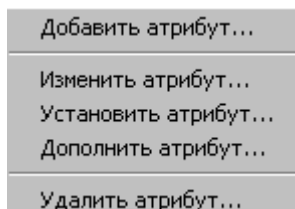


Рисунок 41 Контекстное меню правой части окна подборки

С помощью пункта «Дополнить атрибут...» можно изменить значение выбранного атрибута только у объектов, не имеющих данного атрибута. Если у объекта атрибут существует, то он не будет изменен.

8.4 Удаление объекта из проекта (дерева связей) и БД

Для удаления объекта из проекта его следует выделить его в дереве связей и выбрать пункт «Удалить из проекта» контекстного меню, либо нажать <Delete>.

Удалить объекты из проекта также можно используя вкладки «Входимость» удаляемых объектов, либо «Объекты» родительских объектов. На данных вкладках отображаются, соответственно, родители удаляемых объектов или сами удаляемые объекты. Выделив требуемые сроки на указанных вкладках, для удаления объектов из проектов используйте соответствующие пункты контекстного меню (Рисунок 42).

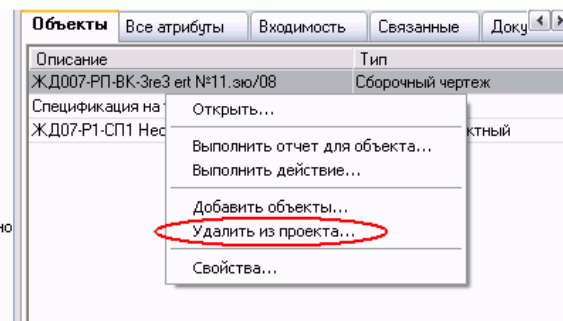
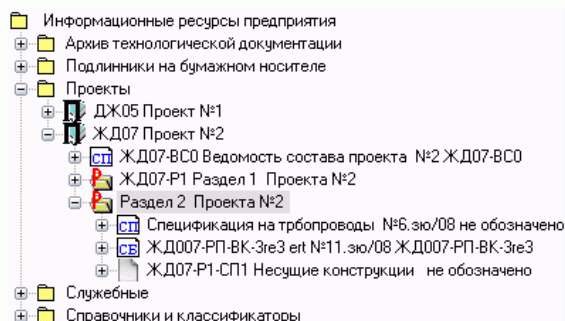
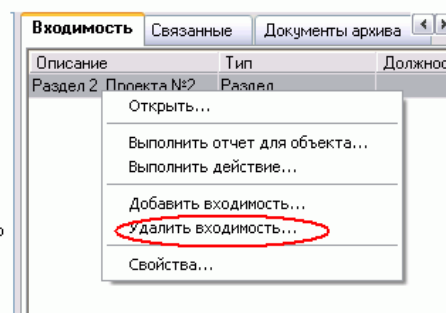
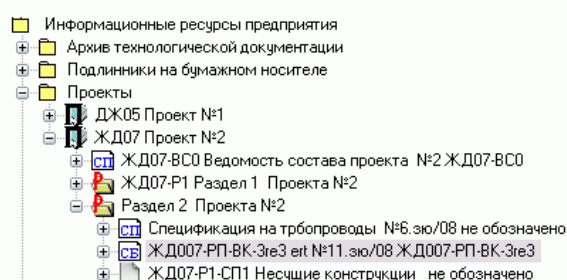
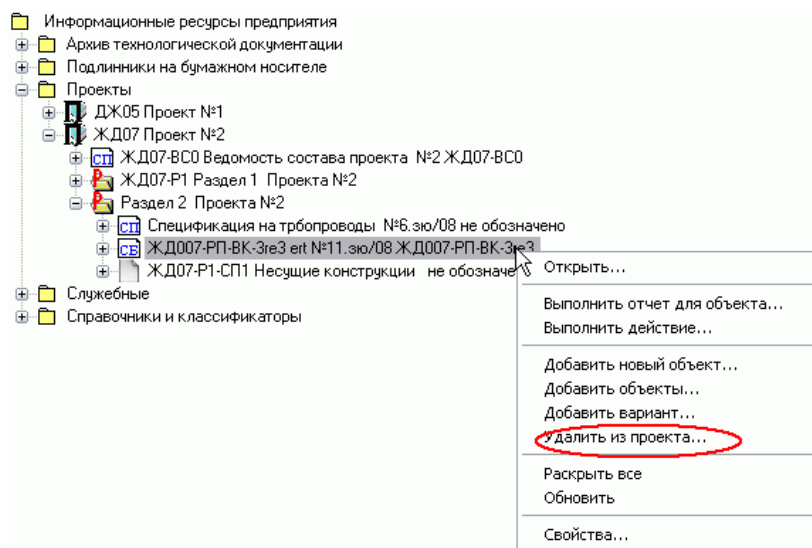


Рисунок 42 Различные способы удаления одного и того же объекта из проекта

Удаление из проекта, другими словами, это удаление подчиненной связи между двумя объектами. Если у объекта нет входящих подчиненных связей – такие объекты не входят ни в один проект, но сами могут содержать другие объекты.

Для удаления объектов из БД необходимо поместить удаляемые объекты в подборку, затем выделить их в подборке и выбрать пункт «Удалить объекты из БД» из верхнего меню «Правка».

В открывшемся Окне с предупреждением надо щелкнуть на кнопке «Да». Далее программа производит проверку наличия у удаляемых из базы данных объектов входимости в другие проекты, а также документов архива и документов архива, помеченных на удаление. Вам будет выданы подтверждения удаления для каждой из этих ситуаций. После удаления объектов из базы данных программа откроет Окно с сообщением об успешном удалении объектов.

Удалить экземпляры объектов из базы данных может только Администратор.





8.5 Сравнение состава проектов и атрибутов

Часто в практической работе приходится иметь дело с незначительно отличающимися по составу проектами (например, исполнениями изделия). При этом трудно визуально определить, если проект достаточно сложный, чем эти проекты различаются. Для решения подобных задач в системе Lotsia PDM имеются функции сравнения их состава и атрибутов.

8.5.1 Сравнение состава проектов

Сравнение состава проектов (входящих объектов) выполняется в следующей последовательности:

- следует выделить объект в дереве связей;
- используя **правую** кнопку мыши, выделенный объект следует перетащить на другой объект в этом же или другом дереве связей (другом Окне проекта);
- в появившемся контекстном меню выбрать пункт «Сравнить проекты».

Сравнение происходит по первому уровню. Сравнимые объекты помечаются пиктограммой с зеленой стрелкой . Несовпадающие объекты (разные) помечаются пиктограммой . Один и тот же объект, но с разным значением атрибута (например, «Количество») помечается пиктограммой . Один и тот же объект с одинаковыми атрибутами помечается пиктограммой .

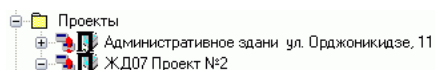
Для удаления дополнительных пиктограмм нажмите  на панели инструментов.

8.5.2 Сравнение атрибутов двух объектов

Сравнение атрибутов двух объектов дерева выполняется в следующей последовательности:

- следует выделить объект в дереве связей;
- используя **правую** кнопку мыши, выделенный объект следует перетащить на другой объект в этом или другом дереве связей;
- в появившемся контекстном меню выбрать пункт «Сравнить атрибуты».


Программа поместит дополнительные пиктограммы в дереве связей перед объектами, и на вкладке «Все атрибуты» перед названиями атрибутов (см. рисунок 43).



Группа	Атрибут	Значение атрибута
Дополнительная	 Статус	Разрабатывается
Основная	 Шифр проекта	1213
	 Обозначение	1213

Группа	Атрибут	Значение атрибута
Дополнительная	 Статус	Разрабатывается
Основная	 Шифр проекта	ЖД07
	 Наименование	Проект №2
	 Обозначение	ЖД07

Рисунок 43. Результаты сравнения атрибутов двух объектов.

Одинаковые значения атрибутов помечаются пиктограммой . Неодинаковые значения атрибутов помечаются пиктограммой .

8.6 Резюме

Для поиска и выбора объектов в БД могут использоваться окно поиска и/или окно выбора объектов. При использовании данных окон возможен поиск объекта по отдельным характеристикам объектов или комбинации характеристик, по дереву проекта, в списке последних открывавшихся объектов и по наличию или отсутствию связей объекта.

Возможно создание поисковых запросов и их сохранение для последующего использования.

Поиск объектов может выполняться также навигацией по дереву проектов, на соответствующих вкладках выделенного в дереве объекта отображается информация о потомках и родителях первого уровня, а также об объектах, связанных с данным объектом горизонтальными связями.

Пользователи могут включать объекты в создаваемые подборки.

Подборки могут быть персональными (только для автора) или общими. Доступ пользователей к общей подборке осуществляется в соответствии с установленными им правами.

Подборка может использоваться для работы с выделенным объектом аналогично окну дерева связей, но в формах (на вкладках) типа «Атрибуты» не отображаются атрибуты проекта.

Подборки могут использоваться для выполнения различных операций одновременно с несколькими выделенными в них объектами.

Пользователь может удалять объекты из проектов (в пределах имеющихся прав доступа). Администратор может удалить объекты из БД. Предварительно такие объекты помещаются в подборку.

Имеется возможность визуального сравнения состава проектов и значений атрибутов.

8.7 Работа с проектом. Учебное задание

На вкладке «Выбор» окна поиска введите начало описания имеющегося в БД объекта, нажмите «Выполнить», убедитесь в правильной работе поиска. Откройте один или несколько найденных объектов. Нажмите «Очистить».

Перейдите на вкладку «Дерево проекта» окна поиска, откройте объект, выбранный в дереве на данной вкладке.

Перейдите на вкладку «Доп. условия», переключатель «Входимость объектов» установите в положение «Объекты, не входящие ни в один проект». Нажмите «Выполнить». По результатам поиска оцените «замусоренность» Вашей БД не входящими в состав проектов объектами.

Выполните поиск по атрибуту. На вкладке «Атрибуты» введите запрос (пример – см. рисунок 44, сформируйте запрос исходя из содержимого Вашей БД) и нажмите кнопку «Ok». Результат поиска приведен на том же рисунке.

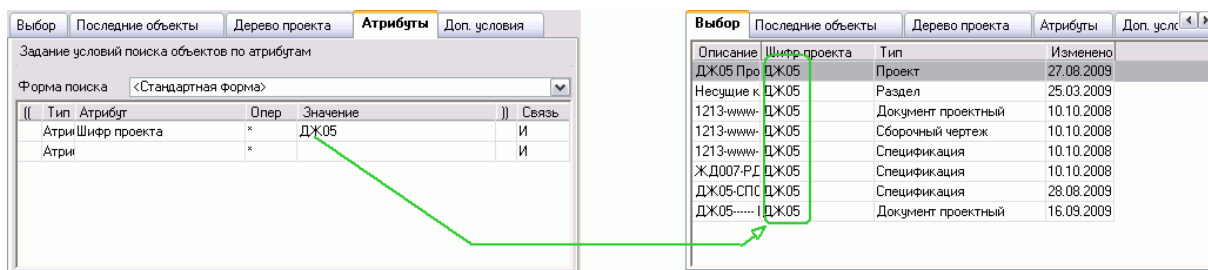


Рисунок 44. Поиск объекта с использованием запроса. Пример

Для перемещения по дереву проекта откройте любой из созданных проектов, выделите корневой объект в левой части окна. Перейдите на вкладку «Объекты», сделайте двойной щелчок левой клавишей мыши по строке в списке дочерних объектов, убедитесь, что соответствующий объект стал выделенным в дереве связей.

Перейдите на вкладку «Входимость». Сделайте двойной щелчок левой клавишей мыши по строке в списке родительских объектов, убедитесь, что открылось окно дерева связей выбранного Вами родителя.

Для ознакомления с использованием подборок объектов – создайте подборку, поместите туда проект целиком. Установите для выделенных в подборке документов проекта атрибут «Статус» – «Разрабатывается».

В учебных целях удалите из БД «лишние» (ошибочно созданные) объекты. Если «лишние» объекты отсутствуют – предварительно создайте несколько объектов различных типов, в присвоении им атрибутов нет необходимости, описания могут быть произвольными. Для удаления из БД «лишних» объектов поместите их в подборку, для чего используйте контекстное меню левой части окна подборки, пункт «Добавить объект...». В открывшемся окне выбора, на вкладке «Доп. условия» установите переключатель в положение «Объекты, не входящие ни в один проект», нажмите «Выполнить». В списке найденных объектов выделите все, кроме корневого объекта (Информационные ресурсы предприятия). Нажмите «Ок» для добавления выделенных объектов в подборку. Теперь объекты можно удалять из БД (см. п. 8.4).

В целях ознакомления с использованием инструментов «Сравнение проектов» и «Сравнение атрибутов», выполните сравнение состава проектов и сравнение атрибутов для двух различных объектов.

8.8 Работа с проектом. Вопросы для самопроверки

1. Как найти объект в БД, если известно его наименование?
2. Как найти все объекты, не входящие в проекты?
3. Как получить информацию о:
 - a. родителях данного объекта?
 - b. потомках первого уровня?
 - c. объектах, связанных указанной горизонтальной связью с данным объектом?
4. Как поместить объект в подборку? Как поместить в подборку проект целиком?
5. Как удалить объект из проекта?
6. Кто и каким образом может удалить объект из БД?

9 Классификаторы. Возможности и особенности реализации

Классификатор – структурированный список заранее определенных значений атрибутов, служит для присвоения атрибутам значений, сформированных с использованием данного списка.

В Lotsia PDM PLUS Классификатор – проект, обычно состоящий из объектов типа «Классификатор» (имеется в виду именно predetermined тип объекта, ID типа объекта -12), имеющих атрибуты с заданными значениями. Такой проект следует предварительно создать. Все объекты классификатора должны иметь присваиваемые с помощью данного классификатора атрибуты.

Классификатор работает:

- для стандартной вкладки «Все атрибуты» в режиме редактирования;
- в формах типа «Атрибуты» в режиме редактирования (см. п.10.2.1);
- в формах действий над объектами (см. п. 17.3.4).

Чтобы значение атрибута выбиралось из классификатора, необходимо при настройке данного атрибута в поле «Редактирование» указать «Классификатор», в поле «Классификатор» указать соответствующий, заранее созданный проект (корневой объект) классификатора (Рисунок 45).

Атрибут - Отрасль хозяйства

Основные | Значения | Типы объектов | Уникальность значений

Описание: Отрасль хозяйства

Группа: Информация о клиенте

Тип данных: Строка ☐ Множественный

Связь: Объект ☐ История изменений

Редактирование: Классификатор ☐ Скрытый

Классификатор: Отрасли хозяйства

Формат:

Доступ: Общедоступный

ID: 100000358444037

Создал: Администратор

Изменил: Администратор

Изменено: 16.10.2007 13:01:00 PM: 3

Рисунок 45 Подключение атрибута к классификатору

Классификатор может использоваться для присвоения атрибутов объекта и атрибутов проекта.

Классификатор позволяет:

- выбрать значение из списка или сформировать его вручную, полностью или частично (см. п.9.1);
- организовывать структурированный список значений атрибута любой вложенности (см. п. 9.2);
- формировать значение атрибута, последовательно собирая его из определенных в ветви классификатора значений (см. п. 9.3);
- формировать значение атрибута из нескольких параллельных, заранее определенных ветвей классификатора, в заданной последовательности (см. 9.4);
- присваивать одновременно значения нескольким атрибутам (см. 9.5);

- присваивать данному атрибуту (атрибутам) значения, с использованием одного и того же, или различных классификаторов для объектов различных типов. Для использования различных классификаторов, в зависимости от типа объекта, необходимо на вкладке «Типы объектов» окна свойств данного атрибута указать классификаторы для типов объектов (Рисунок 46).

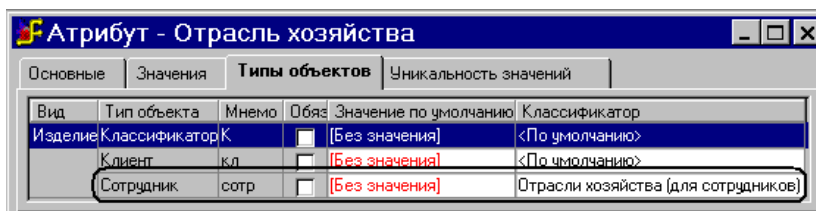


Рисунок 46 Задание отдельного классификатора для типа объекта

Указанные возможности могут комбинироваться по желанию пользователя.

Обычно проект классификатора формируется из predetermined объектов типа «Классификатор», корневым должен быть объект типа «Классификатор».

Использование других типов объектов в проекте классификатора возможно.

Ограничение на использование в проекте классификатора объектов других типов заключается в том, что при настройке такого классификатора, при установке атрибутов с редактированием – «Классификатор», для подобных объектов будет открываться окно классификатора. В отличие от других типов объектов, при редактировании атрибутов для объекта типа «Классификатор» (предetermined код ID типа объекта = -12), стиль редактирования атрибута игнорируется и всегда используется свободный ввод.

Использование действий (будет рассмотрено далее, см. п. 16) для установки значений атрибутов делает данное ограничение несущественным.

Все возможности классификатора можно использовать комплексно, определяющими критериями являются сложность формирования атрибута, количество возможных вариантов значения атрибута и комбинаций значений атрибутов. Учитывается также количество обрабатываемых объектов и атрибутов и особенности стоящей перед пользователем задачи, в части обеспечения требуемой скорости и гибкости работы с классификатором, а так же правильный учет возможности дальнейшего развития системы.

9.1 Формирование значение атрибута из списка возможных значений и ручным вводом

Для реализации данной возможности создается проект «Классификатор», с объектами, имеющими требуемый атрибут с присвоенными значениями.

Таким классификатором можно пользоваться как списком значений, подключая его к различным типам объектов, к одному атрибуту для разных типов объектов можно подключить разные классификаторы (см. рисунок 46).

Для обеспечения возможности ручного ввода значения атрибута, в соответствующий объект классификатора, в формируемый атрибут, вводится {\$} (использовать фигурные скобки). Во время использования классификатора, при выборе этого объекта, пользователь получает возможность ввести значение атрибута вручную.

Можно также использовать выражение вида {\$x}, что обеспечивает возможность ручного ввода значения не более заданной длины, где x – число, определяющее количество знаков, которые могут быть введены.

Если значение атрибута вручную надо формировать лишь частично (например, начало, окончание, промежуточную часть), то, в данный атрибут соответствующего объекта классификатора вводится значение следующего вида:

- $\{\$ \}$ постоянное_значение – ручной ввод перед постоянным значением;
- постоянное_значение $\{\$ \}$ – ручной ввод после постоянного значения;
- постоянное_значение1 $\{\$ \}$ постоянное_значение2 – ручной ввод между двух постоянных значений и т.п.

где «постоянное_значение» – содержимое постоянной части значения атрибута;

Ручной ввод осуществляется после окончания выбора значений атрибутов.

9.2 Организация структурированного списка возможных значений атрибута.

Для реализации данной возможности создается проект «Классификатор» с соответствующей структурой. Окончания ветвей имеют атрибуты с присваиваемыми значениями (также см. п. 9.1). Остальные объекты дерева служат для структурирования списка значений (построения дерева), имеют соответствующий атрибут со значением $\{\}$ – не обрабатывается, или $\{.\}$ – обязательно перейти на следующий уровень (вниз). В значениях атрибутов классификатора следует использовать именно фигурные скобки. Использование символов $\{.\}$ не позволит при использовании данной ветви классификатора закрыть окно классификатора, ограничившись присвоением значения, соответствующего данному объекту классификатора, заставляя пользователя продолжить присвоение значения на следующем уровне классификатора.

9.3 Последовательное формирование значений атрибута.

Для реализации данной возможности создается проект «Классификатор» с соответствующей структурой – на каждом иерархическом уровне объекты классификатора содержат атрибут, из значений которого формируется итоговое значение данного атрибута. При движении по дереву классификатора от корня к концу ветви, формируемое значение атрибута «сцепляется» из значений соответствующего атрибута из каждого задействованного объекта классификатора, расположенного в текущей ветви (за исключением служебных символов, то есть, расположенных внутри фигурных скобок – $\{\}$).

9.4 Формирование значения атрибута из нескольких параллельных, заранее определенных ветвей классификатора, в заданной последовательности

Если значение атрибута формируется из нескольких частей, не связанных между собой, а указанные части значения формируются из различных веток дерева классификатора, чтобы не усложнять общее дерево классификатора, последовательность формирования значения атрибута может быть задана в атрибуте корневого объекта классификатора в следующем виде:

$\{\text{Описание}_1\}$ постоянное_значение1 $\{\text{Описание}_2\}$ постоянное_значение2 ...и т.д.,
где:

Описание_1 - содержимое поля «Описание» корневого объекта, (входящего в данный классификатор), ниже которого в иерархии расположены объекты, служащие для формирования соответствующей части атрибута

Объекты классификатора («Описание_1», «Описание_2» и т.д.) находятся ниже корневого объекта и не входят друг в друга.

Для формирования полного значения атрибута необходимо выбрать объект классификатора в каждой из ветвей («Описание_1», «Описание_2» и т.д.).

Последовательность выбора объектов в различных ветвях классификатора («Описание_1», «Описание_2» и т.д.) не влияет на последовательность расположения частей в формируемом значении атрибута, (она определена в корневом объекте общего классификатора).

Если корневые объекты классификаторов («Описание_1», «Описание_2» и т.д.), имеют значение атрибута {.,}, то задание части значения атрибута, формируемой соответствующим классификатором, станет обязательным – то есть, станет невозможно выйти из классификатора, не сформировав соответствующей части значения атрибута.

9.5 Одновременное присвоение значения нескольким атрибутам.

Если объекты классификатора имеют несколько атрибутов, то значения нескольким атрибутам объекта можно присваивать с использованием одного и того же классификатора (при условии, что для данных атрибутов настроено редактирование – «Классификатор», и в настройках указан один и тот же классификатор для всех этих атрибутов).

Присвоение значений нескольким атрибутам, подключенным к общему классификатору, по умолчанию осуществляется одновременно. Если необходимо присвоить значение только отдельным атрибутам, необходимо отключить «лишние» флажки – см. рисунок 48.

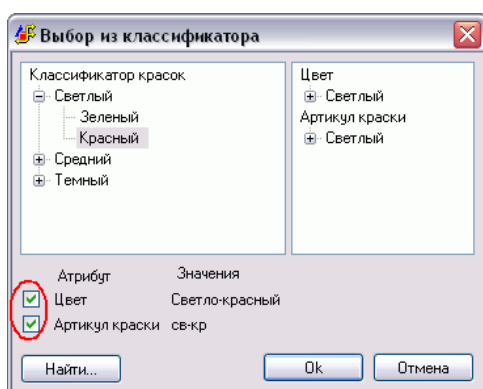


Рисунок 47 Использование классификатора для формирования значений двух атрибутов одновременно

Использование одного объекта классификатора для одновременного формирования значений нескольких атрибутов возможно при наличии жесткого соответствия между значениями различных атрибутов. В том случае, если значению одного атрибута соответствуют несколько (список) возможных значений другого атрибута, следует построить дерево так, чтобы в начале определялся первый атрибут, а потом, на более низком уровне – следующий.

9.6 Использование классификатора в действиях над объектами

Подобное использование для присвоения значений переменным и/или выбора объекта, возможно, применяется, но имеет некоторые особенности – будет рассмотрено далее, см. п. 17.3.4.

9.7 Преимущества и ограничения использования классификатора

К преимуществам классификатора можно отнести простоту настройки, в том числе – возможность применения без написания действий. Указанные преимущества для

создания настоек, используемых конечными пользователями, редко можно считать определяющими.

Ограничения применения классификатора, в основном, следующие:

1. Окно классификатора не настраиваемое, его внешний вид изменить невозможно.
2. Другое ограничение – при выборе значения из классификатора всегда сцепляются значения редактируемых атрибутов из объектов – родителей выбранного объекта классификатора, для родителей всех уровней дерева классификатора. То есть, отсутствует возможность получения атрибута только из выделенного объекта классификатора, находящегося на произвольном уровне в дереве, при наличии родителей, которые содержат атрибуты со значениями, отличными от служебных.
3. Еще одно ограничение – невозможность выбора в окне классификатора сразу более одного объекта.
4. Назначение классификатора – именно присвоение значения атрибуту (атрибутам). При попытке использовать (в действии) окно классификатора для выбора объекта в дереве связей, приходится дополнительно настраивать средства поиска объекта по считанному с выбранного объекта значению атрибута, что не всегда реализуемо приемлемым (удобным для конечного пользователя) образом.

В большинстве случаев в форме действий вместо использования классификатора удобнее применять шаги типа «Окно выбора объектов из дерева» или «Окно выбора объектов из списка» – будет рассмотрено далее: см. п. 17.4. Также вместо классификатора может использоваться выбор объекта, с настройкой редактирования переменной типа «объект» требуемым образом (будет рассмотрено далее: см. п. 17.3.1.3, пример подобной настройки – см. рисунок 136). **На сегодня указанные инструменты обычно применяются для использования конечными пользователями системы взамен классификаторов.**

9.8 Резюме

Классификатор – структурированный список, используемый для присвоения значений одному или нескольким атрибутам. Классификатор представляет собой проект, все объекты которого содержат данный атрибут. Для построения проекта классификатора используется предопределенный тип объекта «Классификатор», в обоснованных случаях допустимо использование объектов других типов.

Для подключения классификатора к атрибуту, необходимо выполнить настройку свойств данного атрибута.

Настройкой проекта классификатора обеспечивается возможность последовательного формирования значения атрибута, а также ручного ввода значения атрибута или его части.

Классификатор может также использоваться для присвоения значений переменным в формах действий над объектами.

На текущем этапе развития Lotsia PDM PLUS, ввиду ограничений классификатора, для работы конечных пользователей вместо классификатора обычно применяются другие средства.

9.9 Настройка классификатора. Учебное задание

Для тренировки и лучшего понимания данной темы, рекомендуется настроить небольшой классификатор произвольного назначения, на котором попробовать последовательно различные возможности данного инструмента. Если затруднительно

придумать несложный классификатор для Вашей предметной области, можно воспользоваться следующим учебным примером:

Пусть из классификатора требуется присвоить два соответствующих друг другу атрибута, например: «Цвет» и «Артикул краски».

Структура данного классификатора для атрибута «Цвет» может быть следующая:

Выбор оттенка из возможных значений, например:

- Светлый
- Средний
- Темный

Далее, для выбранного оттенка, выбираем собственно цвет (для всех оттенков может быть одинаковый или различный список):

- Красный
- Зеленый
-

Для реализации данной настройки создайте атрибуты «Цвет» и «Артикул краски» (см. п. 4.3.4)

Для обоих создаваемых атрибутов:

Тип данных – «Строка»

Связь – «Объект»

Редактирование – «Свободный», либо «Классификатор» без указания классификатора (пока классификатор не создан, указать его не представляется возможным).

Сопоставьте созданным атрибутам следующие типы объектов (см. п. 4.3.5):

1. Классификатор;
2. несколько типов объектов, встречающихся в Вашем дереве связей – например: папка, полка, стеллаж, деталь, сборочная единица и т.д., – объектам данных типов должно быть логично присваивать указанные атрибуты (в учебных целях последним замечанием можно и пренебречь)

Постройте дерево классификатора (Рисунок 48). Корневой объект данного дерева, для удобства его нахождения в дальнейшем, включите в дерево основных проектов (например – в папку «Справочники и классификаторы»).

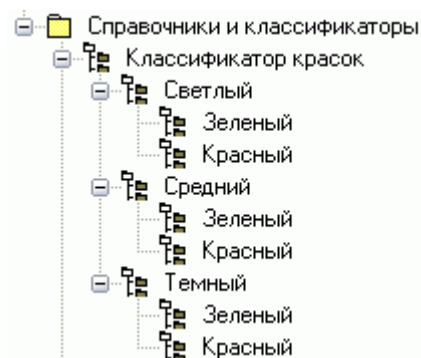



Рисунок 48 Дерево учебного классификатора

Тип всех объектов данного дерева – «Классификатор». Входимость типа «Классификатор» в тип «Классификатор» по умолчанию имеется. Чтобы можно было включить созданное дерево классификатора в общий проект – добавьте для типа объекта «Классификатор» входимость в тип объекта «Папка» по связи «Дерево проектов» (см. п. 4.3.3).

Все объекты типа классификатор () данного дерева должны содержать 2 атрибута: «Цвет» и «Артикул краски». Значения атрибутов могут быть следующими:

Название объекта в дереве – см. рисунок 48	Значение атрибута «Цвет»	Значение атрибута «Артикул краски»
Классификатор красок	{}	{}
Светлый	Светло-	св-
Средний	Средне-	ср-
Темный	Темно-	тм-
Зеленый	зеленый	зл
Красный	красный	кр

Использовать множественную входимость объектов, в данном случае – нижнего уровня, в классификаторах обычно нецелесообразно. Например, для данного случая – при изменении артикула светло – зеленой краски, скорее всего не должен аналогично меняться артикул темно – зеленой и средне – зеленой красок. Вместо множественной входимости, при построении дерева данного классификатора удобно использовать копирование объектов (см. п. 6.4) нижнего уровня.

Для использования данного классификатора, подключите к нему атрибуты.

В свойствах атрибута «Цвет» (Главное меню → Администрирование → Структура данных → Атрибуты → выберите из списка атрибут «Цвет»). На вкладке «Основные», в поле «Редактирование» укажите «Классификатор». В появившемся при этом поле «Классификатор» укажите требуемый объект классификатора – в данном случае «Классификатор красок». Сохраните изменения.

В учебных целях присвойте атрибут «Цвет», например, любому из объектов типа «Папка» (либо объекту другого типа, из тех, кому сопоставлен атрибут «Цвет», но не объекту типа «Классификатор»). При присвоении значения атрибуту откроется окно классификатора. Последовательно делая двойной щелчок левой клавишей мыши по объектам дерева классификатора, присвойте значения атрибуту (Рисунок 49).

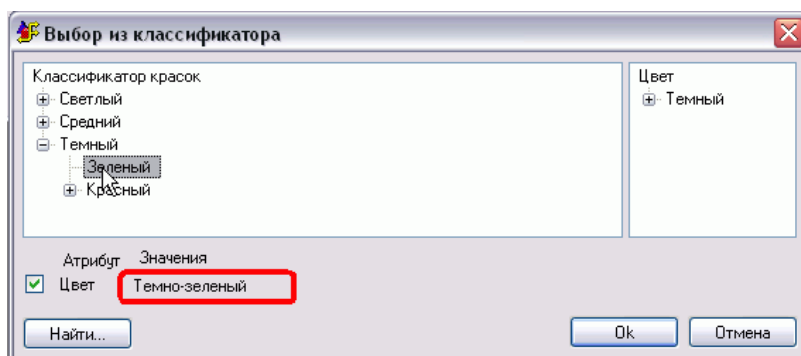


Рисунок 49 Присвоение значения выбором из классификатора. Учебный пример

Рекомендуется, меняя настройку учебного классификатора, на практике ознакомиться со всеми возможностями, перечисленными в начале раздела 9 настоящего документа.

9.10 Классификатор. Вопросы для самопроверки

- Как организовать классификатор с возможностью ручного ввода части значения атрибута?
- Как установить максимальное количество знаков при ручном вводе значений в классификаторе?
- Как можно выбором из классификатора одновременно присвоить значение трем атрибутам?

10 Настройка интерфейса

К настройкам интерфейса можно отнести настройку Главного меню, форм для объектов, внешнего вида стандартных окон и вкладок, контекстного меню объектов, а также верхнего меню, настройку отображаемых в дереве названий объектов, дерева почтовых папок (рассмотрено далее, см. п. 22.2), фильтры и представления проектов, привязки элементов интерфейса к типам объектов.

Кроме того, настраивается внешний вид (конфигурация) окна программы: могут использоваться (или не использоваться) вкладки для открытых окон программы, а также использоваться (не использоваться) закреплённая панель (будет рассмотрено в разделе 10.3.8).

Интерфейс пользователя обеспечивают также так же ряд других настроек, в том числе формы действий (будет рассмотрено далее см. раздел 16) и документооборота (будет рассмотрено далее см. раздел 23).

Применение большинства настроек интерфейса для пользователей определяется в профилях пользователей (рассмотрено далее, см. п. 10.3). Часть настроек выполняется непосредственно редактированием профиля пользователя.

В настройках профиля могут быть разрешены персональные настройки отдельных элементов, в этом случае для указанных элементов пользователя, входящие в данный профиль, могут создавать и использовать персональные настройки.

10.1 Настройка Главного меню и соответствующей инструментальной панели

Конечному пользователю системы не требуется большинство стандартных пунктов Главного меню, поэтому из пользовательских Главных меню такие «лишние» пункты удаляются. Вместе с тем, пользователю часто требуется выполнять в системе свои функции (создание и редактирование требуемых объектов, проектов и атрибутов; запуск работ и выполнение отчетов), реализуемые с помощью действий (создание действий будет рассмотрено далее, см. п. 16). Выполнять некоторые действия пользователям удобнее с использованием созданных для них дополнительных пунктов Главного меню.

При настройке Главного меню следует помнить, что полное (стандартное) Главное меню пользователя, в отличие от полного Главного меню администратора, не содержит папки «Администрирование». Если пользователю требуются пункты меню из данной папки, в Главном меню пользователя следует создать новую папку, и включить туда требуемые пункты. Для обеспечения возможности использования некоторых пунктов Главного меню, пользователю могут потребоваться соответствующие привилегии (будет рассмотрено далее, см. п. 11).

Для создания Главного меню используйте: «Главное меню» → «Администрирование» → «Настройка интерфейса» → «Настройка Главного меню» → кнопка «Создать».

В открывшемся окне настройки меню, с использованием контекстного меню левой части окна (Рисунок 50) можно создавать дополнительные пункты меню, папки, удалять и перемещать имеющиеся пункты.

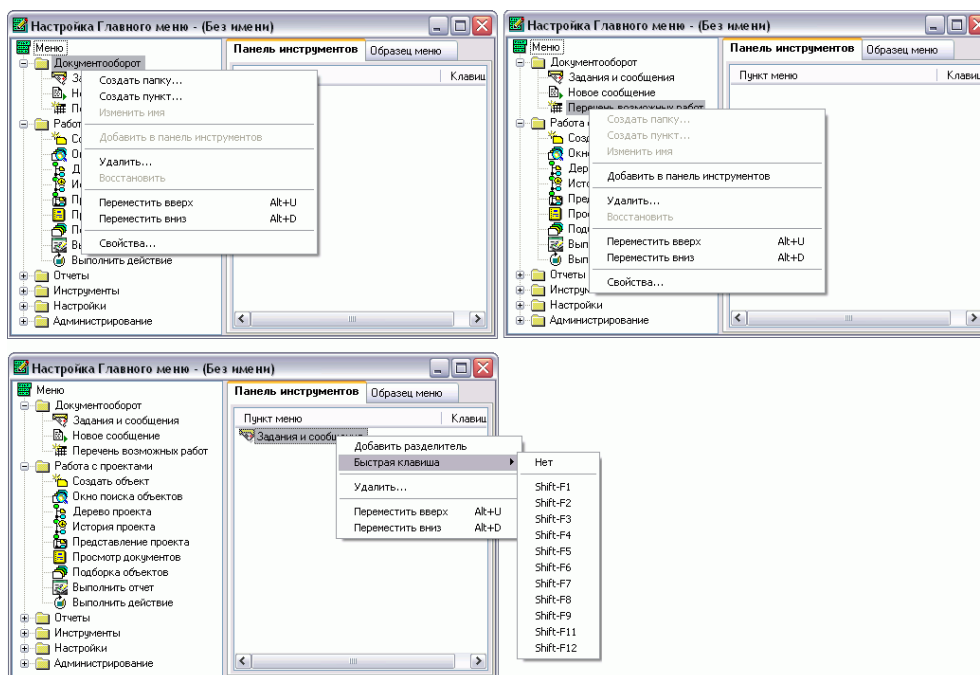


Рисунок 50 Настройка Главного меню

Созданному дополнительному пункту меню можно сопоставить макрокоманду – действие над объектом (настройка действий будет рассмотрена далее – см. п. 16). Для создания дополнительного пункта меню, в левой части окна настройки меню выберите корневой объект или папку, в которую хотите добавить создаваемый пункт, в контекстном меню выберите «Создать пункт...» (Рисунок 50). В открывшемся окне (Рисунок 51) выберите тип – «Действие», введите в поле «Имя» название создаваемого пункта, нажмите кнопку рядом с полем «Заменить значок», выберите подходящую пиктограмму.

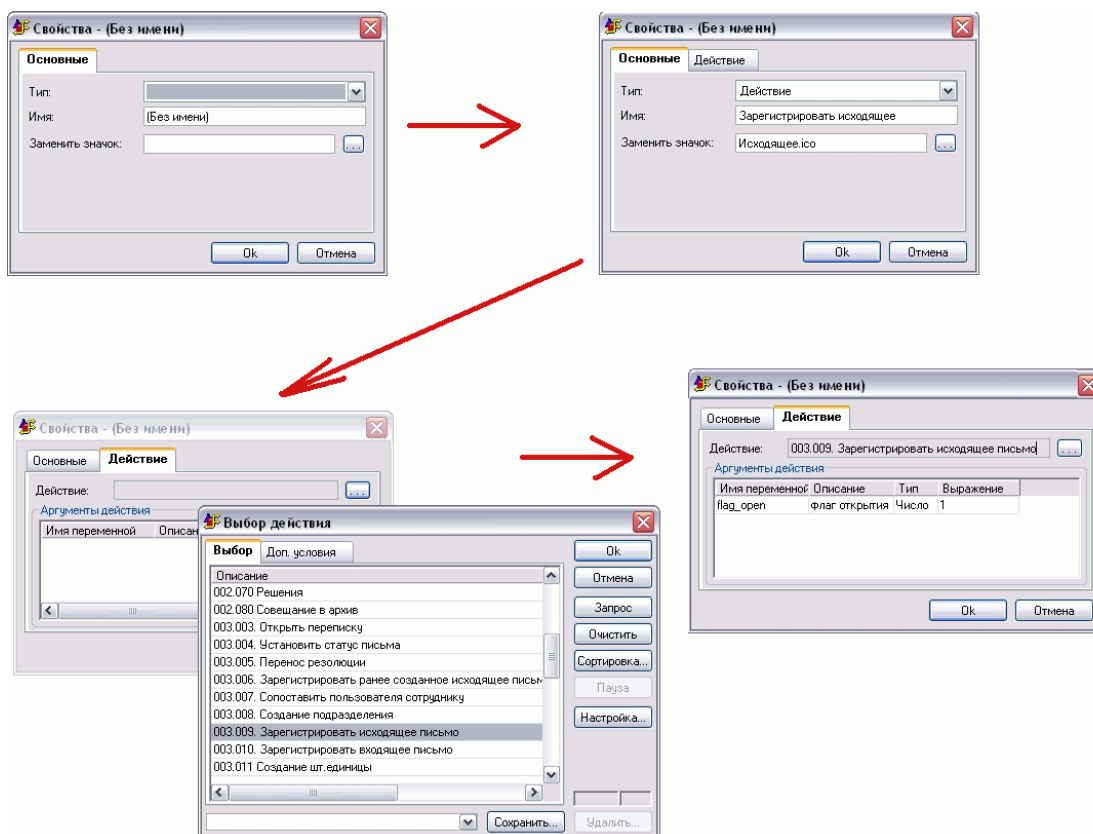


Рисунок 51 Последовательность создания в меню дополнительного пункта типа «Действие»

Перейдя на вкладку «Действие» следует указать требуемое действие, при необходимости добавить аргументы действия (используя контекстное меню поля «Аргументы действия»), далее ввести значения аргументов или выражения для определения указанных значений.

В окне программы могут отображаться две инструментальные панели – «Панель приложения» и «Панель окна» (Рисунок 52). Настроенному Главному меню соответствует «Панель приложения». Вид «Панели окна» зависит от текущего окна программы, настройка панели окна описана в п. 10.3.7 настоящего документа.

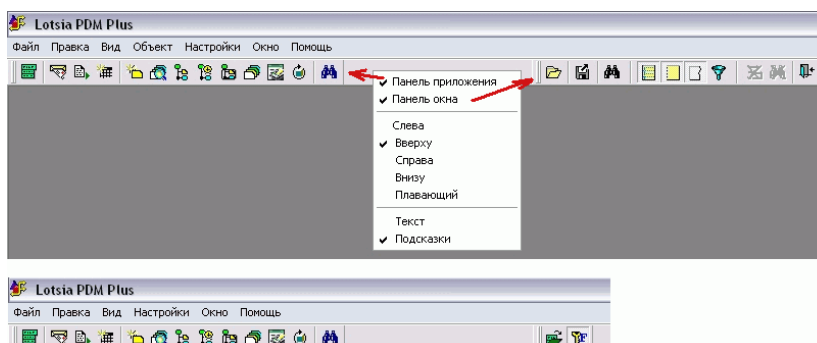


Рисунок 52 Инструментальные панели: Панель приложения и Панель окна

Чтобы пункт Главного меню отобразился на панели приложения, можно использовать контекстное меню (Рисунок 50), либо перетащить, используя drag-and-drop, соответствующий пункт меню в правую часть окна, на вкладку «Панель инструментов». Кнопке панели приложения можно сопоставить быструю клавишу (Рисунок 50).

Сохранить новое настроенное Главное меню можно нажав <Ctrl+S>, после чего ввести название меню, нажать «Ок» или «Применить». Для удаления настроенного главного меню используется пункт верхнего меню «Файл» → «Удалить».

Настроенное Главное меню становится доступным пользователю после подключения меню к профилю данного пользователя (см. п. 10.3).

В зависимости от выполняемых пользователем функций, в некоторых случаях может быть удобным использование одним и тем же пользователем различных Главных меню для выполнения различных задач. Для обеспечения такой возможности, к профилю пользователя подключается требуемое ему количество различных настроенных Главных меню. При настройке профиля определяется также Главное меню, отображаемое по умолчанию («При старте») для данного профиля. Выбор требуемого меню из подключенных к данному профилю осуществляется пользователем при текущем окне Главного меню с помощью пункта верхнего меню либо с использованием панели окна. (Рисунок 53).

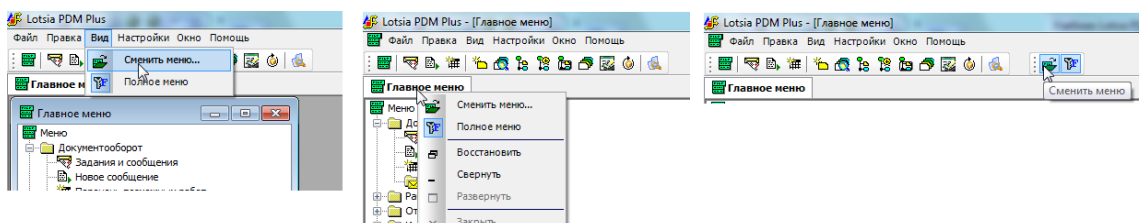


Рисунок 53 Смена Главного меню

Если основное Главное меню удалено из профиля пользователя, пункт «Полное меню» будет для данного пользователя неактивен. Для Администратора указанный пункт активен всегда.

10.2 Формы для объектов, поисковые формы и формы для подборок

Формы (вкладки) для объектов обеспечивают удобную работу пользователя с объектами, позволяя просматривать и изменять атрибутивную информацию объекта, а также работать со связанными объектами, в том числе добавлять и разрывая связи.

Формы для объектов могут быть следующих типов: «Атрибуты», «Дочерние объекты», «Входимость», «Связанные объекты», «Документы архива», «Документы потомков», «Отчет», «Подборка объектов», «Поиск по атрибутам», «План».

Форма типа «Отчет» предполагает наличие в системе отчета (настройка отчетов будет рассмотрена далее, см. раздел 20).

Формы типа «Подборка объектов» используются для отображения левой стороны окна подборки (см. п. 8.3). Форма типа «План» предназначена для отображения информации в виде календарных планов.

При настройке форм вначале выполняется создание формы (см. п. 10.2.1, 10.2.1.1, 10.2.3, 10.2.6). Далее форма сопоставляется типу объекта, либо определяется в качестве поисковой, либо сопоставляется конкретной подборке (только формы типа «Подборка объектов»). Сопоставление форм типам объектов или указание форм в качестве поисковых осуществляется путем настройки профилей пользователей (см. п. 10.3). Одному и тому же типу объектов для данного профиля пользователя может быть сопоставлена одна или несколько различных форм (или не сопоставлено ни одной).

В качестве поисковой может использоваться форма типа «Атрибуты», однако для этой цели имеется специальный тип формы «Поиск по атрибутам». Формы типа «Поиск по атрибутам» обеспечивают возможность поиска по введенному диапазону значений атрибутов или полей формы.

Для работы с объектами могут использоваться как формы типа «Атрибуты», так и формы других типов («Дочерние объекты», «Входимость», «Связанные объекты», «Документы архива», «Документы потомков», «План»), для подборок (см. п. 8.3) используются формы типа «Подборка объектов», кроме того, для выделенного в подборке объекта можно отобразить его атрибутивные формы (вкладки).

Обычно в форме присутствуют поля для отображения значений атрибутов («колонки»). В колонках форм типа «Атрибуты» возможен ввод и редактирование атрибутивной информации, формы других типов не предназначены для изменения значений колонок.

В формах могут присутствовать текстовые и вычисляемые поля, а так же всплывающие подсказки. Использование вычисляемых полей, помимо возможности динамического изменения отображаемой в них информации в зависимости от текущих значений атрибутов или внешних параметров, позволяет вставлять в форму графические изображения в формате *.BMP, *.jpg и др..

В формах типа «Атрибуты» и в отчетах (в том числе, применяемых в качестве форм) возможно создание кнопок. Кнопке сопоставляется действие (настройка действий будет рассмотрена далее, см. раздел 16), которое будет выполняться при её нажатии.

В выражениях для свойств элементов форм и в вычисляемых полях могут быть использованы различные параметры, в зависимости от значений которых внешний вид и свойства данной формы будут изменяться. В качестве указанных параметров могут выступать, например, текущий тип связи, тип текущего объекта, режим работы с формой (редактирование или просмотр), текущий пользователь (можно, например, отобразить в форме имя текущего пользователя или изменить внешний вид и другие свойства объектов формы в зависимости от пользователя, работающего в данный момент с формой).

Данные в полях формы типа «Атрибуты» могут редактироваться пользователем.

При вводе данных в поле формы для объекта (после нажатия «Ok» или «Применить») происходит автоматическое создание соответствующего атрибута (если данный атрибут допустим для данного типа объекта) или редактируется имеющееся значение атрибута.

Таким образом, при наличии настроенных пользовательских форм, пользователю нет необходимости выбирать из списка добавляемый атрибут, и вообще, знать о существовании такого понятия в Lotsia PDM – для повседневной работы конечного пользователя, в большинстве случаев достаточно заполнения полей форм и нажатия кнопок в формах.

При открытии объекта, которому сопоставлены формы, пользователь видит эти формы как вкладки для данного объекта (см. рисунок 54).

Рисунок 54 Пример использования нескольких настроенных форм для одного объекта

Для открытия формы выделенного объекта в окне проекта следует перейти на вкладку соответствующей формы. Для редактирования атрибутов с использованием формы для объекта (в данном случае подразумевается использование формы типа «Атрибуты») необходимо перевести форму в режим редактирования, для чего следует дважды щелкнуть мышью в любом месте на соответствующей вкладке. В открывшемся окне щелкните по требуемому полю, введите значение атрибута. Кнопка «Ok» или «Применить» используется для сохранения изменений, «Отмена» – для отказа от сохранения.

Переход в режим редактирования формы по двойному щелчку может быть отключен в настройке профиля (будет рассмотрено далее, см. раздел 10.3.6).

10.2.1 Настройка форм. Создание форм типа «Атрибуты»

Для создания или изменения формы используется Редактор форм («Главное меню» → «Администрирование» → «Параметры интерфейса» → «Настройка форм для объектов» → кнопка «Создать» → из списка выбрать тип «Атрибуты»).

Подобный рассматриваемому в данном разделе редактор форм используется также при настройке форм для ввода аргументов отчетов (будет рассмотрено далее, см. раздел 20.5.6), создании форм действий (будет рассмотрено далее, см. раздел 17.3) и настройке форм Workflow (будет рассмотрено далее, см. раздел 23.5). В соответствующих разделах

настоящего руководства мы остановимся на особенностях настроек различных форм и их отличиях от форм для объектов.

Левая часть окна редактора форм (Рисунок 55) отображает макет создаваемой формы (см. также Рисунок 57).

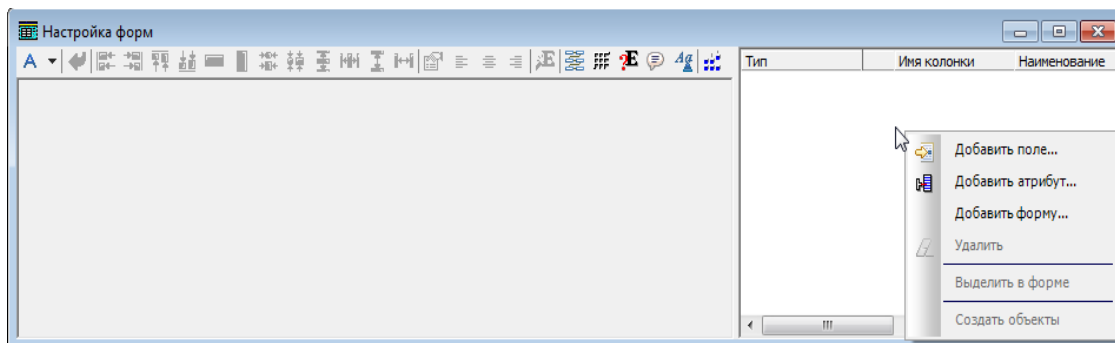


Рисунок 55 Окно редактора форм. Начало создания формы

В правой части окна отображается список переменных (также употребляется слово «Колонки») формы. Для добавления в форму новых колонок используется контекстное меню правой части окна (Рисунок 55). При настройке атрибутивных форм, в данном контекстном меню используются пункты «Добавить поле...», «Добавить атрибут...», «Добавить форму...». При использовании пункта «Добавить атрибут...» добавляемая колонка выбирается из списка имеющихся атрибутов. Пункт «Добавить поле...» позволяет добавить в форму описание объекта и некоторые другие параметры (Рисунок 56).



Рисунок 56 Выбор полей, добавляемых в форму

Пункт «Добавить форму...» позволяет добавить в данную форму другую, уже имеющуюся форму.

При добавлении в форму колонок, они одновременно появляются в списке переменных формы (правая часть окна редактора форм) и в макете формы (левая часть окна).

Добавленные колонки могут быть удалены из формы полностью, либо только из макета формы. Для удаления колонок используется пункт «Удалить» контекстного меню, соответственно, правой или левой части окна. Колонка, удаленная только из макета (левой части окна) формы может использоваться в выражениях вычисляемых полей формы, а также в выражениях для свойств объектов формы. Системную колонку sys\$mode (отображает текущий режим формы, возвращает «E» или «W» – соответственно, при редактировании и просмотре) можно использовать в выражениях и без её добавления в форму.

Отметим, что в выражениях (будет рассмотрено далее, см. п. 10.2.5) вычисляемых полей и свойств объектов формы могут использоваться функции, извлекающие из БД

параметры, отсутствующие в списке колонок данной формы. Так, например, можно получить код текущего пользователя, значение текущей даты и времени, и т. д., либо выполнить SQL-запрос (см. п. 24.3.3.2) и его результат использовать в форме.

В макет формы (левую часть окна редактора форм) могут быть добавлены (контекстное меню левой части окна редактора форм, п. «Создать» → далее выбрать создаваемый элемент (Рисунок 57)) следующие элементы:

- колонки – отображают значения переменных формы. В формах объектов в качестве переменных формы используются атрибуты информационных объектов и некоторые другие параметры. Соответствующий пункт контекстного меню (см. рисунок 57) используется для восстановления удаленных из макета формы (левой части окна редактора форм) колонок
- текстовые поля – отображают введенный при создании формы постоянный текст
- вычисляемые поля – содержат выражения, отображают текущий результат выполнения данного выражения
- кнопки – используются в формах типа «Атрибуты». Данному элементу устанавливается соответствие с макрокомандой («Действием», см. п. 16), которая выполняется при нажатии кнопки. При использовании формы типа «Атрибуты» в качестве поисковой, кнопки становятся неактивными
- линии, геометрические фигуры и рамки – элементы графического оформления формы

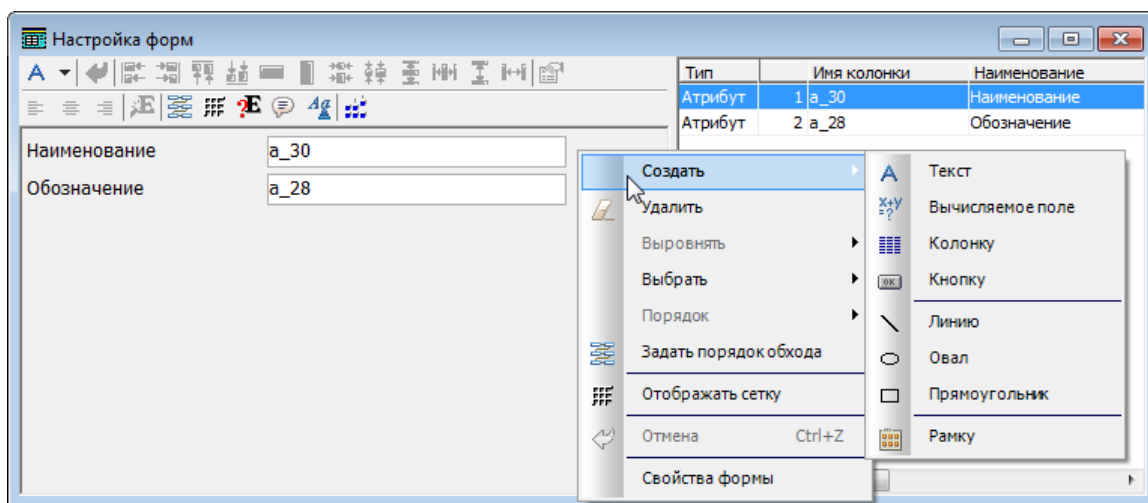



Рисунок 57 Окно редактора форм. Настройка формы типа «Атрибуты»

Добавление элементов и некоторые другие команды управления элементами формы продублированы кнопками, расположенными на панели (слева сверху) окна редактора форм (Рисунок 57). С помощью кнопки  осуществляется пользовательская настройка указанной панели, позволяющая размещать на ней требуемый набор кнопок.

Свойства формы в целом определяются с помощью пункта контекстного меню «Свойства формы» (Рисунок 57). В открывшемся окне (Рисунок 58) можно управлять размерами и графическим оформлением формы. Отметим, что на форме можно разместить изображение, выбрав слева пункт «Фон», в поле «Градиент» – «Изображение», а в поле «Файл» – растровый графический файл (Рисунок 58).

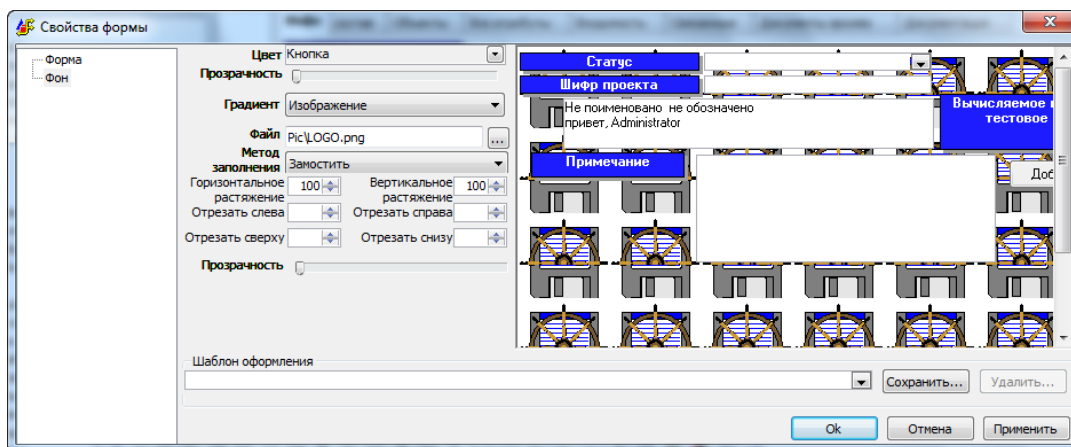


Рисунок 58 Свойства формы (для формы типа «Атрибуты»). Использование графического файла размещения логотипа организации на форме

Настройку свойств объектов формы можно осуществлять, предварительно выделив требуемый элемент (элементы). Для выделения одного объекта достаточно щелкнуть по нему левой клавишей мыши. Выделить несколько элементов в форме можно последовательно, кликая на них левой клавишей мыши при нажатой клавише <Ctrl>, либо выделив в форме область, в которой находятся требуемые объекты формы. Выделение области осуществляется перемещением мыши по диагонали соответствующей области, при нажатой левой клавише мыши и клавише <Alt> клавиатуры (в некоторых случаях <Alt> для выделения области можно и не нажимать).

Перемещение выделенных объектов формы и изменение их размеров возможно как с помощью мыши, так и с использованием клавиатуры. При использовании клавиатуры применяются клавиши со стрелками. Если одновременно со стрелкой нажата клавиша <Shift>, происходит изменение размеров выделенного объекта формы. Если <Shift> не нажат – объект перемещается. Одновременное нажатие клавиши <Ctrl> убыстряет перемещение объекта или изменение его размеров.

Путем задания свойств элементов формы можно управлять их размерами, графическим оформлением, положением, видимостью и другими параметрами. Окно свойств объекта формы (Рисунок 59) открывается по двойному щелчку на соответствующем элементе формы в левой части окна редактора форм, либо из контекстного меню выделенного объекта → пункт «Свойства объекта».

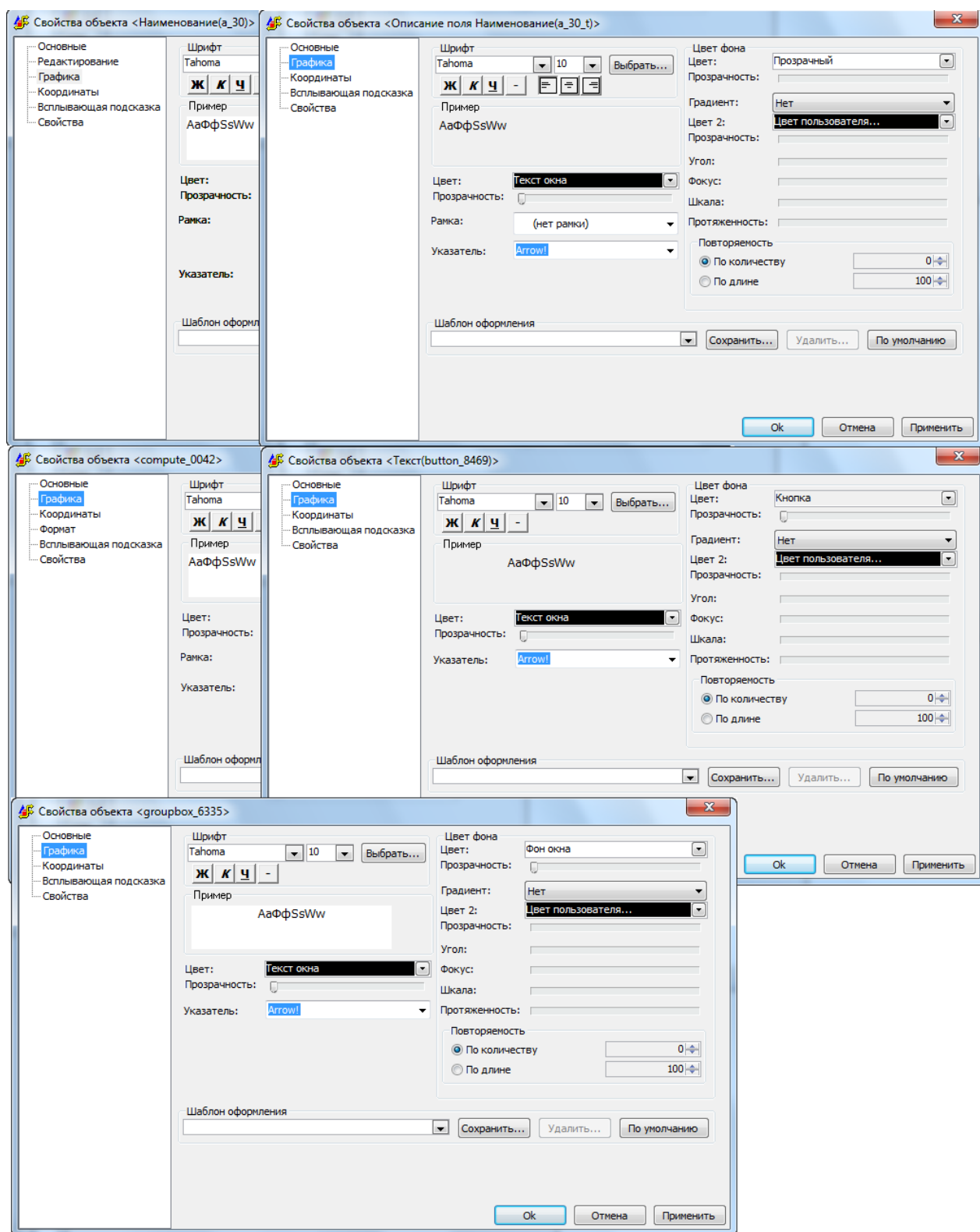


Рисунок 59 Окна свойств для различных элементов формы, слева направо, сверху вниз: колонка, текстовое поле, вычисляемое поле, кнопка, рамка. Выбран пункт «Графика»

Для различных объектов формы пункт «Основные» окно свойств отличается наличием специфических полей, кроме того, некоторые типы полей в окне свойств содержат дополнительные пункты:

- у текстовых полей и рамок – пункт «Основные» содержит поле «Текст», предназначенное для ввода отображаемого в данном поле текста;

- у вычисляемых полей пункт «Основные» содержит поле «Выражение» – для ввода выражений (см. п. 10.2.3). Кроме того, в окне свойств вычисляемых полей имеется пункт «Формат» – для задания формата (маски) отображения данных. Отметим, что для окон свойств колонок такого пункта нет, используется маска, заданная при настройке соответствующего атрибута, либо определенная в поле «Формат отображения» (пункт «Свойства»);
- у кнопок – пункт «Основные» содержит поле «Действие», где указывается и действие, выполняемого по нажатию данной кнопки. Там же имеются поля «Аргументы действия» и «Возврат значений», используемые для передачи значений в действие и возврата в форму результата выполнения действия;

Остальные пункты в окнах свойств в основном сходны (либо полностью идентичны) для различных объектов формы.

При выборе пункта «Общие» в окне свойств, помимо имени, типа и видимости данного объекта отображается информация о его использовании в других объектах формы (Рисунок 60).

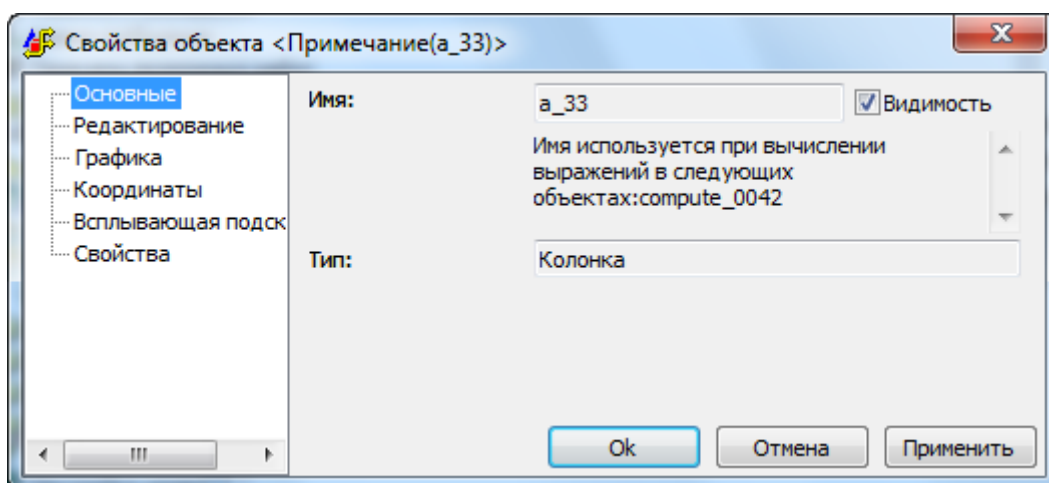


Рисунок 60 Пункт «Общие» окна свойств элемента формы

При выборе пункта «Графика» (Рисунок 59) можно определить достаточно сложное графическое оформление – доступно задание прозрачности (отдельно для текста и фона), градиента (перехода цвета) и другого оформления для выделенных элементов. Для облегчения настройки, графическое оформление элемента может быть сохранено в виде шаблона, после чего шаблон можно использовать для других элементов и форм. Аналогичные настройки имеются и для формы в целом (Рисунок 58).

Свойства объектов формы могут задаваться напрямую – выбирая или указывая требуемые значения при выбранном пункте «Координаты» или «Графика» окна свойств объекта, либо с помощью выражений для отдельных свойств, при выбранном пункте «Свойства» (Рисунок 61).

свойств объектов можно управлять видимостью объектов в форме, их размерами, положением, графическим оформлением, внешним видом курсора и некоторыми другими параметрами – в зависимости от текущих значений аргументов выражения.

Вычисляемые поля содержат выражения (см. п. 10.2.5). Ввод выражения осуществляется при выбранном пункте «Выражение» в окне свойств объекта вычисляемого поля.

Добавляемые в форму кнопки сопоставляются действиям над объектами (см. п. 16).

Сопоставление кнопке действия производится в пункте «Основные» окна свойств соответствующего объекта формы (Рисунок 62).

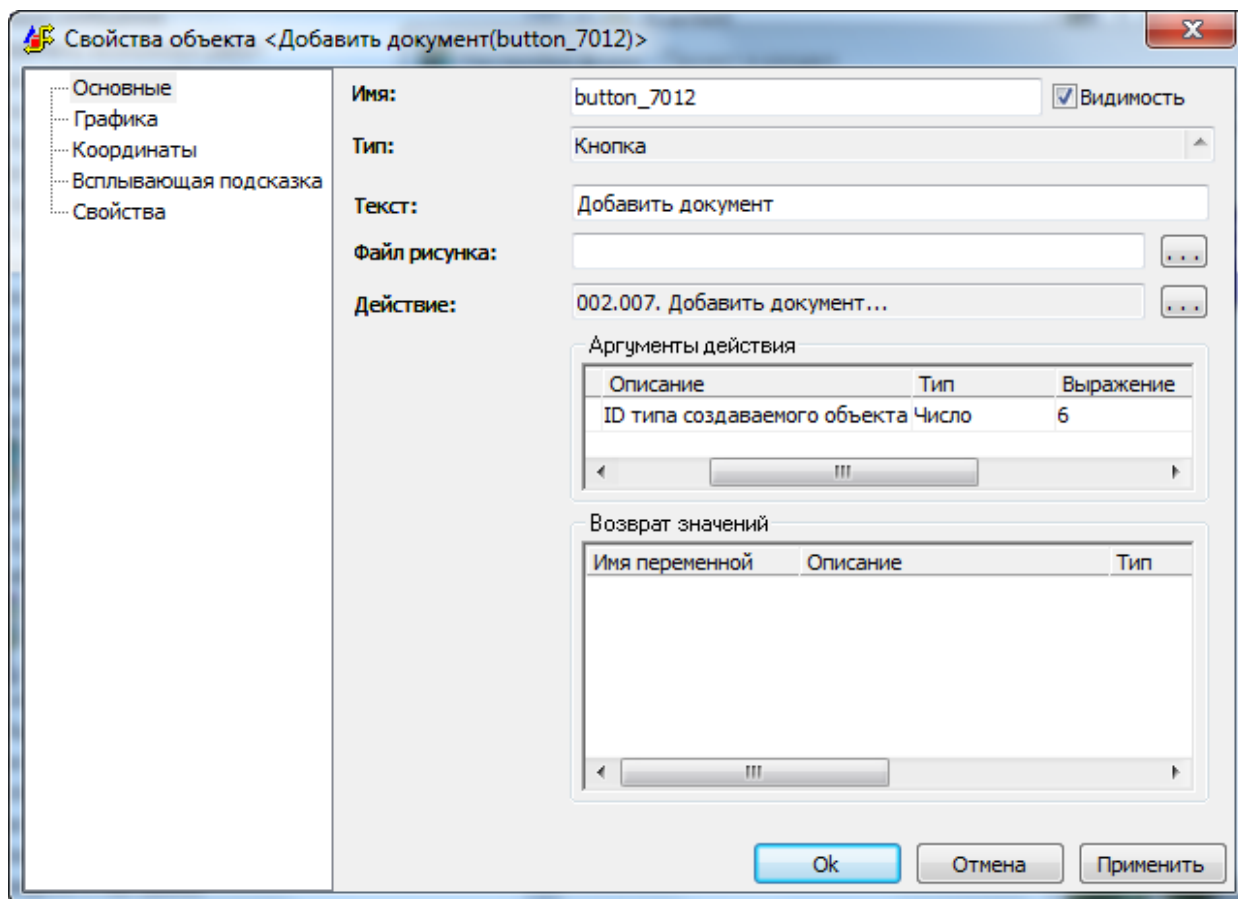



Рисунок 62 Сопоставление действия кнопке атрибутивной формы

При сопоставлении действию кнопки могут быть заданы аргументы, а также указаны переменные формы для возврата в них значений из переменных действия.

Для сохранения созданной формы используйте кнопку  инструментальной панели окна, либо пункт «Сохранить» верхнего меню «Файл». В открывшемся окне свойств формы отдельно указывается название формы и имя вкладки.

Для форм, используемых в качестве поисковых, можно использовать предварительные условия отбора объектов (для форм типов «Атрибуты» и «Поиск по атрибутам»). Для ввода указанных условий: при текущем окне настройки формы, в верхнем меню «Файл» выберите «Свойства...». В открывшемся окне перейдите на вкладку «Поиск» (Рисунок 63).

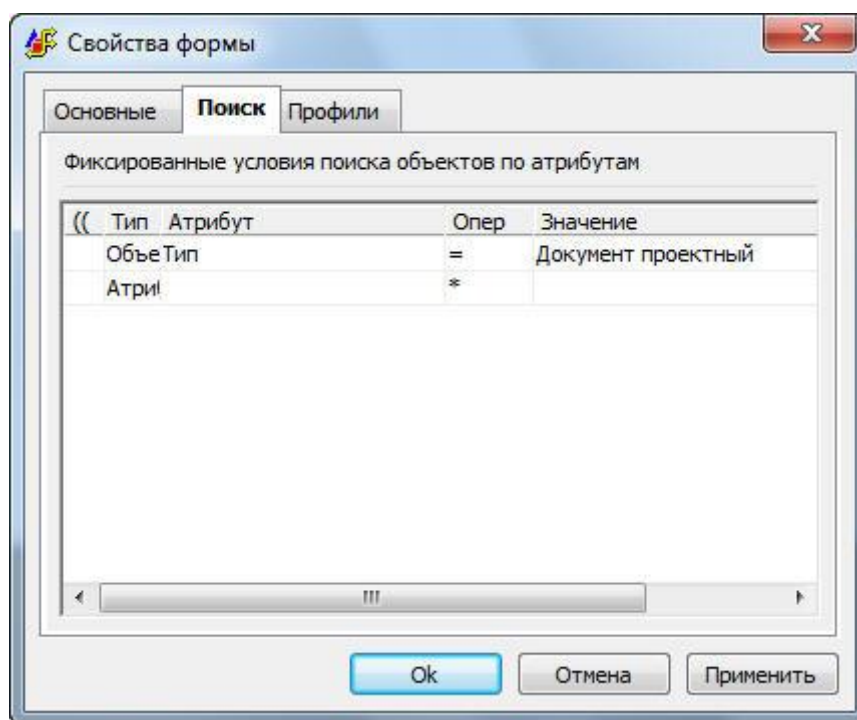


Рисунок 63 Ввод фиксированных условий поиска объектов

На данной вкладке формируются фиксированные условия отбора объектов по значениям атрибутов и другим параметрам, аналогично условиям поиска объектов (см. рисунок 44).

Для полей типа «Колонка» имеется пункт «Редактирование», его использование рассмотрено далее (см. раздел 17.3.1). Отметим, что использование различных стилей для конкретной колонки может быть ограничено, в зависимости от установленного свойства «Редактирование» соответствующего атрибута.

10.2.1.1 Использование вложенных форм

При настройке формы типа «Атрибуты» в нее могут добавляться (внедряться) другие имеющиеся формы. Наиболее востребовано внедрение табличных форм (в том числе – форм отчетов – см. п. 20.3), а также форм типа «Документы архива» и «Документы потомков».

Для добавления в атрибутивную форму вложенной формы, в контекстном меню правой части окна редактора форм выберите «Добавить форму...». В открывшемся окне «Выбор форм» выберите форму, которую вы хотите внедрить в настраиваемую форму. Для добавления предопределенных форм необходимо перейти на вкладку «Дополнительно» (Рисунок 64).

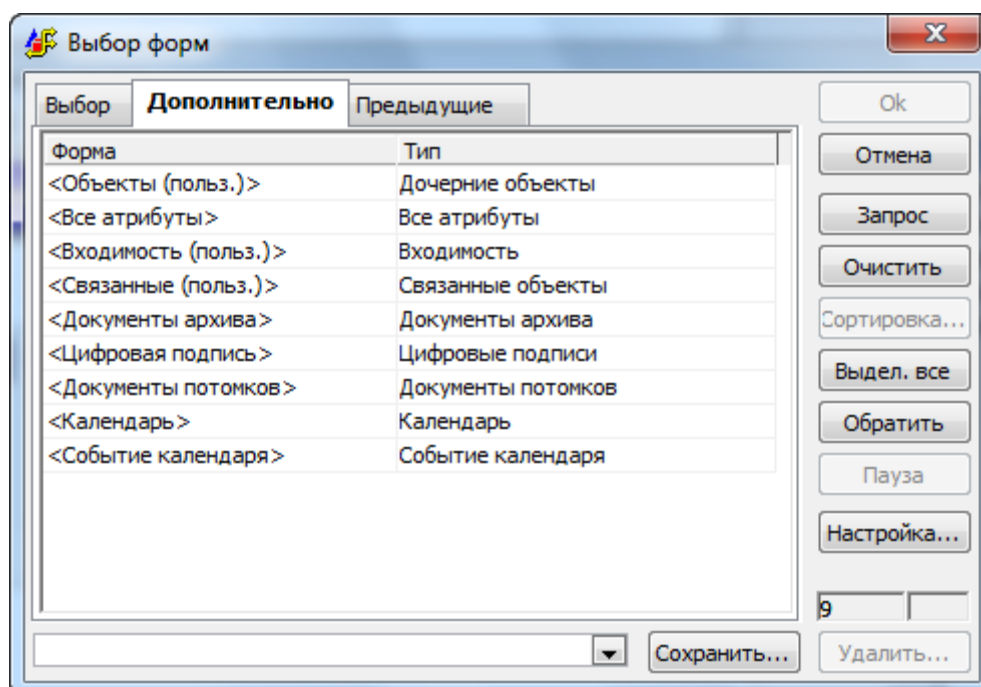


Рисунок 64 Выбор внедряемой формы из predetermined (стандартных) форм

После выбора требуемой формы и нажатия «Ok» выбранная форма будет добавлена в редактируемую форму.

Задание свойств объекта внедренной формы несколько отличается от настройки свойств других объектов формы, в частности, пункт «Редактирование» позволяет выбрать способ выравнивания внедренной формы (Рисунок 65).

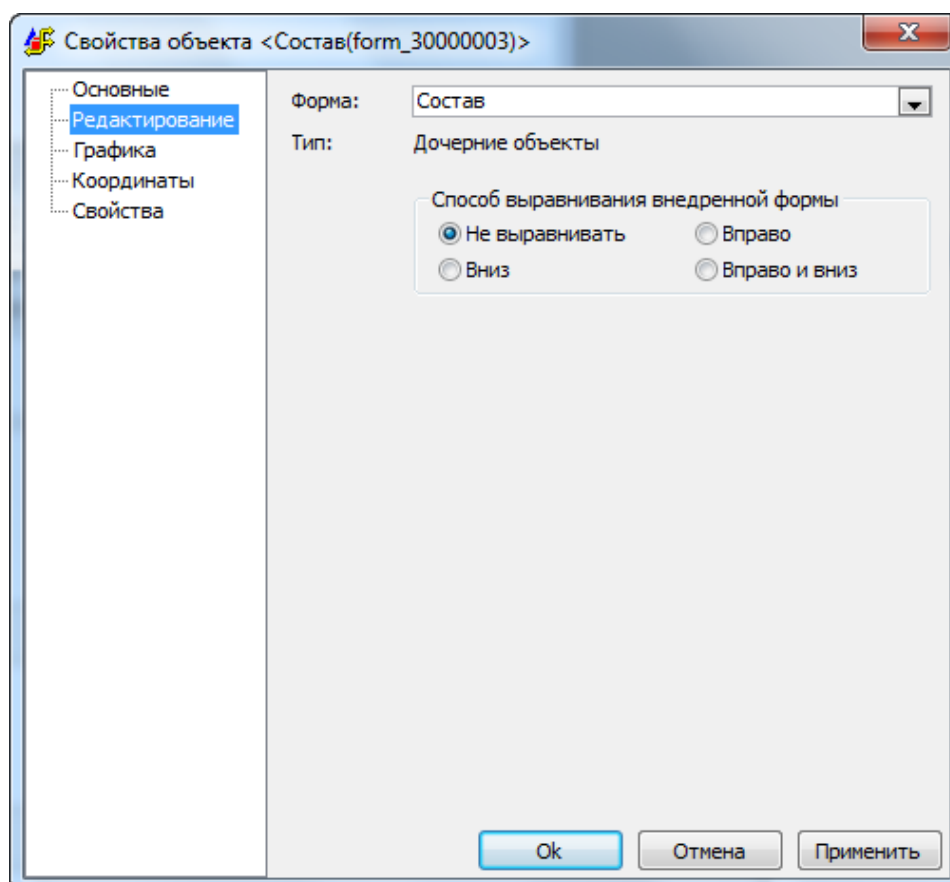


Рисунок 65 Пункт «Редактирование» окна свойств внедренной формы

При выборе способов выравнивания «Вниз», «Вправо» или «Вправо и вниз», поле внедренной формы будет автоматически раздвигаться до границ окна в указанном направлении (направлениях).

10.2.2 Особенности настройки форм типов «Дочерние объекты», «Входимость», «Связанные объекты», «Подборка объектов»

Форма данных типов представляет собой таблицу, строки которой содержат информацию о, соответственно, дочерних, родительских или связанных с текущим объектом объектах, в форме подборки строка соответствует объекту, помещенному в подборку, колонки содержат информацию о соответствующем объекте подборки. В формах указанных типов, в отличие от формы типа «Атрибуты», не предусмотрено редактирование информации, а также использование кнопок.

Настройка форм указанных типов идентична между собой, а также во многом сходна с настройкой отчетов. По своим свойствам, формы первых трех типов соответствуют отчетам с глубиной поиска объектов «Объекты первого уровня», а форма типа «Подборка объектов» – глубине поиска «Выделенный объект» (будет рассмотрено далее, см. раздел 20).

Для настройки формы «Дочерние объекты» выберите: «Главное меню» → «Администрирование» → «Параметры интерфейса» → «Настройка форм для объектов» → кнопка «Создать» → из списка выберите тип «Дочерние объекты». Откроется окно редактора форм, в котором уже имеются предопределенные колонки данной формы. В открывшемся окне свойств формы отдельно указывается название формы и имя вкладки.

Добавление и удаление колонок из правой части формы осуществляется аналогично формам типа «Атрибуты», отличие в том, что имеются предопределенные колонки, которые из правой части формы удалить невозможно. Как добавленные, так и предопределенные колонки можно удалить из левой части окна настройки формы.

В отличие от форм типа «Атрибуты», при настройке форм типов «Дочерние объекты», «Входимость» и «Связанные объекты» и «Подборка объектов» используются области формы. Для ознакомления с данным понятием, поместите курсор в левую часть окна настройки формы. Откройте контекстное меню (Рисунок 66).

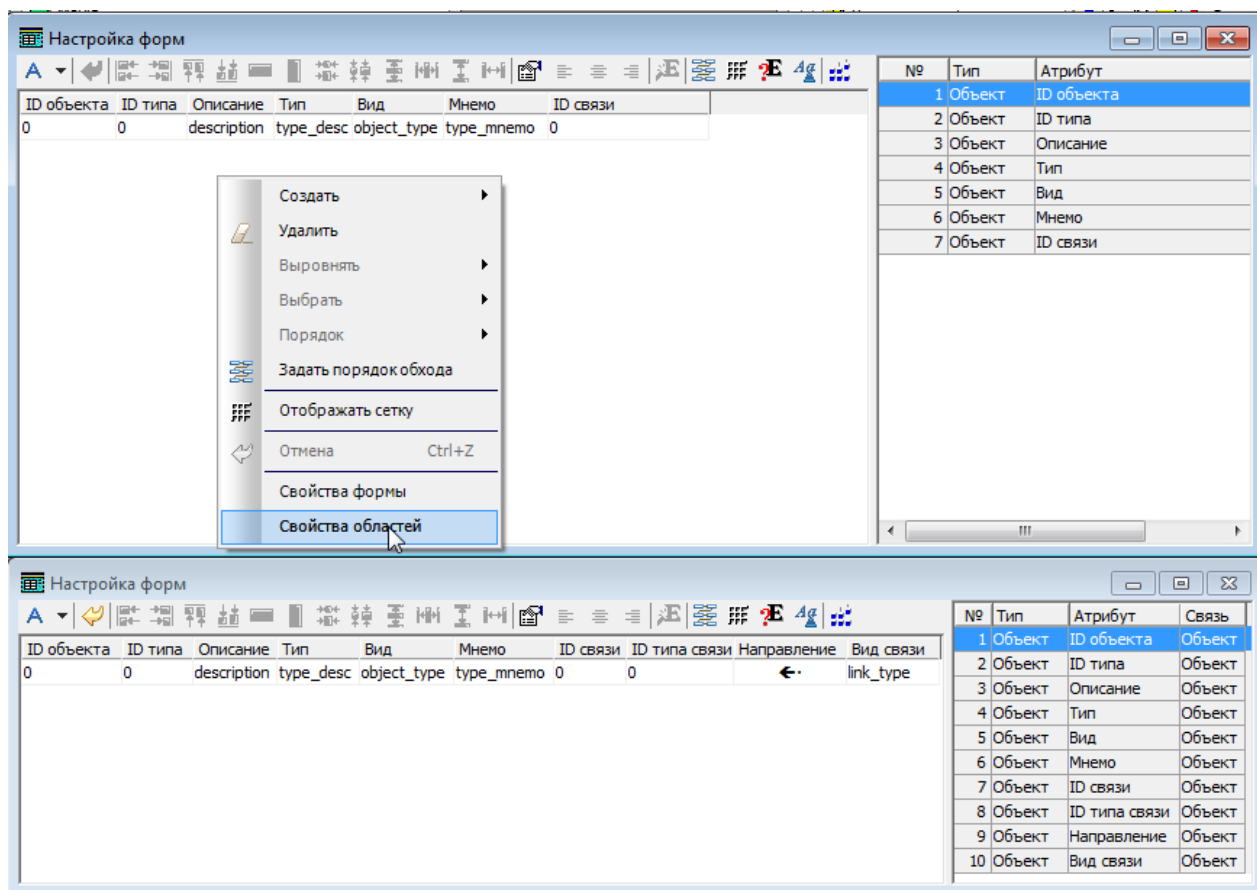


Рисунок 66 Окна настройки форм типов «Дочерние объекты», «Входимость» (вверху) и «Связанные объекты» (внизу)

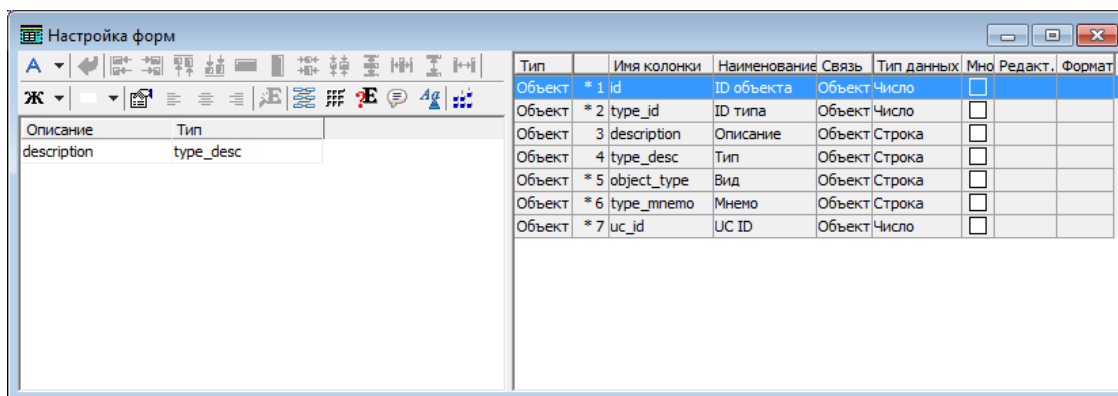


Рисунок 67 Окно настройки формы типа «Подборка объектов»

В контекстном меню выберите «Свойства областей». В открывшемся окне (Рисунок 68) определите цвета областей, для области итогов и нижнего колонтитула укажите высоты около 100 единиц, после чего области формы будут наглядно отображены в окне настройки формы (Рисунок 69).

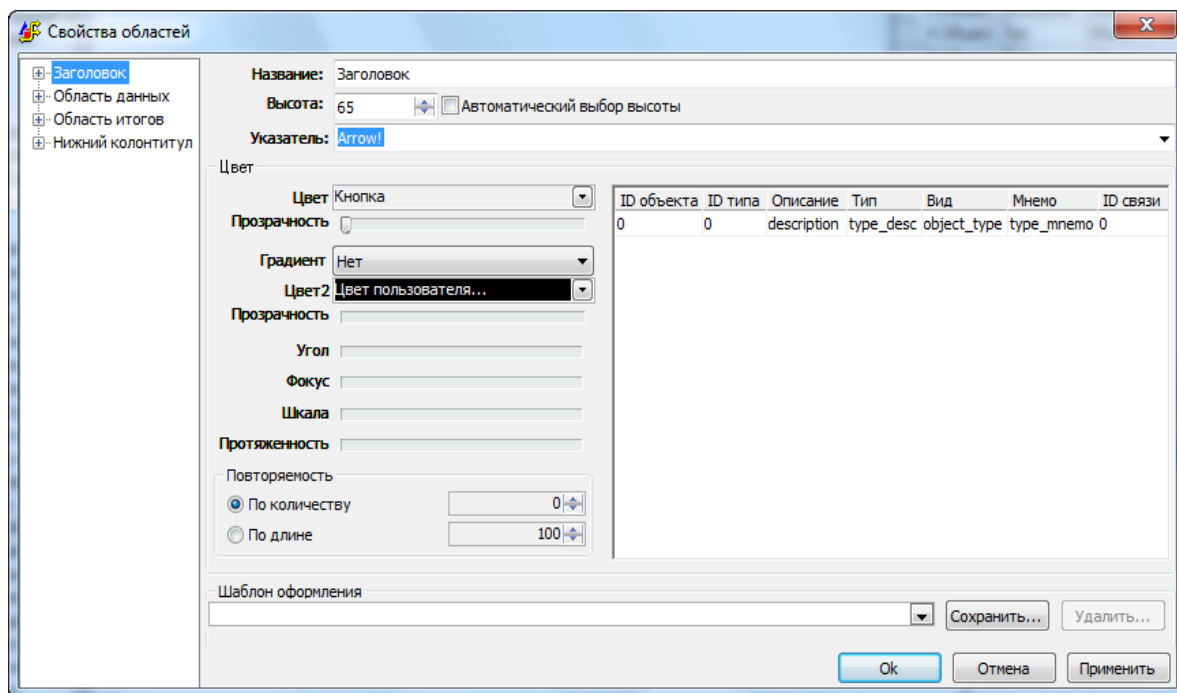


Рисунок 68 Определение свойств областей формы

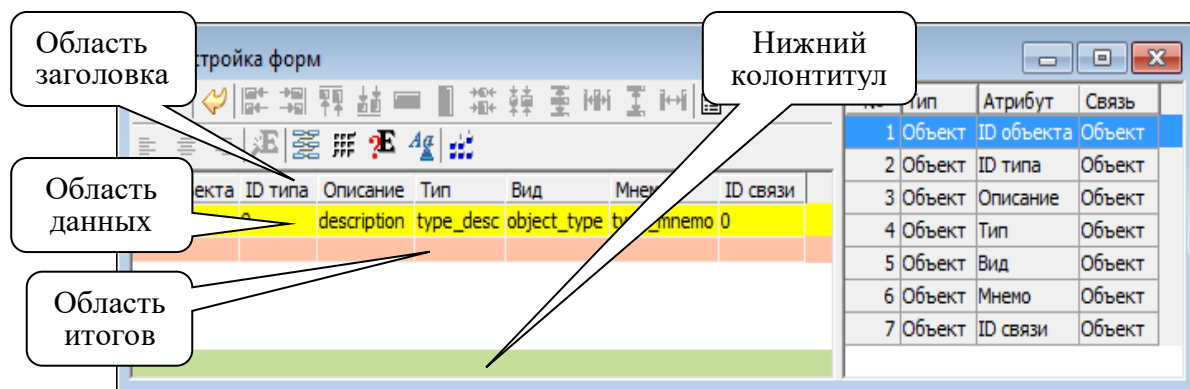


Рисунок 69 Области формы

При добавлении колонки в правую часть формы, в область данных добавляется поле соответствующей колонки, в область заголовка добавляется текстовое поле – заголовок данной колонки. При добавлении вычисляемого поля в область данных, в форму добавляется новая колонка таблицы.

Увеличить высоту области формы удобно перемещением элементов формы, находящихся в данной области, изменить высоту области можно, указав её в свойствах данной области (Рисунок 68).

Переместить объект формы из одной области в другую (например – из области заголовка в область итогов) можно, поместив данный объект вначале на передний план, после чего передвинуть его, расположив над требуемой областью формы и снова указать положение – соответственно требуемой области. Положение объекта формы изменяется в окне свойств данного элемента, в пункте «Координаты» (Рисунок 70).

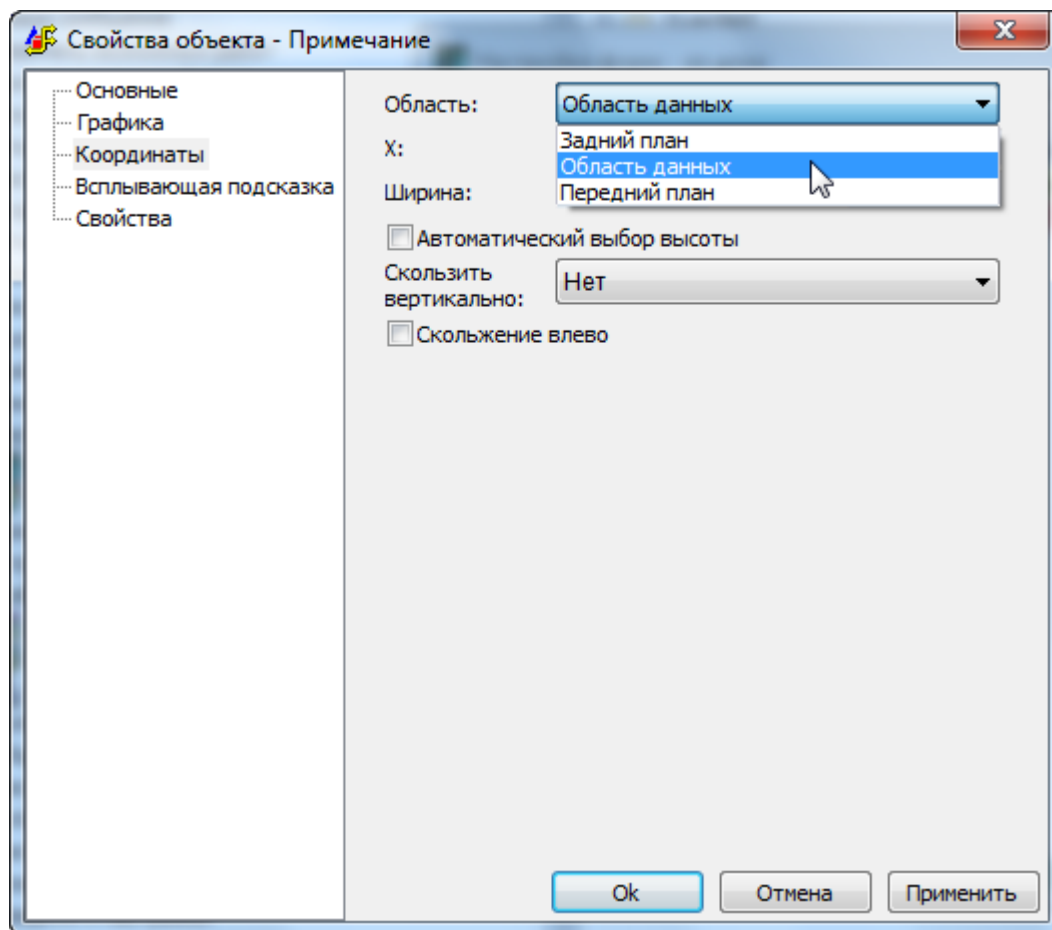


Рисунок 70 Указание области формы, в которую будет помещен данный объект.

Для изменения размеров и положения объектов формы можно использовать клавиатуру или мышь. При использовании клавиатуры, клавиши со стрелками перемещают выделенный объект, при нажатой клавише <Shift> стрелками изменяется размер выделенного объекта, при нажатой клавише <Ctrl> операция осуществляется ускоренно.

Отметим, что табличные формы (кроме форм типа «Подборка», используемых исключительно для подборок) применяются как в виде вкладок объектов, так и в качестве вложенных форм (см. п. 10.2.1.1)

10.2.3 Привязка форм к типам связей

В формах «Дочерние объекты» и «Родительские объекты» по умолчанию поиск родительских и дочерних объектов осуществляется по текущему типу связи (по которому открыто окно проекта), а в формах «Связанные объекты» тип связи для связанных объектов может быть выбран в верхней части формы.

Однако, можно привязать формы указанных типов к определенному типу связи (независимо от типа связи, по которому открыто данное дерево проекта), явно указав тип связи в свойствах формы (верхнее меню «Файл» → «Свойства...» → вкладка «Параметры» открывшегося окна свойств формы (Рисунок 71)).

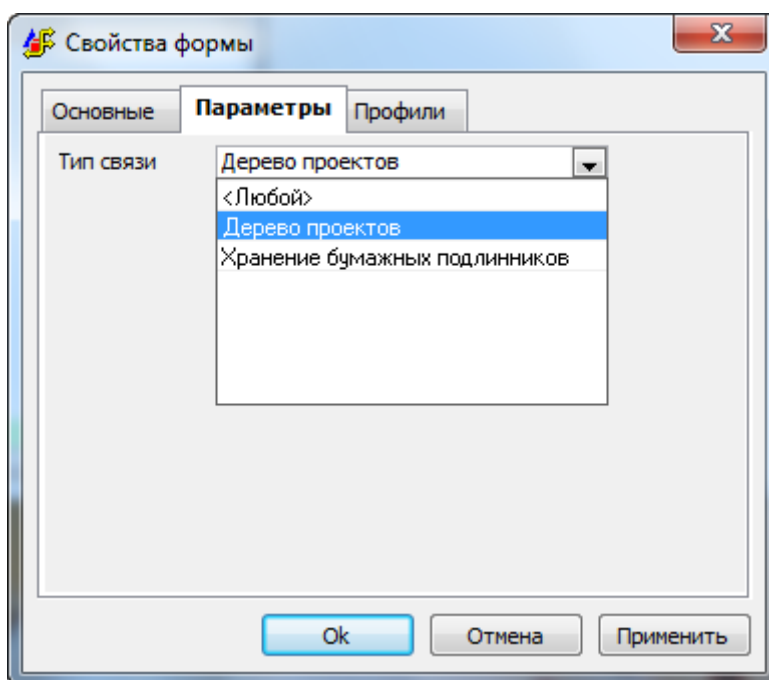


Рисунок 71 Указание типа связи для отбора объектов, информация о которых отображается в форме

Для форм типа «Атрибуты» обычно нет необходимости в подобной привязке. Однако, если это требуется, можно управлять видимостью, положением и другими свойствами объектов формы (Рисунок 61) в зависимости от текущего типа связи. Предварительно в форму следует добавить поле «ID типа связи» (см. рисунок 55, Рисунок 56). Для этого колонка «ID типа связи» используется в выражениях (см. п. 10.2.5) для свойств объектов формы.

10.2.4 Формы типов «Документы архива» и «Документы потомков»

Данные формы предназначены для работы с документами – импортирования файлов в архив, открытия документов архива и т.д. Форма «Документы архива» отображает документы, импортированные в текущий объект, форма «Документы потомков» – документы текущего объекта и потомков первого уровня. Контекстное меню левой части окна настройки данных форм содержит пункт «Форма документов» (Рисунок 72), обеспечивающий возможность специфических настроек форм для работы с документами.

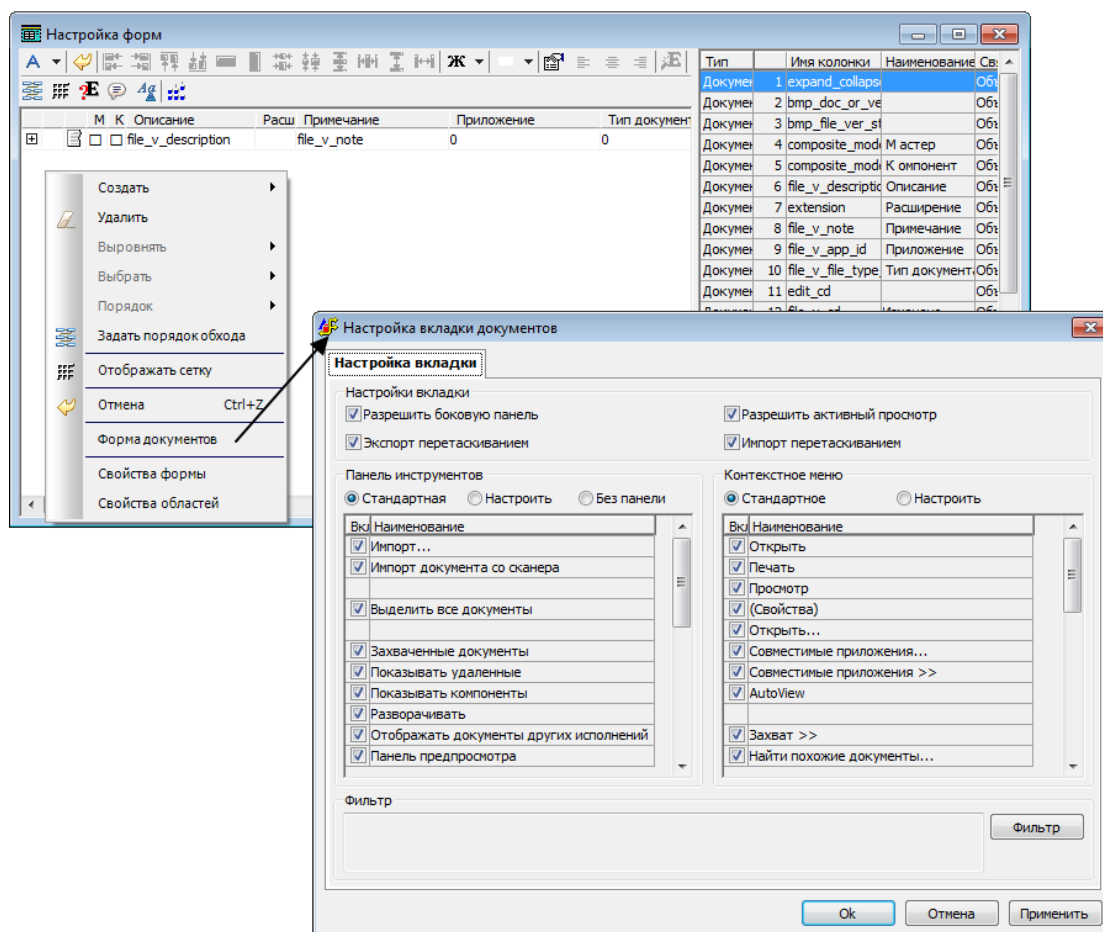


Рисунок 72 Особенности настройки форм типов «Документы архива» и «Документы потомков»

Назначение отдельных элементов окна настройки вкладки документов соответствует названиям элементов.

Отметим, что формы для работы с документами применяются как в виде вкладок объектов, так и в качестве вложенных форм (см. п. 10.2.1.1)

10.2.5 Начальные сведения по составлению выражений

Рассмотрим простейшие правила написания выражений, используемых например, в вычисляемых полях и при задании свойств объектов формы. Выражения будут встречаться постоянно – при задании фильтров, условий и т.д.

Выражения могут составляться ручным вводом или с помощью редактора выражений, где так же возможен ручной ввод. При составлении выражений часто удобно копировать выражения или их фрагменты через буфер обмена (можно использовать клавиши <Ctrl>+<Insert> либо <Ctrl>+<c> для копирования в буфер, <Shift>+<Insert> либо <Ctrl>+<v> для вставки фрагмента). Подробно использование редактора выражений описано в документе «Редактор выражений», имеющемся на Вашем дистрибутивном диске.

Составление выражений начнем рассматривать на примере вычисляемого поля для атрибутивной формы. По умолчанию, формат данных в вычисляемом поле – число. Для составления обычного арифметического выражения, достаточно ввести обрабатываемые числа и арифметические операторы – + (сложение), - (вычитание), * (умножение), / (деление). Можно использовать обычные скобки любой вложенности.

Вычисляемыми полями можно обрабатывать не только числа, но и данные в виде строк, даты/времени и логические значения. Формат данных, не являющихся числами, необходимо явно определять, используя одинарные кавычки для строковых значений, а так же функции преобразования, для определения данных соответствующих форматов.

Обычно* числа, функции и имена переменных вводятся в выражения без кавычек, текст – в одинарных кавычках. При использовании функций необходимо соблюдение синтаксиса данной функции (при выделении функции в списке функций редактора выражений, информация о синтаксических правилах данной функции отображается в поле «Синтаксис»). Даты, строки и числа, с учетом формата, можно преобразовывать друг в друга с использованием соответствующих функций.

Приведем несколько примеров простейших выражений, не содержащих переменных и функций:

Выражение	Результат выполнения
223	223
2+3	5
2*3	6
2+'текст'	2текст
'текст1'+'текст2'	текст1текст2

Как видно из примеров, математические операции выполняются над числами, текстовые строки при суммировании сцепляются (то есть, для объединения строк не требуется применения специальной функции, как например, «СЦЕПИТЬ()» в MS Excel). Для определения и обработки дат следует использовать функции из группы «Даты/времени».

Вместе с прямым указанием значений, в выражениях вычисляемых полей обычно используются переменные. В качестве переменных могут использоваться колонки формы (их имена), в том числе и имена других вычисляемых полей формы (аналогично тому, как это делается в электронных таблицах). Для вставки в выражение имени переменной можно (помимо ручного ввода), скопировать его из свойств соответствующей колонки, или использовать вкладку «Колонки» редактора выражений (см. рисунок 73).

* При использовании функций применение кавычек определяется синтаксическими правилами конкретной функции

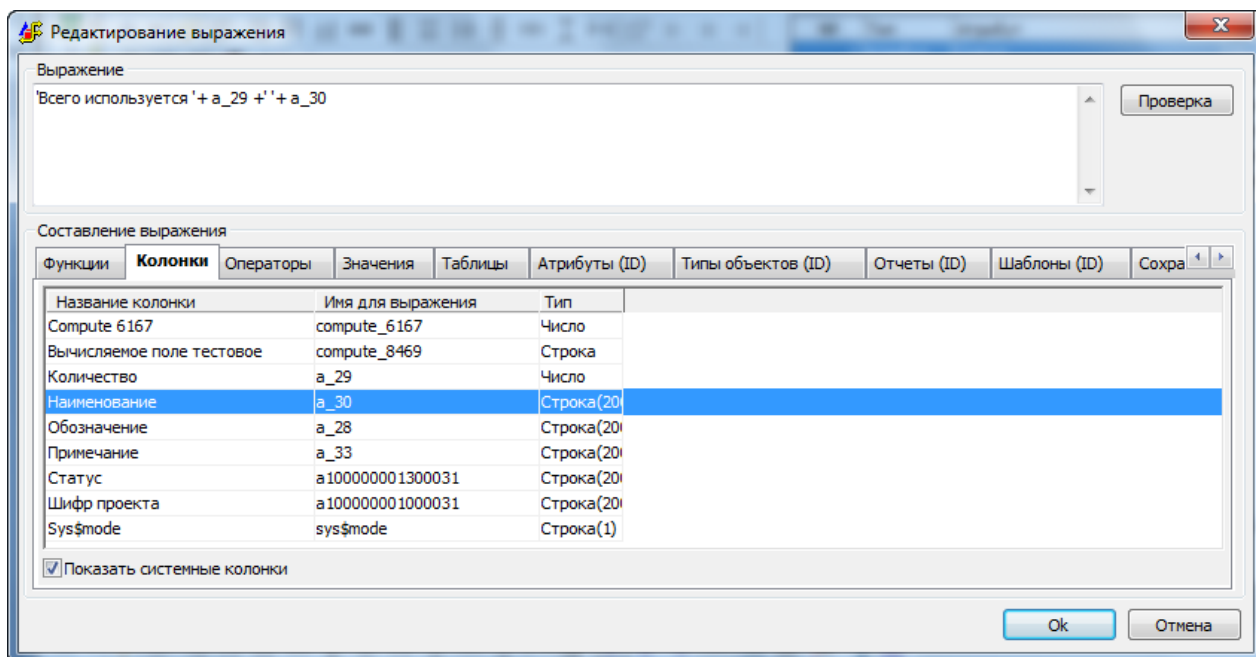


Рисунок 73. Добавление колонки в выражение

Для вставки имени колонки в выражение необходимо дважды щелкнуть левой клавишей мыши по строке с названием соответствующей колонки. Отметим, что при вставке в вычисляемое поле колонок, в качестве вставляемых колонки могут использоваться и другие вычисляемые поля. При этом удобно, если вставляемая колонка имеет понятное название либо имя. Формирование названий колонок будет рассмотрено далее, см. п. 10.2.6. Изменить имя вычисляемого поля можно на вкладке «Общие» окна свойств данного поля (имя должно начинаться с латинской буквы, допустимы латинские буквы, цифры, символ подчеркивания).

Имеется системная колонка sys\$mode, которая возвращает текущий режим, в котором находится форма – «Редактирование» или «Отображение» (колонка возвращает соответственно «E» или «W»). Данная колонка может использоваться в выражениях вычисляемых полей формы или в выражениях для свойств объектов формы, для чего в редакторе выражений на вкладке «Колонки» следует включить флажок «Показать системные колонки» (см. рисунок 73) (добавлять данную колонку в форму для использования её в выражениях не обязательно).

Приведем примеры простейших выражений, в которых используются переменные (имена колонок):

Выражение	Переменная				Результат выполнения
	Название	Имя	Тип данных	Значение	
'Всего ' + a_29 + ' шт.'	Количество	a_29	Число	2	Всего 2 шт.
'Используется ' + a_29 + ' шт. изделий ' + a_30 + ' . ' + a_28	Количество	a_29	Число	2	Используется 2 шт. изделий Скоба прямая. A-12701
	Обозначение	a_28	Строка	A-12701	
	Наименование	a_30	Строка	Скоба прямая	

Заметим, что для корректной работы даже простейших выражений с переменными, необходимо, чтобы значения всех переменных были определены. То есть, при

использовании вышеприведенных выражений в формах для объектов, необходимо наличие у конкретных объектов атрибутов «Наименование», «Обозначение», «Количество».

Если какой – либо из указанных атрибутов у данного объекта отсутствует, соответствующее выражения в вычисляемом поле формы такого объекта выполняться не будет. Для того чтобы правильно обрабатывать случаи отсутствия некоторых данных, в выражениях следует использовать функции, что позволяет формировать различные выражения для возможных случаев наличия или отсутствия данных.

Далее рассмотрим применение некоторых часто используемых стандартных функций.

Для проверки выполнения различных условий может быть использована функция `if()`. Синтаксис данной функции:

`If(<boolean expression> , <if true value> , <if false value>)`

Где: `<boolean expression>` – проверяемое условие; может принимать значения «истина» или «ложь». Здесь и далее под «истина»/«ложь» понимаются значения логического типа.

`<if true value>` – значение, возвращаемое данной функцией, если условие выполняется, то есть, принимает значение «истина»;

`<if false value>` – значение, возвращаемое данной функцией, если условие не выполняется, то есть, принимает значение «ложь».

В функции так же могут включаться различные вложенные функции, (максимальный уровень вложенности системой не ограничивается). В качестве проверяемых условий данной функции могут использоваться различные логические выражения или функции, возвращаемыми значениями могут быть как конкретные значения (константы), так и выражения, включающие переменные и функции.

Например, выражение:

`If(IsNull(a_28), 'Извещение не зарегистрировано', 'Извещение зарегистрировано')`,
где: `IsNull(a_28)` – функция, проверяющая аргумента – колонки «a_28» (атрибут «Обозначение»). Возвращает «истина» если проверяемый аргумент существует, или «ложь», если, в данном случае, атрибут «Обозначение» у данного объекта отсутствует.

Возвращает «Извещение зарегистрировано» при наличии у данного извещения обозначения, и «Извещение не зарегистрировано» в случае, если извещению не присвоено обозначение.

В качестве более сложного примера приведем выражение вычисляемого поля, использованное в настройке «Проектное предприятие», в форме «Чертеж». Данное поле, в зависимости от того, установлен ли атрибут «Дата выполнения (факт)» выводит сообщение о сроке, оставшемся до плановой даты выполнения, или задержке сдачи, относительно плановой даты, или указывает, с опозданием (каким), или досрочно сдан чертеж:

`If(IsNull(a681153613001014) ,`

`Case (DaysAfter(date(today()) , a808870773400000)`

`When is> 0 Then 'До окончания срока осталось' + DaysAfter(date(today()) , a808870773400000)+ ' дн.'`

`When is=0 Then 'Срок истекает сегодня'`

```

When is<0 Then 'Запоздывание составляет ' + Abs( DaysAfter( date(today() ,
a808870773400000 )) + ' дн.'

Else " ) ,

Case ( DaysAfter(a681153613001014 , a808870773400000 )

When is> 0 Then 'Сдано досрочно'

When is=0 Then 'Сдано'

When is<0 Then 'Сдано с опозданием в ' + Abs( DaysAfter( a681153613001014 ,
a808870773400000 )) + ' дн.'

Else ")

```

Где:

a681153613001014 – фактическая дата сдачи (имя колонки)

a808870773400000 – плановая дата сдачи (имя колонки).

Синтаксис и назначение использованных функций вы можете уточнить, выбрав требуемую функцию в редакторе выражений (Рисунок 74).

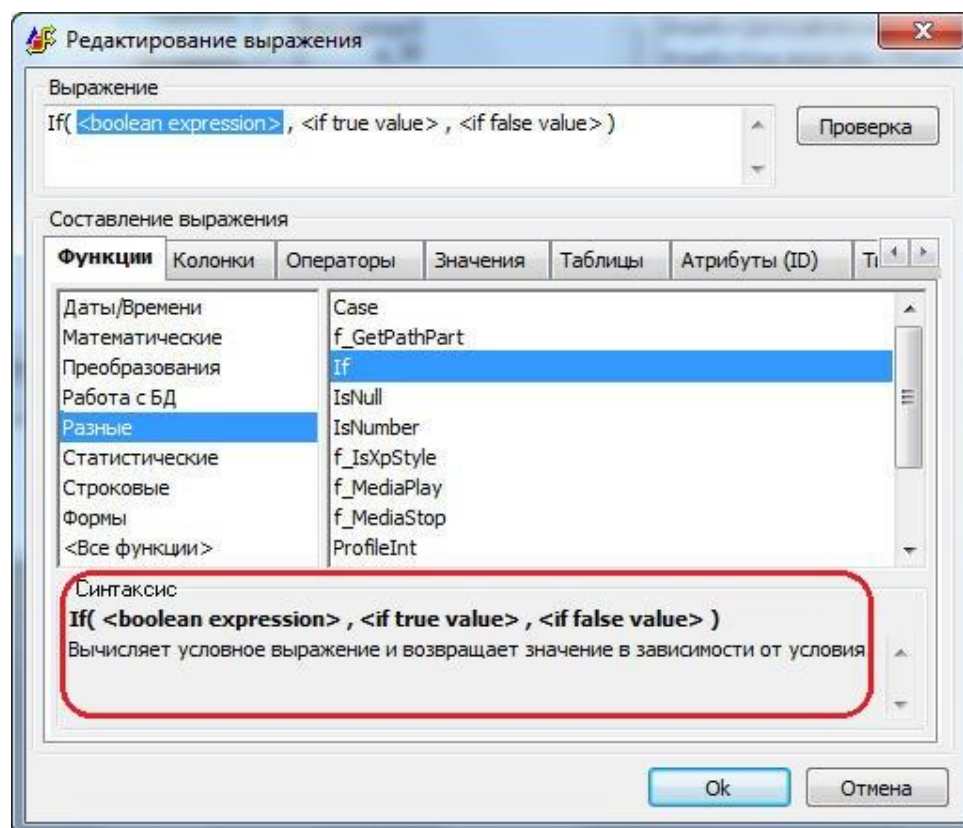


Рисунок 74 Информация о функции, выбранной в редакторе выражений

С помощью вычисляемого поля в форму могут добавляться и графические элементы. Для добавления таких элементов используется функция Bitmap, имя файла с графикой может храниться в атрибуте, полное имя файла может формироваться сцеплением строк.

Выражение, используемое, например, в вычисляемом поле формы, может содержать SQL-запрос (будет рассмотрено отдельно, см. раздел 24.3.3.2), использование которого позволяет отобразить в форме не только информацию о текущем объекте, но и другие данные, содержащиеся в БД.

В выражениях могут использоваться комментарии. Комментарии формируются следующим образом:

`/*Текст комментария*/`. В выражении вычисляемого поля, помимо комментария обязательно должна быть значимая часть – функция или значение. То есть, если необходимо закомментировать вычисляемое поле целиком, в вычисляемом поле должно остаться какое-либо незакомментированное выражение, например, 0. При этом содержимое вычисляемого поля примет следующий вид:

`0/*Выражение вычисляемого поля, которое необходимо закомментировать*/`

В таком виде вычисляемое поле может быть сохранено и отредактировано позднее.

На практике часто встречаются случаи, когда в значении атрибута записан идентификатор какой-либо сущности Lotsia – например: объекта, пользователя, процесса Workflow. Значение идентификатора (может указываться также несколько идентификаторов, в виде строки с разделителями) для пользователя не несет полезной информации. Чтобы вместо значений ID отобразить в форме информацию в удобном для пользователя виде, используется вычисляемое поле, содержащее функцию `f_ExecSQLSelect_3 ()` с соответствующим SQL-запросом. Например, если атрибут содержит строку с идентификаторами объектов, а требуется отобразить в форме описания этих объектов, выражение может быть следующим:

`f_ExecSQLSelect_3 ('Select description from LSDBO.object_reference_view where LSDBO.object_reference_view.ID in ('+ a100000004003031 +')' , " , '~r~n' , 'cols=char(250)' , 10)` где:

`a100000004003031` – имя колонки атрибута (тип – «Строка»), содержащего строку идентификаторов объектов с разделителем – запятой. Приведенный в первом аргументе функции SQL-запрос вернет описания объектов, идентификаторы которых содержатся в значении данного атрибута.

Если атрибут содержит 1 идентификатор пользователя, то, чтобы отобразить в вычисляемом поле имя данного пользователя, можно использовать более простое выражение:

`f_UserName(a100000004003031)` где:

`a100000004003031` – имя колонки атрибута (тип – «Число»), содержащего ID пользователя.

10.2.6 Всплывающие подсказки

Всплывающие подсказки могут создаваться как в неизменном виде (текста), так и в виде вычисляемых полей (выражений).

Для создания всплывающих подсказок в виде простого текста используется пункт «Всплывающая подсказка» окна свойств объекта формы (Рисунок 75).

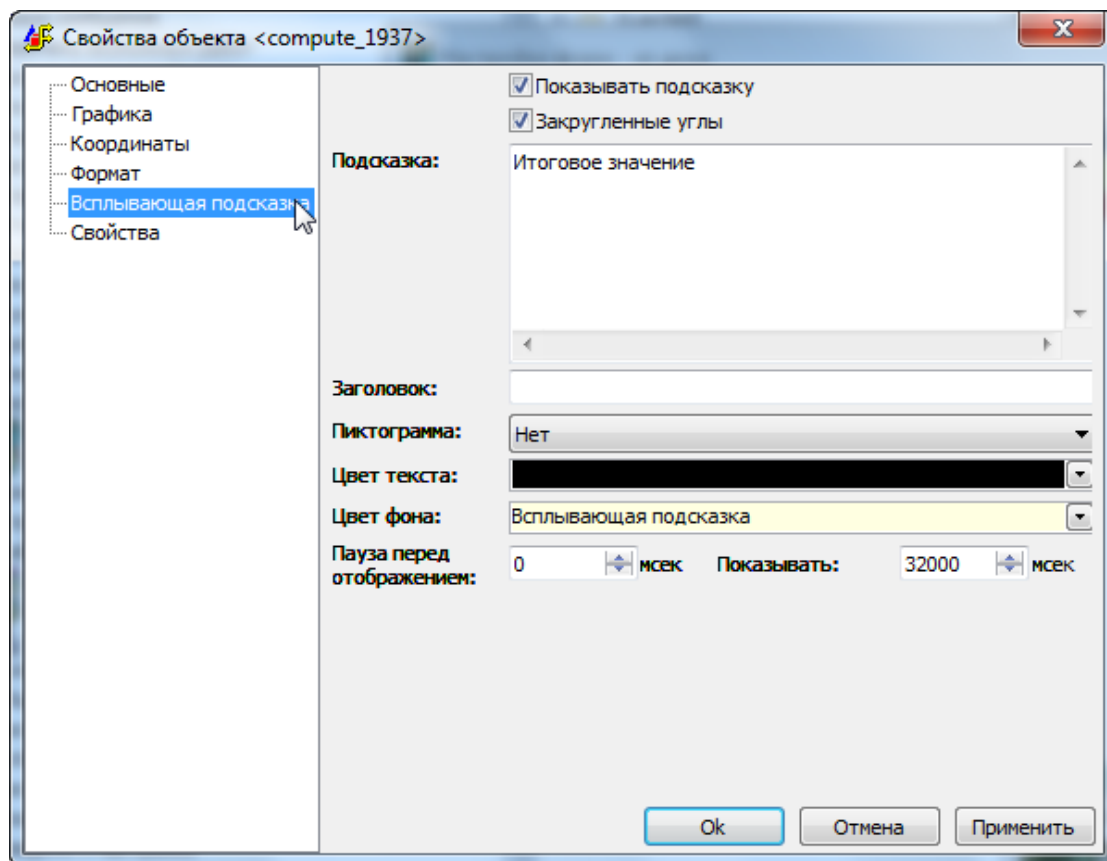


Рисунок 75 Создание всплывающей подсказки

Для отображения подсказки необходимо включить флажок «Показывать подсказку».

Если значение всплывающей подсказки должно формироваться в виде выражения, то указанное выражение следует ввести в поле свойства подсказки «Подсказка.Текст», для чего выбрать пункт «Свойства» окна свойств объекта формы.

10.2.7 Создание названий колонок и вычисляемых полей.

При добавлении вычисляемого поля в форму (или отчет) оно имеет название, формируемое автоматически по его имени. Обычно это что-то похожее на «Compute XXXX», где XXXX – автоматически присваиваемый программой числовой код (см., например, Рисунок 73). При наличии значительного количества вычисляемых полей, возникает потребность формировать их названия в более удобном виде, с использованием символов национального алфавита. Чтобы задать вычисляемому полю произвольное название, необходимо создать текстовое поле, ввести в поле «Текст» пункта «Основные» окна свойств данного текстового поля текст, соответствующий названию вычисляемого поля. Имя данного текстового поля (пункт «Основные») следует изменить, оно должно состоять из имени соответствующего вычисляемого поля, далее, без пробелов указывается «_t» (символ подчеркивания и малая буква t).

При добавлении в форму (или отчет) колонки, аналогичное текстовое поле создается автоматически. Если оно было удалено, то для его восстановления можно создать и настроить текстовое поле, аналогично тому, как рассмотрено выше.

Если текстовое поле добавлять при выделенной в левой части редактора форм колонке или вычисляемом поле, то имя создаваемого текстового поля автоматически формируется в вышеуказанном виде: <Имя колонки>_t.

10.3 Профили и настройки пользователей

Профиль – это настройка, организующая интерфейс пользователя в программе, включающая в себя:

- набор пользовательских меню, одно из которых загружается по умолчанию (см. п. 10.3.1);
- поведение программы при запуске;
- отображаемые названия объектов (см. п. 10.3.2);
- соответствия «Объект – форма» (см. п. 10.3.5);
- список поисковых форм;
- настройку для окна проекта (см. п. 10.3.6);
- контекстные меню для окон объектов (см. п. 10.3.7);
- параметры импорта документов, используемые по умолчанию (см. п. 10.3.9);
- список действий и отчеты, используемых в данном профиле;
- настройки и параметры, относящиеся к работе с сообщениями (см. п. 10.3.11);
- другое.

При создании БД создается один профиль «По умолчанию». Далее могут быть созданы другие профили. Каждый пользователь имеет только один текущий профиль, при назначении пользователю нового профиля, пользователь автоматически исключается из старого профиля. При удалении пользователя из профиля, пользователю автоматически присваивается профиль «По умолчанию», из которого пользователя удалить можно только путем включения в другой профиль.

Пользователю могут быть даны привилегии по созданию и изменению профилей и назначению профиля себе.

Для добавления нового профиля используйте контекстное меню левой верхней части окна «Профили и настройки» (открытие данного окна: «Главное меню» → «Администрирование» → «Профили и настройки»).

Для настройки профиля используйте: «Главное меню» → «Администрирование» → «Профили и настройки» → выбрать требуемый профиль в окне «Профили и настройки» → для выбранного профиля в нижней левой части окна указать требуемую настройку → в правой части окна выполнить соответствующие настройки.

Для части настроек профиля возможно использование персональных настроек (см. рисунок 76 – флажок «Разрешено использовать персональные настройки»).

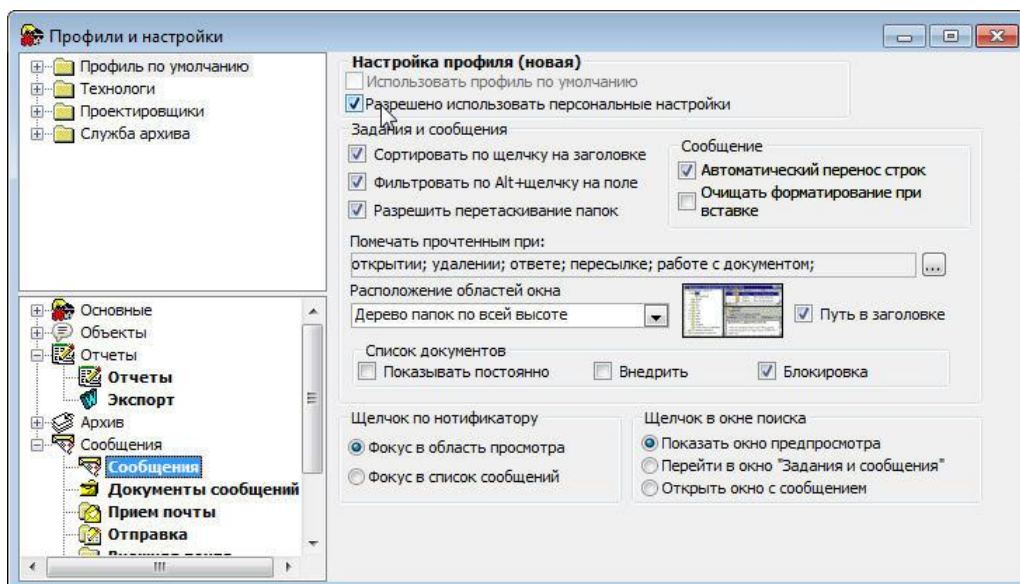


Рисунок 76 Разрешение персональных настроек окна сообщений для профиля по умолчанию

Такие настройки могут устанавливаться персонально отдельным пользователям. Для выполнения персональных настроек требуемому пользователю, выберите соответствующего пользователя, входящего в данный профиль в окне «Профили и настройки» (Рисунок 77).

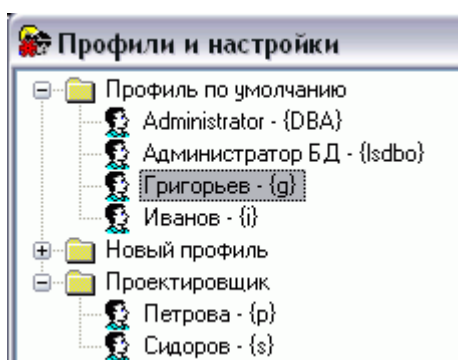


Рисунок 77 Выбор пользователя для установки ему персональных настроек

Для выполнения персональных настроек текущего пользователя выберите: «Главное меню» → «Настройки» → «Настройки пользователя», либо верхнее меню «Настройки» → «Настройки пользователя».

Значительная часть настроек профиля понятна из их названий и дополнительных пояснений не требует, отдельные настройки подробнее описаны далее. Для получения полной информации используйте контекстную подсказку или обратитесь к документации по Lotsia PDM PLUS.

10.3.1 Подключение Главных меню к профилю

Для подключения к профилю настроенного Главного меню (о настройке Главных меню см. раздел 10.1) к нескольким профилям можно, при открытом окне настройки соответствующего меню, использовать пункт верхнего меню «Меню» → «Связь с профилями...» (Подключение настроенного Главного меню к профилям) → в открывшемся окне нажать «Добавить...» → выбрать из открывшегося списка требуемые профили, нажать «Ok».

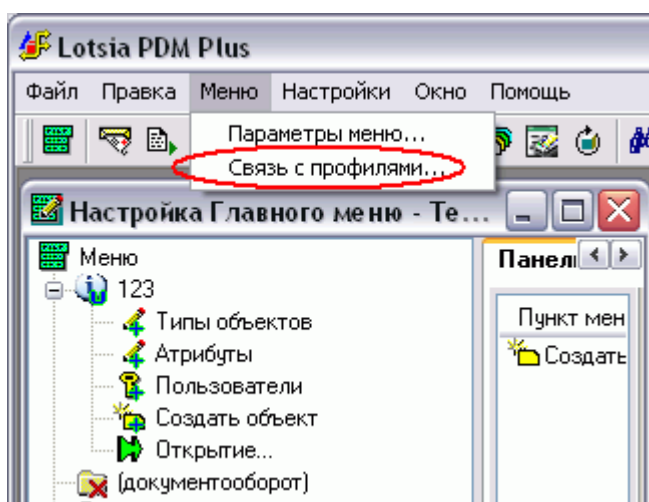


Рисунок 78 Подключение настроенного Главного меню к профилям

Для подключения одного или нескольких меню к профилю, в окне «Профили и настройки» («Главное меню» → «Администрирование» → «Профили и настройки») следует выбрать требуемый профиль → в нижней левой части окна выбрать «Основные» → «Меню» → нажать «Добавить...» → в открывшемся окне выбрать требуемые меню, нажать «Ok».

Задать Главное меню, которое будет отображаться при старте программы можно в поле «приложения» (Рисунок 79).

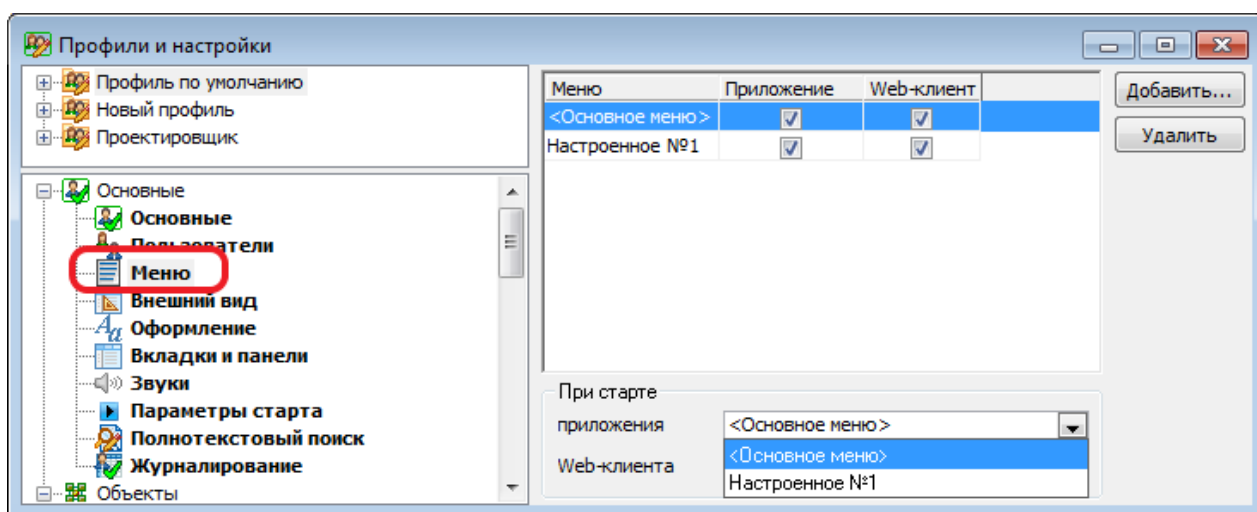


Рисунок 79 Указание Главного меню, отображаемого при старте программы

Из профиля конечного пользователя обычно целесообразно удалить основное меню (удаление любого меню – кнопка «Удалить»). Для добавления в профиль основного меню (в отличие от настроенных меню) используется **кнопка «Основное»** окна выбора меню.

Отметим, что в отличие от обычного пользователя, Администратору основное меню доступно, даже если оно удалено из его профиля.

10.3.2 Формирование названия объекта в дереве

По умолчанию (то есть, без выполнения дополнительных настроек, рассмотренных далее), в качестве названия объекта в дереве связей используется описание объекта. В случае, когда название объекта неизменно, достаточно при создании объекта сформировать описание требуемым образом и не выполнять более никаких настроек. Если название объекта предполагается изменять, более удобным представляется выполнить соответствующие настройки и отображать в названии объекта значения соответствующих

атрибутов, которые могут изменяться. При этом уже нет необходимости при изменении значений атрибутов дополнительно вносить изменения в описание объекта – название изменится автоматически.

Для настройки названий используйте: «Главное меню» → «Администрирование» → «Профили и настройки», выбрать требуемый профиль или Главное меню → «Настройки» → «Настройки пользователя» (аналогично, если разрешены и используются персональные настройки – Верхнее меню «Настройки» → «Настройки пользователя»); далее: «Объекты» → «Названия объектов».

В правой части окна (Рисунок 80) название объектов может быть сформировано для всех типов объектов – «По умолчанию», либо для групп типов объектов. Настраиваемая группа может содержать 1 или более тип объектов.

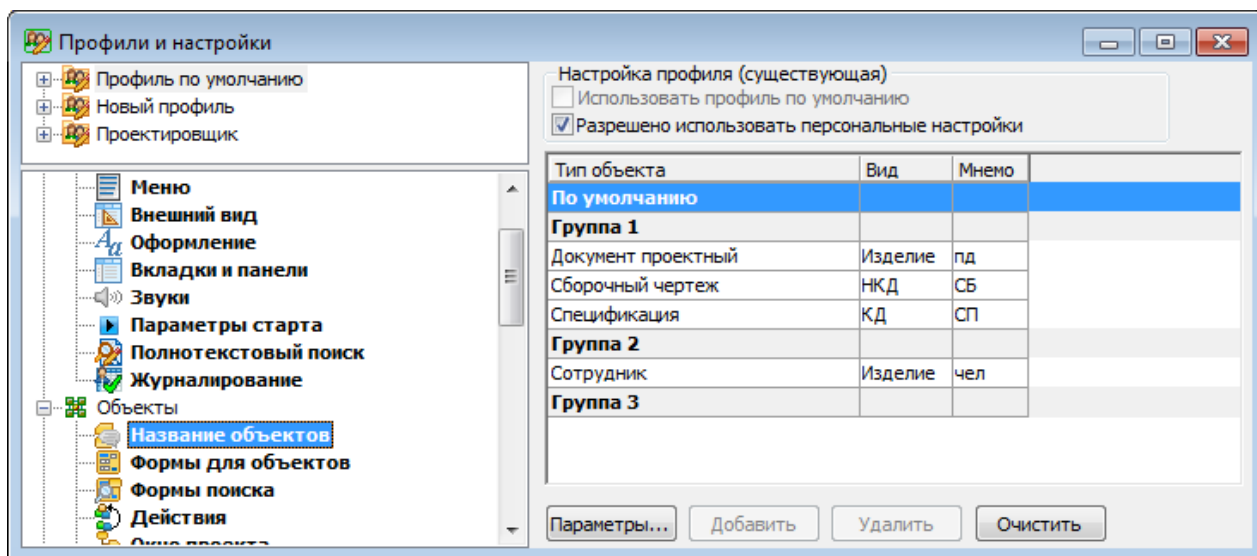


Рисунок 80 Указание типов объектов или их групп, для которых будет настраиваться формирование названия

Для добавления типа объекта в группу используйте кнопку «Добавить», либо контекстное меню выделенной строки на вкладке.

Для формирования названия выделите соответствующую строку в правой части окна и нажмите «Параметры...», либо дважды щелкните по выделенной строке левой клавишей мыши. В открывшемся окне (Рисунок 81) сформируйте список отображаемых в названии объекта значений атрибутов.

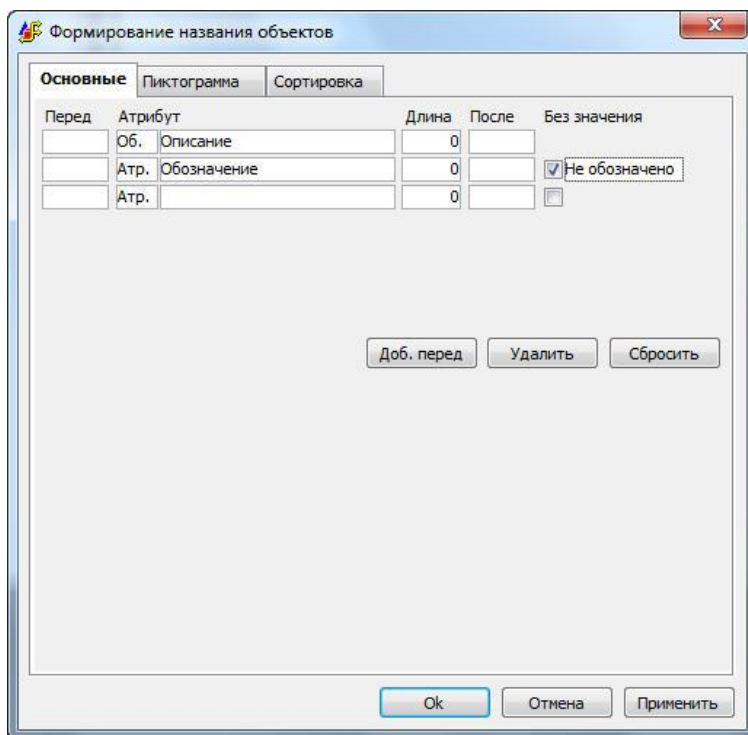


Рисунок 81 Формирование названий объектов

При включенном флажке «Без значения» можно ввести значение, которое будет отображаться в данном месте названия объекта в дереве при отсутствии соответствующего атрибута у объекта.

10.3.3 Настройка отображаемой пиктограммы

По умолчанию (то есть, без выполнения дополнительных настроек, рассмотренных в данном разделе), в качестве пиктограммы для экземпляра объекта в дереве связей используется пиктограмма, указанная в настройках типа данного объекта (см. раздел 4.3.2)

В случае, когда требуется для данного экземпляра объекта отобразить пиктограмму, отличную от заданной по умолчанию, необходимо выполнить соответствующую настройку отображаемой пиктограммы.

Для настройки пиктограммы, отображаемой для конкретного экземпляра объекта, используйте: «Главное меню» → «Администрирование» → «Профили и настройки», выбрать требуемый профиль или Главное меню → «Настройки» → «Настройки пользователя» (аналогично, если разрешены и используются персональные настройки – Верхнее меню «Настройки» → «Настройки пользователя»); далее: «Объекты» → «Названия объектов».

Аналогично настройке названий объектов (см. раздел 10.3.2), пиктограммы могут быть заданы для всех типов объектов – «По умолчанию», либо для групп типов объектов. Настраиваемая группа может содержать 1 или более тип объектов. Для добавления типа объекта в группу используйте кнопку «Добавить», либо контекстное меню выделенной группы.

Для настройки пиктограмм выделите соответствующую строку в правой части окна настройки профиля и нажмите «Параметры...», либо дважды щелкните по выделенной строке левой клавишей мыши. В открывшемся окне перейдите на вкладку «Пиктограмма» и снимите флажок «Использовать стандартную пиктограмму» (Рисунок 82).

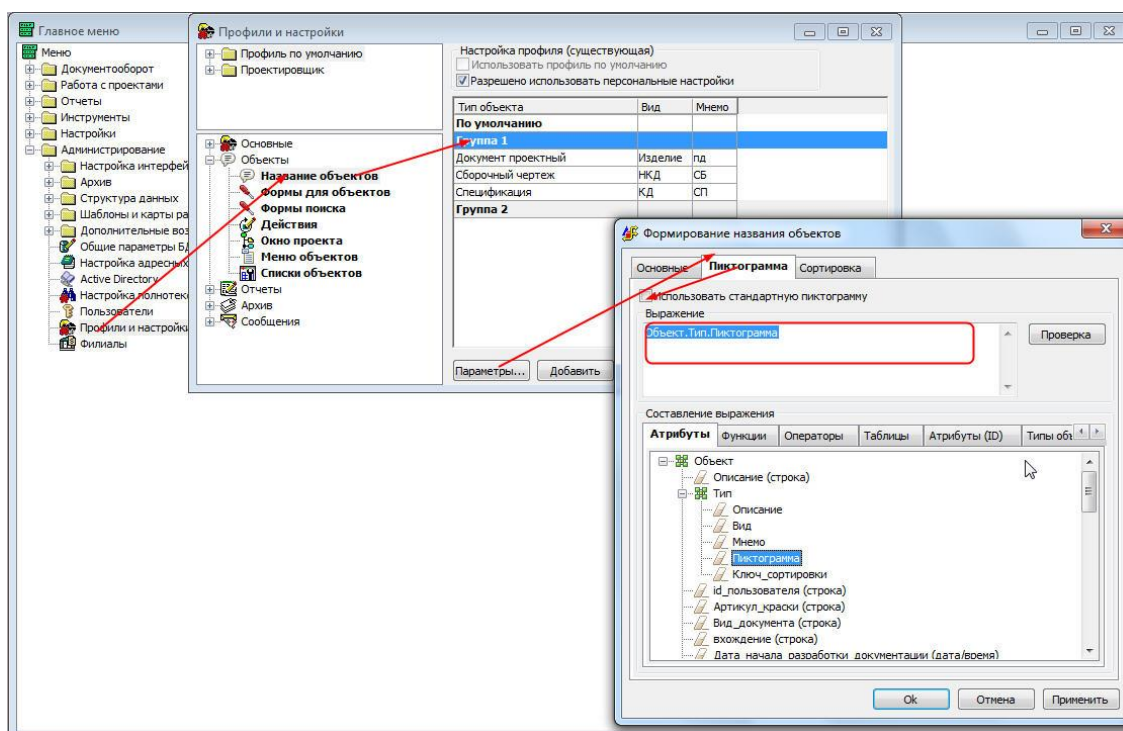


Рисунок 82 Формирование выражения для отображаемой пиктограммы

В окне редактора выражений сформируйте выражение, возвращающее имя файла требуемой пиктограммы, например:

- Объект.Тип.Пиктограмма – используется стандартная пиктограмма для данного типа объекта;
- If(Объект.Вид_документа= 'Аннулированный', 'Аннулированный_док.ico', Объект.Тип.Пиктограмма) – для объектов, атрибут «Вид документа» у которых имеет значение «Аннулированный» отображать пиктограмму «Аннулированный_док.ico», в остальных случаях – отображать стандартную пиктограмму для данного типа объекта;
- If(IsNull(Объект.Доп_пиктограмма), Объект.Тип.Пиктограмма, Объект.Доп_пиктограмма) – при отсутствии у данного экземпляра объекта атрибута «Доп_пиктограмма» отображать стандартную пиктограмму для данного типа объекта, в остальных случаях отображать пиктограмму, имя файла которой содержится в указанном атрибуте.

10.3.4 Настройка сортировки объектов, находящихся в дереве связей на одном уровне

По умолчанию (то есть, без выполнения дополнительных настроек, рассмотренных в данном разделе), сортировка объектов, находящихся в дереве связей на одном уровне, осуществляется следующим образом:

- По виду объектов в следующем порядке: КД, НКД, Прч.Док., Изделие;
- Среди объектов одного вида – по ключу сортировки, определяемому для типа объекта (см. раздел 4.3.2). Объекты с большим значением ключа сортировки располагаются выше, допустимо использовать отрицательные значения ключа сортировки;

- Далее – по названию объектов в дереве связей, в алфавитном порядке. Подробнее о формировании названий объектов см. раздел 10.3.2

Если описанная сортировка объектов не достаточна, либо противоречит предъявляемым требованиям, следует выполнить настройку сортировки.

Для настройки сортировки объектов, находящихся в дереве связей на одном уровне, используйте: «Главное меню» → «Администрирование» → «Профили и настройки», выбрать требуемый профиль или Главное меню → «Настройки» → «Настройки пользователя» (аналогично, если разрешены и используются персональные настройки – Верхнее меню «Настройки» → «Настройки пользователя»); далее: «Объекты» → «Названия объектов».

Аналогично настройке названий объектов (см. раздел 10.3.2), сортировка может быть определена для всех типов объектов – «По умолчанию», либо для групп типов объектов. Настраиваемая группа может содержать 1 или более тип объектов. Для добавления типа объекта в группу используйте кнопку «Добавить», либо контекстное меню выделенной группы.

Для настройки сортировки выделите соответствующую строку в правой части окна настройки профиля и нажмите «Параметры...», либо дважды щелкните по выделенной строке левой клавишей мыши. В открывшемся окне перейдите на вкладку «Сортировка» и снимите флажок «Использовать стандартное выражение» (Рисунок 83)

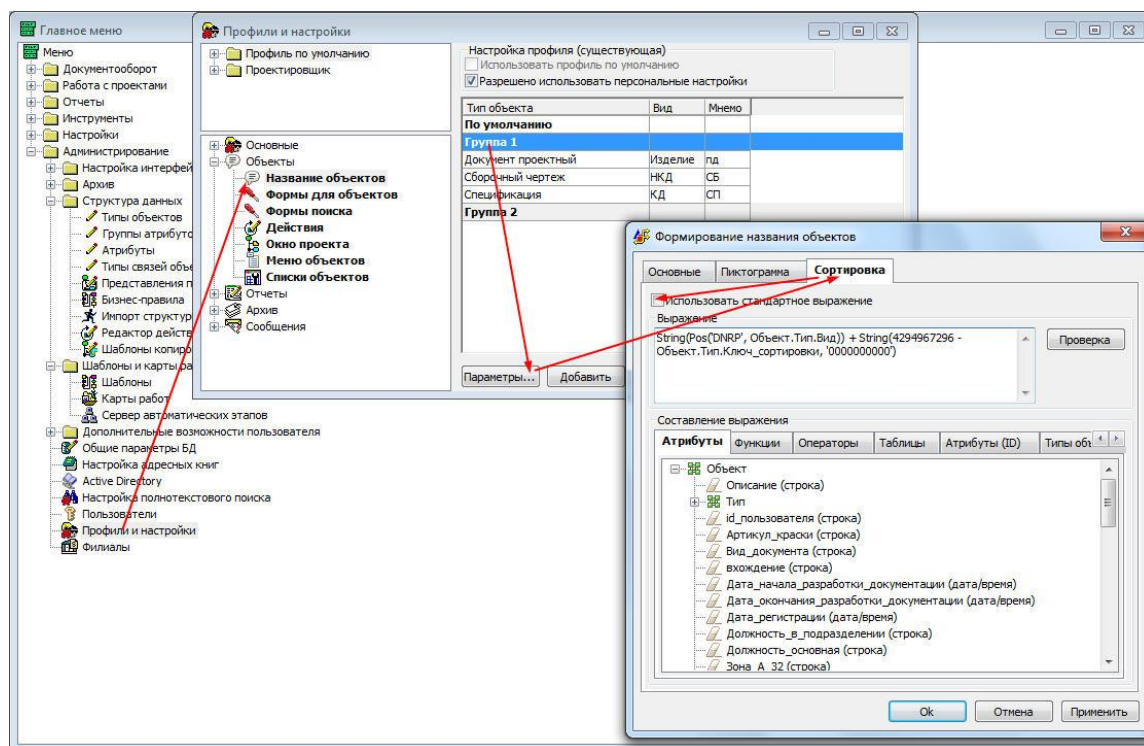


Рисунок 83 Формирование выражения для сортировки объектов

В окне редактора выражений сформируйте выражение, возвращающее строку, используемую при сортировке данного объекта в дереве в алфавитном порядке, например:

- `String(Pos('DNRP', Объект.Тип.Вид)) + String(4294967296 - Объект.Тип.Ключ_сортировки, '0000000000')` – стандартное выражение для сортировки, сортировка по умолчанию. Поясним, что значения параметра «Вид объекта», которое хранится в БД, имеет одно из четырех значений: D,

N, R, P, которые в отображаются в программе, соответственно, в виде: КД, НКД, Прч.Док., Изделие;

- Объект.Порядок_сортировки – сортировка выполняется по возрастанию значению атрибута «Порядок сортировки».

10.3.5 Настройка соответствия форм типам объектов

Для данной настройки используется пункт «Формы для объектов» для выбранного профиля (Рисунок 84) список отображаемых в окне дерева связей форм может быть сформировано для всех типов объектов – «По умолчанию», либо для отдельных типов объектов или их групп.

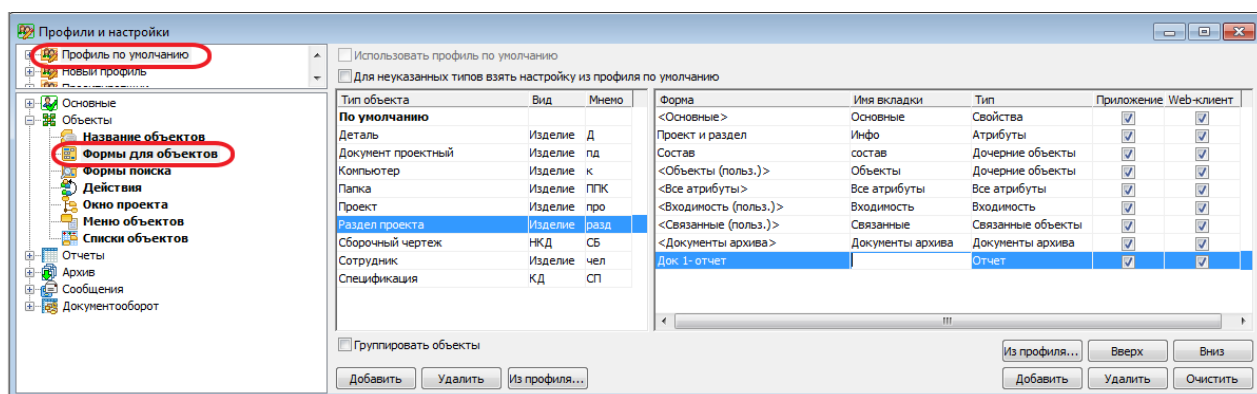


Рисунок 84 Формы для объектов

Для работы с группами типов объектов необходимо включить флажок «Группировать объекты» (Рисунок 84).

Для добавления типа объекта в группу используйте кнопку «Добавить», либо контекстное меню строки, выделенной в левой части вкладки.

Для добавления формы в список сопоставленных данному типу объектов (или группе, или «По умолчанию») форм, выделите соответствующую строку в левой части вкладки и нажмите «Добавить», либо воспользуйтесь контекстным меню правой части окна.

В открывшемся окне (Рисунок 85) выберите требуемую форму из списка.

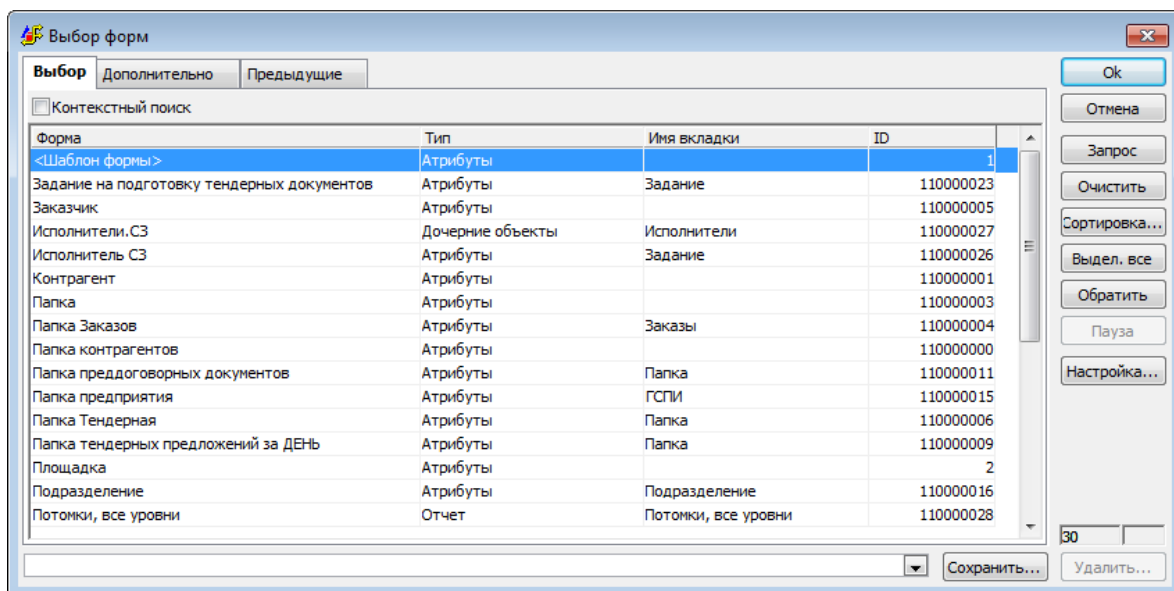


Рисунок 85 Окно выбора форм

Список стандартных (предопределенных) форм (вкладок) находится на вкладке «Дополнительно». О настройке стандартных вкладок см. раздел 10.4.

В качестве форм (вкладок) могут также использоваться отчеты, в которых выполнены соответствующие настройки (будет рассмотрено далее, см. раздел 20.3). Формы типа «Отчет» отображаются в окне выбора форм (Рисунок 85), их сопоставление типам объектов не отличается от сопоставления форм других типов.

Порядок расположения вкладок для данного типа объекта меняется с помощью кнопок «Вверх» и «Вниз» (см. рисунок 84).

Имя вкладки для данного типа объекта (или группы типов, при включенном флажке «Группировать объекты» (Рисунок 84)) может быть изменено непосредственно в окне «Профили и настройки» (Рисунок 84).

Перейти к редактированию формы можно также из окна настройки профиля, воспользовавшись контекстным меню выделенной строки вкладки, пункт «Связанные» (Рисунок 86).

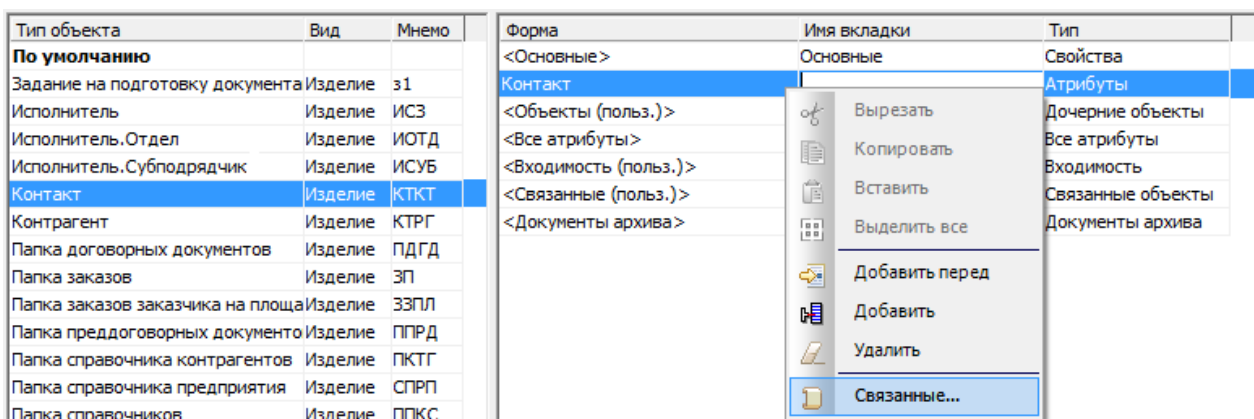


Рисунок 86 Контекстное меню выделенной вкладки. Переход к редактированию соответствующей формы

10.3.6 Настройки окна проекта

Данная настройка (см. рисунок 87) позволяет:

- определить используемый по умолчанию тип подчиненной связи;
- разрешить выделение и копирование через буфер обмена текста в колонках форм без перевода формы в режим редактирования (флажок «Фокус в формах атрибутов»);
- запретить переход в режим редактирования по двойному щелчку в форме типа атрибуты. При выключенном флажке форма позволяет только просматривать данные и нажимать кнопки, расположенные в форме. (Перевод формы в режим редактирования возможен с использованием пункта «Свойства» контекстного меню объекта);
- включить/отключить пункты «Создать» в окнах выбора объектов (флажок «Меню создать»);
- включить/отключить видимость кнопок predetermined фильтров по видам объектов на инструментальной панели и соответствующие пункты верхнего меню «Правка» (флажок «Меню фильтров по видам объектов»);
- при выключенном флажке «Сохранять сортировку и автофильтр» состояние табличных форм (например – формы типа «Дочерние объекты», «Входимость»...) будет всегда умолчательным. Если флажок включен – до закрытия окна дерева связей для каждого выделенного в дереве объекта сохраняются измененные пользователем настройки форм (сортировка и автофильтр, выделенная строка в форме дочерних объектов).
- Выполнение/невыполнение автоматической проверки наличия потомков для объектов развернутого уровня. При отсутствии у объекта потомков, слева от объекта в дереве не отображается индикатор «+» (если флажок «Проверять наличие дочерних объектов» включен).

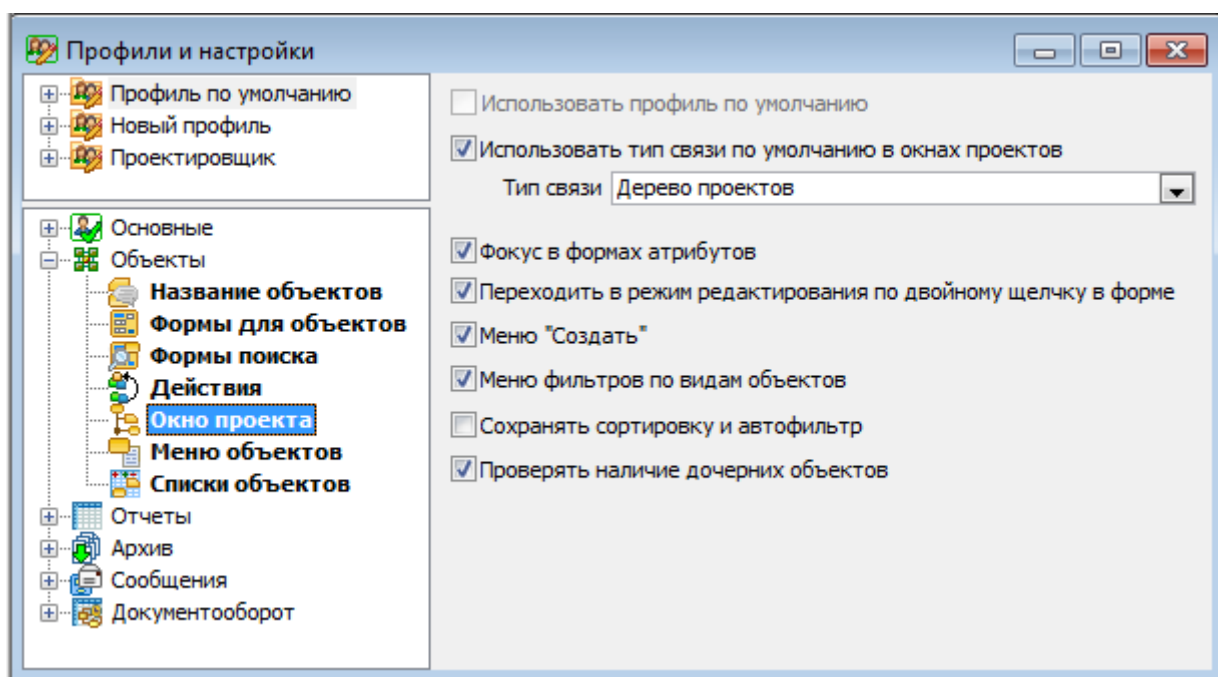


Рисунок 87 Настройка окна проекта

10.3.7 Настройка контекстных меню и инструментальной панели окна

Возможна настройка контекстных меню окон, в которых ведется работа с объектами – Окна проекта, Окна поиска объектов Окна подборки. Одновременно производится и настройка верхних меню «Правка» и «Объект», а также инструментальной панели – «Панели окна».

Для указанной настройки используйте: Главное меню → Администрирование → Профили и настройки → для выбранного профиля – пункт «Меню объектов» (Рисунок 88).

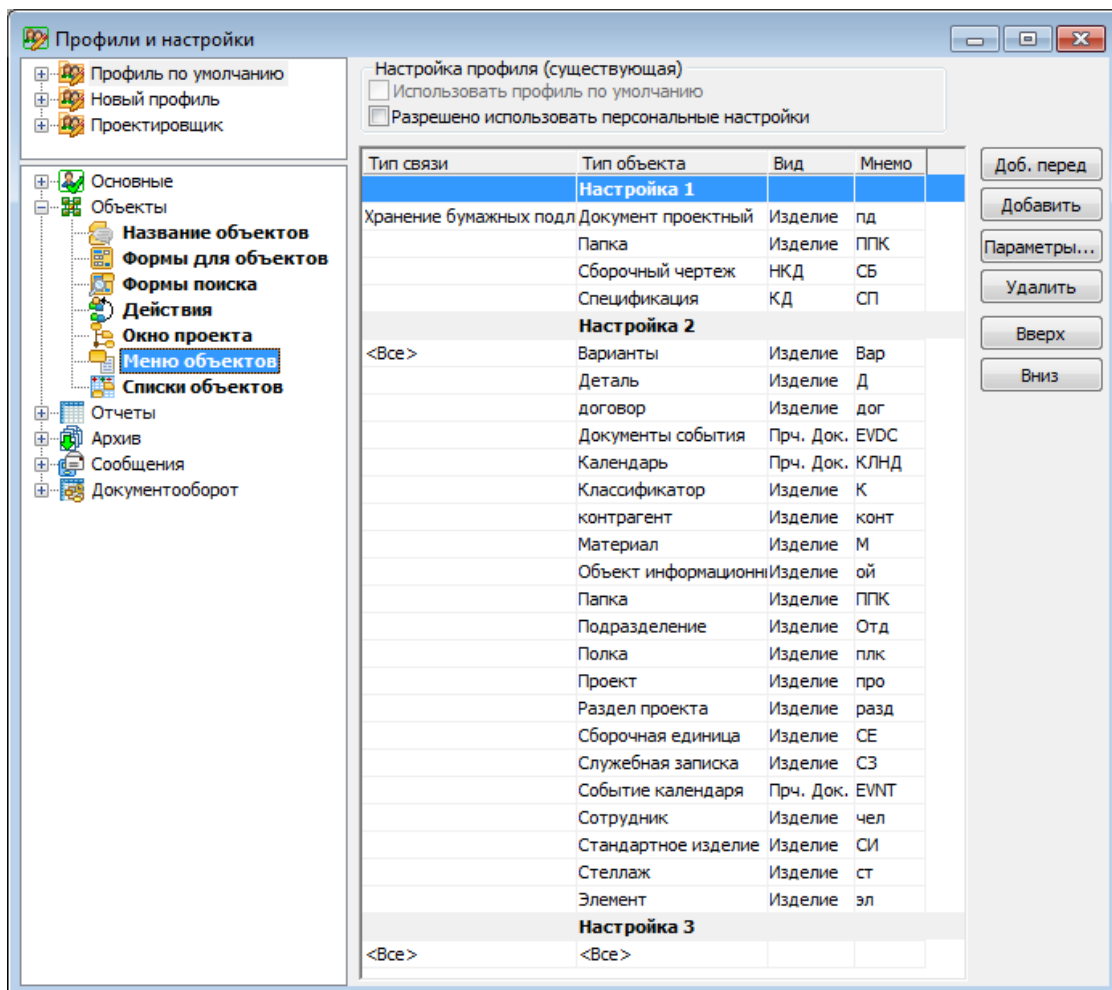


Рисунок 88 Вкладка «Меню объектов»

Могут быть заданы различные настройки меню, привязанные к различным типам объектов и различным типам подчиненных связей. В конце списка настроек всегда присутствует стандартная настройка, привязанная ко всем типам объектов и ко всем типам связей.

Настройки, привязаны к типам объектов и к типам связей могут противоречить друг другу. При наличии таких настроек, для инициализации меню будет использоваться первая сверху настройка, которая удовлетворяет условиям привязки. Изменить положение настройки можно с помощью кнопок «Вверх» и «Вниз».

Для добавления новой настройки используйте кнопку «Добавить», либо одноименный пункт контекстного меню данной вкладки. Для редактирования настройки следует дважды щелкнуть по выделенной строке настройки левой клавишей мыши, либо использовать кнопку «Параметры...» (Рисунок 88).

В открывшемся окне настройки (Рисунок 89) на вкладке «Основные» производится настройка пунктов меню.

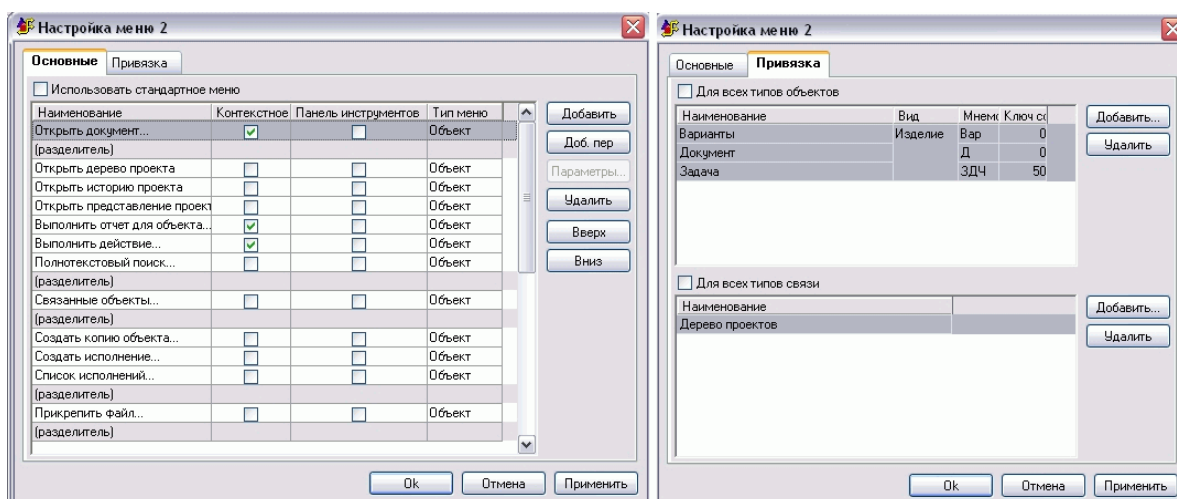


Рисунок 89 Вкладки окна настройки меню

Настройка пунктов меню возможна при снятом флажке «Использовать стандартное меню». Для настройки отображения пункта меню в контекстном меню и на инструментальной панели используются соответствующие флажки на вкладке «Основные» (Рисунок 89). Для добавления и удаления пунктов меню можно использовать кнопки или контекстное меню указанного окна, при этом откроется окно выбора пунктов меню (Рисунок 90).

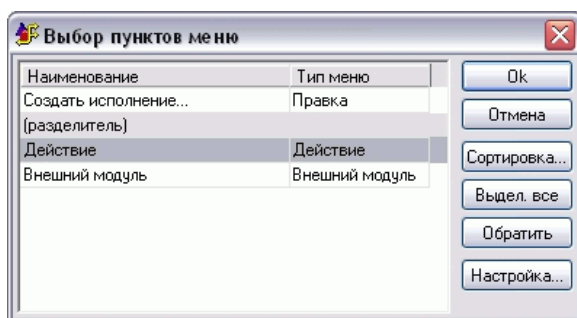


Рисунок 90 Окно выбора пунктов меню

Для добавления доступны ранее удаленные стандартные пункты меню, а также дополнительные пункты меню, которым сопоставляются действия или внешние модули. Создание дополнительных пунктов меню не отличается от создания аналогичных пунктов Главного меню (см. п. 10.1, Рисунок 51).

10.3.8 Настройка вкладок и закрепленной панели окна программы

Использование вкладок и закрепленной панели программы определяется в пункте «Вкладки и панели» настроек профиля или настроек пользователя (Рисунок 91).

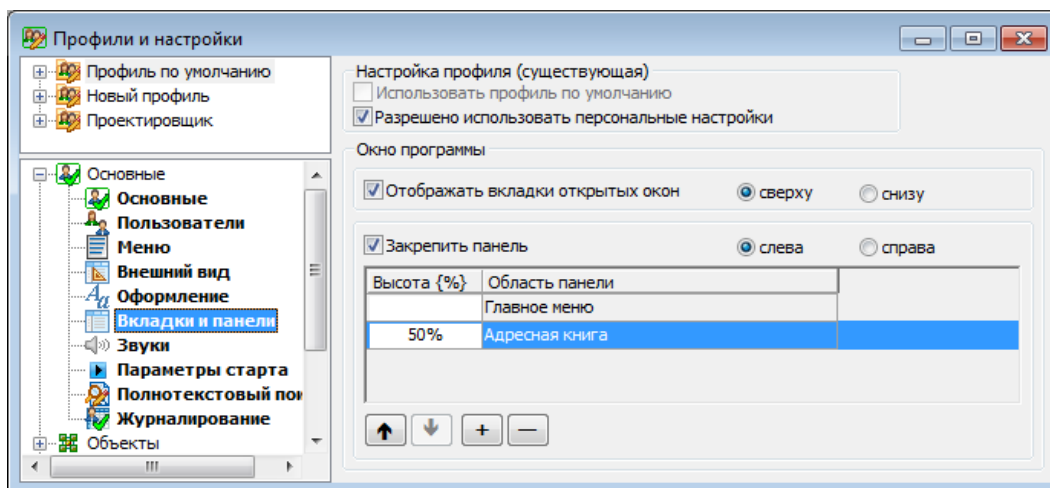


Рисунок 91 Настройка вкладок открытых окон и закрепленной панели

Закрепленная панель не перекрывается другими открытыми окнами программы, высота её областей (в % или единицах PowerBuilder) может быть указана слева от названия области (Рисунок 91).

Применение и местоположение панели и вкладок определяются с помощью соответствующих флажков и переключателей.

Результат использования прикрепленной панели и вкладок – см. рисунок 92

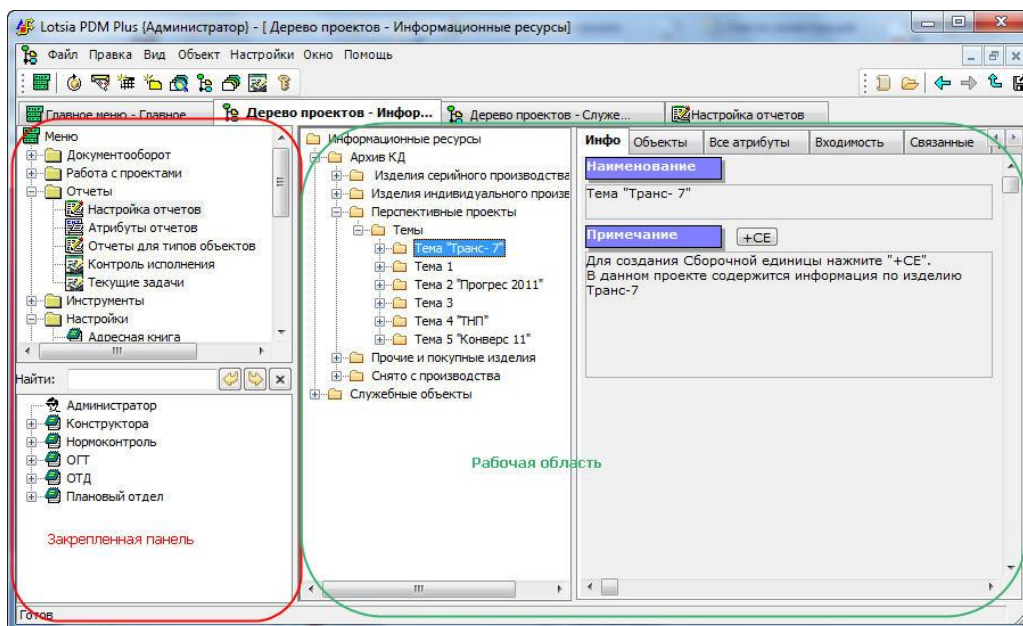


Рисунок 92 Результат включения вкладок и закрепленной панели. Пример

Поведение окон рабочей области программы определяется настройками, выполняемыми в пункте «Профилей и настроек»: «Основные» → «Внешний вид» → выпадающий список «Поведение окон».

10.3.9 Параметры импорта документов

Импорт документов и настройка архива будут рассмотрены далее, см. раздел 19.

Возможно выполнение настроек, используемых по умолчанию при импорте документов в архив и работе с документами архива. Указанные настройки выполняются в окне «Профили и настройки» (либо «Настройки пользователя», если разрешено использовать персональные настройки), с использованием пунктов настроек, входящих в папку в папку «Архив» (Рисунок 93).

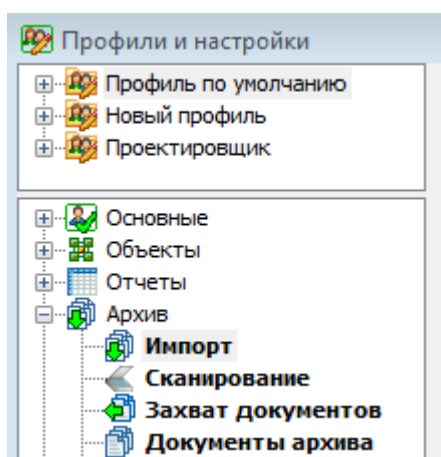


Рисунок 93 Пункты настроек архива

Пункт «Импорт» (Рисунок 94) позволяет задать умолчания для импорта документов, которые включают в себя настройки библиотеки, способа помещения документа в архив, методов формирования описания документа и примечания к нему, права доступа для компонент, внешний вид окна импорта, а также использования импорта действием.

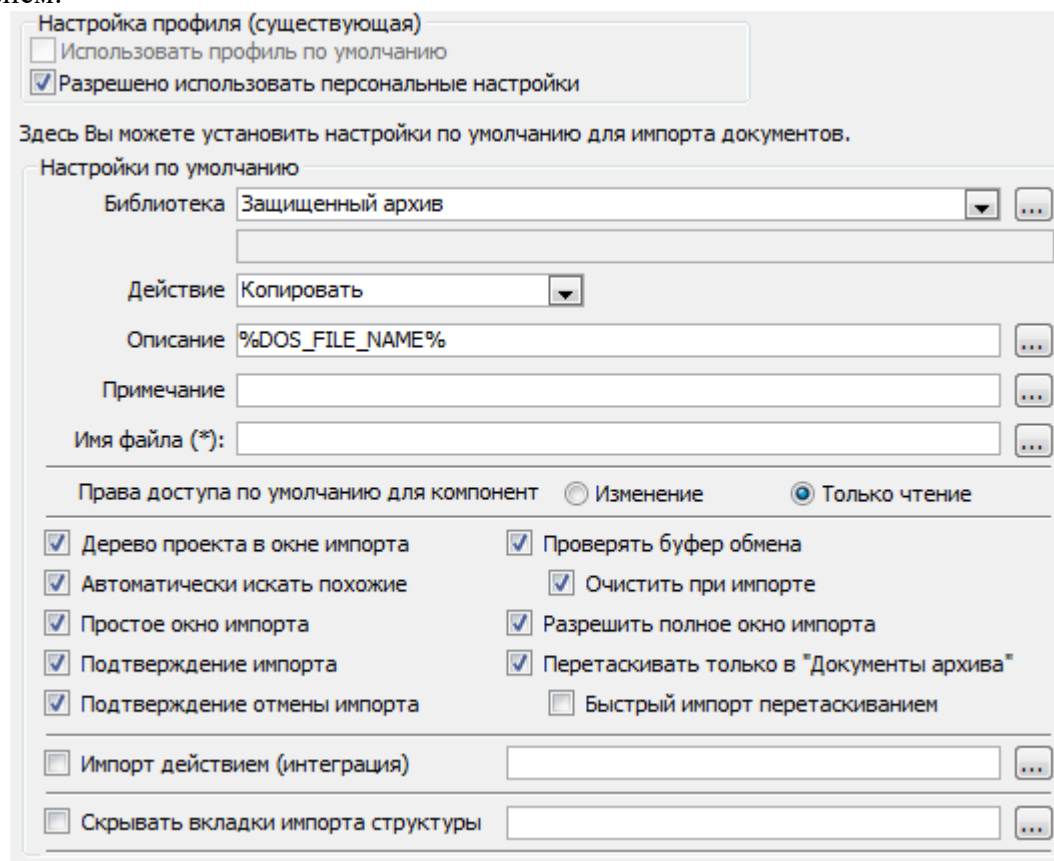


Рисунок 94 Настройки, выполняемые в пункте «Импорт»

Импорт действием – запуск специального действия при сохранении документа из интегрированного приложения, вместо открытия стандартного окна импорта.

Пункт «Сканирование» обеспечивает настройку умолчаний при получении изображений со сканера (требуется, чтобы был установлен модуль Twain-сканер).

Пункт «Захват документов» обеспечивает настройку умолчаний для захвата документов (Рисунок 95).

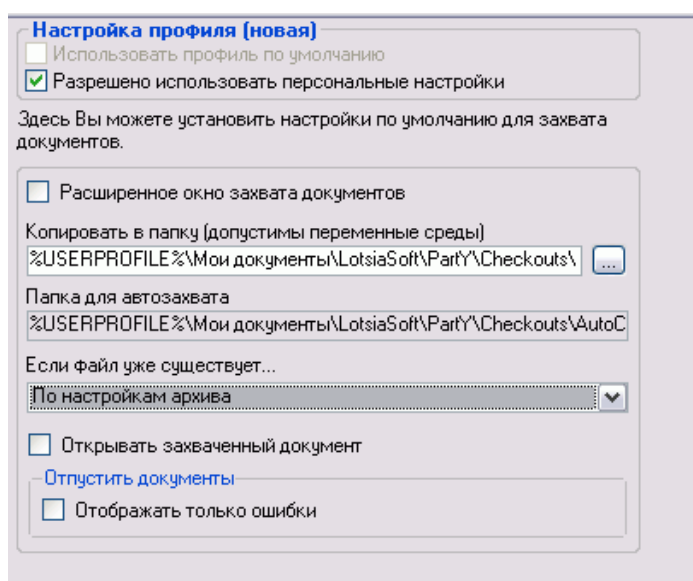


Рисунок 95 Настройка параметров захвата

Пункт «Документы архива» (Рисунок 96) позволяет настроить умолчания для вкладки «Документы архива» окна дерева связей.

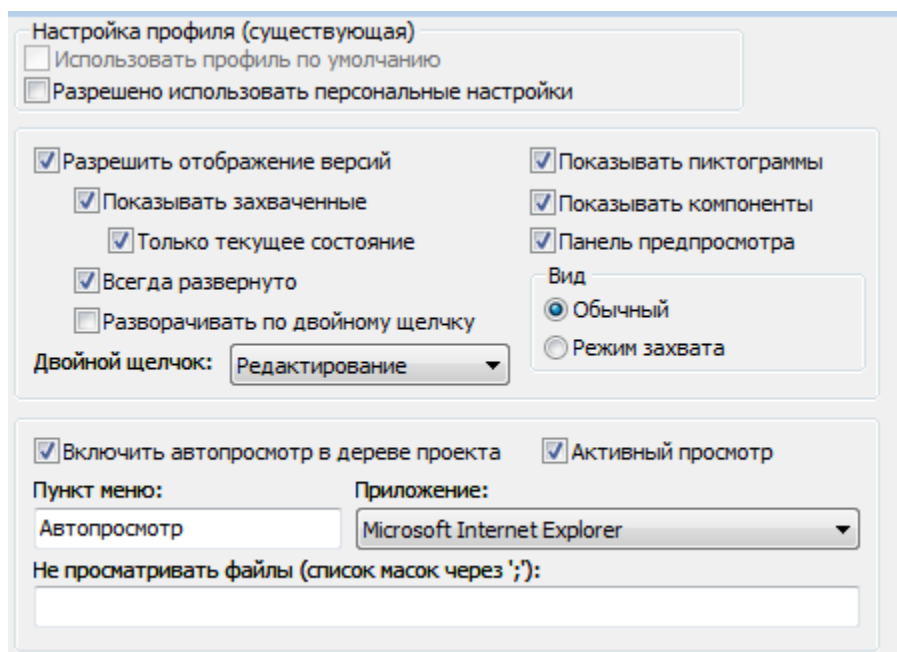


Рисунок 96 Настройка «Документы архива»

Для включения возможности просмотра выделенных на вкладке «Документы архива» документов в плавающем окне следует включить флажок «Включить автопросмотр в дереве проекта». Автопросмотр документов возможен с использованием приложения Microsoft Internet Explorer или AutoVue. В последнем случае необходимо наличие установленной программы AutoVue и установленного ActiveX элемента «AutoVueX».

Подробную информацию об использовании каждой настройки можно получить, воспользовавшись контекстной справкой, либо документацией по системе Lotsia PDM PLUS.

10.3.10 Настройки стандартных окон и вкладок

Возможно выполнение настроек отображаемых в стандартных окнах программы колонок. Для различных окон и вкладок настройка во многом идентична, но запуск настройки выполняется различным образом.

Настройка окон выбора осуществляется следующим образом: «Главное меню» → «Администрирование» → «Профили и настройки», выбрать требуемый профиль (или «Настройки» → «Настройки пользователя») → «Объекты» → «Списки объектов». В правой части (Рисунок 97) путем формирования списка отображаемых параметров и осуществляется настройка соответствующих окон.

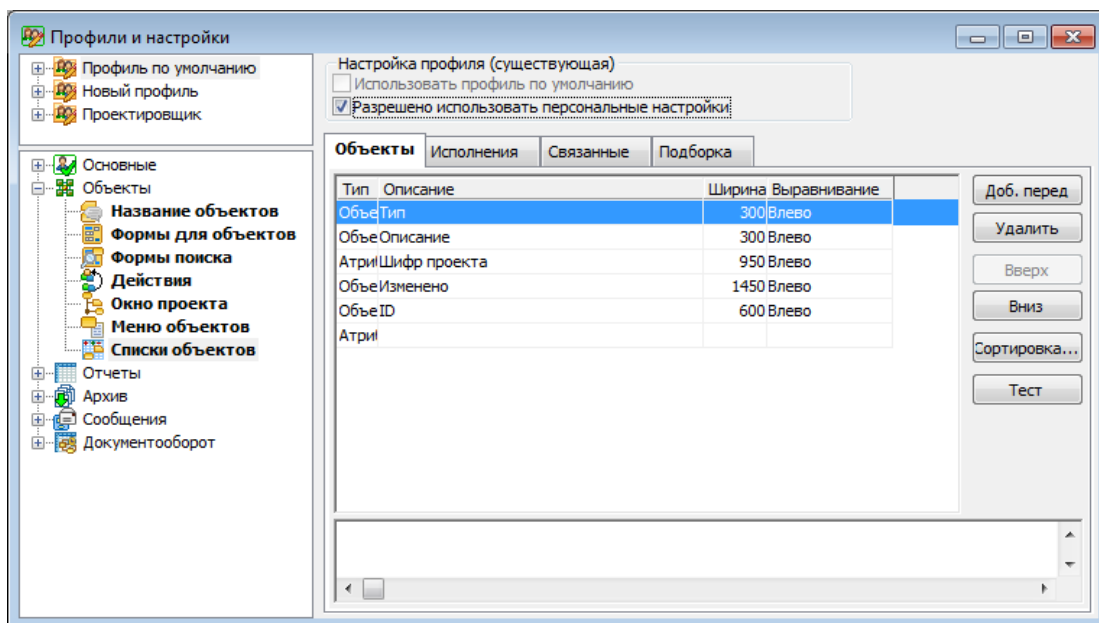


Рисунок 97 Настройка внешнего вида окон выбора

Настройка выполняется с использованием контекстного меню окна и кнопок в правой части окна.

Настройка, выполненная на вкладке «Объекты» применяется для вкладки «Выбор» окна поиска (Рисунок 32) и окна выбора (Рисунок 30) объектов.

Настройка, выполненная на вкладке «Исполнения» применяется для окна со списком исполнений (см. п. 6.3).

Настройка, выполненная на вкладке «Связанные», применяется для окна со списком связанных объектов (Рисунок 37), а также для вкладки «Связанные» окна дерева проектов (Рисунок 36).

Настройка, выполненная на вкладке «Подборка», применяется для окна подборки как форма по умолчанию (см. п. 8.3)

Стандартные вкладки окна дерева проектов («Объекты», «Входимость», «Атрибуты») также настраиваемые. Их настройка описана в разделе 10.4

10.3.11 Настройки и параметры, используемые при работе с сообщениями

Работа с сообщениями будет рассмотрена далее, см. раздел 22.

Настройки и параметры, относящиеся к работе с сообщениями, объединены в папку «Сообщения» окна профилей и настроек (или «Настроек пользователя») – см. рисунок 98

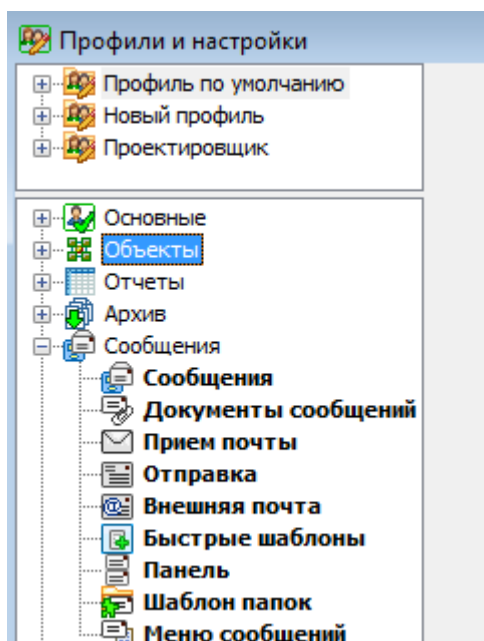


Рисунок 98 Настройки, используемые при работе с сообщениями

- Пункт «Сообщения» обеспечивает настройку окна «Задания и сообщения», а также реакцию программы на щелчок по нотификатору или строке в окне поиска сообщений.
- Пункт «Документы сообщений» позволяет определить места хранения пересылаемых файлов.
- Пункты «Прием почты» и «Отправка» позволяют задать поведение системы, соответственно, при приеме и отправке сообщений.
- Пункт «Панель» определяет конфигурацию всплывающей панели сообщений.
- Пункт «Шаблон папок» (только для окна «Профили и настройки») позволяет настроить дерево почтовых папок, отображаемых по умолчанию для пользователей, входящих в данный профиль.
- Пункт «Меню сообщений» обеспечивает возможность настройки контекстного меню сообщений и соответствующей инструментальной панели, аналогично «Меню объектов» (п. 10.3.7).

Информацию о назначении конкретных параметров настройки можно получить, воспользовавшись контекстной справкой.

10.4 Настройка стандартных вкладок окна дерева проектов

Для настройки стандартных вкладок окна дерева проектов (кроме вкладки «Связанные», о настройке данной вкладки – см. раздел 10.3.10) следует, при выделенном окне дерева проектов, в верхнем меню «Настройки» выбрать «Настройки окна...». В открывшемся окне (Рисунок 99), идентично настройкам окон выбора (Рисунок 35) выполняются настройки для вкладок «Объекты» и «Входимость».

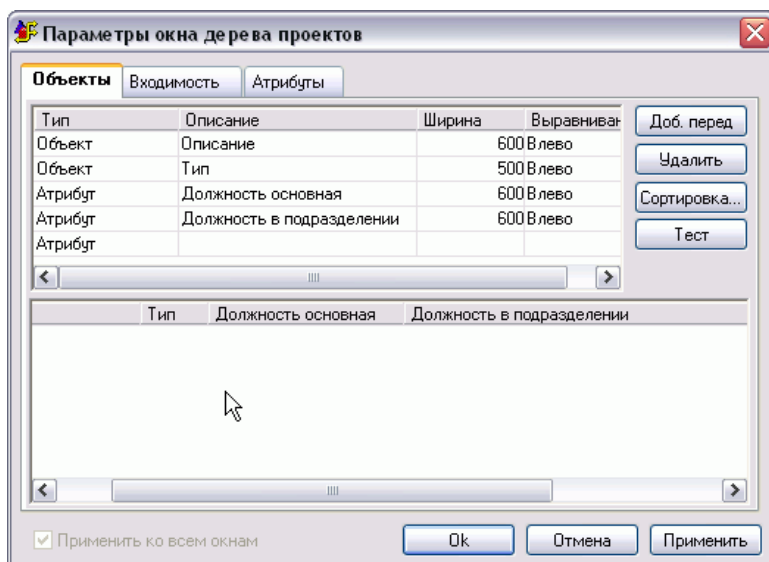


Рисунок 99 Настройка внешнего вида для вкладок окна дерева проектов. Настройка вкладки «Объекты»

Настройка внешнего вида вкладки «Атрибуты» выполняется путем установки флажков для показа соответствующих групп атрибутов (Рисунок 100).

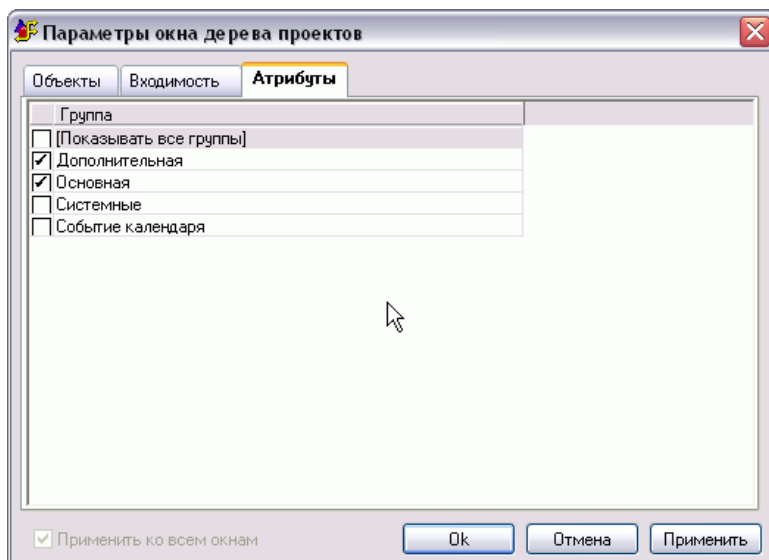






Рисунок 100 Настройка внешнего вида вкладки «Атрибуты» окна дерева проектов

10.5 Представления проекта и фильтры

Кнопки    инструментальной панели окна обеспечивают включение/выключение показа в дереве объектов отдельных видов. Если данные кнопки не используются, их можно убрать из инструментальной панели отключив флажок «Меню фильтров по типам документов» в настройке данного профиля (см. п. 10.3.6). Кнопка  позволяет настроить фильтрацию по типам объектов.

Дерево связей проекта отображается в окне проекта. Если задать определенные правила отображения, проект можно представить и в другом виде, который называют представлением проекта.

В отличие от фильтрации, в представлении могут отображаться все удовлетворяющие заданным условиям объекты, лежащие на любом уровне вложенности в иерархии дерева.

Для создания представлений проекта используется пункт Главного меню «Представления проектов», расположенный в папке «Администрирование» → «Структура данных» стандартного меню Администратора. Кнопка «Создать» в открывшемся окне выбора представления проекта используется для добавления нового представления.

В открывшемся окне, на вкладке «Основные» следует ввести описание создаваемого представления и переключателем указать порядок его формирования (Рисунок 101).

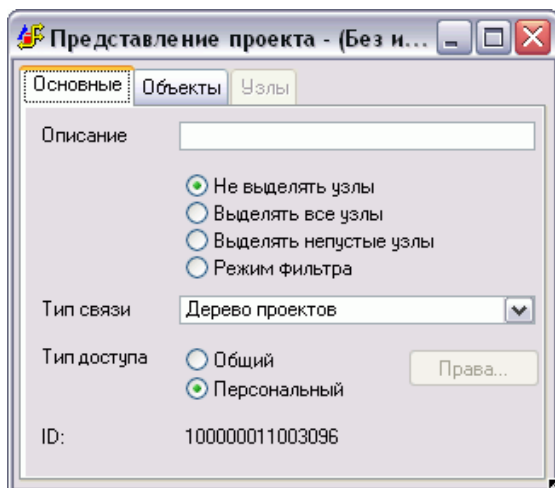


Рисунок 101 Создание представления проекта

Представление можно сделать доступным для других пользователей (переключатель в нижней части вкладки перевести в положение «Общий»). В этом случае следует также определить права (см. п. 16) на данное представление.

Далее, в зависимости от выбранного порядка формирования представления, следует задать условия отбора для объектов и/или узлов.

Для открытия проектов в созданных представлениях используется пункт Главного меню «Работа с проектами» → «Представление проекта».

10.6 Резюме

Профиль – это настройка, организующая интерфейс пользователя в программе.

Настройки профилей выполняются администратором программы, либо уполномоченными на это пользователями. Для создания некоторых настроек, например: Главного меню, инструментальных панелей, атрибутивных форм и др. используются специальные инструментальные средства, запускаемые из Главного меню. Другие настройки интерфейса выполняются непосредственно редактированием профиля. Выполненные настройки интерфейса становятся доступными пользователю после их подключению к его профилю.

В настройках меню и инструментальных панелей можно включить или отключить показ любого из стандартных пунктов, изменить их пиктограммы и названия, а также добавить собственные пункты, сопоставляя им действия или внешние модули.

Сформированное название объектов в дереве может содержать требуемый набор атрибутов, а также информацию, выводимую при отсутствии данного атрибута. Название может формироваться различным образом для объектов различных типов.

Для объектов могут использоваться стандартные, а также настроенные формы следующих типов:

Форма «Атрибуты» содержит информацию о текущем объекте и предназначена, в том числе, для ввода значений атрибутов. Может содержать кнопки, по нажатию которых запускаются действия. Форма данного типа может использоваться и в качестве поисковой формы.

Формы «Дочерние объекты», «Родительские объекты» и «Связанные объекты» представляют собой таблицы, строки которых содержат информацию о, соответственно, дочерних, родительских или связанных с текущим объектом объектах. В формах указанных типов, в отличие от формы типа «Атрибуты», не предусмотрено редактирование данных, а также использование кнопок.

Формы для объектов могут содержать:

- колонки – поля для отображения атрибутивной информации об объекте;
- текстовые поля;
- вычисляемые поля – содержат выражения, в которых могут использоваться колонки (в том числе и другие вычисляемые поля), математические и логические операторы, различные функции;
- линии и рамки;
- кнопки.

К элементам интерфейса также можно отнести фильтрацию дерева проекта и использование представлений проекта. Представления проекта – специальная настройка, изменяющая вид отображаемого дерева проектов. После выполнения необходимых настроек, выбранный проект может быть открыт в требуемом представлении.

10.7 Выполнение настроек интерфейса пользователей. Учебное задание

Создайте один или несколько профилей пользователей, ознакомьтесь с возможными настройками. Определите в качестве используемой по умолчанию связь «Дерево проектов».

Произвольным образом настройте одно или несколько Главных меню пользователей, подключите их к профилю, определите меню, отображаемое при старте системы. Определите порядок формирования названий объектов в дереве для различных типов объектов.

Чтобы проверить работу сделанных настроек, включите Администратора (так как других пользователей в учебной настройке, скорее всего, еще не создано) в настроенный профиль и перезапустите программу. Убедитесь, что сделанные настройки работают.

Создайте формы типа «Атрибуты» для всех используемых типов объектов. Настройка форм может быть произвольной, в учебных целях, для наглядности, все допустимые для данного типа объекта атрибуты должны присутствовать в соответствующей форме.

Подключите созданные формы к типам объектов для используемого профиля (см. п. 10.3.5), для удобства настройки соответствия одинаковых форм различными типам объектов используйте группировку типов объектов (Рисунок 102).

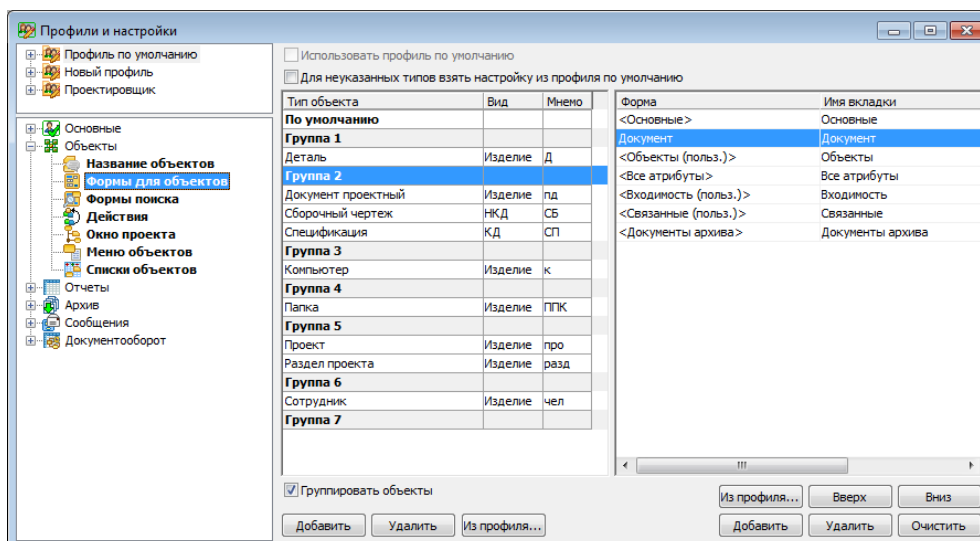


Рисунок 102 Пример настройки соответствия форм для типов объектов

В учебных целях, predetermined вкладки для используемых типов объектов также сохраните – для этого стандартные вкладки придется добавить к добавленным в настраиваемый профиль типам объектов или их группам. В приведенном примере (Рисунок 102) для объектов документов (Группа 4) используется одинаковый набор форм, аналогично – для типов «Проект» и «Раздел» (Группа 3). Сохраните настройки, **перезапустите программу**, ознакомьтесь с работой атрибутивных форм.

Для ознакомления с использованием представлений проектов выполните настройку представления согласно рисунку (см. рисунок 103).

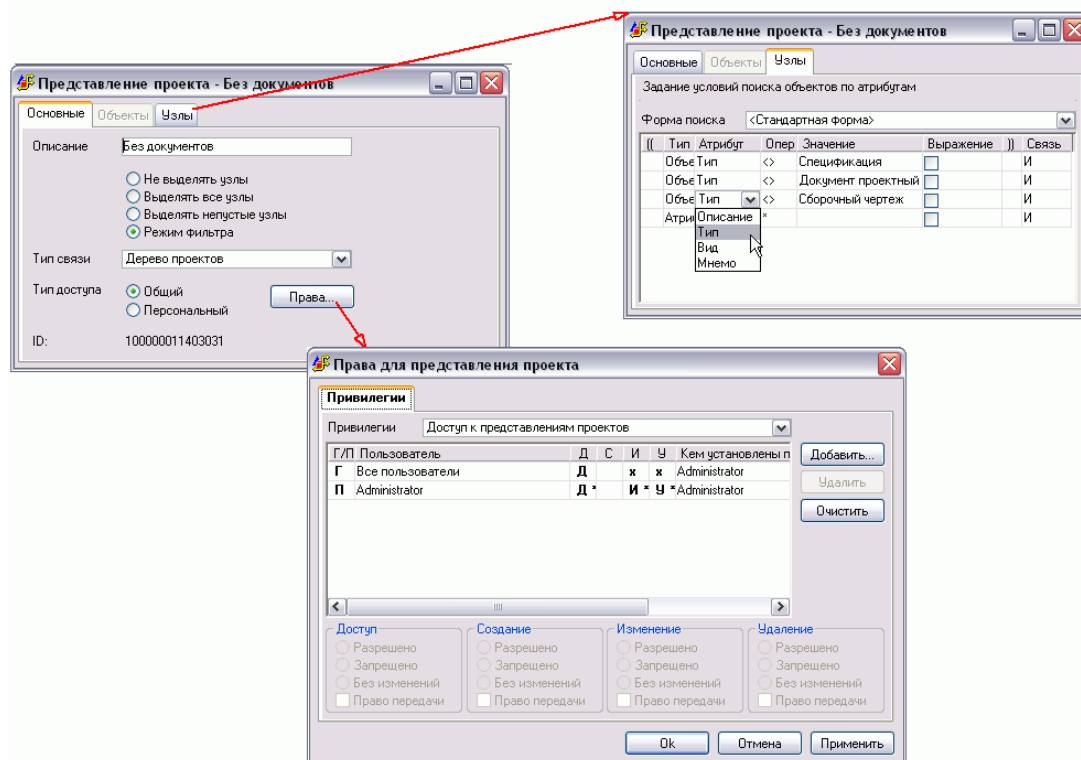


Рисунок 103 Настройка представления проекта. Пример

Откройте в созданном представлении проект, убедитесь, что объекты документов в дереве связей не отображаются.

10.8 Настройки интерфейса. Вопросы для самопроверки

1. Как настроить Главное меню пользователя, чтобы пользователь, не являющийся Администратором, мог выбором соответствующего пункта меню:
 - выполнить конкретное действие;
 - выполнить соответствующий отчет;
 - создать новое действие;
2. Как в форме типа «Атрибуты» можно запретить пользователю изменять значения атрибута?
3. Как задать в поисковой форме условия предварительного отбора объектов, как привязать табличную форму к типу связи, как управлять свойствами объектов в форме типа «Атрибуты» в зависимости от текущего типа связи, текущего режима формы (просмотр, редактирование)?
4. Как обеспечить пользователю возможность использования:
 - настроенного администратором меню;
 - формы для типа объекта;
 - поисковой формы?
5. Как определить пользователю тип подчиненной связи, используемой по умолчанию?

11 Создание пользователей, контактов, адресных книг

Создание пользователей осуществляется средствами Lotsia PDM PLUS, использования при этом средств СУБД не требуется. Пользователь создается непосредственно как пользователь БД, дополнительных прав в БД в процессе работы с Lotsia PDM PLUS пользователь не получает. Таким образом, пользователь, используя прямой запрос к БД, в принципе не сможет получить больше информации, чем ему доступно через интерфейс Lotsia PDM PLUS, в соответствии с правами доступа, определенными средствами Lotsia PDM PLUS.

Пользователь может включаться в произвольное количество групп пользователей. Группы пользователей создаются в требуемом количестве, группа «Все пользователи» – предопределенная, имеется всегда. В данную группу автоматически включаются все пользователи, удалить из данной группы пользователя нельзя. Группа не может входить в другую группу.

Для работы со свободными сообщениями используются не сами пользователи, а сопоставляемые пользователям контакты. Контакты хранятся в Глобальных адресных книгах (АК). На использование глобальных адресных книг, а также отдельных контактов, пользователям или группам пользователей задаются права, и в соответствии с имеющимися правами на контакты и глобальные адресные книги пользователь может настроить собственную адресную книгу. Использование контактов и АК дает возможность ограничивать переписку пользователей между собой, если это требуется. Например, если возможность отправлять свободные сообщения генеральному директору Предприятия должен иметь ограниченный круг сотрудников, то только соответствующие пользователи должны иметь права на контакт, сопоставленный генеральному директору.

Для созданных пользователей и групп могут задаваться и наследоваться права на объекты и документы архива, а также права на сохраненные запросы, представления проектов, карты работ.

При регистрации пользователей и групп могут быть заданы приоритеты, привилегии, доступные библиотеки и другое. Задание административных привилегий (вкладка «Привилегии» окна свойств группы или пользователя) должно сопровождаться предоставлением пользователю соответствующих пунктов Главного меню из стандартной папки «Администрирование» (см. п. 10.1) – если это требуется для реализации соответствующей привилегии.

Приоритеты для пользователей и групп определяют, кто кому может устанавливать права доступа (см. п. 16). Чтобы пользователь мог установить права другому пользователю или группе пользователей, необходимо, чтобы у данного пользователя (или группы, куда он входит), был установлен приоритет над соответствующим пользователем (группой). По умолчанию, для группы «Все пользователи» установлен приоритет над той же группой – то есть, любой пользователь может устанавливать права любому пользователю (и эта настройка не позволяет управлять приоритетами отдельных групп и пользователей – (Рисунок 104)).

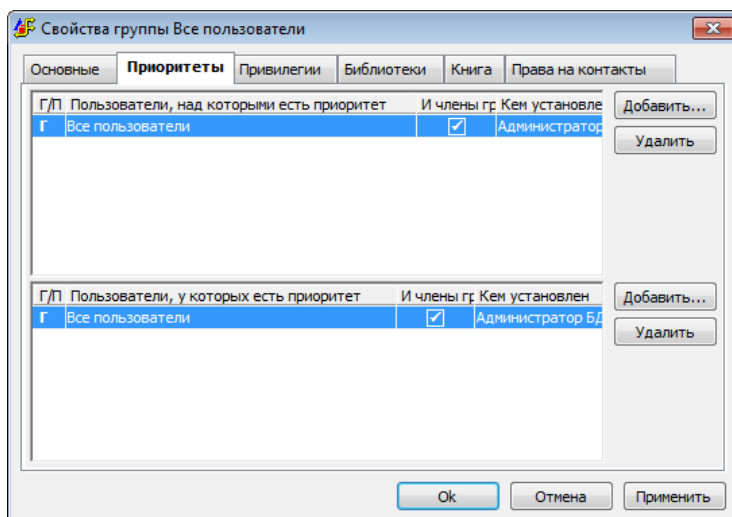


Рисунок 104. Установленные по умолчанию приоритеты для группы «Все пользователи»

Если для решения ваших задач такое положение не подходит, данный приоритет для группы «Все пользователи» необходимо удалить, после чего будут работать приоритеты, установленные для других групп и пользователей.

Создание пользователей, групп и АК может выполняться по-разному и в различной последовательности. Полная информация содержится в документации по системе Lotsia PDM PLUS, здесь кратко рассмотрим наименее трудоемкий для администратора системы способ регистрации пользователей. Рекомендуется выполнять регистрацию пользователей в следующей последовательности:

- Создание групп пользователей
- Создание Глобальных адресных книг и сопоставление их группам пользователей (количество групп и Глобальных адресных книг могут не совпадать, но каждой АК рекомендуется сопоставить группу пользователей)
- Задание прав на Глобальные адресные книги для групп пользователей – выполняется на вкладке «Права на контакты» окна свойств группы. Если возможность переписки ограничивать не требуется – задайте права доступа к адресным книгам для предопределенной группы «Все пользователи».
- Создание пользователей с включением их в требуемые группы.

При указанном подходе Администратор при регистрации новых пользователей выполняет минимум операций: контакт пользователя автоматически создается и включается в сопоставленные группам АК, а права на АК («Права на контакты», то есть, возможность вести переписку с определенным кругом пользователей) зарегистрированный пользователь получает от групп, в которые он входит.

Для удобства администрирования, рекомендуется в максимально возможной степени использовать группы пользователей при задании прав на контакты, привилегий, приоритетов, библиотек. При этом свойства группы удобно передать пользователю путем включения его в соответствующую группу. То же самое относится и к назначению прав доступа (см. п. 16) – правами доступа пользователя удобно управлять, включая его в соответствующие группы, для которых требуемые права заранее определены.

Для групп могут быть заданы параметры, по которым может осуществляться отбор пользователей.

Настройка пользователей и групп осуществляется с использованием окна «Справочник пользователей», для открытия которого предназначен пункт Главного меню «Пользователи», расположенный в папке «Администрирование» полного меню Администратора.

Для создания групп пользователей в левой части окна «Справочник пользователей» следует выделить пункт «Пользователи и группы», выбрать требуемый пункт контекстного меню (см. рисунок 105).

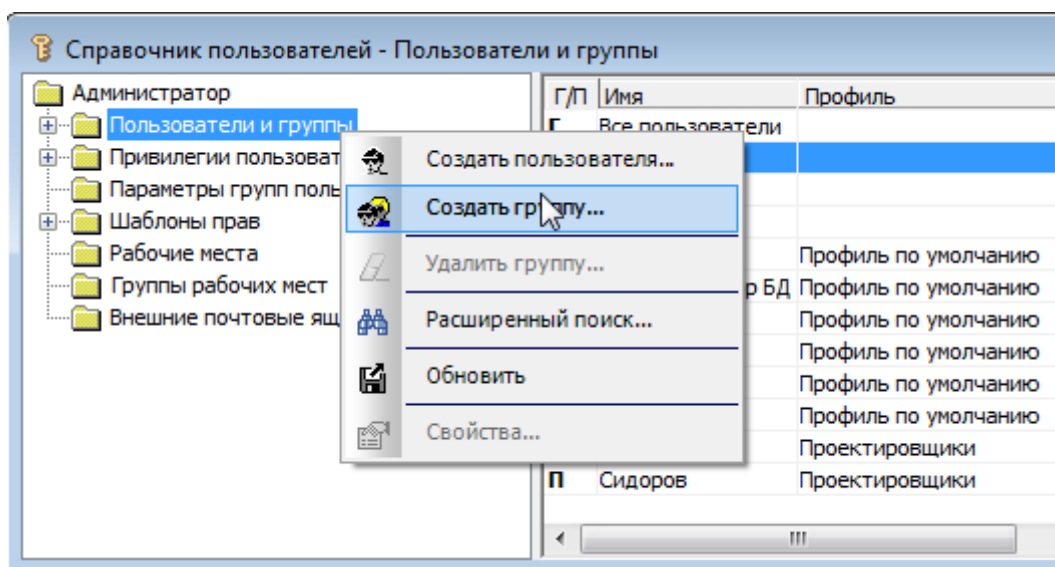


Рисунок 105 Создание пользователей и групп

Для настройки глобальных АК используется пункт главного меню «Настройка адресных книг». Для создания новой книги используется пункт «Создать адресную книгу» контекстного меню правой части окна настройки адресных книг. Количество адресных книг может соответствовать количеству рабочих групп, члены которых должны участвовать в системе документооборота. Сопоставляемая адресной книге группа указывается в поле «Группа пользователей» (см. рисунок 106).

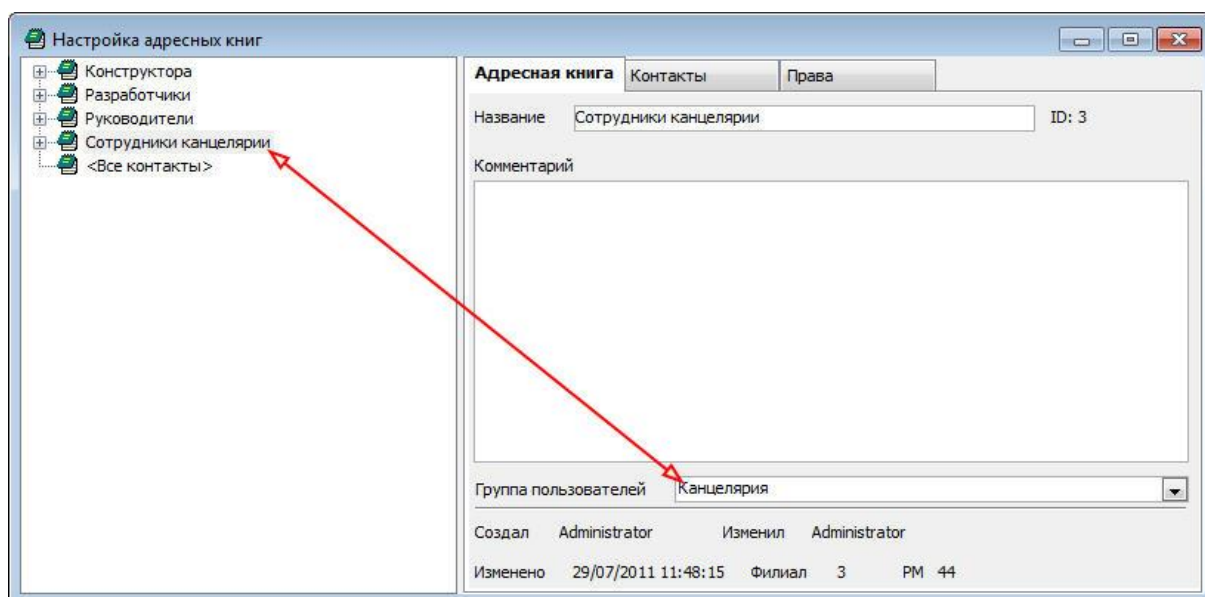


Рисунок 106 Настройка АК

При создании пользователей их контакты будут создаваться автоматически, при включении пользователей в группы, сопоставленные адресным книгам, контакты пользователей будут автоматически включены в соответствующие адресные книги.

Чтобы пользователь получил доступ к контактам в адресной книге, ему необходимы права на отдельные контакты или на соответствующую АК целиком. Права удобнее (и рекомендуется) задавать не для отдельных пользователей, а для групп пользователей. Права на контакты и АК задаются в справочнике пользователей, в свойствах пользователей или групп, на вкладке «Права на контакты» (Рисунок 107), либо, для сопоставленных группам АК – в окне настройки АК, на вкладке «Права» – для группы, сопоставленной выделенной АК. Отметим, что и в последнем случае права на контакты даются именно группе пользователей, сопоставленных данной АК, не зависимо от того, какие контакты фактически имеются в данной АК.

Для обеспечения возможности импорта документов в архив, для пользователей или групп пользователей следует определить список доступных библиотек. Настройка и использование архива будут рассмотрены далее, см. раздел 19.

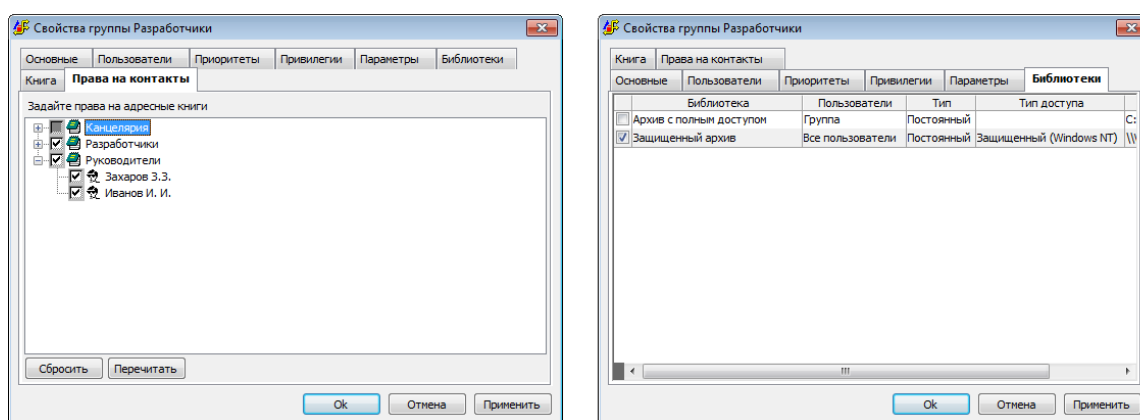


Рисунок 107 Задание прав на контакты и АК и определение списка доступных библиотек для группы пользователей

При создании пользователя, соответствующий контакт создается и сопоставляется пользователю автоматически. Для пользователя определяется профиль: это можно сделать как в свойствах данного пользователя (Рисунок 108), так и при настройке соответствующего профиля (см. п. 10.3).

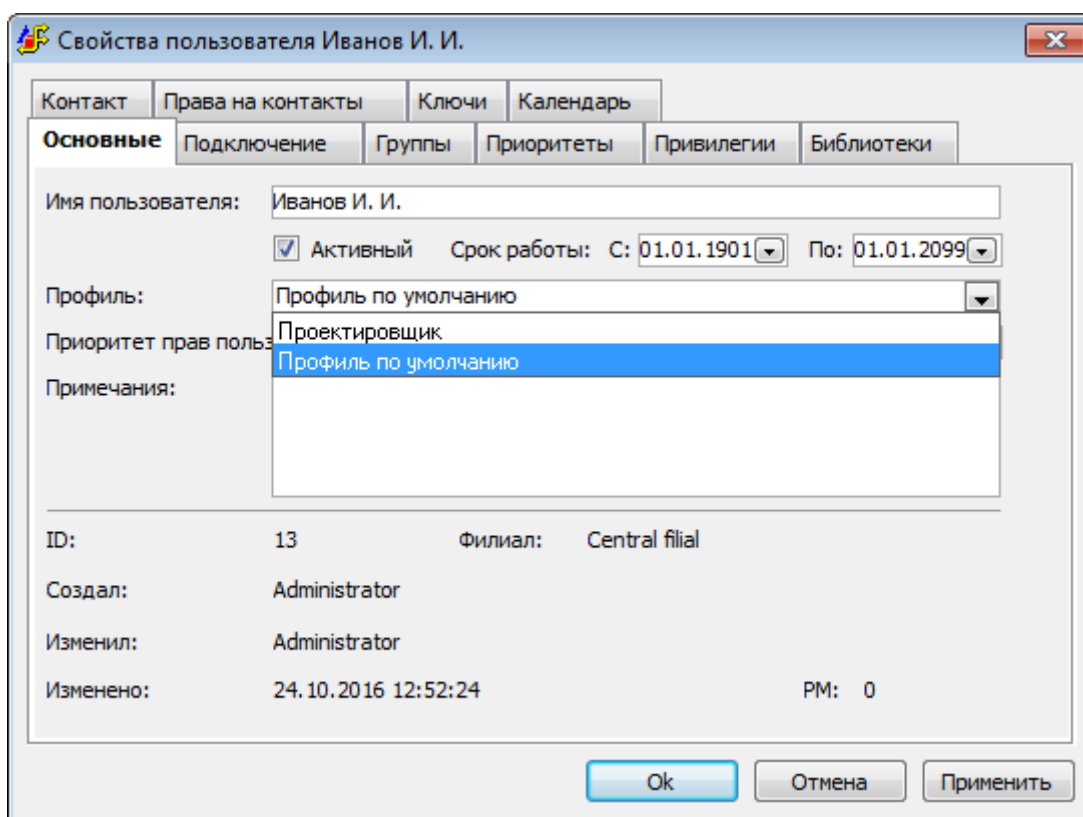


Рисунок 108 Указание профиля пользователя

При вышеописанном способе формирования Глобальных АК, контакт пользователя автоматически попадает в АК, сопоставленные группам. Пользователь автоматически получает от групп права на соответствующие контакты и АК.

В большинстве случаев такой настройки АК достаточно, во всяком случае, к этому следует стремиться при определении политики ведения Глобальных АК и настройке групп. При удалении пользователя из группы, сопоставленной АК, происходит автоматический отбор прав на данную АК у пользователя. Контакт пользователя, при этом остается в АК, сопоставленной группе, из которой удален данный пользователь. Для синхронизации вхождений контакта пользователя в АК, сопоставленные группам, и вхождений пользователя в такие группы, следует нажать кнопку «Из групп». Указанная настройка осуществляется в окне свойств пользователя, на вкладке «Контакт» (Рисунок 109).

Контакт

Сопоставленный контакт
Иванов Выбрать Отменить

Поля контакта

Поле	Значение
Ф.И.О.	Иванов
E-Mail	
Телефон	

Список адресных книг

Адресные книги, в которые входит пользователь

- Разработчики
- Сотрудники канцелярии

Добавить Удалить Из групп Пересчитать

Рисунок 109 Задание вхождения контакта пользователя в различные АК

Программа позволяет так же создавать Глобальные АК, не сопоставляя их группам пользователей. Добавление контактов в такие книги можно осуществлять при настройке свойств пользователя (на вкладке «Книги», кнопка «Добавить»), или при настройке АК (пункт «Добавить контакт» контекстного меню окна настройки АК). При этом необходимо дать права на данную АК пользователям или группам.

Для сопоставления адресной книге непустой группы пользователей и обеспечения автоматического попадания контактов данных пользователей в указанную АК, следует после сопоставления АК группе пользователей и сохранения настройки, при текущем окне настройки глобальных АК использовать пункт верхнего меню «Адресная книга» → «Синхронизация с группами» (Рисунок 110).

Адресная книга Настройки Окно Помощь

- Генерировать контакты...
- Синхронизация с группами
- Обновить поля ЛАК...

Настройка адресных книг

Канцелярия
Разработчики
Руководители
<Все контакты>

Адресная книга Контакты Права

Название <Все контакты>

Комментарий
Список всех контактов для всех адресных книг

Группа пользователей <Все пользователи>

Создал

Изменено 00/00/0000 00:00:00 Филиал PM

Рисунок 110 Добавление контактов пользователей в адресные книги из сопоставленных им групп

11.1 Резюме

Пользователи создаются непосредственно в Lotsia PDM PLUS, дополнительной настройки пользователей средствами СУБД не требуется.

Пользователи могут входить в одну или несколько групп пользователей.

Пользователю сопоставляется контакт (при создании нового пользователя контакт создается и сопоставляется пользователю автоматически).

Контакты используются для управления возможностью обмена сообщениями между пользователями БД – пользователь может для отправки сообщений использовать только те контакты, на которые у него есть права.

Контакты хранятся в Глобальных адресных книгах, права (на контакты) могут назначаться как для отдельных контактов, так и для Глобальной адресной книги, отдельным пользователям либо группам пользователей.

Глобальные Адресные книги могут сопоставляться группам пользователей и синхронизироваться с ними различными способами.

11.2 Создание пользователей и выполнение необходимых настроек. Учебное задание

Создайте в учебной настройке группы пользователей, например: «Руководство», «Канцелярия», «Разработчики» или другие.

Создайте Глобальные адресные книги и сопоставьте их созданным группам.

Дайте группам права на адресные книги.

Добавьте в группы пользователей, если имеются настроенные профили – установите пользователям различные профили.

Зарегистрируйтесь в программе от имени различных пользователей, убедитесь в правильности сделанных настроек.

11.3 Создание пользователей и выполнение необходимых настроек. Вопросы для самопроверки

- 12 Может ли пользователь входить в несколько групп?
- 13 Сколько контактов может быть сопоставлено одному пользователю Lotsia PDM PLUS?
- 14 Может ли пользователь входить в несколько профилей?
- 15 Как дать пользователю или группе пользователей возможность отправлять сообщения конкретному пользователю или группе?

16 Права доступа

16.1 Права доступа. Общая информация

Права доступа могут устанавливаться на различные сущности Lotsia PDM PLUS – объекты, документы архива, сохраненные запросы, представления проектов, шаблоны и карты работ. Права доступа могут устанавливаться как для отдельных пользователей, так и для групп пользователей. С точки зрения удобства администрирования, второй путь предпочтительнее – для назначения отдельному пользователю прав, его, в большинстве случаев, достаточно включить в соответствующие группы.

Важный момент – для обеспечения возможности установки прав доступа указанные сущности должны быть «защищенными», а не «общедоступными». Для представлений проекта – чтобы обеспечить возможность назначения прав, представление необходимо сделать «Общим», для сохраненных запросов – достаточно сохранить запрос. Для документов архива данное свойство определяется настройкой библиотеки, в которую помещен данный документ, для остальных сущностей – свойствами самой сущности. Подробнее данное положение рассмотрим при назначении конкретных прав.

Для различных случаев, при назначении прав может применяться единообразное интерфейсное окно (Рисунок 111), где можно выбрать привилегию (если предусмотрено более одной привилегии для данного случая назначения прав). Открытие данного окна осуществляется по кнопке «Права» в окне свойств соответствующей сущности. Указанная кнопка активна только в случае, когда сущность является защищенной. Для выбранной привилегии определить права на доступ, создание, изменение и удаление соответствующей сущности. Для тех сущностей, для которых отдельные права неприменимы (например – «Удаление» для типов объектов, или «Создание» для имеющегося сохраненного запроса (Рисунок 111)), соответствующие колонки окна назначения прав будут недоступны.

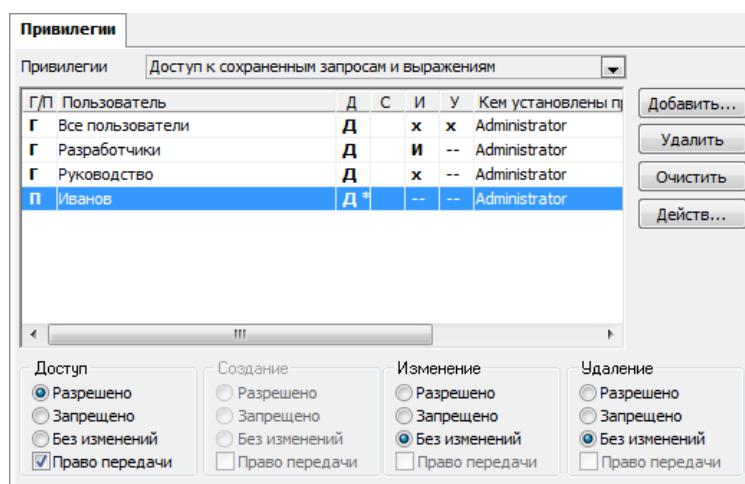


Рисунок 111 Окно назначения прав доступа. Пример

Если разрешено – отображается, соответственно, для доступа – «Д», для создания – «С», для изменения «И», для удаления «У». Если запрещено – в соответствующей колонке отображается «х». Если имеется право передачи (то есть, возможность устанавливать права другим пользователям, в рамках имеющихся полномочий) – отображается, соответственно, «Д*», «С*», «И*», «У*» – то есть, добавляется символ «*». Если для данной колонки установлено «Без изменений» (отображается «--») – это равносильно отсутствию значения в данной строке данной колонки.

Определение действующих прав доступа для конкретного пользователя осуществляется отдельно по каждой привилегии и каждой колонке («Доступ», «Создание», «Изменение», «Удаление») следующим образом:

При определении действующих прав для пользователя вначале проверяются персональные права, установленные для данного пользователя (не групп). Отметим, что в различных случаях права данного пользователя могут быть не назначены, либо назначены один раз или более – то есть, строка для пользователя может отображаться в списке прав и более одного раза (Рисунок 112).

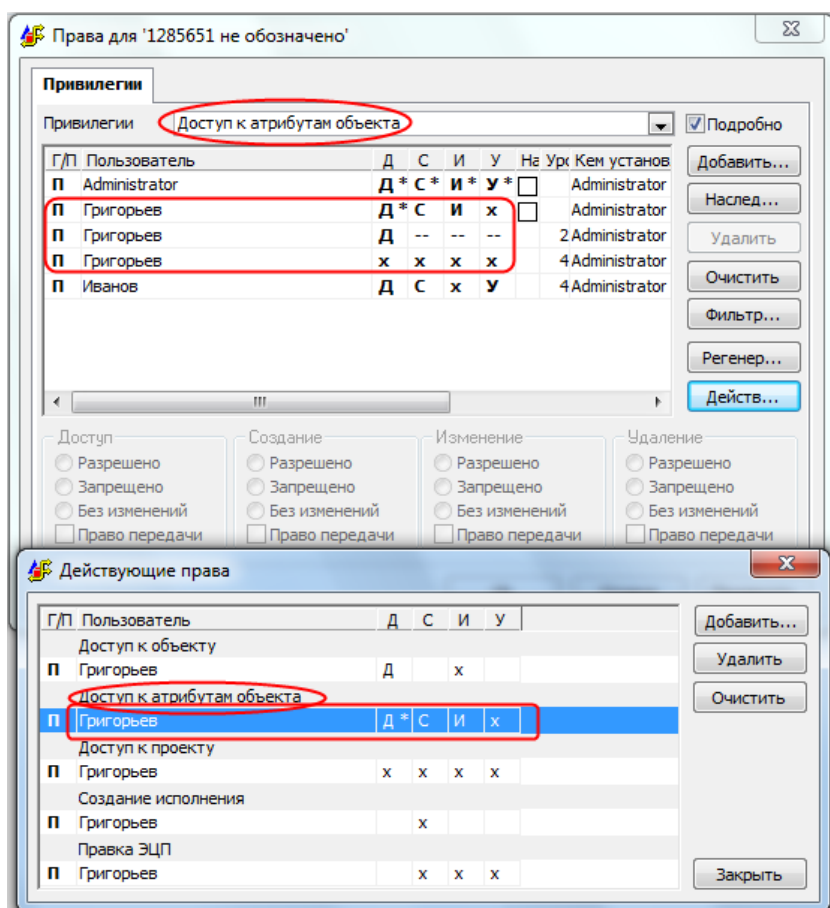


Рисунок 112 Вычисление действующих прав пользователя. Пример

Если строки для данного пользователя (для данной привилегии) имеются, и в данной колонке содержится значение, отличное от «Без изменений» («--»), дальнейшая проверка по данной колонке прекращается, принимается указанное в данной колонке значение. Если таких значений несколько – приоритет у разрешающих прав (то есть, если встречается символ «*» – будет разрешено с правом передачи, если в колонке есть хотя бы одно «Разрешено» – будет разрешено). Если в колонке указано только «Запрещено» («х»), независимо от наличия «Без изменений» («--») – по данной колонке будет запрещено.

Если персональных строк для данного пользователя (по данной привилегии) нет, либо колонка содержит только значения «Без изменений» («--»), проверка по данной колонке продолжается. Проверяются значения для всех групп, куда входит данный пользователь, кроме группы «Все пользователи». Проверка (вычисление действующих прав) выполняется аналогично проверке для пользователя. Если хотя бы одной из групп, куда входит данный пользователь, установлено значение, отличное от «Без изменений» («--»), проверка по данной колонке прекращается и принимается установленное значение. Если таких значений несколько – приоритет у разрешающих прав: если встречается

разрешение с правом передачи – пользователю будет разрешено с правом передачи, если хотя бы один раз установлено «Разрешено» – будет разрешено, если указано только «Запрещено» («х»), независимо от наличия «Без изменений» («--») – будет запрещено.

Если отсутствуют строки для данного пользователя и групп, куда он входит, либо в данной колонке в этих строках содержится только «Без изменений» – аналогично проверяется группа «Все пользователи», приоритет у разрешающих прав. Если для группы «Все пользователи» строк нет, либо в данной колонке имеются только значения «Без изменений» («--») и «Запрещено» («х») – будет запрещено. Если есть значение «Разрешено с правом передачи» – оно будет использовано, если такого значения нет, а есть значение «Разрешено» – будет разрешено.

Остановимся более подробно на управлении доступом к информационным объектам и документам архива.

16.2 Доступ к информационным объектам, проектам, атрибутам.

В Lotsia PDM PLUS имеются следующие возможности по установке прав доступа к объектам:

На уровне структуры данных – путем задания прав доступа к типам объектов и атрибутам;

На уровне экземпляров объектов и проектов – для отдельных объектов и проектов можно управлять правами доступа

16.2.1 Установка прав на уровне структуры БД

Можно сделать тип объекта защищенным, тогда доступ ко всем объектам данного типа будет ограничен, при этом сами экземпляры объектов данного типа не станут защищенными, защита на уровне экземпляров объектов может осуществляться дополнительно и независимо от защиты на уровне структуры БД.

Увеличить уровень доступа к объектам защищенных типов установкой прав на экземпляры объектов невозможно.

Чтобы установить права доступа на тип объекта, тип объекта необходимо вначале сделать защищенным: Главное меню → Администрирование → Структура данных → Типы объектов → выбрать требуемый тип из списка → на вкладке «Основные» в поле «Доступ» указать «Защищенный» (Рисунок 113).

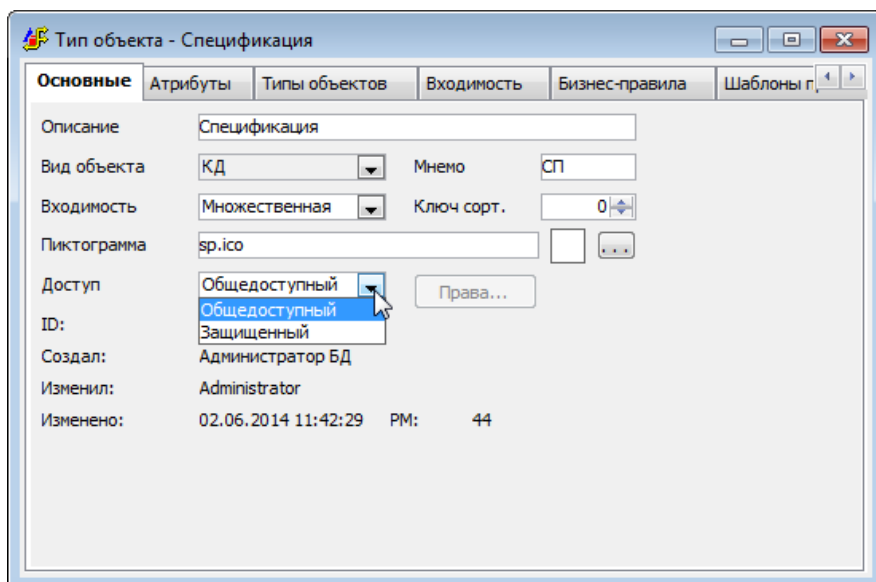


Рисунок 113 Изменение доступа с общедоступного на защищенный (для типа объекта). Пример

После сохранения изменений кнопка «Права...» станет активной. По нажатию указанной кнопки будет открыто окно для установки прав (Рисунок 114).

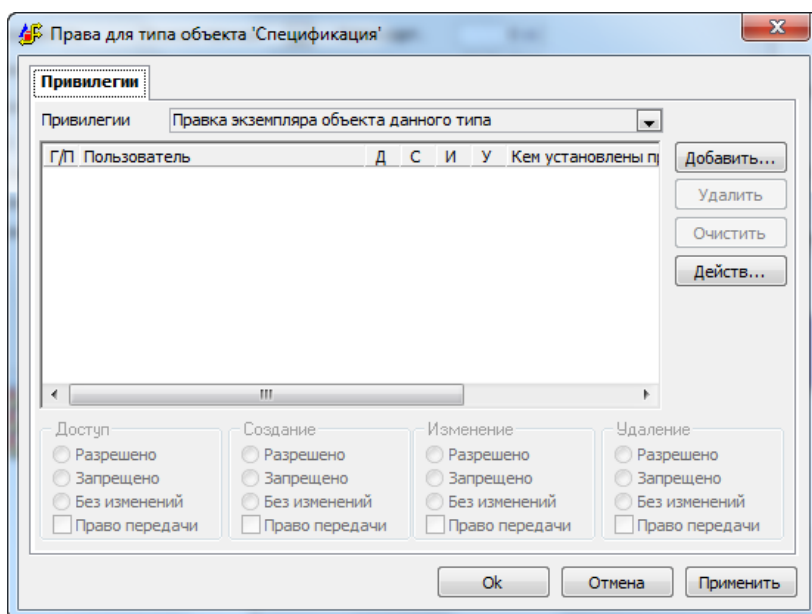


Рисунок 114 Окно прав доступа для типа объекта. Пример

Привилегия в данном случае единственная («Правка экземпляра объекта данного типа»), её выбор не требуется.

Для добавления в список для назначения прав пользователей и/или групп пользователей используется кнопка «Добавить...». Выбранные по нажатию данной кнопки пользователи и/или группы добавляются в список. Для выделенных в списке строк групп и/или пользователей переключателями в нижней части окна устанавливаются требуемые права доступа (Рисунок 115).

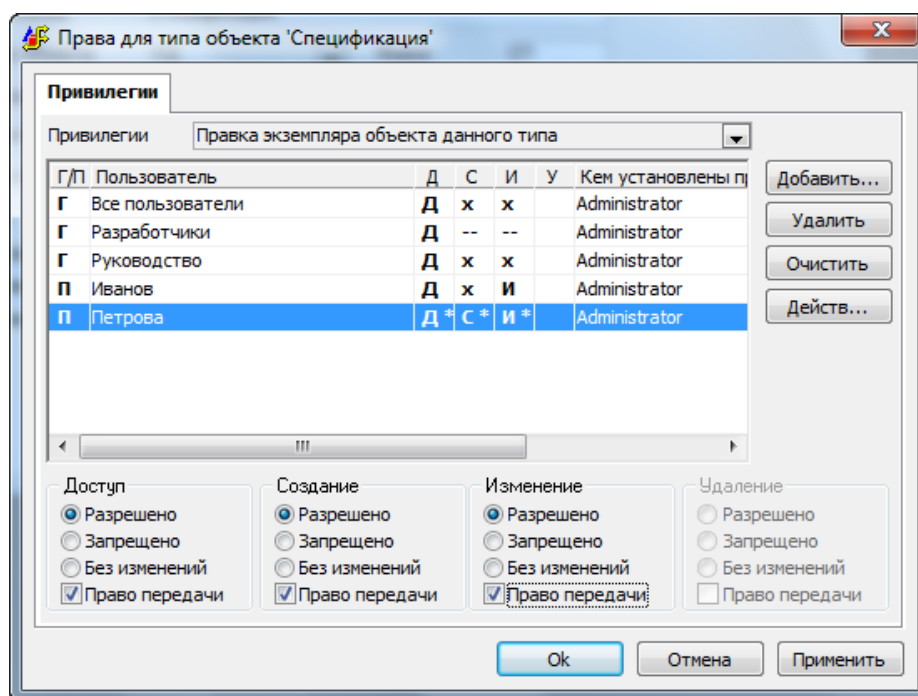


Рисунок 115 Назначены права доступа для типа объекта. Пример

Отметим, что без права на доступ, права на создание и изменение не действуют, колонка «Удаление» не активна, так как удаление экземпляров объектов из БД пользователями (не Администратором) в системе не предусмотрено.

Установка прав доступа на отдельные атрибуты технически отличается от установки прав на типы объектов наличием двух привилегий: «Правка атрибутов у экземпляра объекта» и «Правка списка значений атрибута». Наличие полных прав по первой привилегии не позволит ввести значение данного атрибута, отсутствующее в БД, то есть, для создания новых значений нужны права по второй привилегии.

Чтобы устанавливать права на данный атрибут, необходимо сделать его защищенным: Главное меню → Администрирование → Структура данных → Атрибуты → выбрать требуемый атрибут из списка → на вкладке «Основные» в поле «Доступ» указать «Защищенный». Далее выбрать требуемую привилегию и по нажатию «Добавить...» (Рисунок 116) выбрать пользователей и/или группы, для которых установить права.

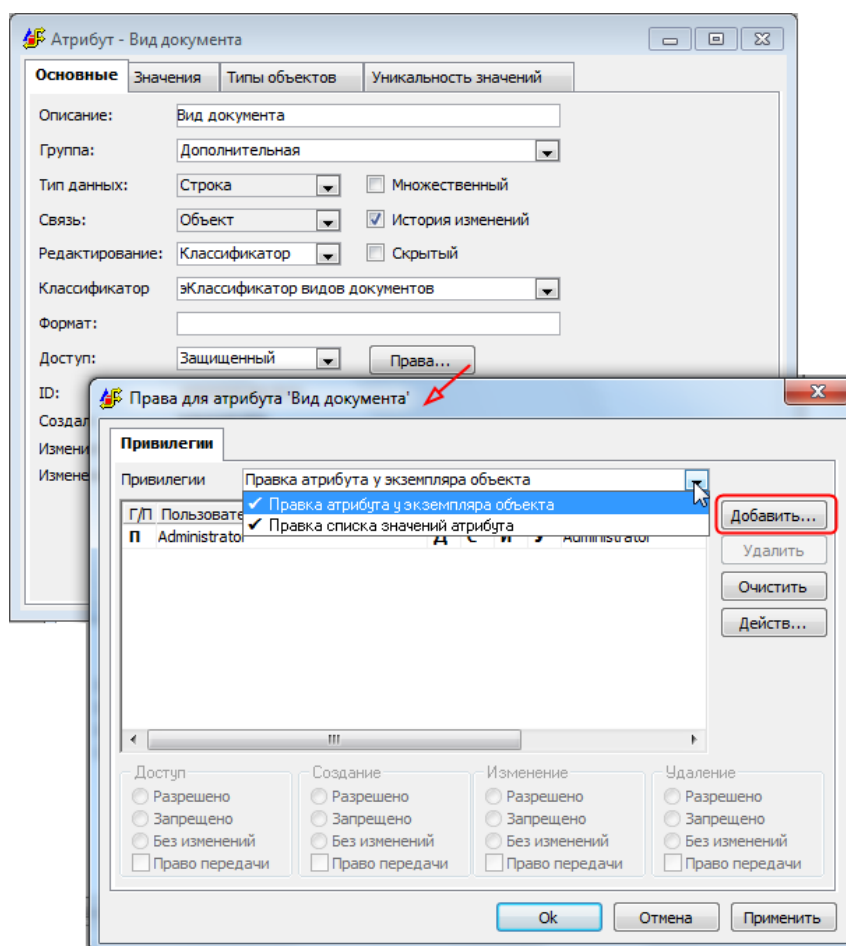


Рисунок 116 Права доступа для атрибута. Пример

Отметим, что при установке прав на отдельные атрибуты, экземпляры объектов с такими атрибутами автоматически защищенными не становятся – сами объекты могут быть как общедоступными, так и защищенными.

Установленные для данного атрибута права будут одинаковыми для всех объектов (экземпляров объектов), у которых имеется данный атрибут. На уровне экземпляров объектов установить отдельно защиту для различных атрибутов невозможно.

Защита объектов на уровне структуры БД используется больше как вспомогательное средство, ввиду недостаточной гибкости данного инструмента – защита действует единообразно для всех объектов данного типа или для данного атрибута у всех объектов.

На практике, как правило, требуется управлять (изменять) правами для отдельных объектов и проектов (см. п. 16.2.2)

16.2.2 Права на уровне экземпляров объектов

Права на уровне экземпляров объектов могут быть:

«Непосредственные» – устанавливаются для данного объекта, действуют только на данный объект, не затрагивая другие объекты;

«Наследуемые» – устанавливаются на объект верхнего уровня и автоматически передаются («наследуются») на все входящие (и добавляемые) в него защищенные объекты, независимо от уровня вложенности и типа подчиненной связи (связей). По горизонтальным связям наследования прав не происходит;

«Унаследованные» – полученные путем наследования. Устанавливаются при установке входящей подчиненной связи любого типа. Для каждого объекта хранится список всех унаследованных прав от всех объектов, даже если эти права одинаковые. Такое хранение необходимо для возможности обработки разрыва связи с объектами, а также для возможности управления этими правами. При удалении входящей подчиненной связи, унаследованные по этой связи права более на данный объект, не действуют. То же самое справедливо для любого экземпляра подчиненной связи верхнего уровня (то есть, вверх по дереву от данного объекта).

Чтобы установить права доступа к объекту, Объект должен быть защищенным. Сделать объект защищенным можно различными способами:

В контекстном меню объекта выбрать «Свойства...» → на вкладке «Основные» переключатель «Доступ» установить в положение «Защищенный» и сохранить изменения;

Если объект создается «в дереве», при добавлении создаваемого объекта в защищенный объект, создаваемый объект автоматически становится защищенным (если объект, в который добавляется новый объект общедоступный – создаваемый объект также будет общедоступным);

При использовании действий (см. п. 17) объект может быть сделан защищенным при использовании соответствующих функций.

Сделать объект защищенным может либо его создатель (если в «Общих параметрах БД», на вкладке «Права» включен флажок «Установка автором защиты объекта»), либо администратор.

Для установки прав доступа к объекту используется кнопка «Права...» (контекстное меню объекта → «Свойства»). В открывшемся окне (Рисунок 117) выбирается привилегия, для каждой привилегии определяется набор прав для пользователей и/или групп.

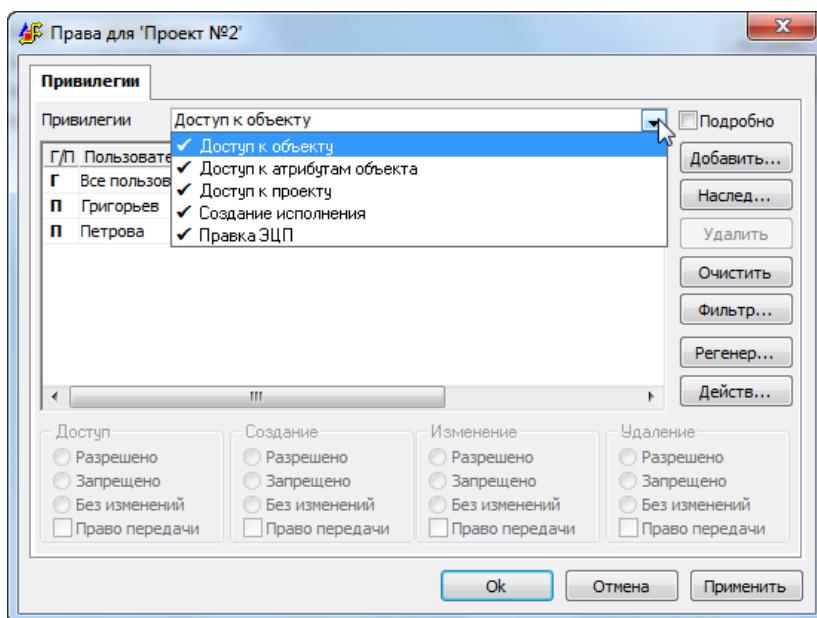


Рисунок 117 Выбор привилегии для установки прав доступа к объекту. Пример

Для отдельных привилегий права назначаются независимо, порядок назначения прав в зависимости от привилегии, с технической точки зрения, не различается (различным, в зависимости от выбранной привилегии, будет число активных колонок для назначения прав – «Доступ», «Создание», «Изменение», «Удаление»).

В то же время есть ряд особенностей, которые необходимо учитывать при назначении прав доступа:

- Если разрешения на доступ отсутствует, то разрешения на создание, изменение, удаление утрачивают силу;
- Если изменение объекта запрещено, то независимо от остальных установленных привилегий запрещено и добавление, удаление и изменение атрибутов данного объекта;
- Удаление обязательных атрибутов запрещено независимо от имеющихся прав доступа;
- Добавление атрибутов (в том числе обязательных при создании объекта) и их удаление, а также документа в архив и удаление документа архива считаются изменением объекта;
- Добавление объекта к другому объекту (в том числе обязательного при создании объекта) и удаление объекта из другого объекта (то есть, удаление экземпляра подчиненной связи) считается изменением проекта.

Помимо указанного:

- Независимо от установленных прав, на Администратора ограничения не действуют – он всегда имеет полные права по всем привилегиям, независимо от того, что отображается в окне прав для объекта. Права Администратору уменьшить невозможно;
- Нельзя увеличить права самому себе, в том числе и путем наследования – если вы добавили объект в проект, вам будет унаследовано не больше прав, чем у вас было изначально. Естественно, невозможно установить (то есть, передать, и только при наличии права передачи) другим пользователям больше прав на данный объект, чем имеется у вас;
- У пользователя, назначающего (изменяющего) права, должно быть право передачи и приоритет (см. п.11) над тем, кому он устанавливает права;
- Если требуется обеспечить полное управление правами доступа к объектам, проектам и документам архива пользователем (не Администратором), это можно реализовать путем настройки соответствующих предопределённых процессов (см. п. 23.12), с использованием серверных действий для назначения прав доступа.
- Если в «Общих параметрах БД», на вкладке «Права» включен флажок «Учет первичной входимости объекта в проекте» (по умолчанию данный флажок отключен), на объект, не имеющий признака первичной входимости, наследуются только права на доступ, остальные права наследуются только по входящей подчиненной связи данного типа, которая была установлена первой (отдельно для каждого типа подчиненной связи). Запрещается удалять объекты из первичного дерева проекта, если данный объект используется в других проектах.

Для того чтобы назначить непосредственные права доступа, в окне назначения прав, для выбранной привилегии (Рисунок 117) используется кнопка «Добавить...». Для установки наследуемых прав применяется кнопка «Наслед...».

Чтобы просмотреть, откуда были унаследованы те или иные права на объект, включите флажок «Подробно» и дважды щелкните левой клавишей мыши по требуемой строке прав (Рисунок 118).

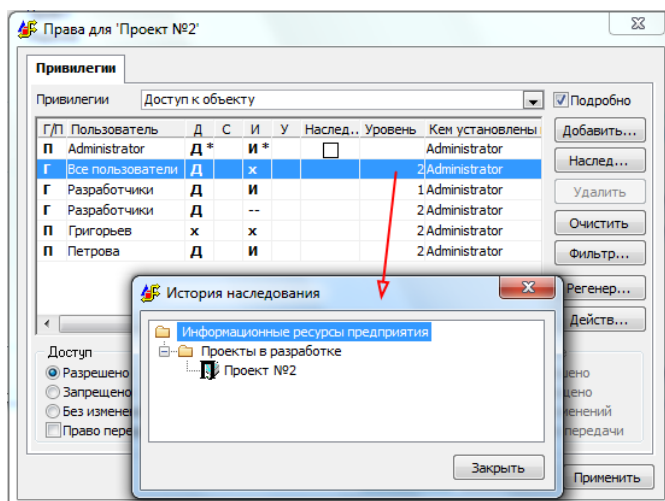


Рисунок 118 История наследования прав. Пример

Таким образом можно просматривать все унаследованные права на данный объект. Кроме того, для выделенной строки прав унаследованные права могут быть изменены (см. п. 16.2.3).

Часто требуется обеспечить, чтобы не все права автоматически распространялись путем наследования. Например, при включении в разрабатываемый проект ранее утвержденного документа, или документа исходных данных (соответствующего объекта с документами архива), вероятно, доступ (возможность чтения) к нему автоматически должны получить участники проекта, но возможность редактирования для такого документа возникать не должна.

В подобных случаях используются фильтры наследования прав. Фильтр устанавливается кнопкой «Фильтр» (либо при выполнении соответствующего действия). Фильтр устанавливается для конкретной колонки (колонок) данной привилегии, действует одинаково на всех пользователей. На объект, на котором установлен фильтр и входящие в него объекты, соответствующие наследуемые права не распространяются. При наследовании прав, в данных колонках устанавливаются значения «--» («Без изменений»).

Для обеспечения работоспособности, фильтр необходимо установить заблаговременно, то есть, до наследования прав – до включения объекта в проект, где будут наследоваться права, либо до установки наследуемых прав на объекте верхнего уровня.

16.2.3 Изменение унаследованных прав доступа

На протяжении жизненного цикла проекта, права доступа к отдельным объектам проекта и/или проекту целиком обычно требуется изменять. Изменения прав может быть выполнено переключателем в нижней части окна прав (Рисунок 119) (при включенном флажке «подробно»).

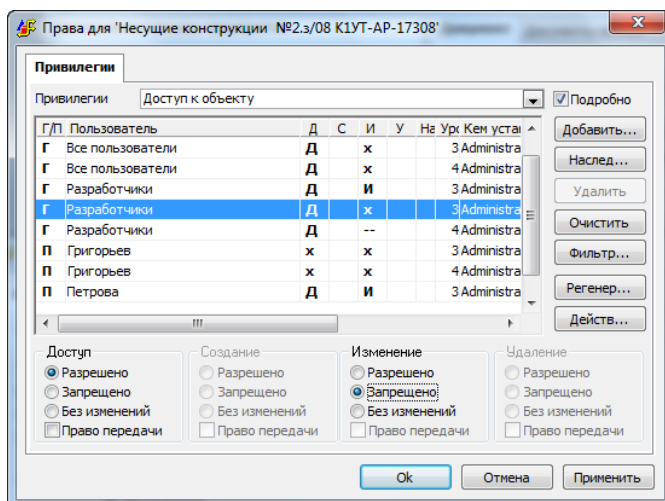


Рисунок 119. Изменение прав доступа к объекту

Изменены могут быть как непосредственные, так и наследуемые и унаследованные права.

При изменении наследуемых или унаследованных прав, в случае, если ранее унаследованные права уже менялись для объектов более низкого уровня, будет открыто окно запроса, содержащее сообщение о том, что права были изменены (Рисунок 120).

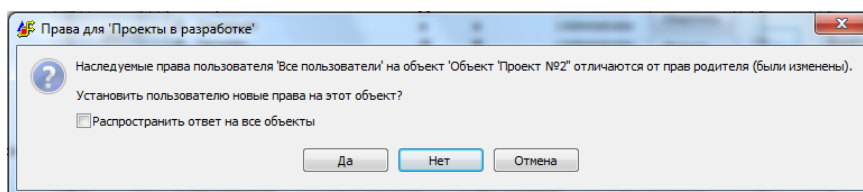


Рисунок 120. Запрос на изменение прав у дочернего объекта.

В этом случае существует два варианта продолжения обработки.

Первый вариант предполагает установку всем правам, унаследованным от изменяемого, нового значения. Для этого в окне запроса на изменения прав необходимо включить флажок «Распространить ответ на все объекты». Соответствующие права у дочерних объектов будут изменены без запросов пользователю.

Второй вариант – когда флажок «Распространить ответ на все объекты» выключен. В этом случае, при обработке каждого права, унаследованного от изменяемого, значение которого отличается от первоначального, программа будет спрашивать у пользователя, вносить ли изменения в это право или оставить его без изменений.

16.2.4 Регенерация прав доступа

Имеется возможность установки требуемых прав на проект, то есть, на все объекты, входящие в данный объект, независимо от имеющихся (установленных и измененных) прав на указанные объекты. Для этого, после установки требуемых наследуемых прав на корневом объекте данного проекта требуется нажать кнопку «Регенер...» (регенерация). Выполнение регенерации подразумевает принудительное распространение (обновление) прав доступа на текущем и всех дочерних объектах. Перед выполнением регенерации открывается окно запроса (Рисунок 121).

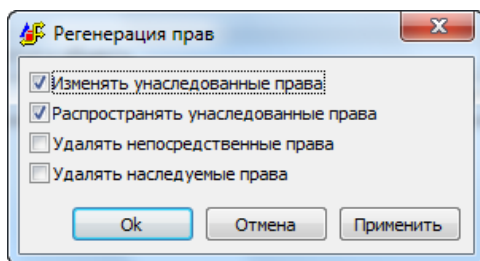


Рисунок 121. Регенерация прав

Можно указать следующие режимы регенерации:

- «Изменять унаследованные права» – установить права (унаследованные) по всем привилегиям по образцу наследуемых и/или унаследованных прав текущего объекта;
- «Распространять унаследованные права» – наследовать на дочерние объекты те права родительского объекта, которые не были ранее унаследованы;
- «Удалять непосредственные права» – удалять все непосредственные права для дочерних объектов;
- «Удалять наследуемые права» – удалять все наследуемые права для дочерних объектов.

Отметим, что наличие фильтров прав учитывается и при выполнении регенерации.

16.3 Шаблоны прав

Не трудно заметить, что на практике могут потребоваться достаточно сложные (громоздкие) настройки прав доступа. Реализация указанных настроек вручную для каждого экземпляра сущности может быть весьма трудоемкой. Одним из способов снизить указанную трудоемкость является использование шаблонов прав.

Шаблоны прав (по видам прав) настраиваются для пользователей или групп пользователей. Как только пользователь, для которого настроен шаблон прав, делает данную сущность защищенной – на нее автоматически устанавливается список значений прав согласно настроенному шаблону.

Изменения в шаблонах прав не влияют на созданные ранее соответствующие защищенные сущности.

Шаблоны прав могут настраиваться для:

- элементов структуры данных – типов объектов и атрибутов;
- экземпляров объектов – только непосредственные права;
- настроек транспортной утилиты;
- представлений проектов;
- сохраненных запросов и выражений

Шаблоны прав создаются и настраиваются в окне справочника пользователей: Главное меню → Администрирование → Пользователи (Рисунок 122).

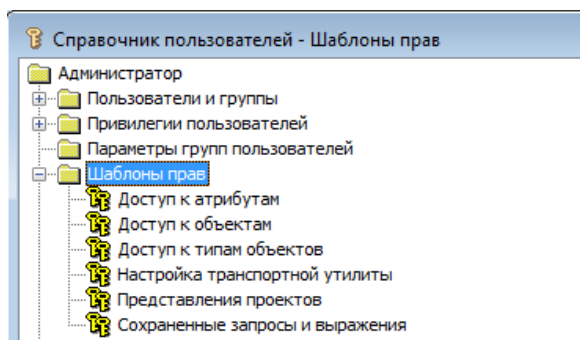


Рисунок 122. Справочник пользователей. Шаблоны прав

Шаблоны прав для различных сущностей создаются единообразно. Для добавления шаблона выделите требуемый пункт в папке «Шаблоны прав». В контекстном меню выберите «Создать...». В открывшемся окне укажите пользователя (или группу пользователей), для которого настраивается шаблон. В открывшемся окне (Рисунок 123) настраивайте шаблон в требуемом виде.

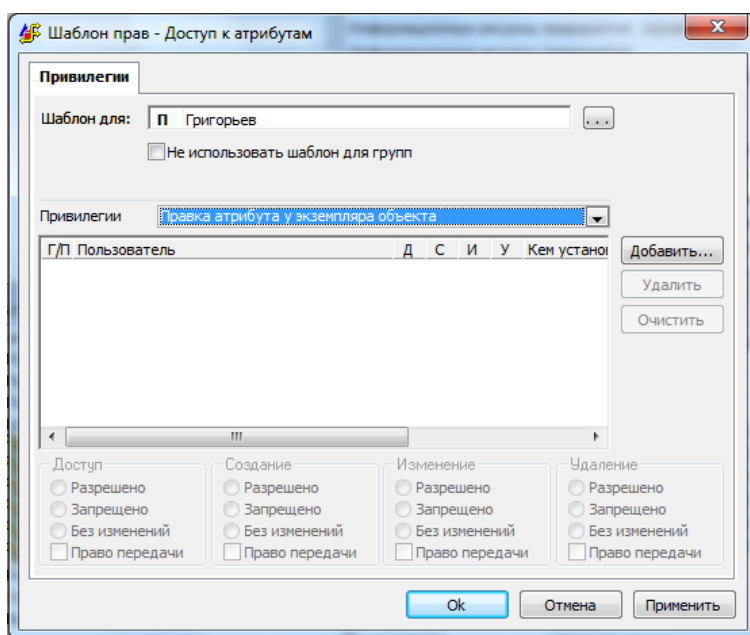


Рисунок 123. Настройка шаблона прав для выбранного пользователя

При настройке шаблона для пользователя имеется возможность отключить для данного пользователя использование шаблонов для групп, а при настройке шаблона для групп – отключить использование шаблона для группы «Все пользователи».

При выборе пользователей или групп при добавлении в настраиваемый шаблон прав (по кнопке «Добавить...»), имеется возможность, помимо обычных пользователей и групп, выбрать специального пользователя «[Creator Owner]». Во время применения такого шаблона для конкретного экземпляра защищенной сущности, данный пользователь подменяется на текущего пользователя – то есть, того, кто сделал данный экземпляр сущности защищенным. (Это актуально для шаблонов для групп пользователей).

Для экземпляров объектов шаблоны прав могут настраиваться также в окне настройки типа объекта, на вкладке «Шаблоны прав» (Рисунок 124).

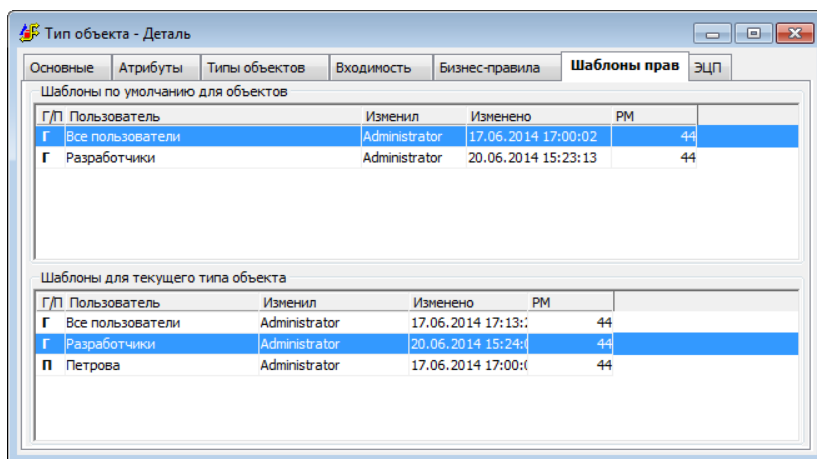


Рисунок 124. Шаблоны прав для типа объекта

В этом случае, шаблон может настраиваться так же как в окне справочника пользователей (то есть, «по умолчанию» – для объектов любых типов) либо только для объектов данного типа (Рисунок 124, нижняя часть окна). Настройка шаблона для текущего типа объекта отличается от настройки шаблона по умолчанию наличием возможности не использовать для данного типа объектов шаблон по умолчанию – имеется соответствующий флажок в окне настройки шаблона (Рисунок 125).

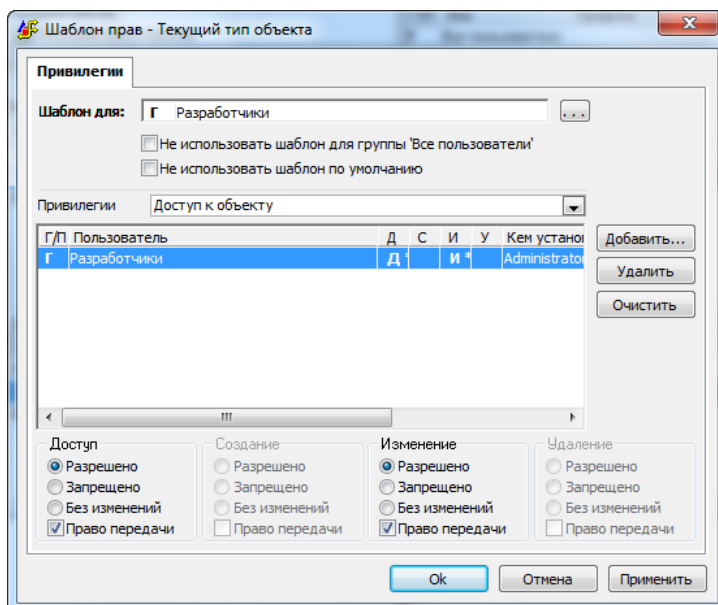


Рисунок 125. Настройка шаблона прав для текущего типа объекта

При наличии шаблонов прав, после того, как соответствующая сущность делается данным пользователем защищенной, права на эту сущность устанавливаются по следующей схеме:

- производится поиск всех шаблонов прав, установленных для текущего пользователя. Если в каком-либо шаблоне установлен флажок «Не использовать шаблон для групп», то найденный шаблон применяется, и дальнейший поиск шаблонов прекращается;
- производится поиск всех шаблонов прав, установленных для групп пользователей, в которые входит текущий пользователь. Если в каком-либо шаблоне установлен флажок «Не использовать шаблон для группы «Все пользователи»», то найденный шаблон применяется, и дальнейший поиск шаблонов прекращается;
- производится поиск всех шаблонов прав, установленных для группы «Все пользователи»;

- права из всех найденных шаблонов независимо (то есть, все строки из всех найденных шаблонов отдельно) копируются на защищаемую сущность и применяются суммарно.

После того, как пользователь сделал объект защищенным, права по шаблонам на данный объект применяются следующим образом:

- Вначале применяются шаблоны для текущего типа объекта – те же три шага:
- Находятся все шаблоны для текущего пользователя.
- Только если таких шаблонов нет, или во всех найденных шаблонах отсутствует включенный флажок «Не использовать шаблон для групп» – поиск продолжается среди шаблонов для групп, куда входит текущий пользователь, кроме группы «Все пользователи».
- Только если вышеуказанных шаблонов нет, или во всех найденных шаблонах для групп отключен флажок «Не использовать шаблон для группы «Все пользователи»» – проверяется шаблон для группы «Все пользователи».

Только если ни в одном из найденных шаблонов для данного типа объекта не установлен флажок «Не использовать шаблон по умолчанию», то предыдущие три шага поиска шаблонов повторяются среди шаблонов по умолчанию.

Права из всех найденных шаблонов независимо копируются на защищаемую сущность и применяются суммарно.

16.4 Права доступа к документам архива (файлам)

Документы архива становятся защищенными автоматически, если документ (файл) импортируется в защищенный архив. Если архив не является защищенным, назначение прав доступа к документам данного архива не имеет смысла, либо невозможно.

Права устанавливаются для документа в целом и отдельно для каждой версии документа. Если права установлены, то считается, что отсутствующие в списке для данного документа (версии) пользователи (группы пользователей) прав не имеют. Для данного пользователя (группы пользователей) отсутствие прав на документ при наличии прав на версию документа понимается программой как запрещение прав на документ. И наоборот: если не установлены права на версию и установлены права на документ в целом – права на версию отсутствуют.

Права для версии документа вычисляются совместно с правами на весь документ. Установленные для версии права на доступ, изменение, удаление являются определяющими для данной версии, их реализация не зависит от установки прав для всего документа.

Право на просмотр документа (версии) реализуется при наличии у соответствующего документу приложения метода запуска – «Просмотр» (в противном случае право на просмотр реализовать не удастся). Разумеется, право доступа («Д») к документу и версии документа у пользователя должно быть.

Для всего документа устанавливаются следующие права:

- Доступ («Д»)
- Создание («С») (создание новых версий документа)
- Изменение («И»)
- Удаление («У»)

Для версии документа могут устанавливаться следующие права:

- Доступ («Д»)
- Изменение («И»)
- Удаление («У»)

Создание новой версии и изменение версии документа реализуется при наличии доступа к версии.

Запрет удаления одной версии приведет к запрету удаления всего документа.

Имеется возможность дополнительно к наследуемым и унаследованным правам, передавать (наследовать) на документы объекта непосредственные права с данного объекта. Для этого следует включить флажок «Наследовать все права на документы» (Рисунок 126). Если указанный флажок отключен, на документ передаются наследуемые и унаследованные права с объекта документа. Если объект документа общедоступный – права на документ передаются с защищенных объектов верхнего уровня. Если данный флажок включен, на документ и его версии также передаются непосредственные права с объекта документа (а на версии – и непосредственные права с документа).

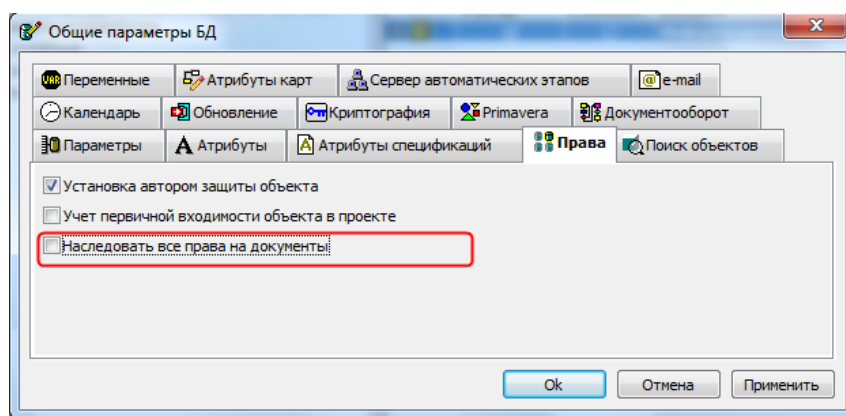


Рисунок 126. Настройка наследования прав на документы объекта

При создании документа или его версии наследование прав производится аналогично наследованию прав в дереве проектов.

Импорт документа в архив рассматривается как изменение объекта, к которому данный документ присоединяется. Поэтому для создания документа архива у пользователя должны быть разрешающие права «Изменение» для привилегии «Доступ к объекту» на тот объект, к которому он присоединяет документ.

На документ передаются права объекта по привилегии «Доступ к объекту». Именно эта привилегия наследуется с объекта на документ.

Право «Доступ» (привилегия «Доступ к объекту») наследуется на документ в виде «Доступ» (привилегия «Правка версий документа») и далее на версию «Доступ» (привилегия «Правка версии документа»). Право «Изменение» (привилегия «Доступ к объекту») наследуется на документы в виде «Создание», «Изменение», «Удаление» (привилегия «Правка версий документа») и далее на версию «Изменение», «Удаление» (привилегия «Правка версии документа») (Рисунок 127).

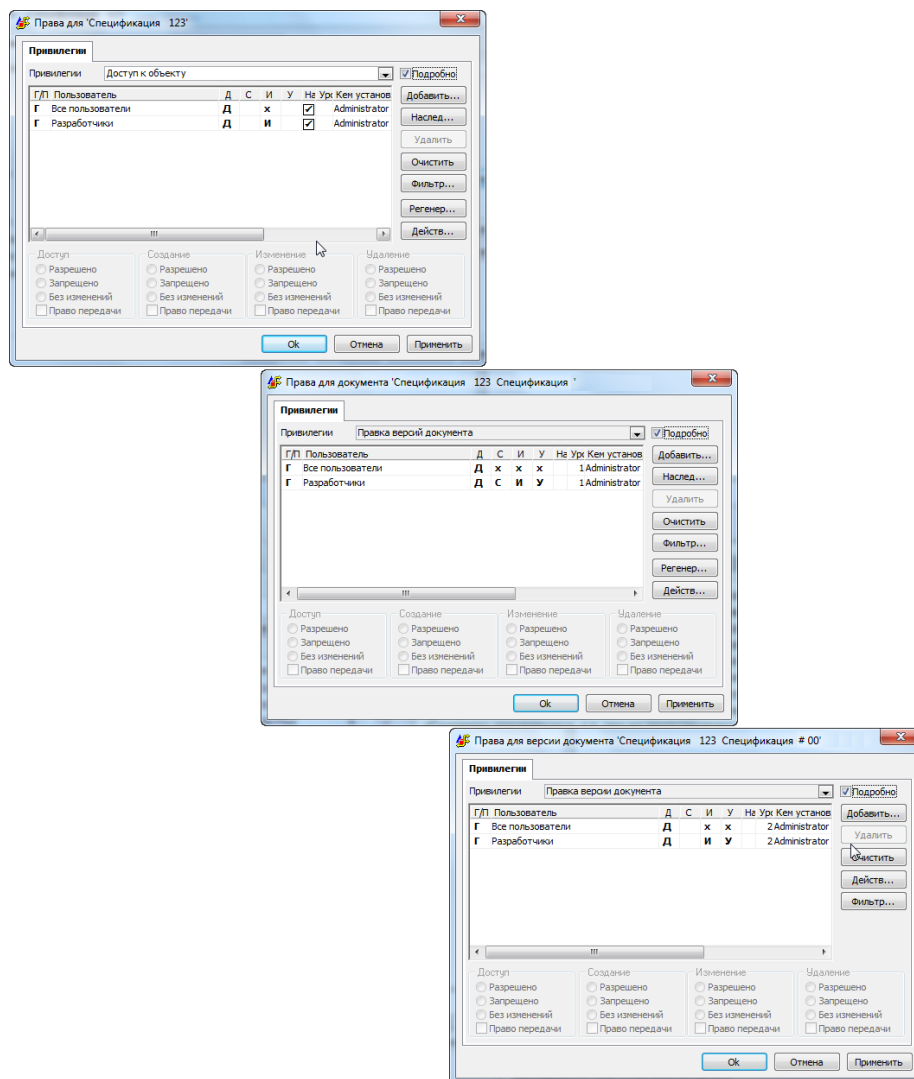


Рисунок 127. Наследование прав доступа к объекту на документ архива и версию документа. Пример

При наследовании прав с документа на его версии право «Создание» игнорируется, так как оно не предусмотрено в правах на версию документа.

Для обеспечения возможности создания версий, у пользователя должны быть наследуемые (при включенном флажке «Наследовать все права на документы» на вкладке «Права» «Общих параметров БД», достаточно непосредственных) права на документ, минимум – «Д», «С», «И».

17 Действия над объектами Lotsia PDM

Действие над объектами Lotsia PDM PLUS – это процедура, автоматизирующая выполнение определенной последовательности операций пользователя (макрокоманда). Выполнение действия может сопровождаться заполнением пользователем одной или нескольких специальных форм. Запускаться действия на выполнение могут различными способами (см. п. 17.2).

17.1 Общий порядок создания действий над объектами

Для создания действий используется «Главное меню» → «Администрирование» → «Структура данных» → «Редактор действий»: → откроется окно «Выбор действия» со списком имеющихся действий → для редактирования имеющегося – выбрать из списка ранее созданное действие, для создания нового – нажать «Создать».

Будет открыто окно редактора действий (Рисунок 128).

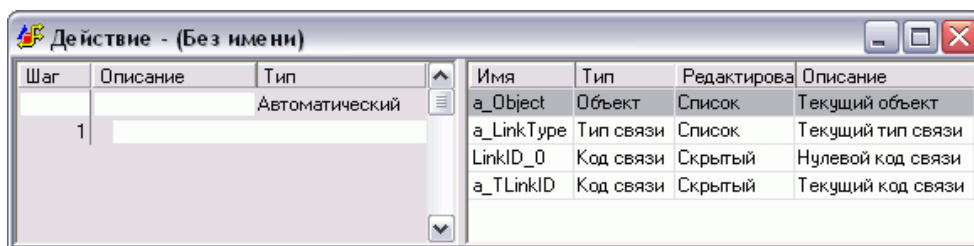


Рисунок 128 Окно редактора действий

В левой части окна вводятся функции, описывающие действия пользователя. Ввод функций осуществляется с использованием специального мастера, при этом функции выбираются из списка, предлагаемого программой. В открывшейся форме для выбранной функции указываются аргументы, при вводе значений аргументов, подразумевающих использование выражений, может использоваться редактор выражений. Если использования данного механизма оказывается недостаточно – используется ручной ввод или копирование одной или нескольких выделенных строк действий через буфер обмена. Для выделения нескольких строк действия удерживайте нажатой клавишу <Shift>.

Функции, из которых состоит действие, объединяются в шаги. Единственный шаг в действии есть всегда, можно добавить произвольное количество шагов. Смысл использования шагов следующий:

1. К каждому шагу можно создать отдельную пользовательскую форму (см. п. 17.3). Такие формы могут последовательно открываться перед выполнением очередного шага. Данные, введенные в указанной форме, могут использоваться в функциях на данном шаге, а также в функциях шагов, выполняемых после данного шага;
2. Перед выполнением очередного шага может быть запущено другое (вложенное) действие, в которое могут быть переданы аргументы и из которого можно вернуть в исходное действие значения переменных. Глубина вложенности действий не ограничивается;
3. Перед выполнением очередного шага может быть выполнен скрипт;
4. Перед выполнением очередного шага может быть открыто окно выбора;
5. Любой шаг можно снабдить уникальной для данного действия строковой меткой. По метке может быть осуществлен переход с использованием функции GoTo(). Можно реализовать переход к разным меткам, в зависимости от условий, в том числе циклическое выполнение шагов (см. п. 17.6);

6. В некоторых случаях промежуточный результат выполнения действия необходимо записать в БД. Это делается с помощью функции Update(), причем данная функция должна выполняться в конце шага, результат её выполнения может использоваться на следующем шаге действия.

Шаг может содержать произвольное количество функций (в том числе не содержать ни одной). Шаги выполняются последовательно, кроме случаев применения функции GoTo() (Перейти к шагу).

В правой части окна действия отображаются переменные, использующиеся в функциях. При открытии окна, в его правой части содержатся предопределенные переменные с именами a_Object, a_LinkType, a_LinkID и LinkID_0. Использование данных переменных не является обязательным, они могут быть удалены, изменены, созданы вновь после удаления. Однако, в большинстве случаев их использовать удобно. Имена предопределенных переменных несут функциональную нагрузку:

Переменная a_Object используется для обозначения текущего объекта (это применимо, если действие запущено по выделенному объекту – например, из контекстного меню, либо нажатием кнопки в атрибутивной форме, либо по строке выполненного отчета, содержащей колонку с именем Isibject_id). Данной переменной может быть присвоено иное значение, а при запуске действия способом, отличным от вышеуказанных, значение для данной переменной не задано (Null, если при написании действия не определено иное).

Переменная a_LinkType используется для обозначения текущего типа связи. Значение по умолчанию – «Дерево проектов», хотя и установлено, однако, при использовании переменной с именем a_LinkType в условиях отбора (см. рисунок 141) данное значение игнорируется, применяется текущий тип связи – тот, по которому открыто дерево, откуда было запущено данное действие. При запуске действия по строке отчета, содержащего колонку link_type_id, значение указанной колонки автоматически присваивается переменной a_LinkType.

Переменная LinkID_0 (задано значение по умолчанию – «Нулевой код связи») – для обозначения кода связи несуществующей связи. Предназначена для использования в функциях, при получении и установке атрибутов объекта. Для атрибутов проекта следует использовать конкретное, непустое (не «Нулевой код связи») значение кода связи, а для атрибутов объекта – «Нулевой код связи».

Переменная a_TLinkID – для обозначения связи между объектами. Значение по умолчанию для a_TLinkID – нулевой код связи. При запуске действия по выделенному в дереве не корневому объекту, переменная с именем a_TLinkID автоматически принимает значение кода связи выделенного в дереве объекта с его родителем (по данному коду связи действием можно определить, например, родительский объект). При запуске действия по строке отчета, содержащего колонку treelink_id, значение указанной колонки автоматически присваивается переменной a_TLinkID.

В действие могут добавляться требуемые переменные, их количество программой не ограничивается.

Для добавления переменных используется пункт контекстного меню правой части окна редактора действий «Добавить». В открывшемся окне (см. рисунок 129) определяются свойства добавляемой переменной.

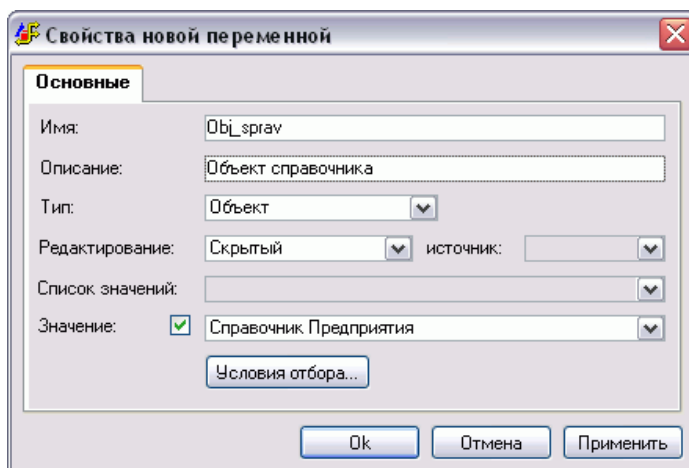


Рисунок 129. Окно свойств переменной. Пример

Имя переменной должно начинаться с латинской буквы, может содержать латинские буквы, цифры и символ подчеркивания. Имена переменных отображаются в функциях действия, поэтому рекомендуется задавать их таким образом, чтобы из имени переменной были ясны её тип и предназначение. Если данная рекомендация выполняется – чтение действия, а значит и его отладка и модификация, значительно облегчаются.

Описание переменной может отсутствовать, но рекомендуется вводить туда информацию о назначении данной переменной.

Тип переменной определяет тип данных, хранимый в данной переменной, в том числе, какой из сущностей Lotsia PDM данная переменная соответствует.

Тип переменной может быть одним из следующих:

- Строка
- Число
- Дата/время
- Атрибут
- Объект
- Тип объекта
- Тип связи
- Код связи
- Документ
- Версия документа
- Пользователь
- Отчет
- Приложение
- Тип документа
- Шаблон работы
- Библиотека
- Шаблон копирования
- Свободное сообщение

Редактирование переменной зависит от типа переменной, для переменных некоторых типов возможно также использование условий отбора (см. п. 17.3.2).

Переменной может быть присвоено значение по умолчанию, для этого флажок «Значение по умолчанию» следует включить, и в поле «Значение» указать требуемое значение.

Для добавления функций в действие с помощью мастера, в контекстном меню левой части окна редактора действий следует выбрать «Вставить функцию». Будет открыто окно «Редактирование функции» (Рисунок 130).

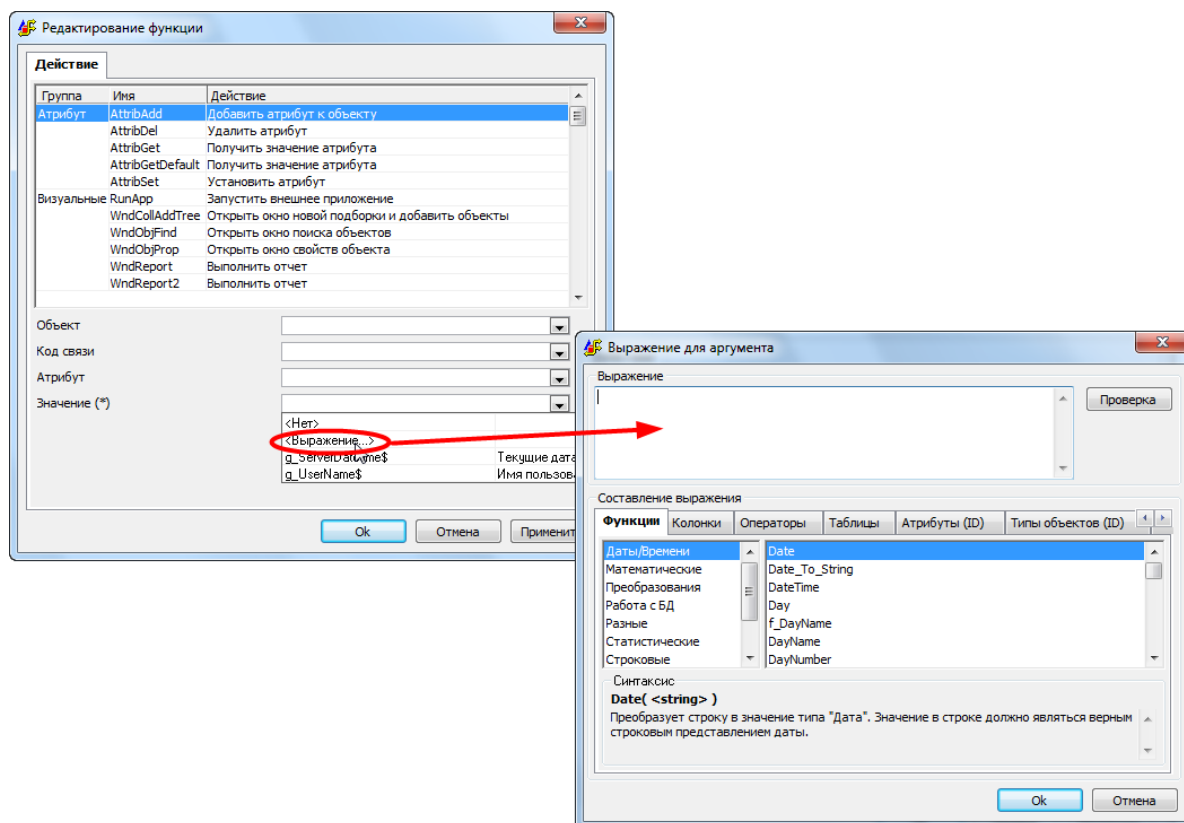



Рисунок 130. Окно «Редактирование функции» редактора действий

В верхней части данного окна следует выбрать требуемую функцию, в нижней части – указать аргументы. Для аргументов, отмеченных знаком «*» может использоваться редактор выражений. По нажатию «Применить» или «Ок» строка с данной функцией вставляется в действие. Назначение и синтаксис отдельных функций описаны в разделе «Типы переменных и функции Редактора действий» документа «Lotsia PDM Руководство администратора», расположенного в каталоге /Manuals вашего дистрибутивного диска. Для оперативного получения информации по синтаксису конкретной функции рекомендуется использовать встроенную справочную систему.

Для сохранения действия используется сочетание клавиш <Ctrl>+<s>, либо кнопка  инструментальной панели.

Для удаления введенной функции (строки) или её фрагмента, следует выделить удаляемую строку (фрагмент) и нажать <Delete>, либо <Ctrl>+<x>, либо выбрать в контекстном меню «Вырезать».

Для копирования (перемещения) фрагмента действия следует выделить, используя мышь или клавиши со стрелками, копируемый фрагмент при нажатой клавише <Shift>, далее использовать контекстное меню: «Копировать», «Вставить», либо сочетания клавиш: <Ctrl>+<c> (для перемещения – <Ctrl>+<x>), далее – <Ctrl>+<v> для вставки

перемещаемого или копируемого фрагмента. Отметим, что таким образом можно копировать (перемещать) одновременно несколько шагов действия.

Для одновременного удаления нескольких строк действия их необходимо выделить, держа нажатой клавишу <Shift>, далее вырезать, используя пункт контекстного меню или сочетание клавиш <Ctrl>+<x>.

Для вставки нового шага в действие перед имеющимся шагом, добавления и удаления шагов используйте контекстное меню левой части окна редактора действий.

Для копирования действия целиком – при открытом окне редактора действий, в верхнем меню «Файл» выбрать «Сохранить как...».

Примеры написания действий рассмотрены в разделе 17.11

17.2 Различные способы запуска действий

Действия могут запускаться на выполнение:

- при настройке данного действия, открытого в текущем окне редактора действий – из верхнего меню «Действие» → «Выполнить». Используется при отладке действия;
- из пункта Главного меню или соответствующей ему кнопке инструментальной панели (и соответствующей быстрой клавиши). В папке «Работа с проектами» Главного меню имеется стандартный пункт «Выполнить действие», откуда можно запустить любое действие.
- из контекстного меню выделенного объекта или сообщения;
- из контекстного меню выделенных объектов подборки или выделенных строк выполненного отчета;
- из верхнего меню Объект → Выполнить действие при выделенном одном или нескольких (в подборке или окне поиска) объектах;
- при настройке главного меню (см. п. 10.1) или контекстного меню объектов (см. п. 10.3.7) можно создать собственный пункт меню, подключив к нему действие. Тот же пункт меню может быть использован в качестве кнопки на инструментальной панели;
- автоматически при входе пользователя в систему. Такое выполнение задается в «Параметрах старта» при настройке профиля (см. п. 10.3) либо персональных настроек пользователя;
- по кнопке: в атрибутивной форме (см. п. 10.2; Рисунок 62), в форме Workflow (см. п. 23.15.2), в форме действия (см. п. 17.3.5), в отчете или форме аргументов отчета;
- При выполнении шага действия может запускаться вложенное действие (см. п. 17.8);
- при выполнении предопределенных процессов (Workflow) (см. п. 23.16) – автоматически и после выполнения этапов или при работе с формой предопределенного сообщения – по событиям формы, например: после открытия формы, изменении значения переменной, выполнения задачи и др....
- В ходе выполнения предопределенных процессов (Workflow), действия могут выполняться автоматически (на сервере автоматических этапов), от

имени специального пользователя, обладающего всеми необходимыми правами и приоритетом – (см. п.23.16.5)

При использовании некоторых способов, до начала выполнения действия в него могут быть переданы заранее заданные значения аргументов.

Для конечного пользователя рекомендуется обеспечивать выполнение необходимых ему действий только по нажатию кнопок в формах или на панели инструментов, событиям форм Workflow, либо использованием специально настроенных пунктов Главного меню, а также в автоматическом режиме – при входе пользователя в систему или выполнении предопределенных процессов.

В любом случае, для конечного пользователя выбор требуемого действия из списка следует исключить.

17.3 Настройка и использование форм шагов действий

Редактор форм, используемый в действиях, во многом похож на редактор атрибутивных форм (см. п. 10.2). Для добавления формы к шагу следует установить курсор в левой части окна редактора действий, в любую строку требуемого шага. В контекстном меню выбрать «Настройка шага...», откроется окно, в котором выбирается тип шага. По умолчанию установлено «Автоматический шаг», что означает отсутствие формы для данного шага. Для настройки формы шага в поле «Тип» следует указать «Свободная форма».

Еще один возможный тип формы – «Автоформа». При использовании автоформ настройка формы сводится к добавлению колонок в правой части окна, редактирование в левой части окна недоступно. Использование автоформ возможно, но в большинстве случаев лишено практического смысла, данный тип формы обеспечивает совместимость с ранними версиями программы, может использоваться в качестве «отладочных» форм при тестировании работы действия.

Удаление созданной формы осуществляется установкой значения «Автоматический шаг».

Остановимся подробнее на особенностях настройки свободных форм и дополнительных (в сравнении с атрибутивными формами, п. 10.2) возможностях редактирования значений. Способ редактирования колонки в форме действия может осуществляться заданием свойств переменной (см. п. 17.3.2) и определением свойств соответствующего объекта формы (см. п. 17.3.1).

Возможности настройки колонки соответствующей переменной средствами редактора форм зависят от типа переменной, а также от того, какие параметры установлены в свойствах соответствующей переменной.

В некоторых случаях удобным может быть использование автоматического завершения (то есть, закрытия без вмешательства пользователя) форм (см. п. 17.3.3).

Полную информацию по настройке форм можно найти в документе «Редактор форм», расположенном в каталоге /manuals Вашего дистрибутивного диска, либо воспользовавшись системой контекстной справки.

17.3.1 Настройка редактирования колонок средствами редактора форм

Настройка редактирования конкретной переменной зависит от типа переменной. Рассмотрим отдельно настройку редактирования средствами редактора форм колонок с типом значения «Число», «Строка» и «Дата/время» (см. п. 17.3.1.1) и переменных, редактирование которых подразумевает использование окон выбора. Отметим, что на практике для редактирования в форме действия, из переменных таких типов используется

переменные типа «Объект». Окно выбора объектов можно настраивать, обеспечивая выбор требуемого объекта в удобном для конечного пользователя виде (см. п. 17.3.1.3). Прямое редактирование в форме действия переменных других типов (например – «Тип объекта», «Тип связи», «Шаблон копирования» и т.д.), как правило, не используется или даже не предусмотрено (например – для переменной типа «Код связи»).

Вышесказанное, однако, не означает, что при выполнении действия нельзя или не требуется устанавливать значения переменным указанных типов. Значения таким переменным должны присваиваться автоматически, при этом могут анализироваться и использоваться данные, вводимые пользователем при редактировании числовых, строковых, датавременных значений, результат выбора файла (см. п. 17.3.2.4) или информационных объектов. Некоторые инструменты для присвоения значений подобным переменным рассмотрены далее (см. пп. 17.3.4, 17.3.5, 17.4).

По умолчанию, при работе с формой пользователь не может закрыть (завершить) форму очередного шага действия до тех пор, пока не присвоены значения всем переменным, добавленным в список переменных данной формы (правая часть окна редактора форм). Можно сделать ввод значения переменной необязательным, для этого необходимо снять соответствующий флажок (см. рисунок 131).

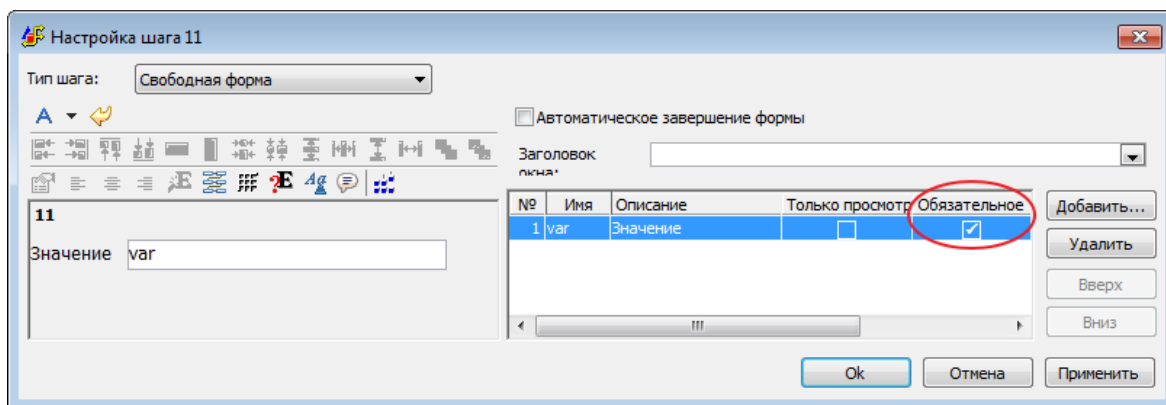


Рисунок 131 Флажок обязательности значения переменной

Отметим, что во избежание появления сообщений об ошибках, необходимо правильно обрабатывать случаи отсутствия значения у переменной (см. п. 17.7).

17.3.1.1 Настройка редактирования переменных с типом значения «Число», «Строка» и «Дата/время» с типом редактирования «Свободный»

Для обеспечения использования всех инструментальных средств данной возможности, в поле «Редактирование» свойств соответствующих переменных следует указать «Свободный» (см. рисунок 164). При установке в указанном поле значений «Список» или «Выпадающий список» также имеется возможность настройки редактирования колонки средствами редактора форм.

В редакторе форм шагов действий, в окне свойств колонок формы имеется пункт «Редактирование», при выборе которого можно выбрать стиль редактирования колонки (Рисунок 132).

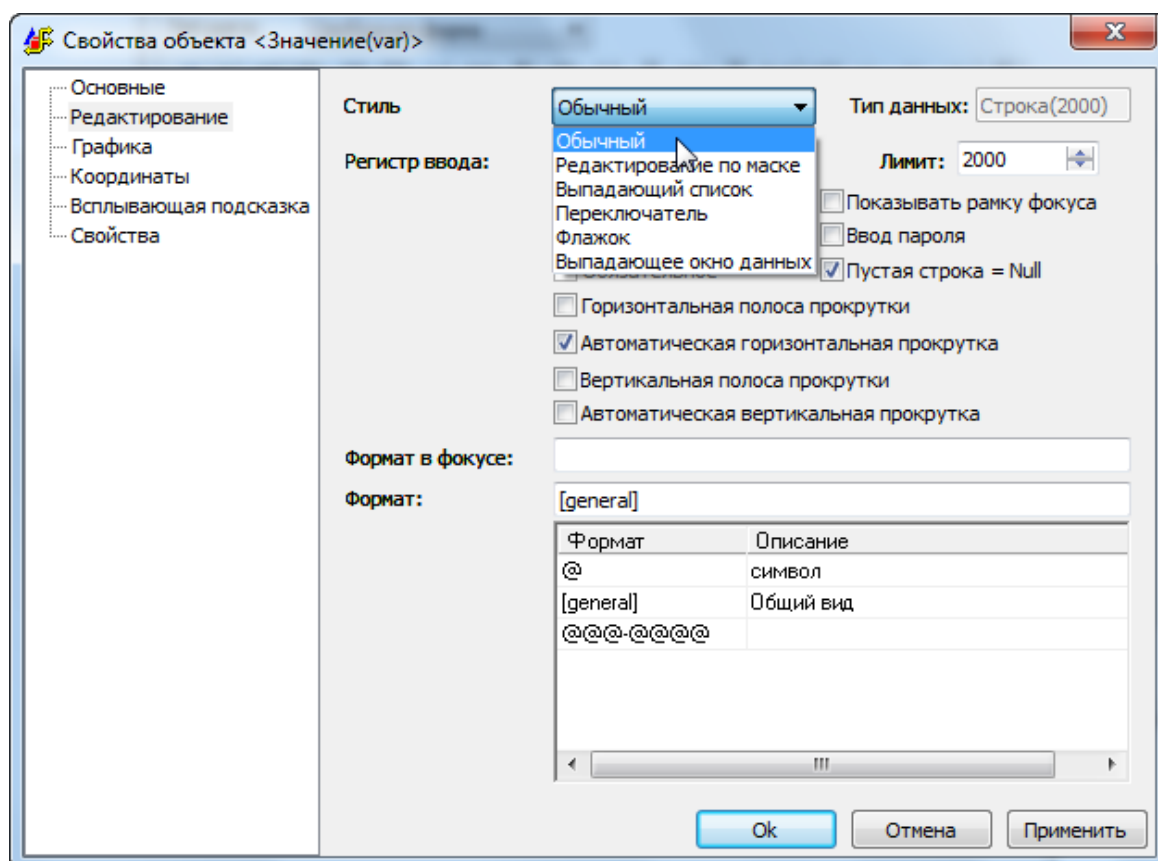


Рисунок 132 Выбор стиля редактирования колонки

При выборе стилей «Выпадающий список», «Переключатель», «Флажок» следует сформировать список присваиваемых переменной значений и соответствующие этим значениям видимые в форме значения (для флажка – указать присваиваемые значения для каждого состояния флажка), см. рисунок 133.

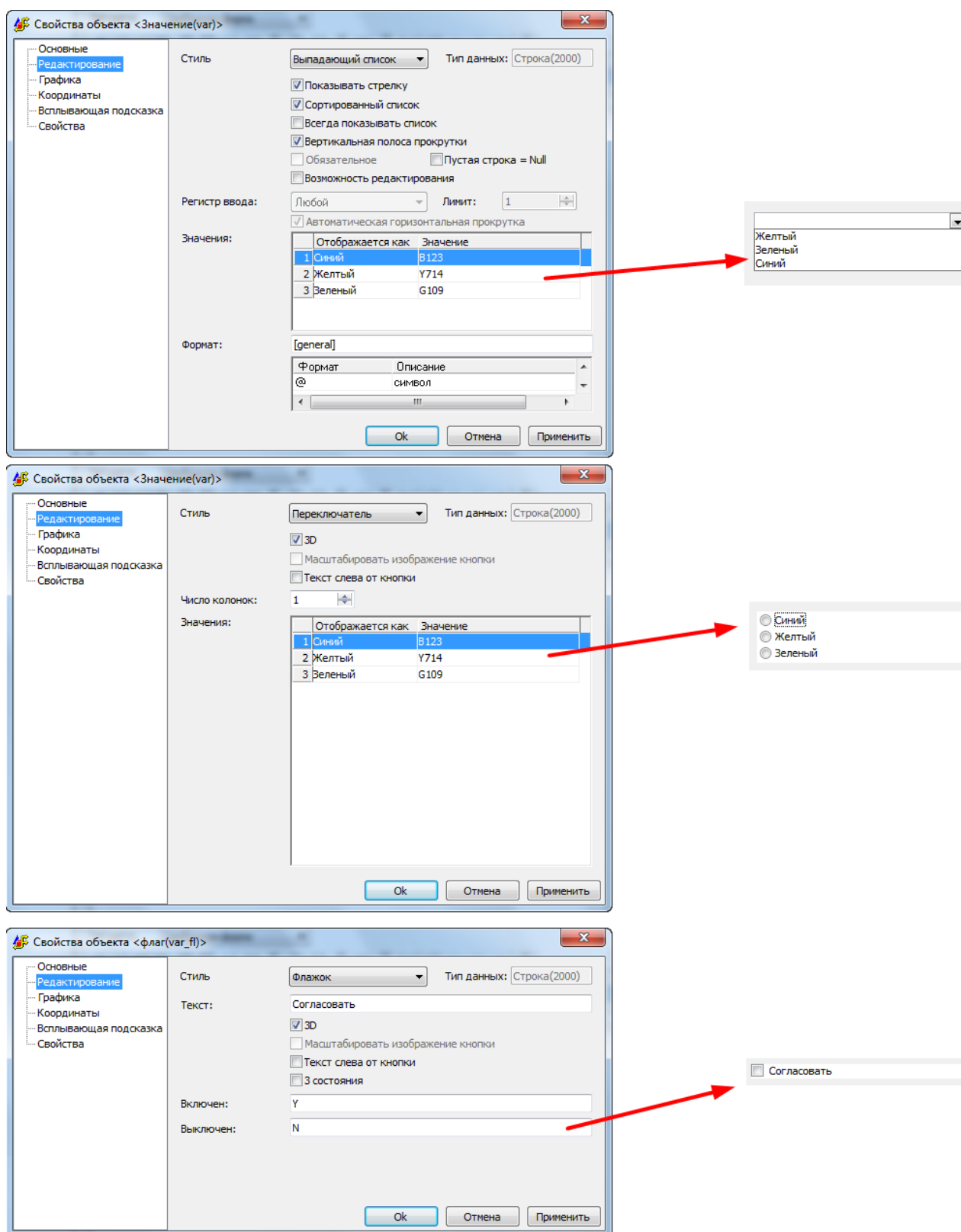


Рисунок 133 Указание значений переменной для различных стилей редактирования и пример отображения колонки в форме для выбранного стиля (по стрелке)

Для переменных с типом значения «Дата/время» возможен выбор значения из календаря. Для использования данной возможности следует выбрать стиль редактирования – «Редактирование по маске», а в поле «Выбор значения» указать «Календарь» (Рисунок 134).

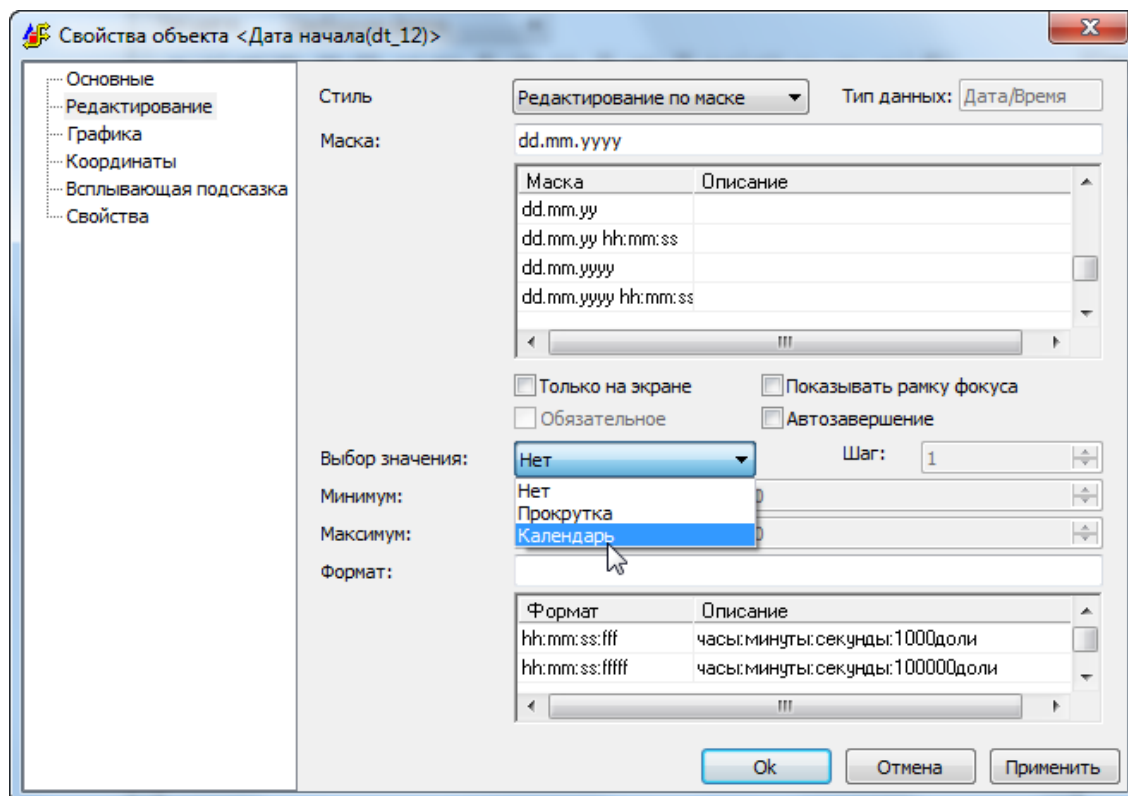


Рисунок 134 Задание способа редактирования даты – «Календарь»

Стили редактирования: «Переключатель», «Выпадающий список», «Флажок» применимы к переменным типом значения «Строка», «Число» и «Дата/время». Однако, возможно (путем дальнейшей обработки присвоенных значений в действии) их использование и для определения сущностей, имеющих идентификатор – объектов, типов объектов и других, для этого:

- в форме, с использованием стиля редактирования «Переключатель», «Выпадающий список» или «Флажок» осуществляется присвоение значения переменной. Используется переменная с типом данных «Число» или «Строка». При этом данной переменной присваивается значение ID соответствующей сущности Lotsia PDM;
- далее с помощью функции SetByID автоматически устанавливается значение переменной, с соответствующим типом значения

Пример подобного действия – см. п. 17.11.2

17.3.1.2 Настройка редактирования переменных с типом значения «Строка» и с типом редактирования «Выбор файла»

Для обеспечения использования данной возможности, в поле «Редактирование» свойств соответствующих переменных следует указать «Выбор файла» (см. рисунок 139).

В этом случае, при задании свойств колонки данной переменной в форме действия, возможна настройка окна выбора файла. Данная настройка осуществляется с использованием пункта «Выбор файла» окна свойств колонки (Рисунок 135).

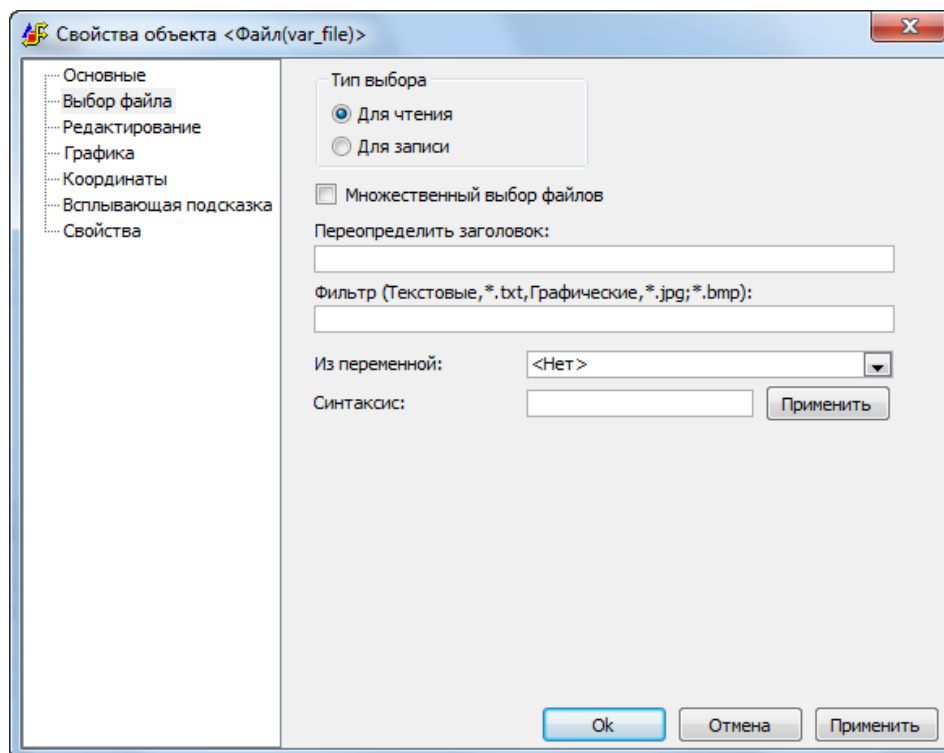


Рисунок 135 Пункт «Выбор файла»

Может быть задан тип выбора файла («Для чтения» – может быть указан только существующий файл, «Для записи» – может быть введено имя несуществующего файла), переопределен заголовок окна выбора файла и значение фильтра по расширениям файлов.

Если требуется возможность выбирать более 1 файла (при нажатой клавише <Ctrl>), необходимо включить флажок «Множественный выбор файлов».

Результат выбора возвращается в данную переменную в виде строки – полного имени (имен) выбранного файла (файлов). Если в полном имени выбранного файла имеется пробел – имя этого файла автоматически помещается в кавычки. При множественном выборе в качестве разделителя имен выбранных файлов используется символ «|» (вертикальная черта).

Настройку окна выбора файла можно изменять при выполнении действия, считывая требуемое значение из строковой переменной, указанной в поле «Из переменной» (Рисунок 135). Для формирования соответствующих значений указанной переменной выполните настройку на данной вкладке требуемым образом, после чего нажмите «Применить» в правом нижнем углу окна, скопируйте в буфер обмена строку, расположенную ниже поля «Из переменной» (Рисунок 135). Указанная строка содержит информацию о сделанных настройках окна выбора файлов. Используйте полученные указанным образом строки в выражениях для присвоения значений переменной при выполнении действия.

Имеющуюся (ранее сохраненную) строку со значением указанной переменной можно использовать и для настройки других аналогичных объектов формы. Для этого значение переменной следует вставить в поле «Синтаксис», после чего нажать «Применить» рядом с указанным полем.

17.3.1.3 Использование окон выбора объектов в формах действий над объектами

При использовании в формах действий над объектами окон выбора для переменных с типом значения «Объект» возможна настройка окна выбора средствами

редактора форм. Настройка осуществляется с использованием пункта «Редактирование» соответствующей колонки формы (см. рисунок 136).

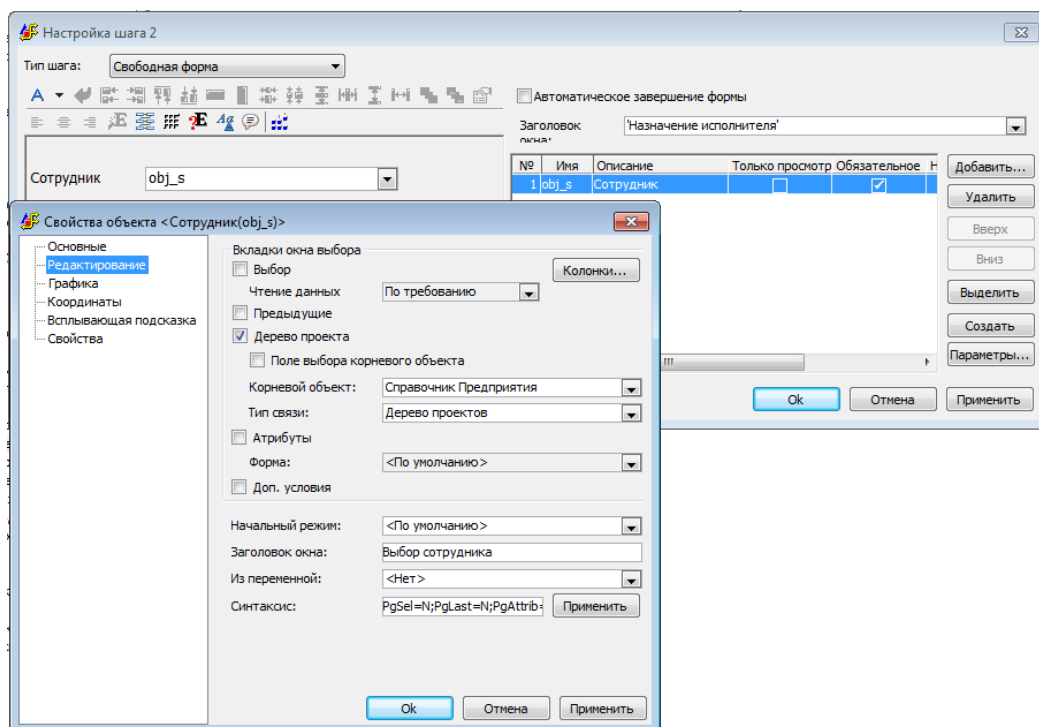


Рисунок 136 Настройка окна выбора объекта – из заданного дерева (Пример)

Настройку окна выбора объекта можно изменять при выполнении действия, считывая требуемое значение из строковой переменной, указанной в поле «Из переменной» (см. рисунок 136). Для формирования соответствующих значений указанной переменной выполните настройку редактирования на данной вкладке соответствующим образом, нажмите «Применить» в правом нижнем углу окна, после чего скопируйте в буфер обмена строку, расположенную ниже поля «Из переменной» (см. рисунок 136). Указанная строка содержит информацию о текущих настройках данного окна. Используйте полученные указанным образом строки в выражениях для присвоения значений переменной при выполнении действия.

Имеющуюся (ранее сохраненную) строку со значением указанной переменной можно использовать и для настройки других аналогичных объектов формы. Для этого значение переменной следует вставить в поле «Синтаксис», после чего нажать «Применить» рядом с указанным полем.

При настройке окна выбора возможно использование условий отбора для переменной, которая используется в качестве объекта формы при открытии данного окна выбора. Например – при выборе объекта ограничить выбор заданием допустимых типов объекта и/или значениями атрибутов (Рисунок 137).

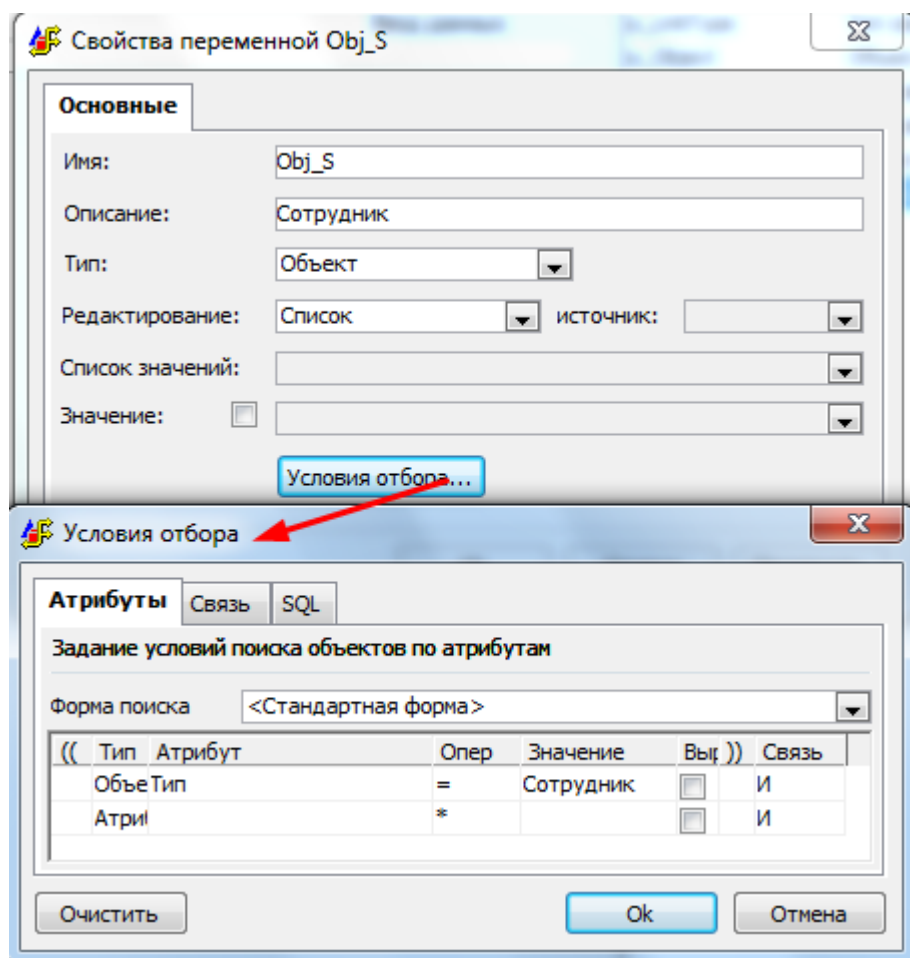


Рисунок 137 Задание условий отбора для переменной действия (Пример)

Если выбранное в окне выбора значение не удовлетворяет заданным условиям отбора – его невозможно присвоить переменной, кнопка «Ok» будет неактивной (Рисунок 138).

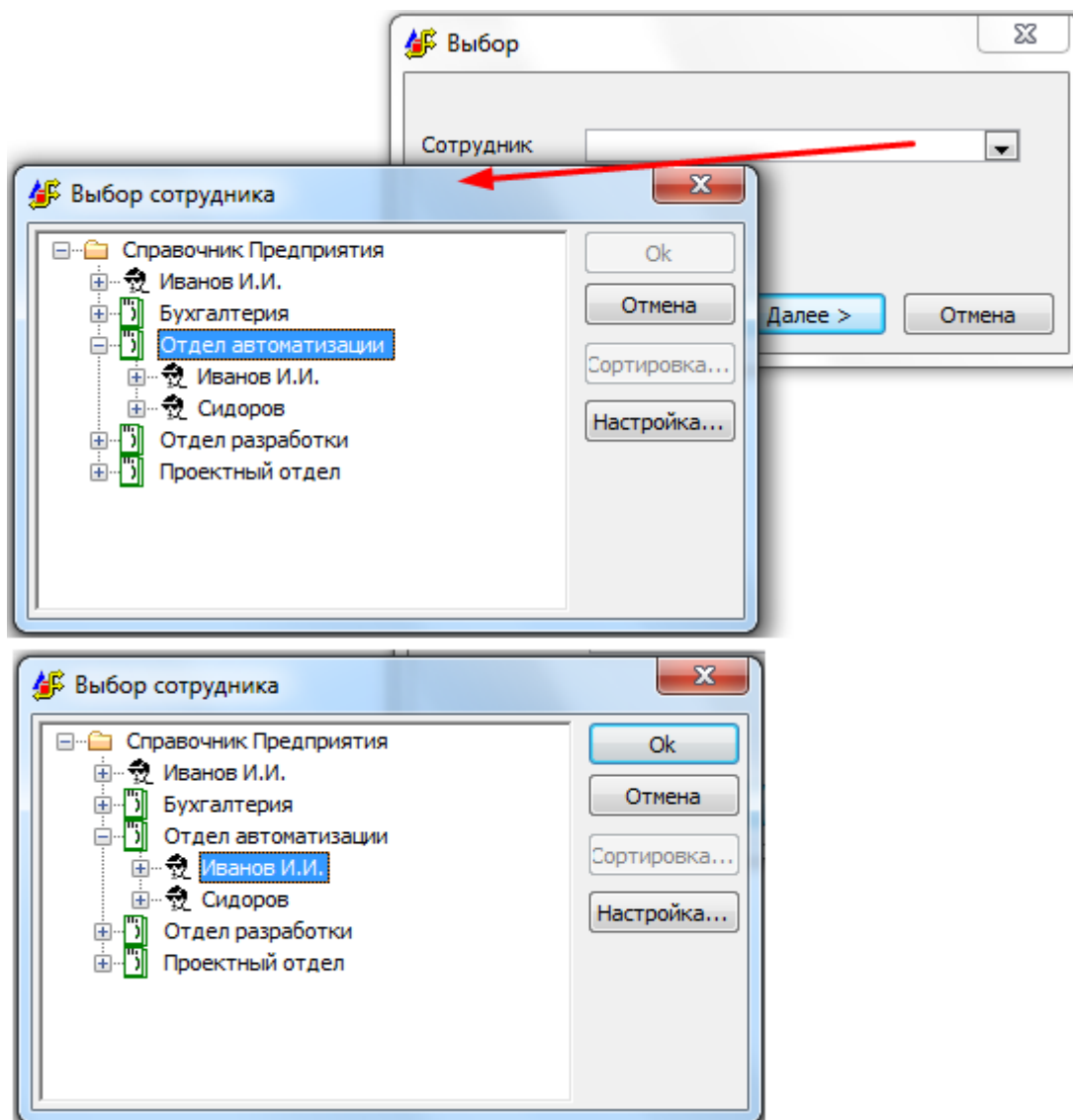


Рисунок 138 Результат применения условий отбора для переменной действия (Пример)

17.3.2 Настройка редактирования заданием свойств переменной: способы редактирования, условия отбора

Если в свойствах переменной действия параметр «Редактирование» установлен в положение «Свободный», то способ редактирования может быть определен настройкой конкретной формы действия (см. п. 17.3.1.1).

Изменять параметр «Редактирование» в свойствах переменной следует до её добавления в форму. Если изменить указанный параметр для переменной, которая уже добавлена в форму, следует выделить данную переменную в списке переменных формы в правой части окна и нажать кнопку «Создать». Если этого не сделать, во многих случаях форма не будет работать правильно. Данная особенность относится и к изменению типа значения переменной.

Выполнение настройки редактирования колонки средствами формы возможны не для всех значений редактирования, определенных в свойствах переменной.

Если в свойствах переменной в поле «Редактирование» указано «Скрытый» (по умолчанию так и установлено), то при добавлении данной переменной в форму шага

действия, она добавляется только в список переменных в правой части окна редактора форм. В левой части окна такая переменная не отображается, её использование в форме возможно только в выражениях вычисляемых полей или свойств объектов формы.

Если в свойствах переменной в поле «Редактирование» указано «Просмотр», редактирование значения такой переменной во всех формах, где данная переменная используется, будет невозможно. Более гибко запрет редактирования переменной можно реализовать средствами формы, но при этом в свойствах переменной не должно быть указано «Скрытый» или «Просмотр».

При определении свойств переменной действия, для различных типов данных перечень возможных способов редактирования различается, (см. рисунок 139).

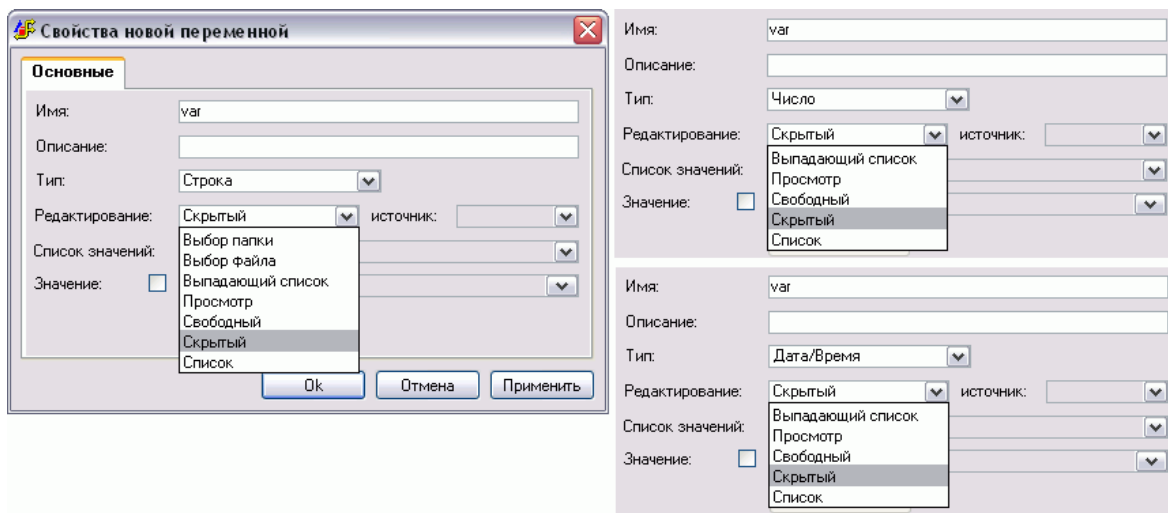


Рисунок 139 Определение свойств переменных для типов «Строка», «Число» и «Дата./Время»

Если в свойствах переменной в поле «Редактирование» указано: «Список», «Выпадающий список» «Выбор папки» или «Выбор файла», указанный способ редактирования невозможно изменить средствами редактора форм, однако дополнительная настройка редактирования таких переменных средствами редактора форм во многих случаях возможна и применяется.

Использование редактирования «Список» различается в зависимости от типа значения переменной.

17.3.2.1 Формирования списка возможных значений переменной: использование имеющихся значений атрибута

Для переменных с типом значения «Строка», «Число» или «Дата/время» при использовании способа редактирования «Список» (или «Выпадающий список») требуется указание источника используемого списка. Источником для формирования списка может быть атрибут или выражение.

Для списка с источником «Атрибут» может использоваться список значений атрибута, указанного в поле «Список значений» (Рисунок 140).

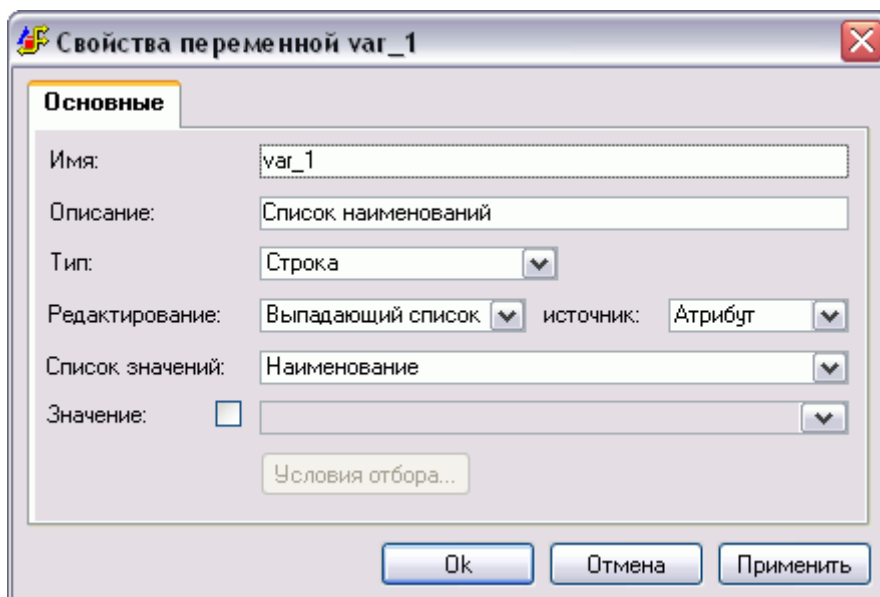


Рисунок 140 Указан атрибут, из списка значений которого будут присваиваться значения переменной действия.

Способ редактирования, указанный в настройке выбранного атрибута имеет значение, если атрибут присваивается из классификатора, а способ редактирования переменной «Список» (использование классификаторов в действиях будет рассмотрено далее, см. раздел 17.3.4). В других случаях отображается список фактически имеющихся в БД значений выбранного атрибута, независимо от того, какой способ редактирования установлен в настройках данного атрибута – «Список», «Свободный», «Классификатор» или «Внешнее». Список, отображаемый при выборе значения переменной из списка значений атрибута можно ограничить, (см. рисунок 13) исключив часть значений.

Для списка с источником «Выражение» список значений формируется с использованием выражения (см. п. 17.3.2.2).

Для переменных с типом значения «Атрибут», «Объект», «Тип объекта», «Тип связи», «Документ», «Версия документа», «Пользователь» и некоторых других типов при использовании способа редактирования «Список» в форме будет открываться окно выбора соответствующей сущности. За редким исключением, для конечных пользователей использование стандартных окон выбора с большим списком значений – не самый удобный вариант. Дополнительной настройкой окна выбора средствами редактора форм (см. п. 17.3.1.3) можно обеспечить конечному пользователю достаточно удобный интерфейс. Во многих случаях оптимальным будет вместо переменных с указанными типами значений, добавлять в форму переменные, соответствующие ID указанных сущностей (подробнее будет рассмотрено далее, см. п. 17.4 и пример действия п.17.11.2).

17.3.2.2 Формирования списка возможных значений переменной: использование выражений

Для использования выражений при формировании списка значений переменной, в поле «источник» следует указать «Выражение».

Само выражение вводится в поле «Значение». Выражение представляет собой строку следующего формата:

<тип>[;<параметр>[;<параметр>...]]~t<синтаксис>

где:

<тип> может принимать значения list или select.

Значение list может применяться только для формирования значений выпадающего списка. Параметры не используются, то есть, для значения «list» выражение может быть следующего формата:

list~t<синтаксис>

где <синтаксис>- строка следующего формата:

<отображаемое значение>~t<реальное значение>/<отображаемое значение>~t<реальное значение>...

Примеры выражений:

'List~tКрасный~t11/Зеленый~t22/Желтый~t33'

Данное выражение формирует список выбора из значений: «Красный», «Зеленый», «Желтый». При выборе в выпадающем списке одного из указанных значений, присваивается, соответственно: «11», «22», «33».

'List~tКрасный~t11/' + str_0 + '~t' + num_0 + '/Зеленый~t22/Желтый~t33/' + str_1 + '~t' + num_1

Данное выражение формирует список выбора из значений: «Красный», значение переменной str_0, «Зеленый», «Желтый», значение переменной str_1. При выборе в выпадающем списке одного из указанных значений, присваивается, соответственно: «11», значение переменной num_0, «22», «33», значение переменной num_1.

Для типа «select» в выражении могут применяться поля <параметр>.

Тип select может применяться для формирования значений списка или выпадающего списка при помощи SQL-запроса. Формат указания параметров:

<имя_параметра>=<значение_параметра>

Если параметров более одного, между параметрами используется разделитель – <;> (точка с запятой).

Допускаются следующие параметры:

Cols – перечисление через запятую типов колонок результата запроса в следующем формате 'cols=type1,type2,type3,...'. Допустимые типы: char(n), date, datetime, decimal(n), long, number, time, где «n» – количество символов.

Titles – заголовки колонок, разделенные символом «|». Используются, если указан параметр Cols, и имеет смысл только для типа редактирования "Список".

DataColumn – имя колонки результата запроса, используемой как данные. Если параметр не указан, то используется первая колонка результата запроса. Тип данных колонки должен соответствовать типу данных редактируемой переменной.

DisplayColumn – имя колонки результата запроса, используемой для отображаемого в форме значения. Если параметр не указан, то используется колонка данных. Используется только для выпадающего списка. Тип данных может быть любой.

В качестве имени колонки в параметрах DataColumn и DisplayColumn можно использовать её номер в запросе: #1, #2 и т.д.

Для типа select – <синтаксис> – строка, содержащая SQL-запрос или вызов хранимой процедуры. Количество возвращаемых колонок не ограничено, но более двух имеет смысл использовать только для типа редактирования «Список», чтобы отобразить результат в таблице в диалоговом окне выбора значения атрибута.

Пример выражения для типа select:

'select; DataColumn=#2;DisplayColumn=#1~tselect Name, id from LSDBO.EmplUsers_v USR where USR.Active=~'Y~' and USR.UserGroup=~'N~'

Данное выражение для числовой переменной сформирует и отобразит в форме действия список имен активных пользователей, без групп. Переменной будет присвоено значение идентификатора выбранного в списке пользователя.

17.3.2.3 Условия отбора

Для переменных типа «Объект», «Документ» или «Версия документа», а также «Атрибут», «Тип объекта», «Тип связи», «Пользователь», «Приложение», «Тип документа», «Шаблон работы», «Шаблон копирования» могут быть заданы условия отбора (Рисунок 141).

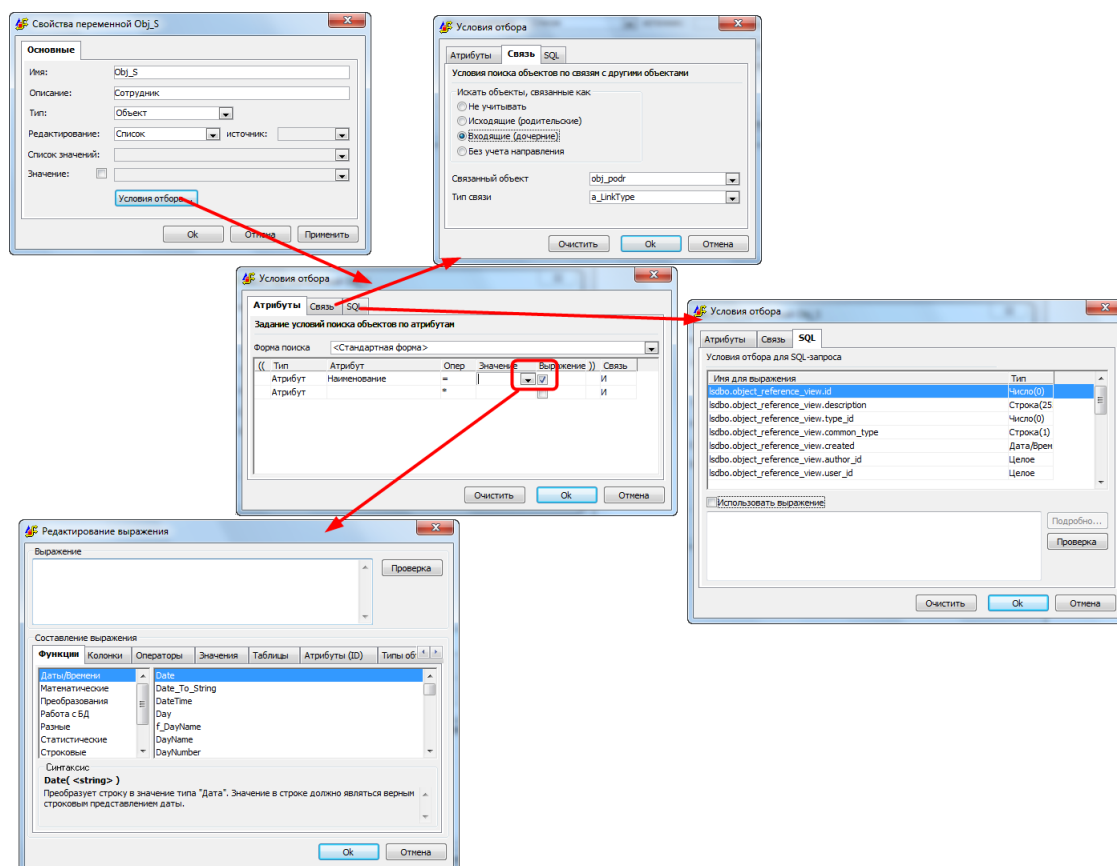


Рисунок 141 Задание условий отбора

Как правило, переменные с условиями отбора используются в формах действий, а целью указания условий отбора является ограничение перечня возможных значений в окне выбора значения данной переменной. Для реализации данной возможности, в свойствах данной переменной, в поле «Редактирование» должно быть установлено «Список».

Для объектов отбор может выполняться по характеристикам объекта – значению атрибутов, описания и др., или/и по связям с другими объектами. Для документов отбор производится по принадлежности документа объекту, для версий документов – по объекту и документу, к которому относится данная версия.

Можно заданием условий отбора обеспечить последовательный выбор, например, для объектов или документов, указывая в условиях отбора очередной переменной предыдущую переменную. Например, требуется вначале выбрать раздел справочника, потом – подраздел, и, наконец, элемент справочника. В данном примере – раздел, подраздел, элемент справочника – объекты, соответственно входящие друг в друга. Для

последовательного выбора, в условиях отбора подраздела должно быть указано, что он является «Входящим (дочерним)» по отношению к разделу, а элемент, соответственно, к подразделу (Рисунок 142)

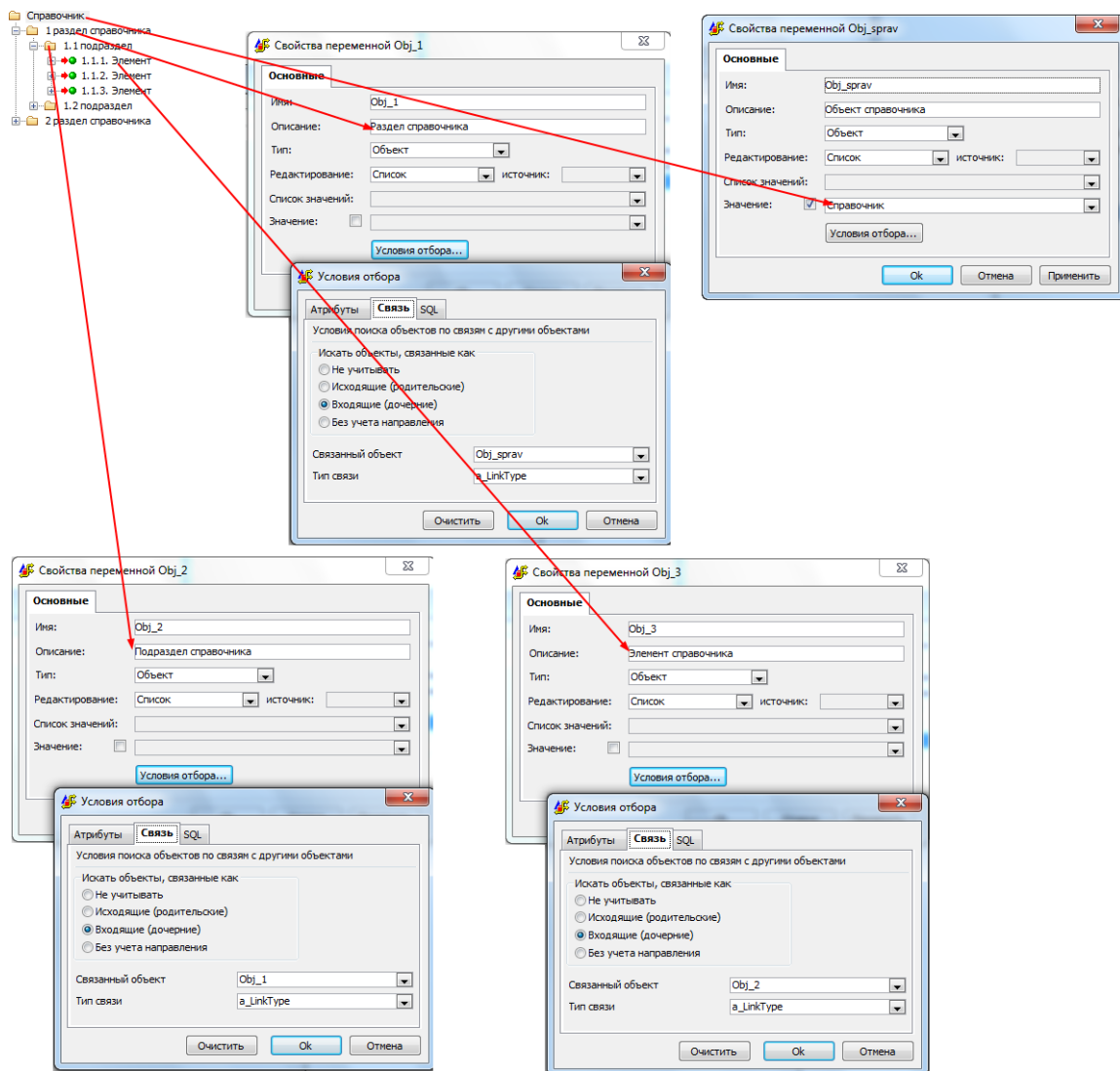


Рисунок 142 Пример указания условий отбора при выборе дочерних объектов

После выполнения соответствующей настройки, форма действия для реализации рассмотренного примера может выглядеть следующим образом – см. рисунок 143

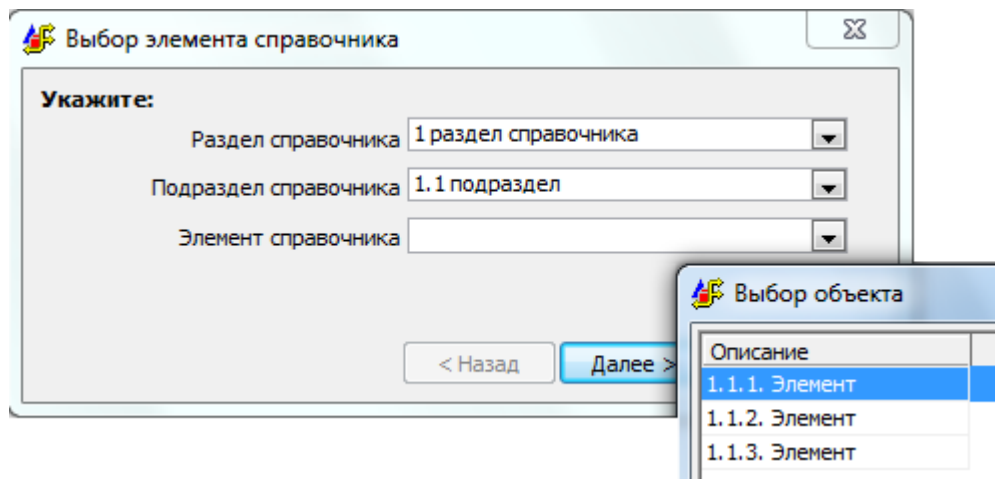


Рисунок 143 Пример формы действия для выбора элементов справочника

Далее остановимся на некоторых особенностях использования условий отбора:

При указании в условиях отбора типа связи `a_Link_Type`, поиск связанных объектов осуществляется по текущему типу связи, присвоенное переменной `a_Link_Type` значение игнорируется – то есть, для поиска по конкретному типу связи следует создать и использовать соответствующую переменную с типом «Тип связи» и именем, отличным от `a_Link_Type`. При задании условий отбора по характеристикам объекта возможно использование выражений – для этого необходимо включить соответствующий флажок, см. рисунок 141. В выражениях могут использоваться переменные, значения которых определяются при выполнении действия.

Если значения, удовлетворяющие условиям отбора, не найдены, окно выбора будет пустым и присвоить переменной значение в форме будет невозможно.

По умолчанию, поиск значений, удовлетворяющих условиям отбора, производится и при открытии окна формы, и при открытии каждого окна выбора для присвоения значения соответствующей переменной. Если найдено единственное значение, удовлетворяющее условиям отбора, такое значение показывается в форме сразу, до открытия окна выбора. Если таких значений несколько – требуется открыть окно выбора и выбрать значение. При открытии окна выбора также производится поиск в БД значений согласно условиям отбора данной переменной.

Если значений, удовлетворяющих условиям отбора, всегда более одного, то при открытии формы поиск значений для таких переменных имеет смысл не выполнять, в результате несколько ускорится открытие формы. По мере открытия окон выбора для каждой из переменных будет выполняться поиск соответствующих значений. Отказ от поиска значений согласно условиям отбора выполняется включением флажка «Не заполнять автоматически» (см. рисунок 144).

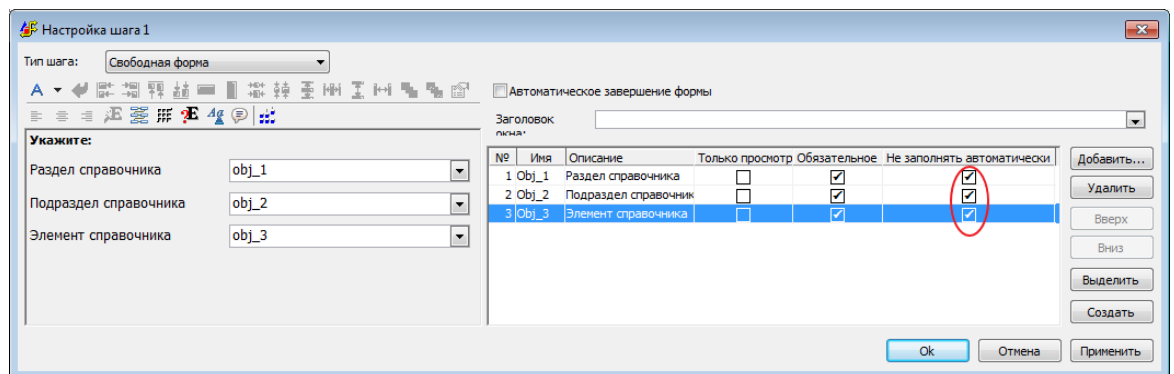


Рисунок 144 Установлен флажок «Не заполнять автоматически»

Использование условий отбора и автоматического завершения формы (см. п. 17.3.3) позволяет осуществить автоматическое присвоение значения соответствующей переменной, либо проверку наличия значения у переменной. Отметим, что для присвоения переменной значения, задания только условий отбора, позволяющих однозначно определить соответствующее значение переменной (например – объект Lotsia PDM) не достаточно, необходимо еще и в ходе выполнения действия открыть форму, в которую добавлена соответствующая переменная (достаточно использовать форму с автоматическим завершением см. п. 17.3.3).

Если в свойствах переменной заданы условия отбора, то эти условия проверяются не только при выборе значения переменной в форме, а также в других случаях присвоения значения, например, при использовании функции `SetByID`. В случае если присваиваемое значение не соответствует условиям отбора, будет выдано сообщение об

ошибке. Если в условиях отбора данной переменной используются переменные, значения которых на момент присвоения значения данной переменной не определены, при присвоении значения данной переменной будет выдано сообщение об ошибке. Таким образом, если для переменных применяются условия отбора, содержащие переменные, то при присвоении значений таким переменным необходимо обеспечить такой порядок присвоения значений каждой переменной, чтобы на момент присвоения значения данной переменной были определены все переменные, использующиеся в условиях отбора данной переменной. Естественно, присваиваемое значение должно удовлетворять условиям отбора.

Например: пусть имеются 2 переменных типа «Объект», Obj_1 и Obj_2. Условия отбора для Obj_2 заданы таким образом, что в условиях отбора Obj_2 используется значение Obj_1 (например, Obj_2 является потомком Obj_1).

Если с помощью функции SetByID присвоить значение переменной Obj_2 раньше, чем будет присвоено значение Obj_1, будет выдано сообщение об ошибке. Если же вначале присвоить значение переменной Obj_1 а потом – переменной Obj_2 – ошибки не будет (в случае, если присвоенное значение удовлетворяет условиям отбора).

17.3.2.4 Выбор файла или папки

При задании свойств строковых переменных доступны также способы редактирования «Выбор файла» и «Выбор папки». При использовании указанных способов редактирования, при добавлении переменной в форму, для присвоения значения будут использоваться стандартные окна Windows: «Выбор файла» или «Обзор папок».

Указанные окна могут быть дополнительно настроены средствами редактора форм (см. п.17.3.1.2), в том числе, возможен выбор одновременно нескольких файлов.

17.3.3 Автоматическое завершение форм

Автоматическое закрытие формы (автоматическое завершение) – автоматическое закрытие формы шага действия без участия пользователя. Для использования данного инструмента в окне редактора формы для шага действия следует установить флажок «Автоматическое завершение формы» (Рисунок 145).

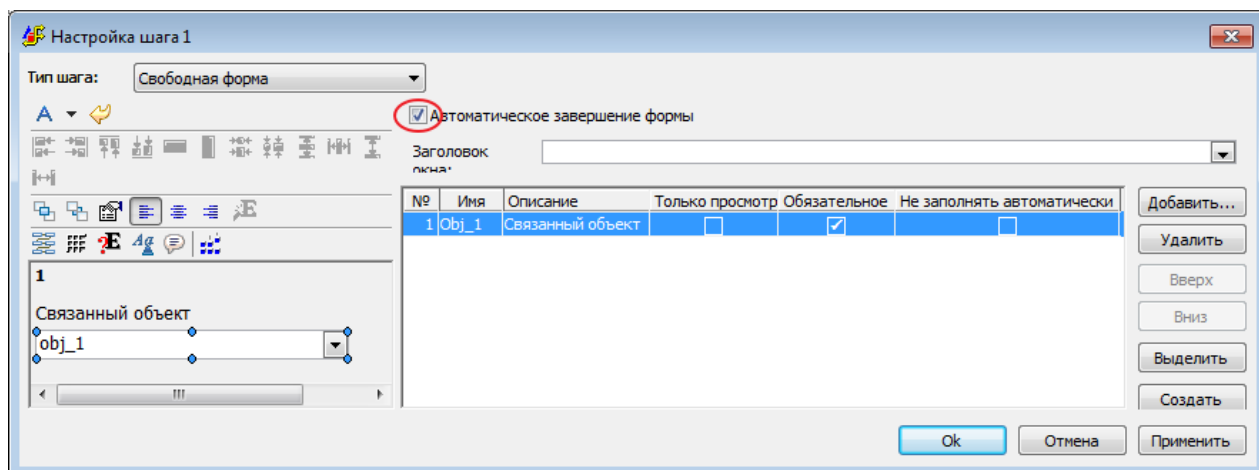


Рисунок 145 Включен флажок «Автоматическое завершение формы»

Базовое назначение данного инструмента – автоматически, без вмешательства пользователя присваивать переменной значение и закрывать форму, если условиями отбора (см. п.17.3.2.3) значение всех переменных данной формы определено однозначно, а если найдено несколько значений, удовлетворяющих условиям отбора – оставлять форму шага открытой для обеспечения возможности указания значения пользователем.

Помимо указанного назначения, формы с автоматическим завершением могут применяться и для автоматического присвоения значений переменным в соответствии с условиями отбора. В этом случае подразумевается, что условия отбора определены таким образом, что значение, удовлетворяющее условиям отбора единственное.

Для указанного применения следует учитывать следующие особенности автоматического закрытия формы:

В случае если при настройке формы для данной переменной флажок обязательности (см. рисунок 131) не отключался (по умолчанию он включен), форма с автоматическим завершением будет работать следующим образом:

- если условия отбора однозначно определяют отбираемую сущность, форма будет автоматически закрываться, а переменной присваиваться соответствующее значение;
- если условия отбора не позволяют однозначно определить отбираемую сущность, форма останется открытой, позволяя пользователю выбрать значение переменной из окна выбора, где отображаются найденные значения удовлетворяющих условиям отбора (если такие значения найдены, в противном случае список значений будет пустым и присвоить значение в форме будет невозможно). Следует учитывать, что список объектов, доступных пользователю для выбора, зависит и от прав доступа пользователя.

В случае если при настройке формы для данной переменной флажок обязательности (см. рисунок 131) был снят, форма с автоматическим завершением будет во всех случаях автоматически закрываться. При этом:

- если условия отбора однозначно определяют отбираемую сущность – переменной присваиваться соответствующее значение;
- если условия отбора не позволяют однозначно определить отбираемую сущность – значение переменной присвоено не будет, независимо от того, найдено ли в БД таких значений более одного или не найдено совсем.

В случае если флажок обязательности снят и включен флажок «Не заполнять автоматически» (см. рисунок 144) (по умолчанию выключен), в форме с автоматическим завершением значение переменной с заданными условиями отбора автоматически присвоено не будет, даже если найдено единственное значение.

Таким образом, применение форм с автоматическим завершением для автоматического присвоения значений переменным возможно при наличии единственного значения, удовлетворяющего условиям отбора. В остальных случаях требуется выполнение необходимых проверок, либо действий пользователя.

17.3.4 Использование классификаторов в формах действий

Чтобы обеспечить присвоение переменной значения из классификатора, следует для строковой переменной действия, в свойствах переменной, установить редактирование – «Список» (не путать с «Выпадающим списком»). В появившемся поле «Список» указать соответствующий атрибут, редактируемый из классификатора. После добавления данной переменной в форму действия, её значение будет выбираться из классификатора. При этом настройка стиля редактирования средствами редактора форм не производится (вкладка «Редактирование» в свойствах соответствующего элемента формы не отображается).

В действиях используется только классификатор по умолчанию, заданный для данного атрибута.

Если в форме действия имеются другие переменные, подключенные аналогичным образом к тому же самому классификатору, то, при редактировании любой из указанных переменных, значения присваиваются всем переменным данной формы, подключенным к общему классификатору – аналогично использованию классификатора для присвоения нескольких атрибутов объекту.

В большинстве случаев, наряду с использованием классификатора в форме действия, для решения сходных задач следует также рассмотреть и другой инструмент – выбор объекта (и, например, последующее считывание с него атрибутивной информации), с соответствующей настройкой окна выбора, например – см. рисунок 136.

Решение об использовании того или иного инструмента принимается в зависимости от особенностей решаемых данной настройкой задач.

17.3.5 Использование кнопок в формах действий

Использование кнопок в формах действий не отличается от использования кнопок в атрибутивных формах. Для добавления кнопки используйте контекстное меню левой части окна редактирования свободной формы действия (Рисунок 146).

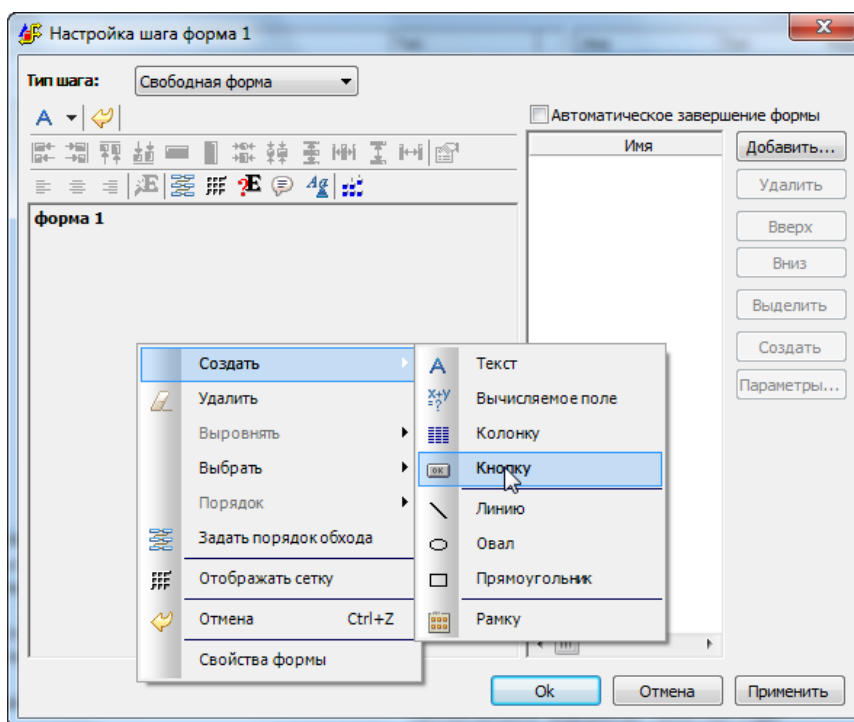


Рисунок 146 Добавление кнопки в форму действия

Используя пункт «Основные» окна свойств добавленной кнопки (Рисунок 147) можно выбрать действие, выполняемое по данной кнопке.

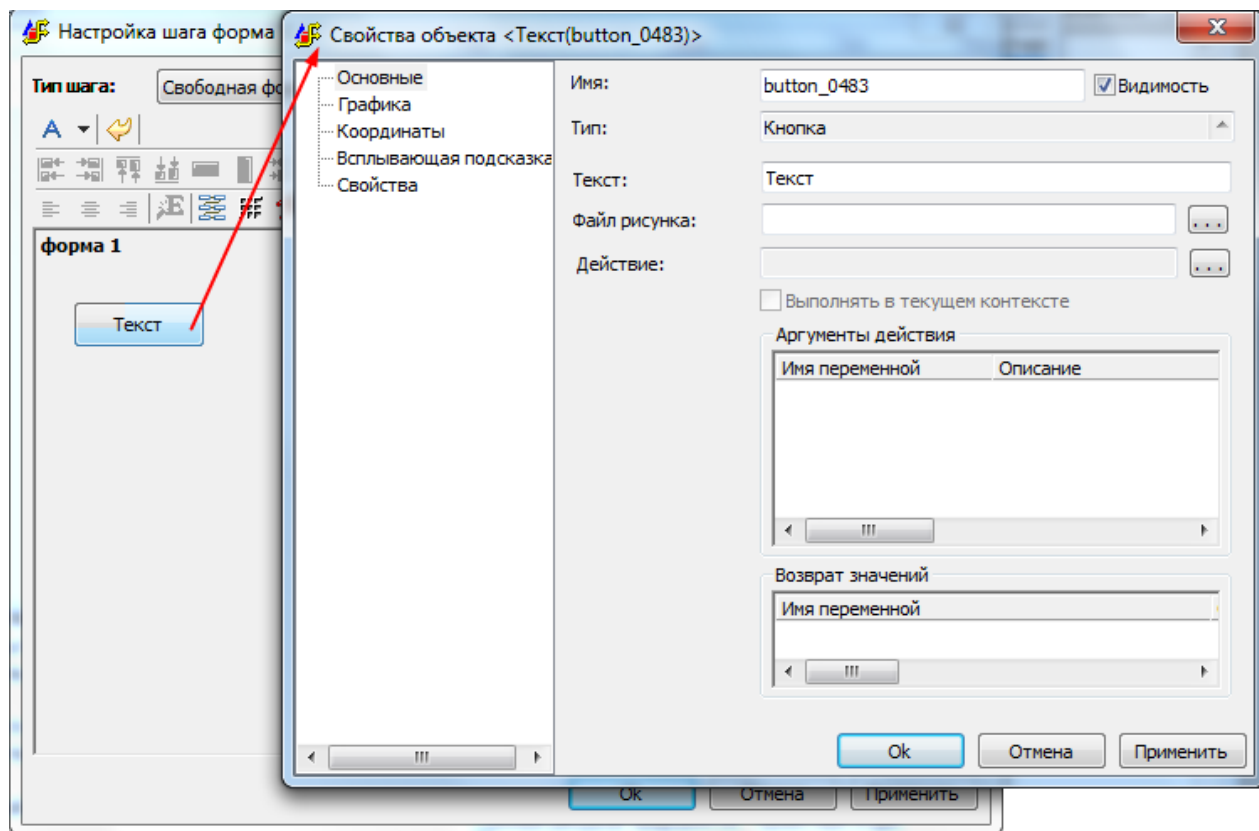



Рисунок 147 Задание свойств кнопки, добавленной в форму действия

Для выбора выполняемого действия используйте кнопку , расположенную справа от поля «Действие». В поле «Текст» введите отображаемый на кнопке текст, при необходимости можно выбрать файл рисунка. Переменным запускаемого по нажатию данной кнопки действия можно передать исходные значения (то есть, значения до выполнения запускаемого действия), для чего используется контекстное меню поля «Аргументы действия».

Результат выполнения запускаемого по кнопке действия (то есть, значения переменных запускаемого действия после его выполнения) можно вернуть в переменные исходного действия, для чего используется контекстное меню поля «Возврат значений».

Запускаемое по кнопке действие может выполняться «В текущем контексте» – то есть, при выполнении такого действия запись результата его выполнения в БД производится после завершения выполнения основного действия (либо выполнением функции «Update»). Если запускаемое по кнопке действие выполняется вне текущего контекста, запись результатов его выполнения в БД будет выполнена по завершении выполнения запускаемого по кнопке действия. Для выполнения действия в текущем контексте, после добавления действия включите флажок «Выполнять в текущем контексте».

17.4 Использование шагов типов «Окно выбора объектов из дерева» и «Окно выбора объектов из списка»

Данный инструмент предназначен для обеспечения удобного выбора пользователем одного или сразу нескольких информационных объектов. Шаг «Окно выбора...» обеспечивает следующие возможности:

- отобразить дерево или список объектов, сформированные по заданным условиям;

- выбирать один или сразу несколько объектов;
- ограничить возможность выбора по заданным условиям;
- по умолчанию отобразить дерево полностью раскрытым;
- по умолчанию выделить заданный объект

Результат выбора возвращается:

- для окна с единичным выбором – в переменную типа «Объект» или «Строка» (в строку возвращается ID выбранного объекта);
- для окна с множественным выбором – в переменную типа «Строка» (в строку записываются ID выбранных объектов, разделенные запятой; длина строки не должна превышать 2000символов).

Для создания шага «Окно выбора...» поместите курсор в любое место шага, в контекстном меню левой части окна выберите «Настройка шага...», в поле «Тип шага» открывшегося окна настройки шага укажите «Окно выбора объектов из дерева» или «Окно выбора объектов из списка». Если выбран тип «Окно выбора объектов из дерева» требуется также указание переменных для корневого объекта и типа связи (для построения дерева, из которого будет осуществляться выбор) (Рисунок 146).

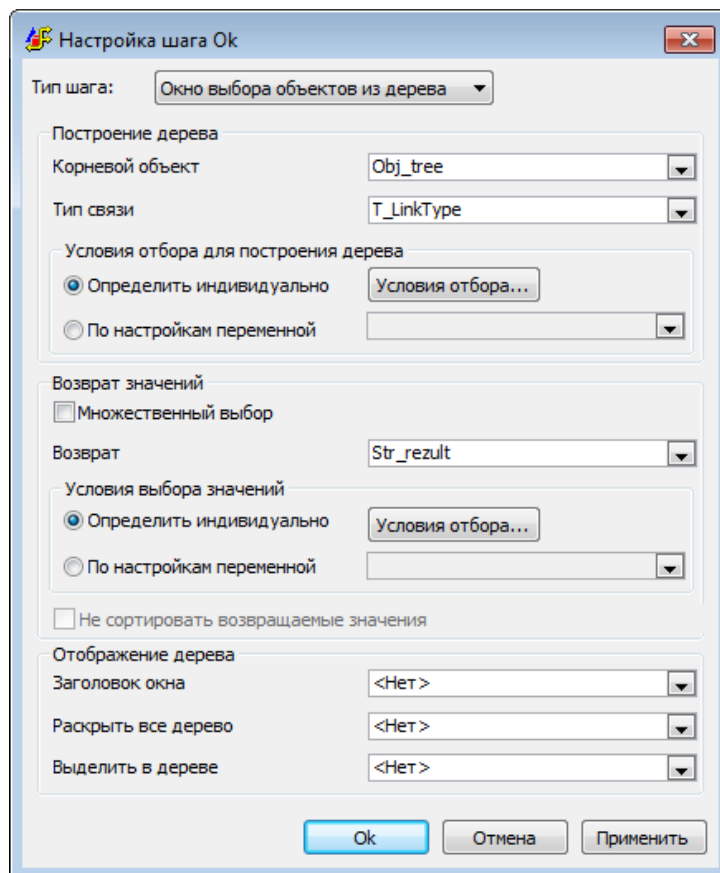


Рисунок 148 Окно настройки шага типа «Окно выбора»

Условия отбора для построения дерева можно:

- не определять (переключатель в положении «определить индивидуально», но условия отбора не заданы) – в этом случае в окне выбора будут отображаться все объекты-потомки (все уровни), входящие в корневой объект по указанному типу связи

- определить индивидуально – при положении переключателя «Определить индивидуально» нажать кнопку «Условия отбора...», в открывшемся окне указать условия отбора объектов для построения дерева
- определить по настройкам переменной – при положении переключателя «По настройкам переменной» указать переменную типа «Объект», условия отбора для которой будут использоваться при построении дерева в окне выбора

При включении флажка «Множественный выбор» в окне выбора будет отображаться накопитель, куда помещаются выбранные объекты.

При включенном флажке «Не сортировать возвращаемые значения» выбранные значения будут переданы в переменную, указанную в поле «Возврат значений» в том порядке, в котором они помещались в правую область окна выбора («накопитель»).

Условия выбора значений, то есть, условия, при которых объект, отображаемый в дереве окна выбора, может быть выбран, задаются аналогично условиям построения дерева.

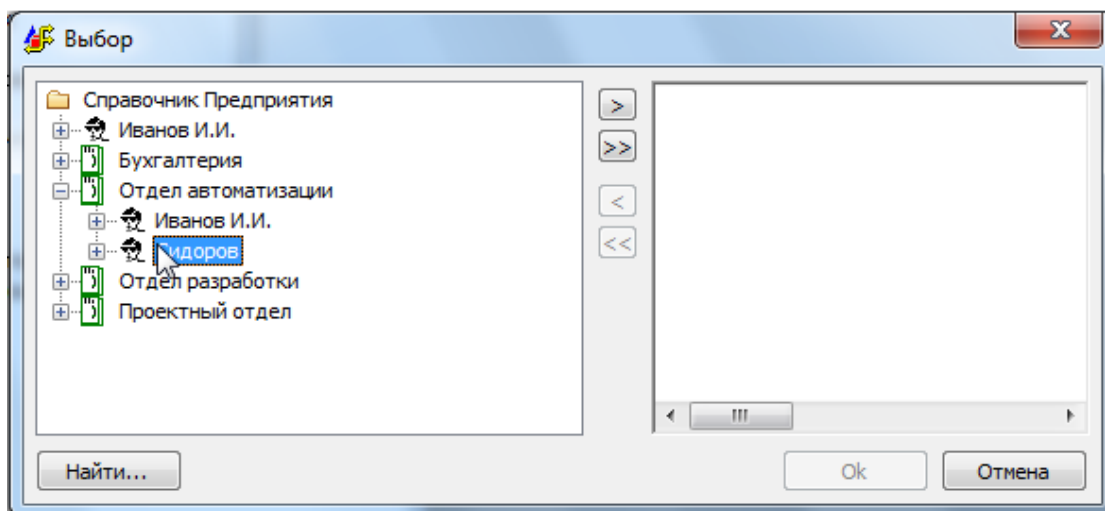
Заголовок окна выбора можно определить в виде выражения, или указав соответствующую строковую переменную.

Необходимость раскрытия всего дерева связей в окне выбора можно указать явно, выбрав значение «<Да>» или «<Нет>», задать в виде выражений или указанием соответствующей строковой переменной. Если необходимо раскрывать все дерево, выражение (или указанная переменная) должно вернуть значение «Y».

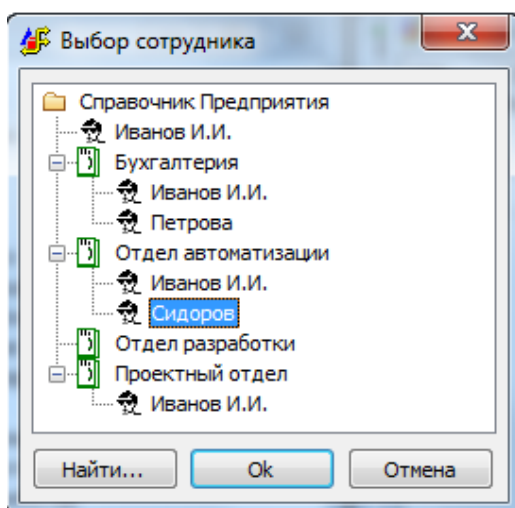
Если требуется выделить в дереве (или списке выбора) объект, необходимо указать соответствующую переменную типа объект.

Для примера, укажем в качестве корневого объекта справочник предприятия, тогда окно выбора, открываемое перед выполнением данного шага, может выглядеть следующим образом – см. рисунок 149

1



2



3

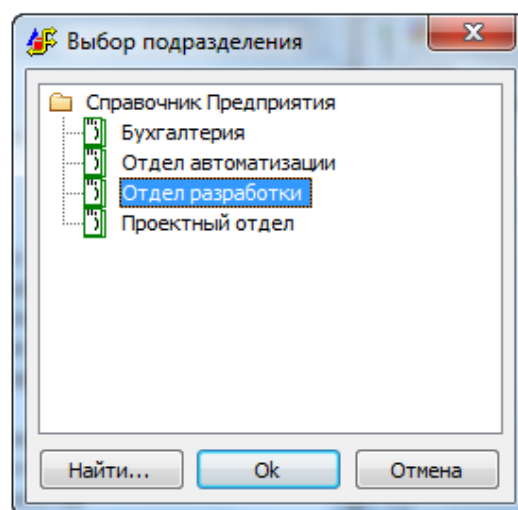


Рисунок 149 Окно выбора исполнителей. Примеры

Пояснения к рисунку:

- 1 – включен флажок «Множественный выбор», в правой части окна отображается накопитель. Заголовок окна не задан, в поле «Раскрыть все дерево» установлено «<Нет>»;
 - 2 – выключен флажок «Множественный выбор», задан заголовок окна «Выбор сотрудника», в поле «Раскрыть все дерево» установлено «<Да>»;
 - 3 – выключен флажок «Множественный выбор», условия отбора для построения дерева определены в виде:
- | Тип | Атрибут | Опер | Значение |
|--------|---------|------|-----------|
| Объект | Тип | <> | Сотрудник |
- , указан заголовок окна «Выбор подразделения».

Пример настройки окна выбора объектов из списка – см. рисунок 150.

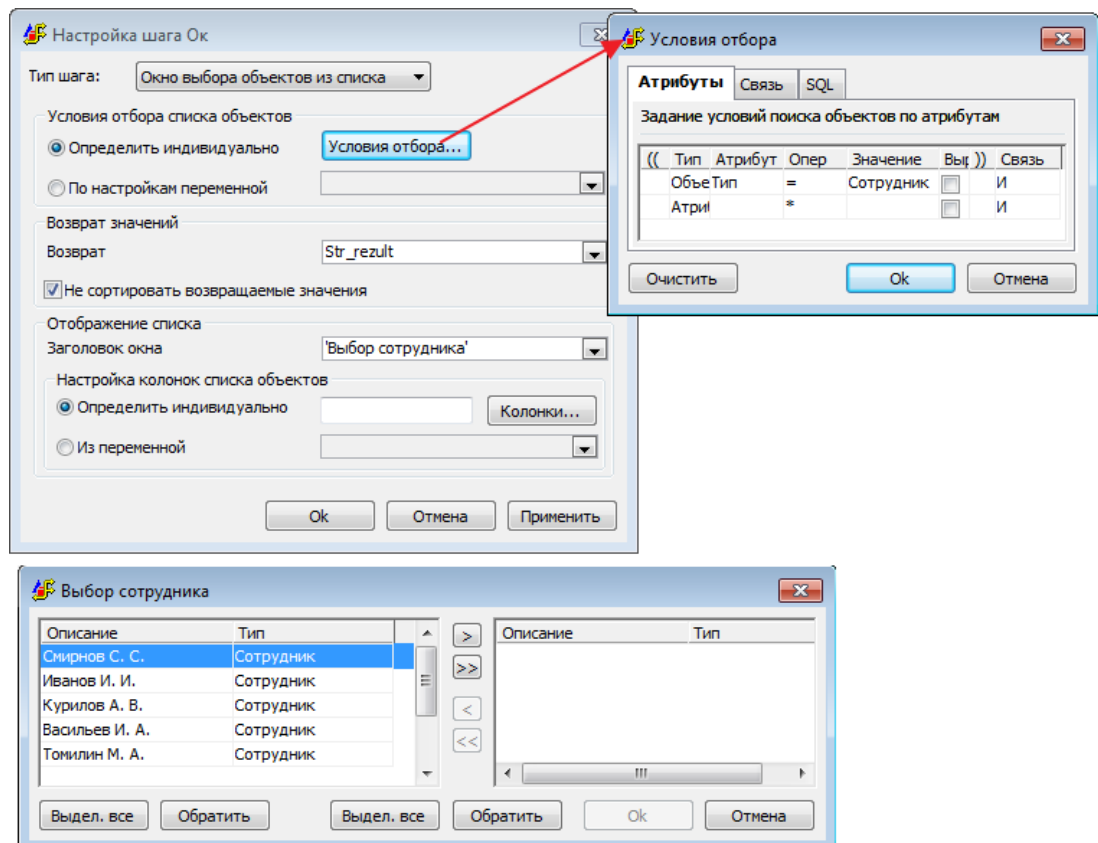


Рисунок 150 Пример (вверху) и результат (внизу) настройки окна выбора объектов из списка

Далее приведем несколько практических замечаний по применению окон выбора.

В большинстве случаев требуется отображение в форме действия более одного поля, с возможностью редактирования одного или нескольких полей из окна выбора. В этом случае окно выбора открывается другим действием, запускаемым по кнопке (или кнопкам) в форме основного действия (см. п. 17.3.5). В этом случае в форме действия обычно требуется отобразить результат выбора не в виде строки идентификаторов, а в более удобном для пользователя виде. Результат выбора в форме отображается вычисляемым полем, с использованием функции `f_ExecSQLSelect_3`. Выражение вычисляемого поля может быть, например, таким:

`f_ExecSQLSelect_3 ('Select description from LSDBO.object_reference_view where LSDBO.object_reference_view.ID in ('+str_result+)', ", '~r~n', 'cols=char(250)', 10)` где:

`str_result` – переменная действия (тип – «Строка»), указанная для возврата значения из окна выбора (Рисунок 146). Приведенный в первом аргументе функции SQL-запрос вернет описания объектов, идентификаторы которых указаны в переменной `str_result`.

Аналогичные вычисляемые поля могут использоваться в атрибутивных формах и отчетах, что позволяет в удобном пользователю виде показывать значения атрибутов, содержащие идентификаторы объектов.

Часто требуется выполнить обработку выбранных объектов, идентификаторы которых содержатся в строке, указанной для возврата значения из окна выбора. Данная задача может успешно решаться путем циклической обработки указанного строкового значения (см. п. 17.6).

17.5 Использование и получение идентификаторов (кодов ID) в действиях. Присвоение значений сущностям, имеющим ID

Часто при написании действий такие сущности, как тип объекта, атрибут, тип связи и т.п. присваиваются переменным по умолчанию, и в дальнейшем указанные переменные используются в действии в качестве констант. При необходимости реализации однотипных действий, например, для создания объектов различных типов или установки различных атрибутов, действия приходится копировать и изменять значения констант. Количество действий при этом возрастает, их администрирование затрудняется. Подобные однотипные действия часто можно заменять общими универсальными, где указанные сущности определяются.

Использование в формах действий окон выбора для таких сущностей, как: типы объектов, отчеты, экземпляры объектов (за исключением случаев их выбора из дерева или использования условий отбора) и т.д. в большинстве случаев не является оптимальным, этого рекомендуется избегать. Вместо самих указанных сущностей, в формах действий обычно определяются их ID (используются переменные с типом значения «Число», либо «Строка» с последующим преобразованием в число). При настройке форм действий, для таких переменных, как правило, используются стили редактирования «Выпадающий список», «Переключатель», «Флажок», что делает работу конечного пользователя вполне комфортной. Разумеется, в форме вместо значений ID отображают понятные пользователю названия (пример такого действия – см. пп. 17.11.2, 17.11.4).

При запуске указанных действий с использованием аргументов действия, удобно передавать в действие конкретные значения ID в качестве аргументов, используя таким образом единственное действие, например, для выполнения различных отчетов, установки различных атрибутов и т.п.

При выполнении действия, функцией SetByID по идентификатору устанавливается значение переменной, тип которой соответствует данной сущности.

Функция SetByID используется также если требуется присвоить значение одной сущности, имеющей ID, другой такой же сущности, например: значение одной переменной типа «Объект» (a_Object) присвоить другой переменной (Object_1) того же типа:

```
Object_1 = SetByID(a_Object)
```

При таком использовании функции SetByID, её аргумент (в приведенном примере – a_Object) следует вводить, используя редактор выражений.

Для получения значения идентификатора конкретной сущности Lotsia PDM можно скопировать его через буфер обмена с вкладки «Основные» окна свойств данной сущности (Рисунок 151).

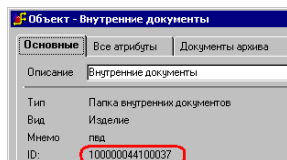


Рисунок 151. Идентификатор объекта может быть скопирован с вкладки «Основные»

Чтобы получить идентификатор автоматически при выполнении действия следует использовать функцию Set. Например, чтобы присвоить числовой переменной ID_obj значение, равное идентификатору текущего объекта, используется выражение:

```
ID_obj = Set(a_Object)
```

Аргумент `a_Object` (или другая переменная, соответствующая сущности имеющей ID) вводится с использованием редактора выражений.

Зачастую наиболее приемлемым способом получения требуемого идентификатора при выполнении действия бывает использование SQL-запроса (см. п. 24.3.3.3).

17.6 Использование меток шагов и функции GoTo. Циклическая обработка в действиях над объектами

Для организации циклической обработки внутри действий над объектами для возврата к началу цикла и выхода из цикла используется функция `GoTo`.

Аргументом функции `GoTo` может быть выражение, возвращающее строку, являющуюся меткой шага, к которому осуществляется переход, либо строковая переменная, которой было заранее присвоено значение, совпадающее с меткой шага.

Например, пусть для шага, к которому требуется перейти, установлена метка «начало» (без кавычек, допускаются русские и латинские буквы, цифры, подчеркивания).

Имеется строковая переменная `metka_1`. Для перехода к шагу следует использовать функции:

```
metka_1 = Set('начало')  
GoTo (metka_1)
```

Или:

```
GoTo('начало')
```

Отметим, что в зависимости от конкретной задачи, можно использовать выражения как внутри функции `Set`, так и внутри `GoTo`.

Присвоить значение переменной `metka_1` можно и другими способами, например, используя форму шага действия (см. п. 17.3).

Проверка условия выхода из цикла может осуществляться, например, с использованием функций `If`, либо `Case`. Проверка условий выполняется внутри выражений действия над объектами, например:

```
GoTo(If(...));  
...=Set(If(...));  
и других.
```

В зависимости от конкретного действия, могут использоваться различные способы организации циклической обработки, например:

- Использовать счетчик цикла. При выполнении очередного шага цикла вычисляется очередное значение счетчика, которое может быть использовано, в том числе для нахождения очередного обрабатываемого объекта. Для нахождения очередного объекта согласно условиям отбора в таких случаях используются формы шагов действий с автозавершением, либо SQL-запросы (см. 24.3.3.3);

- При наличии строки, состоящей из значений, указанных через разделитель, и позволяющих однозначно идентифицировать, например, объекты с помощью условий отбора или функции SetByID, можно использовать строковые функции для «вырезания» из подобной строки очередного значения. После обработки полученного таким образом значения, в следующем цикле «вырезается» очередное значение и обработка повторяется. Получить подобную строку можно по-разному, например, обработкой массива значений средствами Workflow (см. п. 23.20.4), выполнением SQL-запроса (см. 24.3.3.3), использованием окон выбора (см. п.17.4) и другими способами. Указанная строка может быть записана, например, в атрибут объекта и при необходимости автоматически считана в ходе выполнения действия. Следует учитывать, что если строка записывается в атрибут объекта, её длина не может превышать 2000 символов, если строка используется только внутри действия и не показывается в формах, то её длина может достигать, в различных случаях, 32000 или 64 000 символов;
- В некоторых случаях в использовании подобной строки нет необходимости. Очередной объект или другой элемент БД можно получать, используя SQL-запрос (см. п. 24.3.3.3) внутри цикла. В таком запросе для получения очередного единственного значения, следует применять агрегатные функции, такие как min, max и др. Подобный запрос в очередном цикле должен возвращать один очередной, еще не обработанный искомым элемент. Следует учитывать, что при каждом выполнении запроса осуществляется обращение к БД, что дополнительно нагружает сервер БД, как правило, такое действие выполняется значительно дольше.

Обработка объектов или строковых фрагментов, выполняемая в цикле, может быть, в зависимости от решаемых задач, различной, например: нахождение соответствующего объекта и его использование – получение и/или установка значения атрибута, установка и/или удаление связи, копирование найденного объекта по шаблону, импортирование документа, отправка сообщения, запуск работы и т.д.

17.7 Работа с NULL – значениями

Под Null-значением переменной понимается отсутствие у данной переменной значения. Пустая строка для строковой переменной или 0 для числовой не являются Null-значениями. Присвоить Null-значение – то же самое, что удалить у переменной значение.

Правильное использование Null-значений при написании действий в некоторых случаях позволяет значительно упростить действие и/или сделать его более универсальным.

Функцию AttribSet можно использовать как для установки, так и для удаления атрибута. В последнем случае аргумент функции, передающий значение атрибута, должен иметь значение Null. Таким образом, если соответствующей переменной было присвоено значение, атрибут будет установлен, а если значение не присвоено – атрибут будет удален. Таким образом, в ряде случаев удастся избежать создания отдельного шага действия с функцией AttribDel.

В некоторых случаях требуется присвоить переменной значение Null. Использовать для этой цели функцию редактора действий Null не всегда удобно, так как если требуется присвоение Null-значения осуществлять при определенных условиях, потребуется создание дополнительного шага действия и использование условного перехода. Однако,

возможно использование в действиях выражений, при выполнении которых производится присвоение как Null, так и не Null-значений, в зависимости от выполнения условий. Например, выражение:

```
var= Set(if(a = b, var_1, null_var))
```

присваивает переменной var значение var_1, или null_var, в зависимости от выполнения условия a = b. В данном примере – var, var_1, null_var переменные одинакового типа (строки, числа). Если у переменной null_var значение Null (то есть, значение не присвоено), при невыполнении условия a = b переменной var также будет присвоено Null, а при выполнении условия – переменной var будет присвоено значение var_1.

Отметим, что переменная null_var будет иметь Null-значение в случае, если ей не задано значение по умолчанию и не присвоено значение еще каким-либо образом: не передано в качестве аргумента, не указано в форме, не установлено функциями редактора действий. Также Null-значение можно присвоить явно, добавив в действие отдельную строку:

```
Null (null_var)
```

Отметим следующую особенность: если в действие добавить строку, которая напрямую, без использования вложенных функций присваивает переменной Null-значение:

```
Var = Set(null_var),
```

соответствующий шаг действия выполняться не будет (если null_var имеет значение Null), будет выдано сообщение об ошибке.

Чтобы не допускать подобных сообщений, во всех случаях, когда Null – значение на практике может иметь место, но его появление недопустимо с точки зрения правильного выполнения данного действия, следует использовать соответствующую проверку значения переменной, например:

```
GoTo(if (IsNull(var),'end', ''))
```

В результате такой проверки, в случае, если переменная var примет Null-значение, будет осуществлен переход к шагу, имеющему метку «end» (в метке шага кавычек быть не должно). Если же значение переменной var не Null, выполнение действия будет продолжено.

17.8 Вложенные действия

Помимо использования кнопок в форме действий (см. п. 17.3.5), имеется возможность автоматически запускать одно действие из другого (то есть, «вкладывать одно действие в другое»). При этом глубина вложенности не ограничивается. Данная возможность значительно упрощает создание и отладку сложных действий, делает возможным заимствование ранее написанных действий во вновь создаваемые.

Вложенное действие добавляется к шагу действия. Для добавления вложенного действия установите курсор в любой строке выбранного шага, в контекстном меню выберите «Свойства шага...». В открывшемся окне выберите тип шага – «Действие» (Рисунок 152).

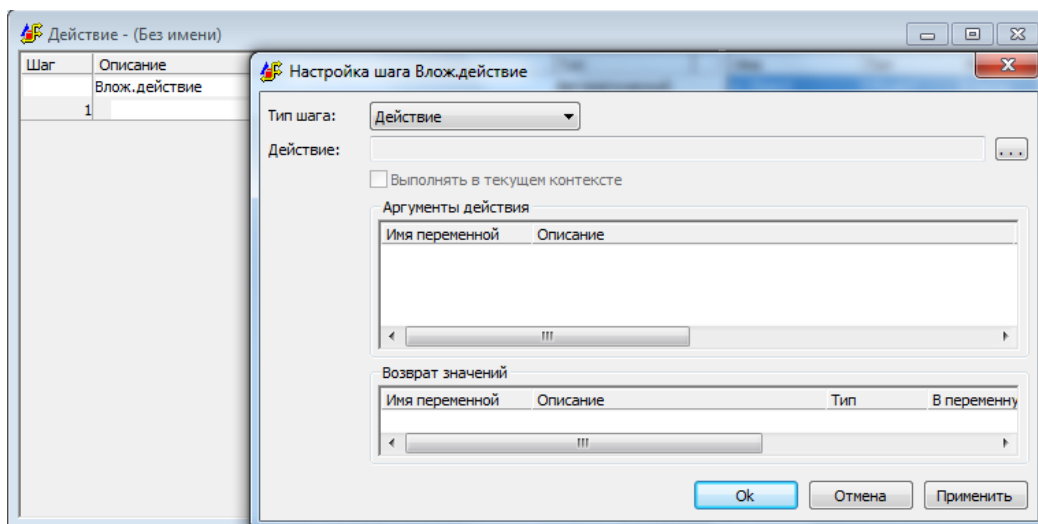



Рисунок 152. Настройка шага с вложенным действием

Вкладываемое действие выбирается из списка действий, для открытия которого нажмите кнопку , расположенную справа от поля «Действие».

Переменным вложенного действия можно передать исходные значения (то есть, значения до выполнения вложенного действия), для чего используется контекстное меню поля «Аргументы действия».

Результат выполнения вложенного действия (то есть, значения переменных вложенного действия после его выполнения) можно вернуть в переменные исходного действия, для чего используется контекстное меню поля «Возврат значений».

Вложенное действие может выполняться «В текущем контексте» – то есть, при выполнении такого вложенного действия запись результата его выполнения в БД производится после завершения выполнения основного действия (либо выполнением функции «Update»). Если вложенное действие выполняется вне текущего контекста, запись результатов его выполнения в БД будет выполнена по завершении выполнения вложенного действия. Для выполнения действия в текущем контексте, после добавления действия включите флажок «Выполнять в текущем контексте».

Пример добавления вложенного действия – см. п.17.11.6.

17.9 Некоторые свойства и особенности действий над объектами

В отличие, например, от привязок к профилям пользователей атрибутивных форм, привязки действий к профилям и типам объектов не ограничивают возможность применения любым пользователем всего имеющегося набора действий, а служат для обеспечения удобства работы.

Действие может быть скрыто (то есть, не показываться по умолчанию в окне выбора действия) – это следует делать для действий, применяемых только в документообороте, либо выполняемых нажатием кнопки в атрибутивной форме, а также для действий, добавленных как пункт Главного или контекстного меню. Для скрытия действия следует открыть его в окне редактора действий, в верхнем меню «Файл» выбрать «Свойства», в окне свойств установить флажок «Скрытое». Скрытие действия, также как и отсутствие для данного действия привязки к профилю или типу объекта, само по себе не ограничивает возможность выполнения данного действия пользователем.

Чтобы исключить возможность запуска действий пользователями, следует полностью удалить из профилей данных пользователей пункты меню, используемые для запуска действий, обеспечив возможность использования только специально настроенных

для данного профиля пользователя инструментов, обеспечивающих запуск конкретных действий. Для конечных пользователей программы рекомендуется именно такой порядок использования действий.

Показ окна действия при его выполнении можно отключить, указав в поле «Окно» окна свойств действия (Рисунок 153) соответствующее значение.

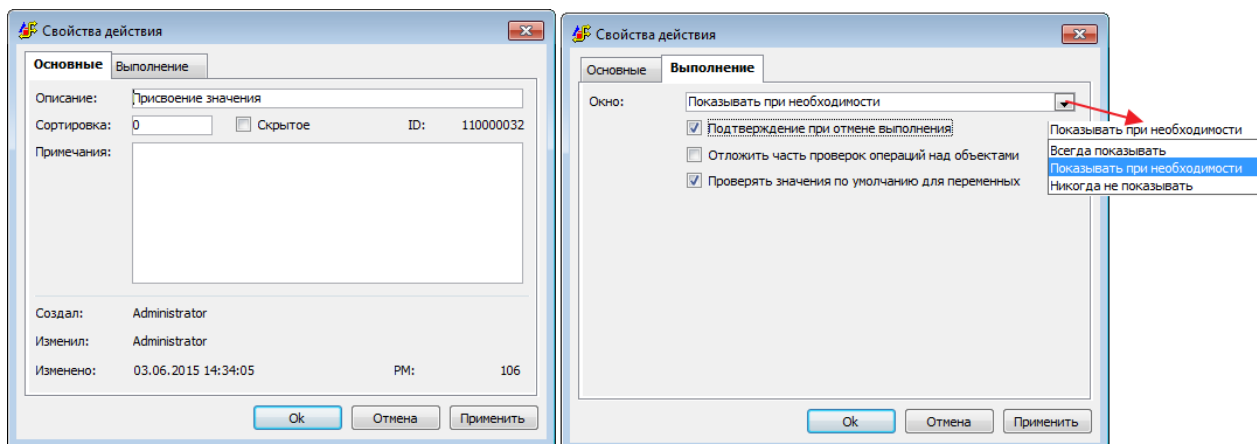


Рисунок 153 Окно свойств действия

Доступны следующие значения:

«Всегда показывать» – используется чаще при тестировании, чтобы быть уверенным, что действие выполняется;

«Показывать при необходимости» – окно действия показывается только при открытии формы шага действия, данное значение установлено по умолчанию;

«Никогда не показывать» – используется для действий без форм, как правило – для сервера автоматических этапов. Если в таком действии используется форма шага – будет выдано сообщение об ошибке;

При снятом флажке «Подтверждение при отмене выполнения» не запрашивается подтверждение при нажатии пользователем кнопки «Отмена» в форме шага выполняемого действия.

Флажки «Отложить часть проверок операций над объектами» и «Проверять значения по умолчанию для переменных» позволяют несколько ускорить выполнение действий, обеспечивая режимы быстрой загрузки и быстрого выполнения действий (при включенном первом и отключенном втором флажке). Данные режимы могут быть практически полезны для случаев, когда действие по локальной сети выполняется значительно быстрее, чем при использовании более медленного подключения, например, по VPN. Для отладки действий указанные режимы рекомендуется отключать. Более подробная информация по указанным режимам имеется в документации и в системе контекстной справки по Lotsia PDM PLUS.

При копировании действия, для созданной копии привязки исходного действия не сохраняются.

При копировании фрагментов действий, при наличии у копируемых шагов форм, копируются и формы этих шагов.

Для обеспечения возможности открытия в отдельном окне созданного в данном действии объекта, для учета внесенных в данном действии изменений при выполнении поиска объектов и др., следует после внесения указанных изменений, по завершении очередного шага выполнить функцию Update – записать изменения в БД. Изменения,

сделанные до выполнения данной функции, остаются в БД и в случае принудительного прекращения выполнения действия после выполнения данной функции.

При получении значений атрибутов (с использованием функций `AttribGetDefault`, `AttribGet`), полученные значения атрибутов запоминаются. Для ускорения работы программы, при повторном считывании значений тех же атрибутов будут использованы ранее сохраненные данные. Учитывая сказанное, при необходимости неоднократного (в ходе выполнения действия) получения текущих значений одного и того же атрибута (для одного и того же объекта (проекта)), следует использовать SQL-запросы (см. п. 24.3.3.3) либо вложенные действия, выполняемые вне контекста текущего действия.

17.10 Резюме

Действие над объектами Lotsia PDM – это процедура, автоматизирующая выполнение определенной последовательности операций пользователя (макрокоманда).

Для настройки действий используется редактор действий («Главное меню» → «Администрирование» → «Структура данных» → «Редактор действий»).

Действия состоят из шагов. В шагах содержатся функции действия. На шаге может располагаться метка, к которой возможен переход. Шагу может сопоставляться форма, окно выбора, вложенное действие, скрипт.

В действиях используются переменные различных типов, данные переменные могут использоваться и в формах действий. Имеется стандартный набор предопределенных переменных, может создаваться произвольное количество пользовательских переменных.

Запуск действий на выполнение может осуществляться выбором требуемого действия из списка действий, либо с использованием настроенного пункта меню или кнопки инструментальной панели, либо по кнопкам в различных формах, либо по событию при выполнении предопределенного процесса, либо автоматически при старте программы.

В действиях возможна циклическая обработка данных, использование SQL-запросов.

Формы шагов настраиваются как средствами редактора форм, так и заданием свойств соответствующих переменных, используемых в форме.

Присвоение значений переменным в форме действия может осуществляться ручным вводом, с использованием списков значений и классификаторов, настраиваемых окон выбора объектов, переключателей, флажков, окон выбора файлов или папок. Можно использовать в формах действий кнопки для запуска других действий, с возвратом результатов их выполнения в поля формы.

Шаги типов «Окно выбора объектов из дерева» или «Окно выбора объектов из списка» могут обеспечивать выбор одного или нескольких значений одновременно.

Имеется возможность формирования в ходе выполнения действия списков возможных значений переменных, с последующим выбором пользователем значений из указанных списков.

Окна выбора в формах действия могут настраиваться, настройки данных окон могут быть фиксированными, или изменяться путем формирования значений переменных.

Привязки действий к профилям пользователей ограничивают списки действий, которые по умолчанию видит данный пользователь. Указанные привязки не являются ограничением возможности запуска пользователем любого из имеющихся действий.

17.11 Примеры (упражнения по написанию) действий

Далее рассмотрим несколько примеров создания действий, которые рекомендуется воспроизвести в учебной настройке. Подобные действия или их элементы обычно используются в настройках системы, применительно к различным предметным областям.

17.11.1 Простейшее действие по открытию объекта

Данное действие будет открывать заданный объект, например, пусть это будет объект «Информационные ресурсы» – корневая папка, куда входят все основные проекты.

Откройте окно редактора действий (см. п. 17.1). Добавьте переменную, соответствующую открываемому объекту: в контекстном меню правой части окна редактора действий выберите «Добавить», в открывшемся окне свойств переменной определите свойства переменной в следующем виде:

Имя – Obj_1;

Описание – Открываемый объект;

Тип – объект;

Включите флажок «Значение», в поле «Значение» укажите требуемый объект;

После нажатия «Применить» окно свойств переменной будет выглядеть примерно так – см. рисунок 154.

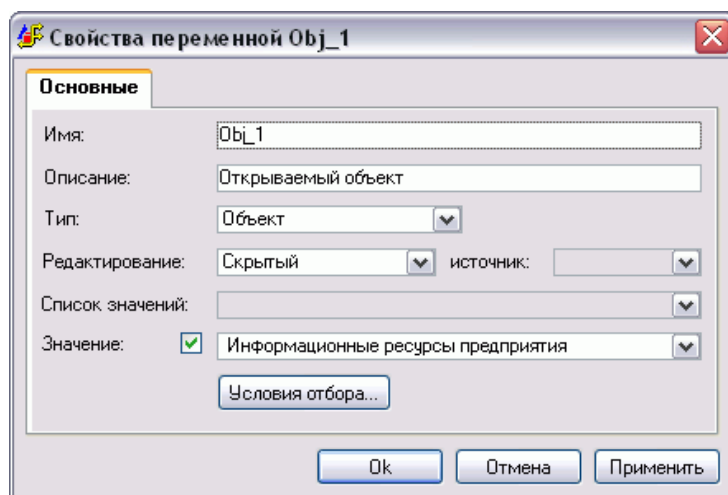


Рисунок 154 Окно свойств добавленной переменной

Нажмем «Ok» для завершения добавления переменной.

Окно с заданным по умолчанию объектом будем открывать по связи «Дерево проектов», соответствующая переменная уже имеется – это предопределенная переменная «a_LinkType», её свойства – см. рисунок 155.

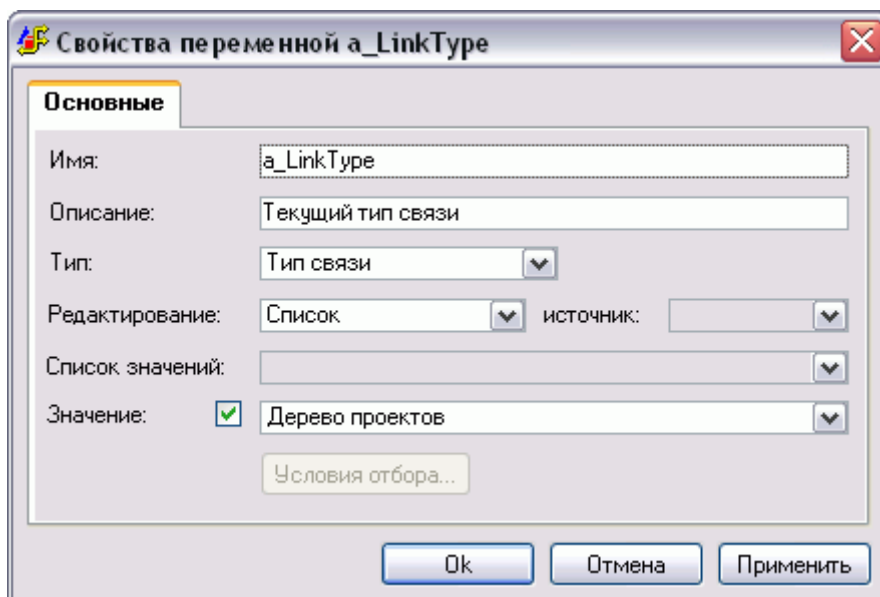


Рисунок 155 Окно свойств переменной «a_LinkType».

Теперь добавьте функцию, выполняющую открытие окна дерева связей.

В контекстном меню левой части окна редактора действий выберите «Вставить функцию» (Рисунок 156).

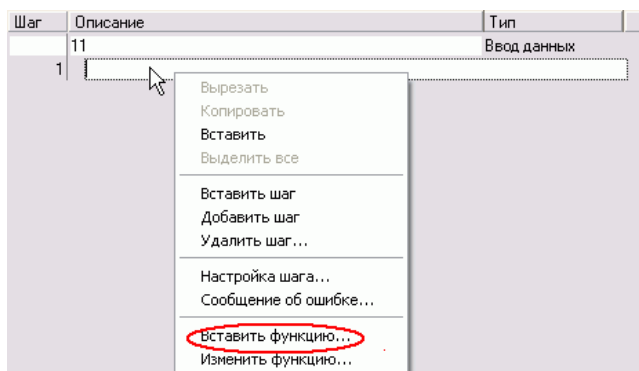


Рисунок 156 Начало добавления функции

В открывшемся окне выберите функцию «WndTree», в нижней части окна укажите открываемый объект и тип связи (Рисунок 157).

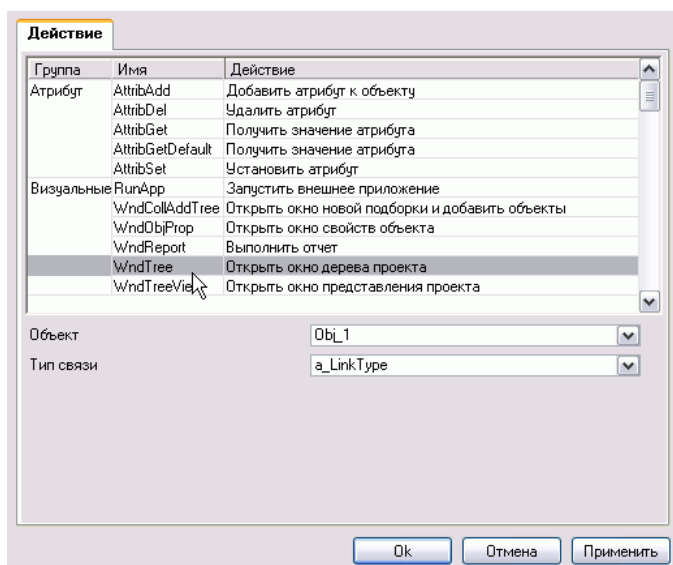


Рисунок 157 Добавление функции в действие – продолжение.

По нажатию «Ok» в действие будет добавлена строка:

WndTree (Obj_1 , a_LinkType)

Введите описание шага действия (например, «Открываем окно дерева проектов»). Нажмите <Ctrl>+<s> для сохранения действия, в открывшемся окне «Свойства действия», в поле «Описание» введите: «Открытие объекта», нажмите «Ok». Действие сохранено.

Для проверки работы действия: при текущем окне действия, в верхнем меню «Действие» выберите «Выполнить» – будет открыто соответствующее окно дерева проектов.

В Lotsia Workflow данное действие можно использовать и для открытия различных объектов, а не только заданного по умолчанию для переменной Obj_1. Для этого переменной Obj_1 достаточно передать в качестве входящего значения соответствующий документ шаблона – см. п. 23.16.5.

В большом числе практических случаев использование документов шаблона не является возможным (или самым удобным) решением. Чтобы осуществлять открытие различных объектов, подключая данное действие к кнопкам атрибутивных форм, событиям работ Lotsia Workflow, пунктам Главного или контекстного меню, данное действие следует усовершенствовать, обеспечив возможность указывать в качестве входящего параметра действия не объект, а число – идентификатор открываемого объекта.

Потребуется, действуя аналогично добавлению переменной Obj_1, добавить в действие еще одну переменную – id_Obj_1 – идентификатор открываемого объекта (Рисунок 158).

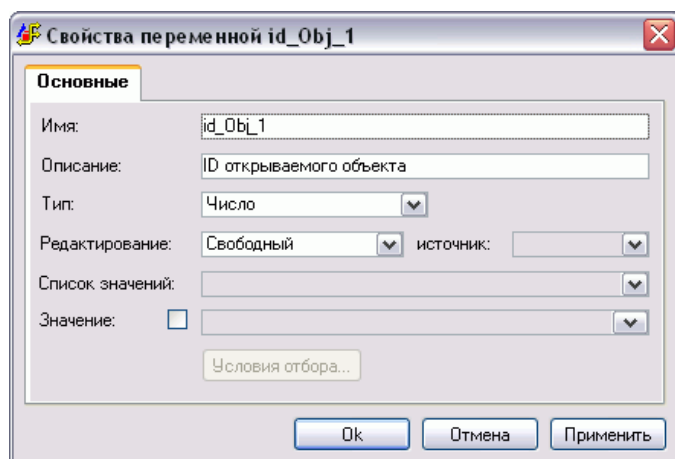


Рисунок 158 Свойства добавляемой переменной «id_Obj_1»

Далее, в левой части окна редактора действий, перед имеющейся функцией открытия дерева проектов, следует вставить функцию, определяющую открываемый объект по его идентификатору. Для этого:

Установите курсор в левую часть окна редактора действий, в строку с имеющейся функцией WndTree. В контекстном меню выберите «Вставить функцию...», выберите в списке функций «SetByID», установите параметры согласно рисунку (Рисунок 159).

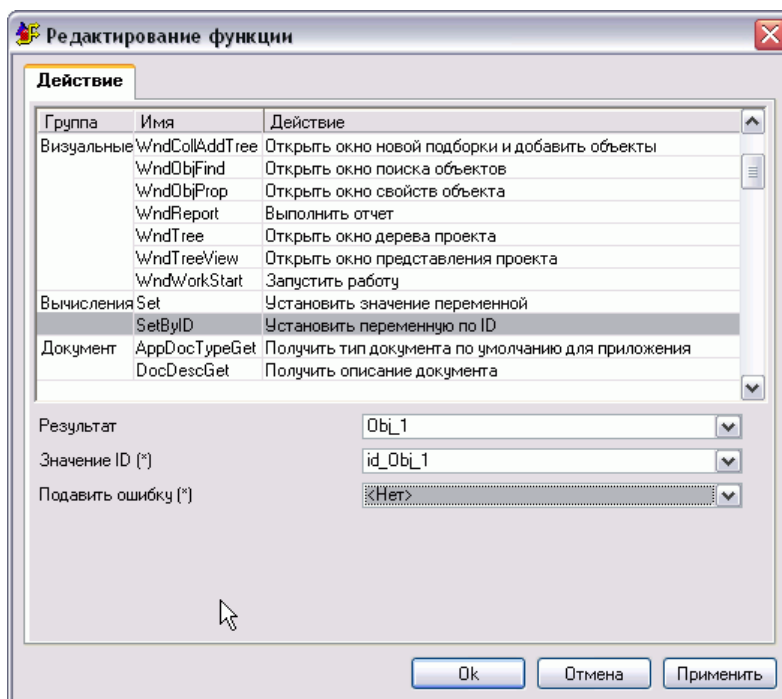


Рисунок 159 Добавление функции SetByID в действие

Нажмите «Ok» для добавления строки в действие. Окончательный вид действия – см. рисунок 160.



Рисунок 160 Окончательный вид действия

Для обеспечения работы действия необходимо присвоить значение ID объекта переменной id_Obj_1. Конечно, можно установить это значение для данной переменной по умолчанию (аналогично тому, как это делалось выше, для переменной Obj_1), но одной из наиболее востребованных будет возможность использования данного действия в качестве пунктов меню или выполнения его по нажатию кнопок в формах, для открытия различных объектов. Пример добавления действия в Главное меню – см. рисунок 51. В качестве аргумента действия следует указать переменную id_Obj_1 и присвоить ей требуемое значение.

Аналогичное действие можно использовать и для выбора открываемого объекта с помощью выпадающего списка или переключателя – см. п. 17.11.2.

17.11.2 Выбор объекта переключателем

Пусть имеется список проектов Lotsia PDM, которые часто требуется открывать, например: «Служебные», «Информационные ресурсы Предприятия», «Проекты в разработке», «Внутренние документы».

Создадим действие, которое бы позволяло выбрать открываемый объект из перечисленных и открыть его.

Переменные действия и функции, используемые в действии, идентичны переменным и функциям, рассмотренным при создании действия по открытию объекта (см. п. 17.11.1).

В действии используется 2 переменных: Obj_1 – тип данных: объект и Id_obj_1- тип данных: «Число», редактирование: «Свободный». Добавим в форму шага действия (контекстное меню левой части окна редактора действий для первого шага → «Настройка шага...» → в открывшемся окне указать тип шага «Свободная форма» → в контекстном меню правой части окна выбрать «Добавить» → выбрать переменную Id_obj_1). Результат – см. рисунок 161.

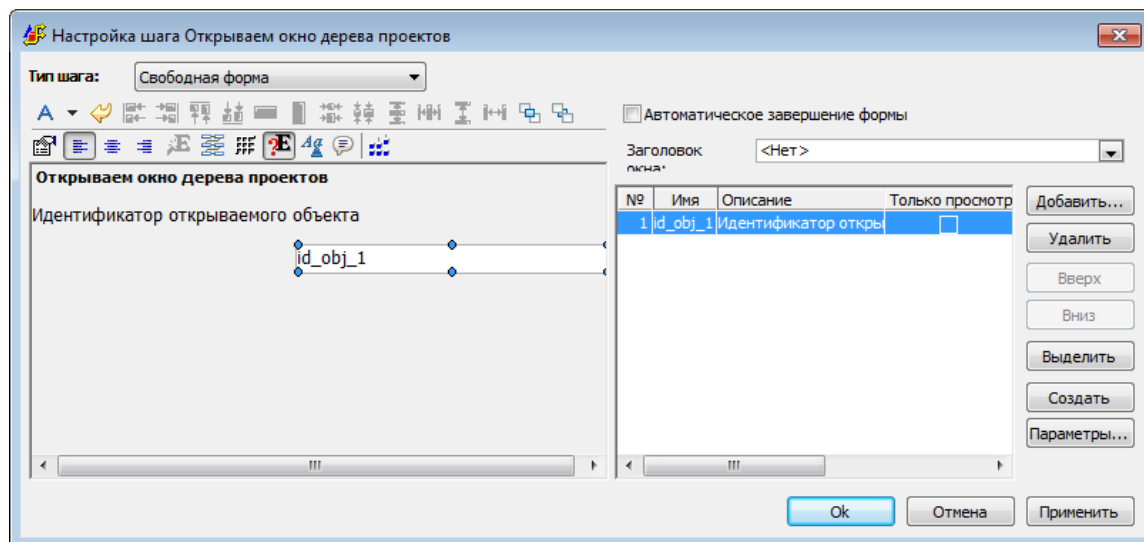


Рисунок 161 Добавлена переменная в форму действия

Дважды щелкните левой клавишей мыши по колонке Id_obj_1 в левой части окна редактора форм. Выберите пункт «Редактирование» открывшегося окна свойств объекта выберите стиль – «Переключатель», пользуясь контекстным меню окна, добавьте возможные значения переменной Id_obj_1 (Рисунок 162).

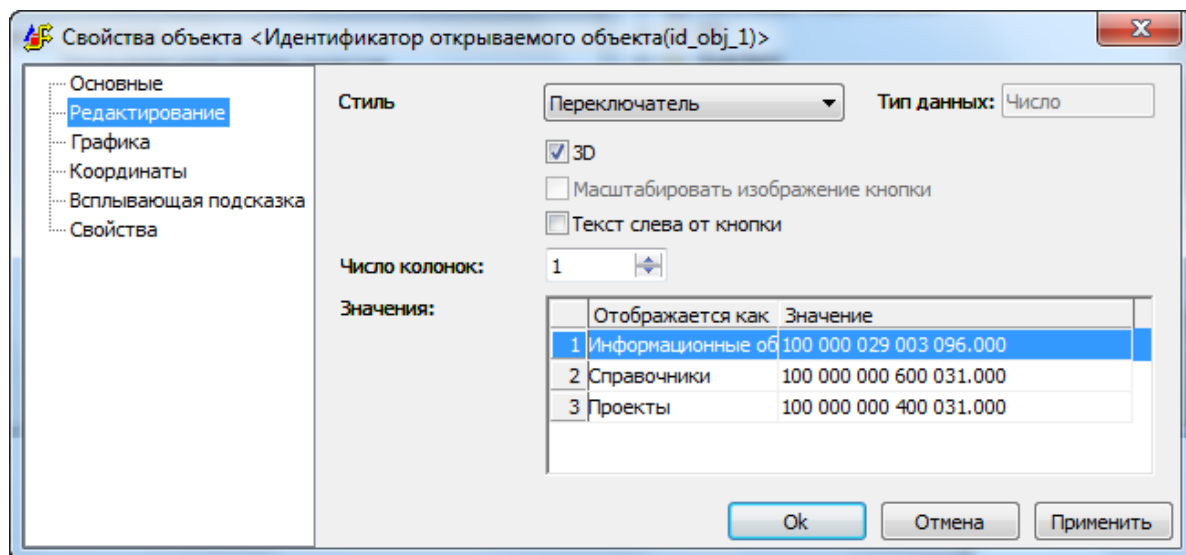


Рисунок 162 Сформирован список возможных значений переменной

В колонку «Отображается как» вводится значение, которое будет отображено в форме, в колонку «Значение» – ID соответствующего объекта. Для ввода ID удобно открыть еще одну копию приложения Lotsia PDM PLUS и копировать данные через буфер обмена. Значения идентификаторов объектов в Вашей БД вероятнее всего не будут такими, как на рисунке.

После сохранения свойств переменной, формирования соответствующих текстовых полей и настройки графики, форма приобретает окончательный вид (пример – см. рисунок 163).

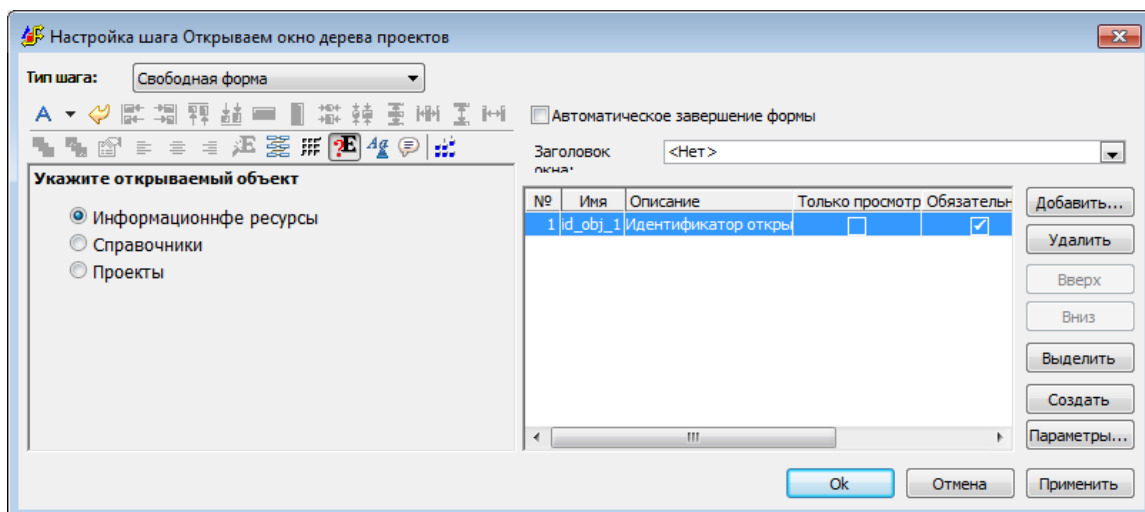


Рисунок 163 Созданная форма шага действия

Сохраните форму, добавьте в действие строки, если это не было сделано ранее:

```
obj_1 = SetByID (Id_obj_1)
```

```
WndTree (obj_1, a_LinkType)
```

Первая строка присваивает открываемому объекту значение, соответствующее выбранному в форме идентификатору, вторая – открывает объект.

Сохраните действие. Не закрывая окна редактора действий, в верхнем меню выберите «Действие» → «Выполнить» и убедитесь в правильной работе действия.

17.11.3 Действие по добавлению объекта служебной записки

Рассмотрим создание действия «Добавление служебной записки», которое будет добавлять объект служебной записки к текущему объекту.

Добавим переменные:

Obj_sz – объект служебной записки;

Type_Obj_sz – тип создаваемого объекта;

Opisanie – строка описания

(см. рисунок 164).

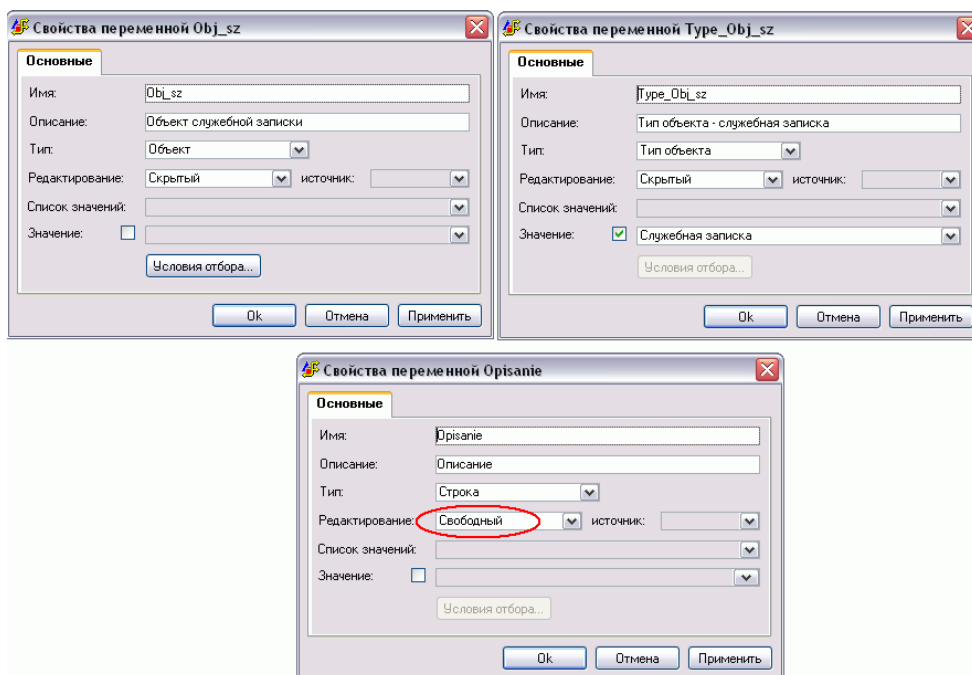


Рисунок 164 Переменные, добавляемые в действие

В поле «Редактирование» для переменной «Opisanie» необходимо указать «Свободный». Тип объекта «Служебная записка» должен быть создан, настроена входимость данного типа объекта и соответствие атрибутов данному типу объекта.

На первом шаге данного действия пользователь будет вводить описание создаваемой служебной записки. В контекстном меню первого шага выберите «Настройка шага...». В поле «Тип шага:» открывшегося окна выберите «Свободная форма», в контекстном меню правой части окна выберите «Добавить», и в открывшемся списке переменных выберите «Opisanie». В результате будет создана форма шага, которая, после минимальной настройки внешнего вида элементов может выглядеть следующим образом – Рисунок 165.

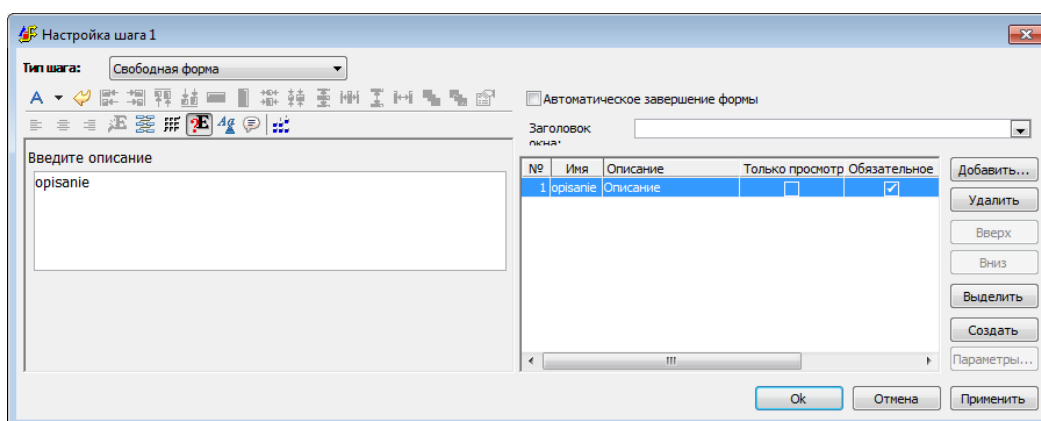


Рисунок 165 Добавление формы к шагу действия.

Нажмите «Ok», сохраните действие.

Создание формы для данного шага завершено.

Данные, введенные в форме шага можно использовать в функциях действия уже на данном шаге.

Для ввода функций в контекстном меню левой части окна, выберите «Вставить функцию». В открывшемся окне выберите функцию ObjCreate («Создать объект») – см. рисунок 166.

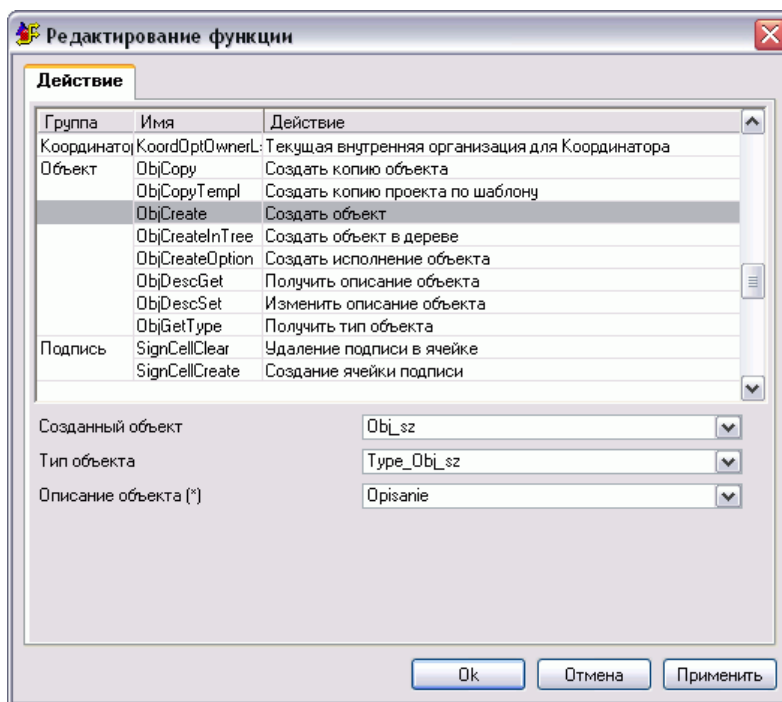


Рисунок 166 Добавление функции в шаг действия

Из выпадающих списков укажите переменные данной функции, нажмите «ОК» – созданная функция

`Obj_sz = ObjCreate (Type_Obj_sz, Opisanie)`
будет добавлена в текущий шаг действия.

Данной функцией создается объект `Obj_sz` типа «Служебная записка» (данный тип задан как значение по умолчанию для переменной `Obj_sz_Type`). Описание созданного объекта было введено пользователем при заполнении формы первого шага.

Созданный объект служебной записки включите в текущий объект `a_Object` – используется функция `LinkAdd`:

`a_TLinkID = LinkAdd (a_Object, Obj_sz, a_LinkType)`

Добавление в действие данной функции производится аналогично добавлению предыдущей функции. Создайте связь с текущим кодом связи – `a_TlinkID`, текущий тип связи – `a_LinkType`, – имеет значение по умолчанию «Дерево проектов».

Сохраните действие.

Для обеспечения возможности привязки действия к конкретному профилю, в «Профилях и настройках» для данного профиля, необходимо выбрать пункт «Действия» и отключить флажок «Показывать все действия» (Рисунок 167).

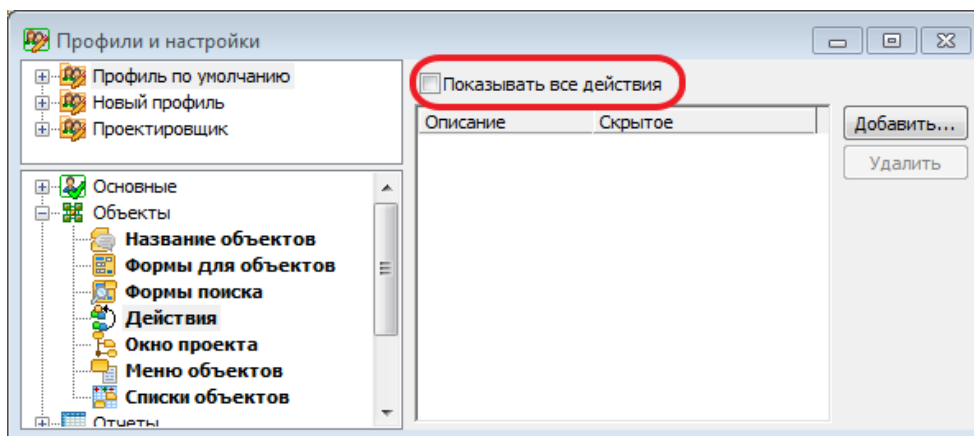


Рисунок 167 Отключен флажок «Показывать все действия»

Выполните привязку созданного действия к профилю и типам объектов (при действии, открытом в окне редактора действий, верхнее меню «Действие»→ «Привязка...»).

Добавьте профиль «Профиль по умолчанию», в качестве типов объектов укажите возможные типы для текущего объекта – любые необходимые, но не противоречащие входимости для типа объекта «Служебная записка», например, «Папка» и т.д.

Для выполнения данного действия выберите в дереве объект одного из типов, к которому привязано данное действие, в контекстном меню выберите «Выполнить действие...», выберите из списка созданное действие «Добавление служебной записки».

В форме действия введите описание создаваемой служебной записки, после нажатия «Далее» к текущему объекту будет добавлен новый объект типа «Служебная записка».

Данное действие можно усовершенствовать – сформировать и присвоить создаваемой записке необходимые атрибуты, добавить открытие её в отдельном окне и т.п.

Выполнение подобного действия целесообразно осуществлять по кнопке, расположенной в атрибутивной форме текущего объекта (см. п. 10.2.1).

17.11.4 Добавление объекта документа

Усложним задачу – пусть требуется добавлять в проект различные виды документов (соответственно, создавать объекты разных типов). Кроме того, добавляемому документу требуется присваивать атрибуты – наименование и обозначение. Обозначение должно включать в себя обозначение родительского объекта и обозначение непосредственно данного документа, и разделитель между ними (см. п. 5.4.1). Подобная ситуация типична, например, при добавлении документа в раздел проекта, а также при использовании предметной системы обозначений в машиностроении.

Очевидно, что текущий объект, в который добавляется документ, должен обязательно иметь обозначение – целесообразно проверить наличие у текущего объекта соответствующего атрибута, при его отсутствии вывести информационное сообщение и завершить выполнение действия.

При наличии обозначения у текущего объекта, в форме действия пользователь выбирает тип создаваемого документа, вводит наименование и обозначение документа, после чего создается объект документа, с включением его в текущий объект и присвоением атрибутов.

Создайте следующие переменные:

№	Имя	Описание	Тип данных	Значение по умолчанию	Редактирование
1.	Obj_1	Создаваемый объект документа	Объект	-	Скрытый
2.	Type_obj_1	Тип создаваемого объекта	Тип объекта	-	Скрытый
3.	id_Type_Obj_1	ID типа создаваемого объекта	Число	-	Свободный
4.	Atr_naim	Атрибут «Наименование»	Атрибут	Наименование	Скрытый
5.	naim	Наименование	Строка		Свободный
6.	Atr_obozn	Атрибут «Обозначение»	Атрибут	Обозначение	Скрытый
7.	obozn_0	Обозначение текущего объекта	Строка	-	Просмотр (или другой, но не «Скрытый»)
8.	Obozn	Обозначение документа	Строка	-	Свободный
9.	a_LinkType	Предопределенная переменная – текущий тип связи	Тип связи	Дерево проектов	Список (можно указать любой)

Для считывания с текущего объекта обозначения, добавьте в первый шаг действия строку: `obozn_0 = AttribGetDefault (a_Object , LinkID_0 , Atr_obozn , 'no')`

При отсутствии обозначения переменной `obozn_0` будет присвоено значение 'no'.

Следующая строка выполняет проверку наличия обозначения текущего объекта:

`GoTo (If(obozn_0 = 'no' , 'ERR' , "))`

Смысл данной строки – при отсутствии обозначения текущего объекта (выполняется условие «`obozn_0 = 'no'`») перейти на шаг с меткой «ERR», а при наличии у текущего объекта обозначения (в этом случае условие не выполняется) – продолжить выполнение действия.

На шаге с меткой «ERR» создайте свободную форму с текстовым полем – информационным сообщением об отсутствии обозначения текущего объекта.
Единственная строка данного шага:

`GoTo ('end')`

Обеспечивает завершение действия – переход к шагу с меткой «end».

Первый шаг действия завершается строкой:

`GoTo ('form')`

Данная строка обеспечивает переход к шагу с меткой «form».

На шаге с меткой «form» создайте форму, где пользователь будет вводить информацию о добавляемом документе (возможный пример настройки указанной формы – см. рисунок 168).

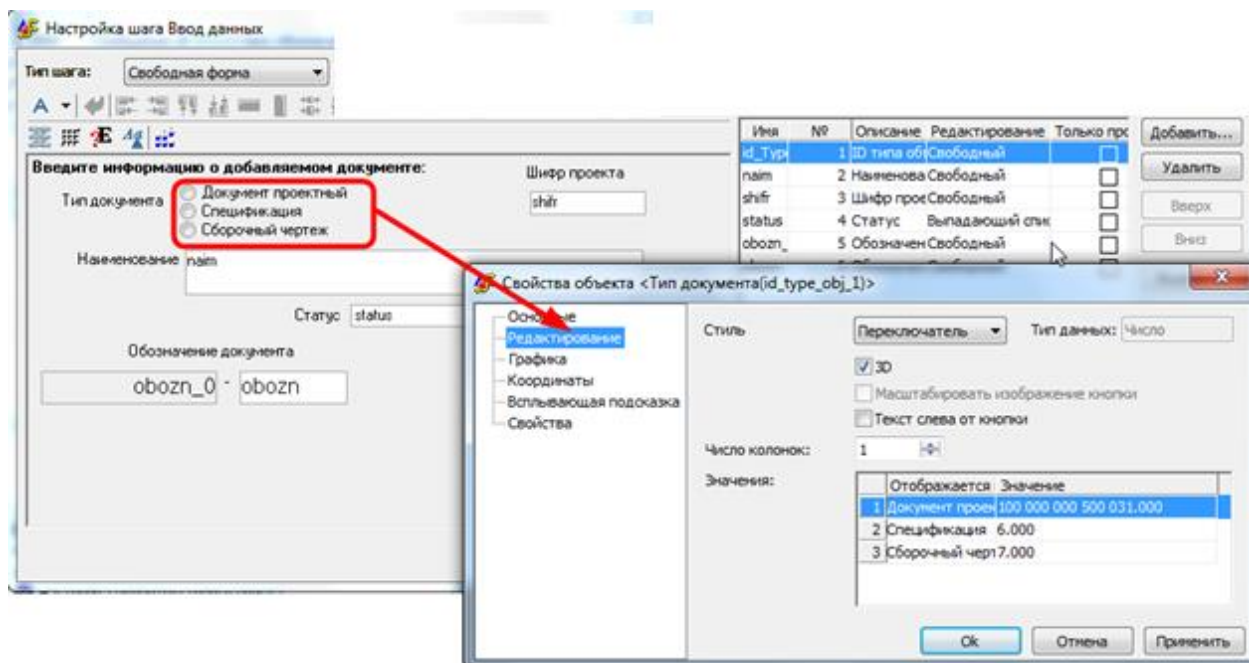


Рисунок 168 Форма для ввода информации о добавляемом документе. Пример

Данный шаг должен содержать следующие функции:

Type_obj_1 = SetByID (id_Type_Obj_1) – определение типа создаваемого объекта в соответствии с выбранным в форме идентификатором типа – id_Type_Obj_1 (см. также п.17.4).

obozn = Set (obozn_0+'-'+ obozn) – формирование обозначения документа, состоящего из обозначения текущего объекта, разделителя (в данном примере это «-», то есть, дефис) и обозначения собственно документа, введенного пользователем в форме – см. рисунок 168.

Obj_1 = ObjCreate (Type_obj_1 , obozn +' '+naim) – создание объекта документа. Описание созданного объекта будет состоять из обозначения и наименования, разделенных пробелом.

a_TLinkID = LinkAdd (a_Object , Obj_1 , a_LinkType) – связывание созданного объекта с текущим.

AttribSet (Obj_1 , LinkID_0 , Atr_naim , naim)

AttribSet (Obj_1 , LinkID_0 , Atr_obozn , obozn) – установка созданному объекту атрибутов – наименования и обозначения.

По окончании редактирования действие должно выглядеть, как показано на рисунке (см. рисунок 169).

Шаг	Описание	Тип
	Проверка наличия обозначения текущего объекта	Автоматический
1	<pre> obozn_0 = AttribGetDefault (a_Object , LinkID_0 , Atr_obozn , 'no') GoTo (If(obozn_0 = 'no' , 'ERR' , ")) GoTo ('form') </pre>	
ERR	Сообщение об отсутствии обозначения	Ввод данных
2	GoTo ('end')	
form	Ввод данных	Ввод данных
3	<pre> Type_obj_1 = SetByID (id_Type_Obj_1) obozn = Set (obozn_0+' '+ obozn) Obj_1 = ObjCreate (Type_obj_1 , obozn +' '+naim) a_TLinkID = LinkAdd (a_Object , Obj_1 , a_LinkType) AttribSet (Obj_1 , LinkID_0 , Atr_naim , naim) AttribSet (Obj_1 , LinkID_0 , Atr_obozn , obozn) </pre>	
end	Завершение	Автоматический
4		

Рисунок 169 Действие по добавлению объекта документа

Выполним данное действие по объекту (см. п. 17.2), не содержащему атрибут «Обозначение». Должна открыться форма с информационным сообщением, после чего действие можно завершить.

При запуске действия по объекту, содержащему обозначение:

1. будет открыта форма для ввода данных о добавляемом документе, после чего
2. в текущий объект будет добавлен новый объект указанного в форме типа
3. добавленному объекту будут установлены атрибуты «Наименование» и «Обозначение»

17.11.5 Присвоение регистрационного номера с помощью действий

Регистрировать (присваивать специальные номера) на практике приходится достаточно часто. Примеры: регистрация входящей и исходящей корреспонденции, приказов, распоряжений; инвентарный учет технической документации, имущества и т.д. Условно назовем соответствующий присваиваемый атрибут «Регистрационным номером», создайте данный атрибут, сопоставьте его типам регистрируемых объектов.

Общей особенностью таких номеров является наличие в значении номера части, соответствующей порядковому номеру – то есть, очередной присваиваемый номер отличается от предыдущего возрастающим (или убывающим) порядковым номером. Рекомендуемый тип данных такого атрибута – «Строка».

В некоторых случаях регистрационный номер состоит только из порядкового номера, но это не является достаточным поводом делать тип данных числом, так как в подобном случае, при возникновении потребности вставить между номерами, например, 3917 и 3918 номер «3917а» придется создавать новый атрибут.

Для определенности, рассмотрим пример присвоения регистрационного номера в следующем виде:

NNN-BB/YY , где:

NNN – цифровой очередной порядковый номер, изменяется для каждого следующего объекта по возрастанию;

ВВ – буквенная часть номера (например, серия). Изменяется оператором по мере необходимости, в нашем случае будет постоянной;

YY – последние 2 цифры года;

Символы «-» и «/» – разделители

Номер присваивается выделенному объекту документа.

Для хранения значения текущего (последнего присвоенного) номера создайте специальный служебный объект – «Счетчик регистрационных номеров». Тип объекта может быть любым, например, «Папка».

Создайте и присвойте объекту счетчика 2 служебных атрибута:

Первый – «Число номера». Тип данных – «Число», единичный, связь – «Объект», редактирование – «Свободный». Значение для объекта счетчик – «0».

Второй – «Буквенная часть». Тип данных – «Строка», единичный, связь – «Объект», редактирование – «Свободный». Значение для объекта счетчик – произвольное, для определенности – «JS».

Создайте следующее действие:

Переменные:

Используйте предопределенные переменные a_Object, LinkID_0, а также создайте следующие переменные:

№	Имя	Описание	Тип данных	Значение по умолчанию	Редактирование
1.	Atr_n_reg	Атрибут "Номер регистрационный"	Атрибут	Номер регистрационный	Скрытый
2.	n_reg	Номер регистрационный	Строка	-	Свободный
3.	Atr_CH	Атрибут "Число номера"	Атрибут	Число номера	Скрытый
4.	CH	Порядковый номер	Строка	-	Скрытый
5.	Atr_BB	Атрибут "Буквенная часть"	Атрибут	Буквенная часть	Скрытый
6.	BB	Буквенная часть номера	Строка	-	Скрытый
7.	Obj_CH	Объект счетчика	Объект	Счетчик регистрационных номеров	Скрытый
8.	opisanie	Описание текущего объекта	Строка	-	Скрытый

Ведите любое значение в описание первого шага, сохраните действие. Пусть, для определенности, оно будет называться «Присвоение регистрационного номера».

Пользовательских форм в действии не будет, действие, в простейшем виде будет состоять из единственного шага. Создайте описанное ниже действие.

Описание работы действия и используемые функции:

Чтение из объекта счетчика значения буквенной части номера:

BB = AttribGet (Obj_CH , LinkID_0 , Atr_BB)

Чтение из объекта счетчика значения последнего (предыдущего) порядкового номера:

CH = AttribGet (Obj_CH , LinkID_0 , Atr_CH)

Увеличение текущего значения порядкового номера на 1:

CH = Set (CH +1)

Возврат в счетчик нового значения порядкового номера:

```
AttribSet ( Obj_CH , LinkID_0 , Atr_CH , CH )
```

Формирование строкового значения регистрационного номера:

```
n_reg = Set ( CH+'-'+BB+'/' + Right ( string ( Year( Today() ) ) , 2 ) )
```

Функция Today() возвращает текущую дату, функция Year() возвращает год (от текущей даты), функция string() преобразует данное значение в строку, функция Right () выделяет указанное количество (два) правых символа переданной в качестве первого аргумента строки.

Присвоение регистрационного номера текущему объекту:

```
AttribSet ( a_Object , LinkID_0 , Atr_n_reg , n_reg )
```

Можно дополнить существующее описание объекта его регистрационным номером. Чтение имеющегося описания текущего объекта:

```
opisanie = ObjDescGet ( a_Object )
```

Изменение описания текущего объекта – добавление к нему символ « № » и значения регистрационного номера:

```
ObjDescSet ( a_Object , opisanie +' № '+n_reg )
```

Сохраните созданное действие, выполните его, запустив по выделенному объекту документа: из контекстного меню объекта выберите «Выполнить действие...» и выберите действие из списка. Если действие в списке не отображается, перейдите на вкладку «Доп.условия» окна выбора действия, задайте требуемые условия отбора, нажмите «Выполнить», и выберите действие из сформированного по заданным условиям списка действий.

В результате выполнения действия:

- текущему объекту будет присвоен соответствующий атрибут «Номер регистрационный» и изменено описание;
- атрибуту «Число номера» объекта счетчика будет присвоено очередное значение.

17.11.6 Создание объекта документа с его автоматической регистрацией. Пример использования вложенного действия

Рассмотрим создание действие, которое будет добавлять объект документа в текущий объект и присваивать объекту документа регистрационный номер.

При создании данного действия используйте рассмотренные ранее действия – добавление объекта документа (см. п. 17.11.4) и присвоение регистрационного номера (см. п. 17.11.5).

Добавьте имеющееся действие «Присвоение регистрационного номера» в действие «Добавление объекта документа» – используйте действие «Присвоение регистрационного номера» как вложенное. Для этого:

Откройте действие «Добавление объекта документа» (см. рисунок 169) для редактирования.

Вставьте перед последним шагом действия новый шаг (из контекстного меню последнего шага выбрать «Вставить шаг»), введите его описание – «Присвоение регистрационного номера».

В контекстном меню нового шага выберите «Настройка шага...», укажите тип шага – «Действие», в поле «Действие» укажите действие «Присвоение регистрационного номера».

В контекстном меню поля «Аргументы действия» выберите «Добавить», в открывшемся списке переменных вложенного действия выберите «a_Object», в колонке «Выражение» укажите для данной переменной соответствующую переменную родительского действия – «Obj_1» (Рисунок 170).

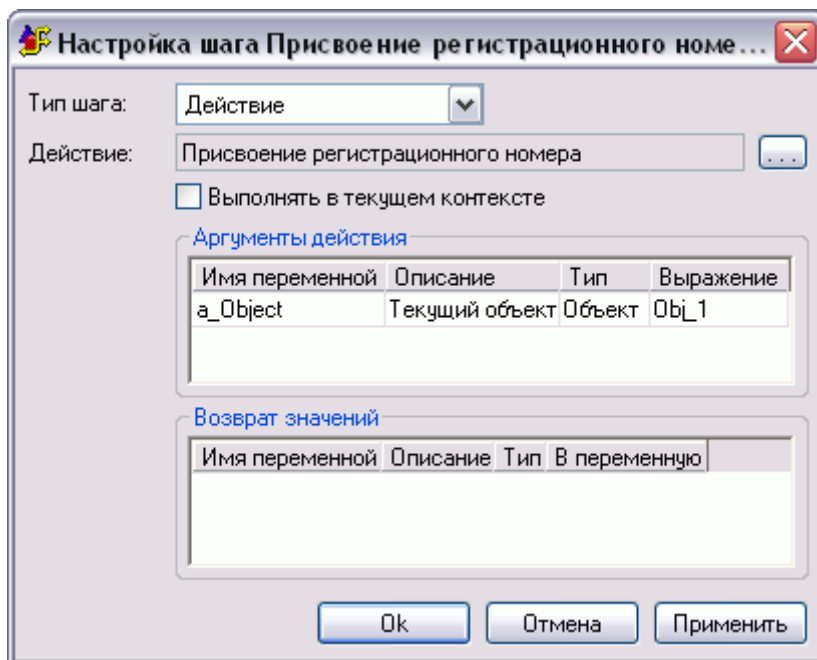


Рисунок 170 Добавление вложенного действия

Нажмите «Ok», сохраните действие.

Теперь действие «Добавление объекта документ» выполняет создание объекта документа с присвоением ему регистрационного номера.

17.12 Создание действий над объектами. Учебное задание

Создайте действие для открытия основных проектов по выбору (см. п. 17.11.2). Добавьте данное действие в виде отдельного пункта Главного меню и инструментальной панели.

Создайте действия по добавлению документа (см. п. 17.11.4) и действие для присвоения регистрационного номера (см. п. 17.11.5).

Создайте в атрибутивной форме для проекта и раздела проекта кнопку «Добавить документ», а в форме документа – кнопку «Зарегистрировать». Настройте запуск указанных действий по соответствующим кнопкам.

Используя созданные действия, добавьте в Вашу настройку некоторое количество документов и присвойте им регистрационные номера. Указанные объекты документов могут быть использованы в дальнейшем, при настройке учебных отчетов.

17.13 Действия над объектами. Вопросы для самопроверки

1. Как добавить в действие новую переменную?
2. Как можно реализовать в действии выбор значения переменной из списка (фиксированного, формируемого в действии, значений атрибута)?

3. Как присвоить переменной типа «Объект» значение с помощью условий отбора?
4. Как добавить форму к шагу действия?
5. Как добавить вложенное действие?
6. Какими способами можно запустить действие на выполнение по данному объекту?
7. Чем отличается создание объекта в дереве и создание объекта с последующим добавлением входящей подчиненной связи – какие функции используются, чем отличается результат?
8. Что будет происходить при выполнении действия с пользовательской формой, если в свойствах данного действия, в поле окно установлено «Никогда не показывать»?

18 Импорт объектов. Резюме

Импорт объектов и атрибутов предназначен для ввода внешних данных в БД, при уже созданной структуре данных (элементы структуры также можно импортировать в БД – см. п. 7).

В Lotsia PDM имеется два способа импорта объектов:

- «Импорт объектов» – способ импорта из текстовых файлов без заголовков, с разделителями – символами табуляции. Файлы импорта должны быть заранее приведены к требуемому стандартному виду;
- «Расширенный импорт» – инструмент, позволяющий осуществлять импорт из различных источников данных с проведением дополнительной обработки импортируемых данных. Для загрузки различных по структуре данных заранее создаются отдельные настройки расширенного импорта. Созданная настройка расширенного импорта позволяет многократно выполнять импорт данных, имеющих соответствующую одинаковую структуру. Данные источника преобразуются при выполнении импорта определенным в используемой настройке расширенного импорта образом.

Источником данных при расширенном импорте может быть база данных, данные из которой извлекаются при выполнении импорта. SQL-запрос, возвращающий требуемые колонки данных, хранится в настройке расширенного импорта. В качестве источника данных может использоваться также XML-файл, файл excel или текстовый файл, содержащий колонки данных. Импорт можно начинать с произвольной строки файла, в качестве разделителей колонок для текстовых файлов могут задаваться различные символы.

При выполнении импорта могут быть загружены (то есть, созданы в БД) объекты соответствующих типов, связи между объектами (как загружаемыми, так и уже имеющимися в БД), описания объектов, атрибуты объектов и атрибуты проектов. В ходе обработки каждой строки файла импорта может выполняться действие над объектами, значения в колонках данной строки могут передаваться в действие в качестве аргументов.

Подробно оба способа описаны в документации пользователя Lotsia PDM.

В некоторых случаях импорт с предварительной выгрузкой данных может использоваться как инструмент, позволяющий осуществлять редактирование данных внешними средствами. Для выгрузки данных можно использовать отчет, с сохранением результатов его выполнения в текстовый файл. Далее полученный файл может быть отредактирован, например, с помощью MS Excel, после чего импортирован в БД. В некоторых случаях данный метод может оказаться наиболее эффективным для решения конкретной задачи по изменению большого объема данных.

19 Архив документов

Архив предназначен для хранения документов, присоединенных к объектам Lotsia PDM. Понятие «Архив документов» включает в себя библиотеку, настроенную в Lotsia PDM PLUS и сопоставленное данной библиотеке хранилище, находящееся на сетевом ресурсе.

Защищенная библиотека с сопоставленным ей защищенным хранилищем образуют защищенный архив (Рисунок 171).

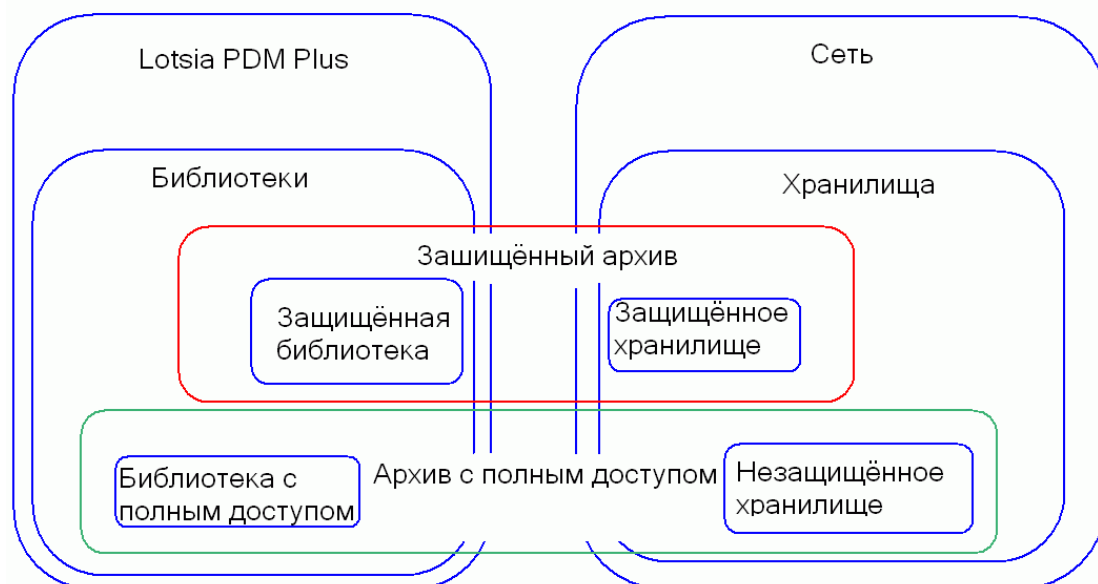


Рисунок 171 Иллюстрация к понятию «Архив»

Соответственно, архив с полным доступом включает библиотеку с полным доступом и хранилище с полным доступом. На документы, импортируемые в защищенный архив, обязательно назначаются права доступа. На документы, импортированные в архив с полным доступом, права доступа не назначаются.

При настройке интеграции с приложениями документов, доступ к документам архива осуществляется непосредственно из Lotsia PDM, с автоматическим запуском приложения документа. При использовании защищенного архива, доступ к документам архива возможен только из Lotsia PDM, в соответствии с установленными в Lotsia PDM правами доступа к документу и его версиям. Максимально возможное количество версий (до 99) устанавливается отдельно для каждого типа документа.

Пользователь работает с версией документа. Версию можно редактировать, сохранить как новую версию, как новый документ, прикрепив его к любому существующему или новому объекту. При создании нового документа, всегда создается первичная версия документа.

При удалении документа архива из программы происходит его физическое удаление из библиотеки. Пользователь, не являющийся администратором, может удалить документ архива или его версию только при наличии соответствующих прав доступа к данному документу или версии.

Для создания архива необходимо:

- Создать хранилище (каталог на сетевом ресурсе). Если это необходимо, выполнить создание защищенного хранилища. Для этого, средствами операционной системы создать специального пользователя PartyArc (в

зависимости от используемой ОС процедура несколько отличается, потребуется выполнение некоторых дополнительных настроек). Создать каталог со специальными правами доступа, после чего создать и запустить специальную службу (или, в зависимости от используемой сетевой ОС, внести соответствующие изменения в конфигурационный файл). Для поддерживаемых сетевых ОС подробно создание защищенного хранилища описано в документации администратора Lotsia PDM PUS;

- Настроить библиотеки для хранения документов;
- Указать в свойствах каждой библиотеки путь до соответствующего хранилища;

Чтобы обеспечить возможность работы с архивом (то есть, импорт документов в архив, открытие документов архива) следует также:

- Зарегистрировать типы документов;
- Зарегистрировать приложения для работы с документами архива, настроить методы их запуска;
- Произвести дополнительную настройку архива (при необходимости).

19.1 Создание защищенного хранилища, подключение его к библиотеке

Рассмотрим установку модуля защиты документов для MS Windows 2000 (2003) и более новых версий серверных операционных систем Microsoft.

Установка модуля защиты документов для MS Windows осуществляется администратором MS Windows на разделе с файловой системой NTFS.

Для установки модуля защиты файлов Lotsia PDM PLUS для сервера MS Windows 2000 (2003) необходимо:

Создать (Settings → Control Panel → Users and Passwords → Add...) пользователя с именем PartyArc. При создании данного пользователя в поле Domain указать имя реального домена.

Для пользователя PartyArc установить неограниченное время действия пароля, рекомендуется запретить пользователю PartyArc изменение пароля.

Добавить пользователя PartyArc в локальную группу Administrators.

дать следующие права пользователю PartyArc:

- «Log on as a service» («Вход в качестве службы»),
- «Take ownership of files or other objects» («Овладение файлами или иными объектами»)
- «Access this computer from network» («Доступ к компьютеру из сети»).

Для каждого защищенного хранилища необходимо создать отдельный каталог. Каталог может быть создан на любом NTFS-разделе локального диска. Если на компьютере предполагается наличие нескольких защищенных хранилищ, то необходимо располагать соответствующие каталоги так, чтобы они не были вложены друг в друга. Каталоги могут располагаться на разных разделах одного диска или на разных дисках.

Для каждого каталога необходимо обеспечить разделение доступа и установить права доступа. Для этого в проводнике открыть окно свойств каталога и перейти на вкладку Sharing. В этой вкладке необходимо ввести имя разделяемого ресурса. Имя может быть любым, отвечающим требованиям Windows

Для каталогов необходимо установить права пользователей в следующем виде:

Разрешения доступа к разделяемому каталогу – Access Through Share Permissions (кнопка Permissions на вкладке Sharing):

- для пользователя PartyArc разрешить полный доступ к каталогу и файлам (Full Control);
- для всех пользователей (Everyone) разрешить полный доступ (Full Control).

Задать права доступа на уровне NTFS (нажать «Advanced» на вкладке «Security»):

- запретить распространения на каталог наследуемых от родительского каталога прав;
- для пользователя PartyArc разрешить полный доступ к каталогу и файлам (Full Control) см. рисунок 172;
- для всех пользователей (Everyone) установить специальные права доступа

Подробнее см. «Руководство по администрированию и настройке программы Lotsia PDM».

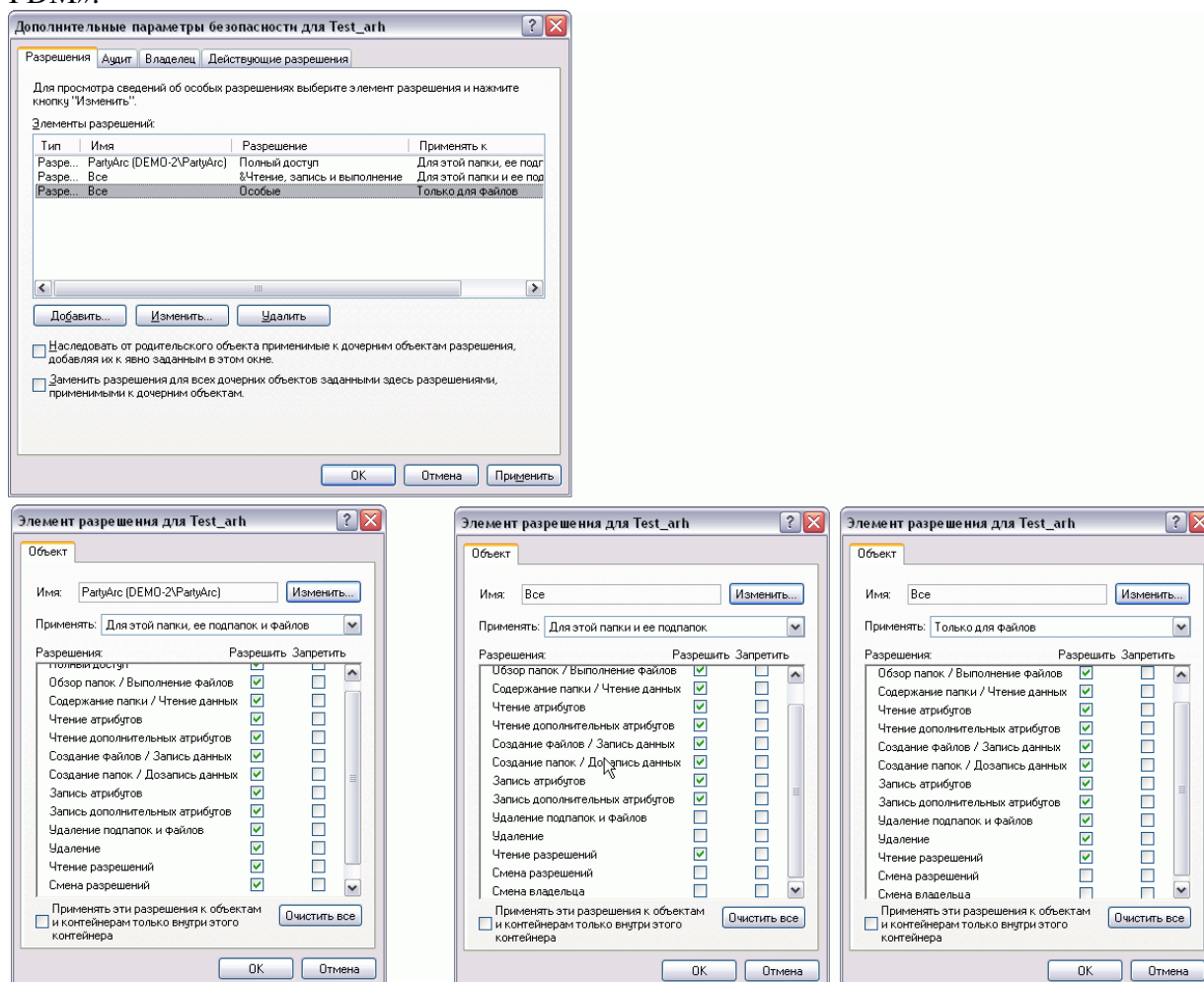


Рисунок 172 Права доступа к хранилищу на уровне NTFS

Далее следует скопировать на жесткий диск компьютера программу «partynt.exe».

Для установки сервиса доступа к документам запустить «partynt.exe» с параметрами:

partynt.exe /i:<Passw> [Share1 Share2 ... ShareN]

где <Passw> – пароль MS Windows NT для пользователя 'PartyArc';

Share1 Share2 ... ShareN – перечень созданных разделяемых ресурсов

Например:

partynt.exe /i:Password_PartyArc PartyArc1 PartyArc2

Запустить программу Services (Службы) в Control Panel MS Windows (Settings → Control Panel → Administrative Tools → Services). С её помощью запустить сервис (службу) «PartyArc Document Access Service».

Просмотром журнала событий убедиться в отсутствии ошибок, связанных с данной службой. При наличии ошибок, в общем случае, рекомендуется:

- остановить службу «PartyArc Document Access Service»;
- удалить службу защиты, выполнив команду partynt.exe /r ;
- устранить ошибки при создании пользователя PartyArc и задании свойств хранилища, после чего повторить установку и запуск службы «PartyArc Document Access Service», добившись отсутствия ошибок при её запуске.

Для создания защищенного архива созданное защищенное хранилище следует подключить к библиотеке и для данной библиотеки установить тип доступа: «Защищенный (WindowsNT)» (см. п. 19.2).

Подробное описание процесса настройки защищенного архива для различных ОС содержится в Руководстве по администрированию Lotsia PDM PLUS.

19.2 Создание библиотеки

В настройке библиотек определяют места физического хранения документов архива. Библиотеки архива могут быть с полным доступом или защищенные средствами Novel Netware или Windows NT.

Для регистрации (создания) библиотеки в Главном меню используются пункты «Администрирование» → «Архив» → «Библиотеки». Для создания библиотеки следует нажать «Создать» в открывшемся окне «Библиотеки» (Рисунок 173).

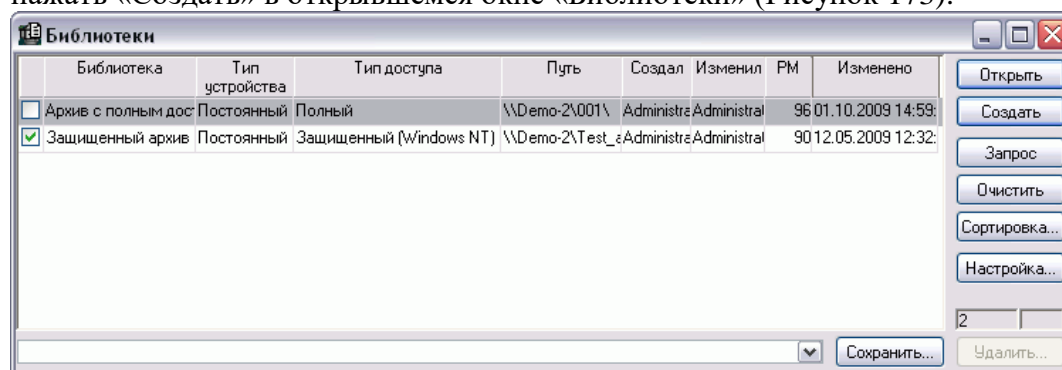


Рисунок 173 Окно библиотек

При создании новой библиотеки следует указать полный путь к месту хранения документов. В качестве места хранения должен использоваться сетевой ресурс, который будет доступен со всех рабочих станций, где планируется использование архива.

В ходе настройки библиотеки вводится описание, тип доступа – см. рисунок 174, Рисунок 175.

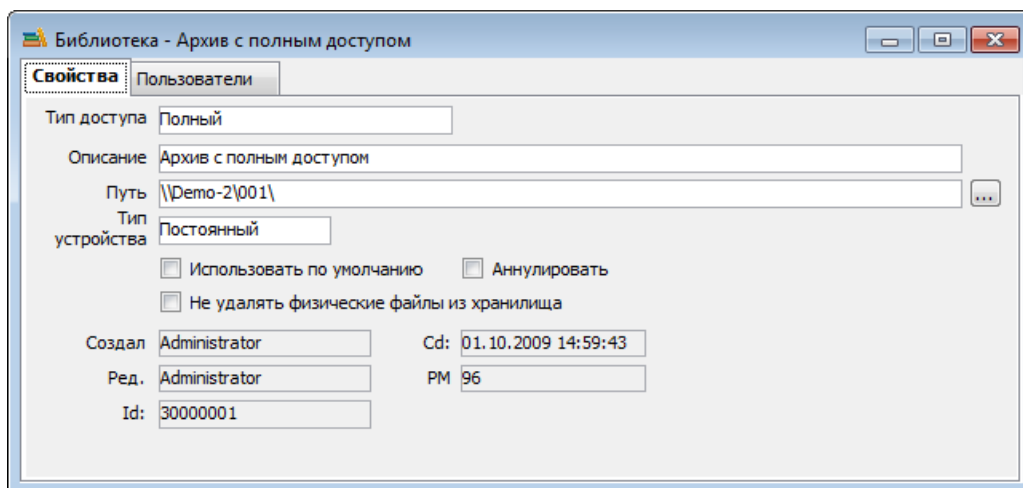



Рисунок 174 Библиотека с полным доступом

Для сохранения созданной библиотеки следует нажать <Ctrl+S> либо кнопку  инструментальной панели.

На вкладке «Пользователи» определяется список пользователей данной библиотеки. В том случае, если пользователь может иметь доступ к нескольким библиотекам, одна из них должна быть установлена «по умолчанию». Можно также, при определении свойств пользователя или группы (см. «Создание пользователей»), доступ к библиотекам и библиотеку по умолчанию определять там, для данного пользователя или группы.

Для того чтобы архив документов был защищенным, необходимо выполнение следующих условий:

- Создание и запуск специального модуля защиты для данного хранилища (см. п. 19.1)
- Установка для данной библиотеки типа доступа в положение «Защищенный (NetWare)» или «Защищенный (Windows NT)» в соответствии с используемой сетевой ОС (см. рисунок 175).

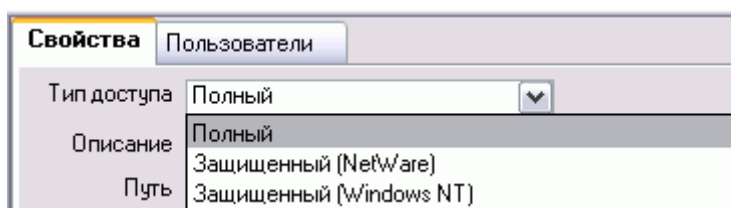


Рисунок 175 Установка типа доступа для библиотеки

19.3 Регистрация типов документов

Для добавления типа документа необходимо в Главном меню выбрать пункт «Администрирование» → «Архив» → «Типы документов», в контекстном меню выбрать «Добавить». Выбрать приложение для документа (при отсутствии приложения в списке – ввести название вручную). Ввести описание типа документа (если не сформировано автоматически). Сформировать список масок файлов для данного типа документа (если не сформировано автоматически) и указать, если требуется, ограничение максимально возможного количества версий (по умолчанию – 99) Пример задания типов документов – см. рисунок 176

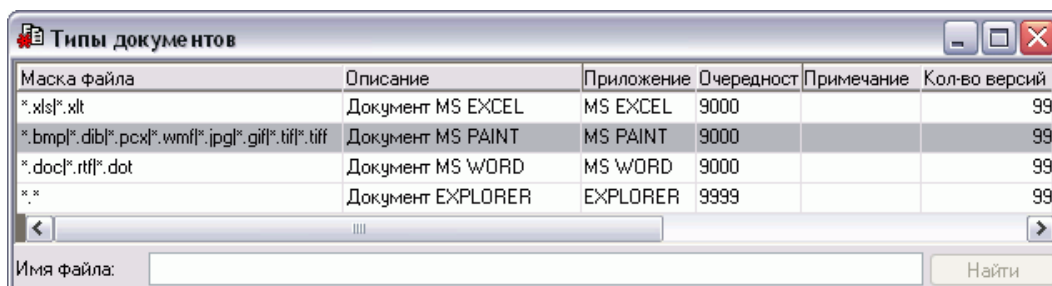


Рисунок 176 Пример задания типов документов

19.4 Регистрация приложения и метода (-ов) его запуска

Для регистрации приложения, в Главном меню следует выбрать пункт «Администрирование» → «Архив» → «Настройка приложений». В открывшемся окне нажать «Создать» для добавления приложения, или дважды щелкнуть на строке соответствующего приложения для настройки его свойств (см. рисунок 177).

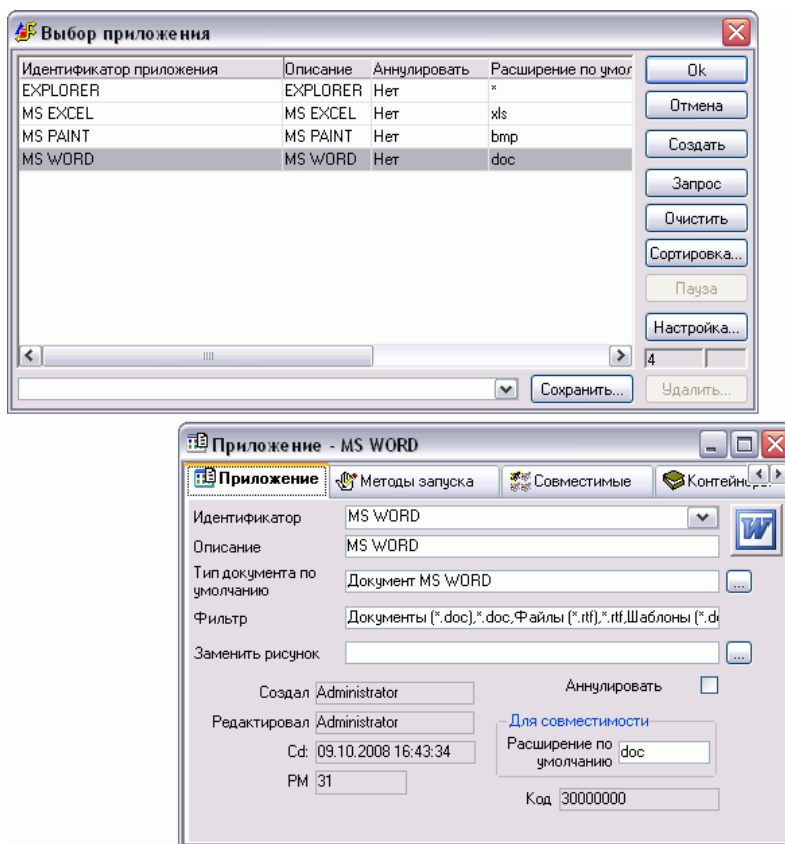


Рисунок 177 Настройка приложений

Для каждого из приложений определяются методы запуска (на вкладке «Методы запуска» окна свойств приложения). Для интегрированных приложений основные методы запуска создаются автоматически. Для остальных случаев методы запуска следует создавать, используя пункт «Добавить» контекстного меню вкладки «Методы запуска»

При настройке методов запуска (см. рисунок 178) следует иметь в виду, что:

- если для приложения не настроен метод запуска с типом «Просмотр», то в контекстном меню документа будет отсутствовать соответствующий пункт;
- для одного из методов запуска должен быть включен флажок «Использовать по умолчанию»;

- не желательно указывать полный путь к исполняемому файлу, так как на рабочих станциях пользователей он может оказаться различным;
- если требуется запускать приложение с различными параметрами командной строки, следует создать несколько методов запуска – с соответствующими значениями параметров;
- использование ODMA (поле «Интеграция», возможные значения «ODMA», «Стандартная», «Нет») возможно в том случае, если ODMA поддерживается данным приложением. Если приложение использует стандартные окна выбора файлов для открытия или «сохранения документа как...», возможно использование стандартной интеграции.

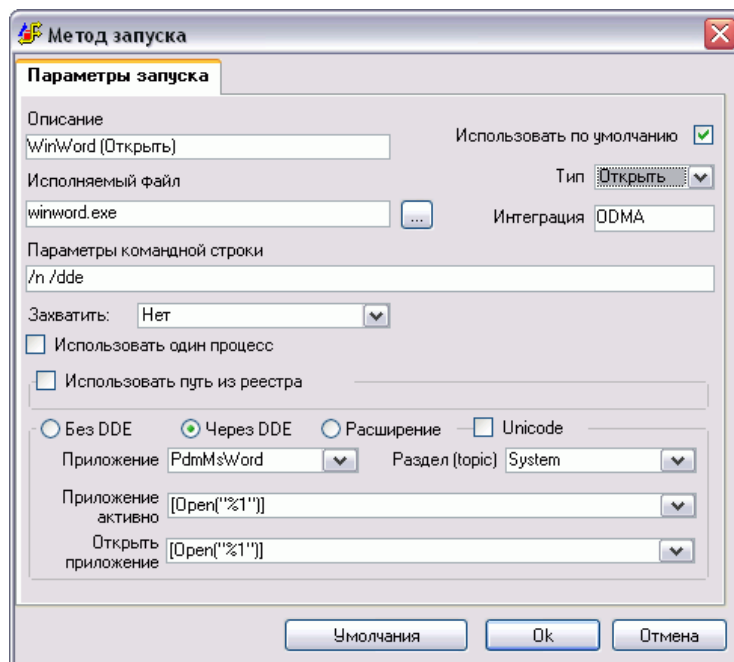


Рисунок 178 Настройка метода запуска приложения

Возможно открытие документа архива непосредственно из хранилища, либо в режиме захвата. «Захват» – копирование документа архива на рабочую станцию пользователя. При захвате документа на изменение, документ в хранилище блокируется, остается доступным только для просмотра. По окончании работы с захваченным на изменение документом, документ в хранилище разблокируется. При завершении работы с захваченным на изменение документом с сохранением внесенных изменений (т.н. «отпускании» документа), измененный документ с рабочей станции переписывается поверх документа в хранилище. Если документ архива компонентный (то есть, в документе содержатся ссылки на другие документы), то при захвате документа захватываются и все компоненты (ссылочные документы) данного документа.

В методе запуска приложения (Рисунок 178, поле «Захватить») могут быть указаны различные режимы захвата:

«Нет» – документ открывается согласно настройкам архива (см. п.19.5);

«Запрещен» – захват документа не выполняется, несмотря на установки места открытия документа в настройках архива (см. п.19.5);

«Автозахват» – документ автоматически захватывается, по завершении работы – автоматически отпускается;

«Захватить/разблокировать» – при первом захвате документ копируется на локальную рабочую станцию пользователя. При отпускании документа документ

разблокируется, но локальная копия не удаляется, документ остается захваченным на чтение. В отличие от автозахвата, при последующих захватах, если документ в архиве не изменялся, повторного копирования документа из хранилища не производится, используется имеющаяся локальная копия документа, документ в хранилище блокируется. Данный режим позволяет минимизировать сетевой трафик, особенно эффективен при работе с компонентными документами. Кроме того, при использовании данного метода захват документа осуществляется постоянно в одну и ту же папку, что, в ряде случаев, дает дополнительные преимущества: если приложение – обработчик при открытии документов создает служебные файлы и папки, то их количество будет возрастать при каждом открытии данного документа архива в минимальной степени.

Подробную информацию по настройкам приложений можно получить, воспользовавшись контекстной справкой по системе Lotsia PDM PLUS, либо обратившись к «Руководству по администрированию Lotsia PDM».

19.5 Другие настройки, относящиеся к работе с архивом

Для задания общих настроек архива используется пункт «Настройка архив» («Главное меню» → «Администрирование» → «Архив» → «Настройка архив»). В открывшемся окне (Рисунок 179) могут быть определены значения указанных настроек.

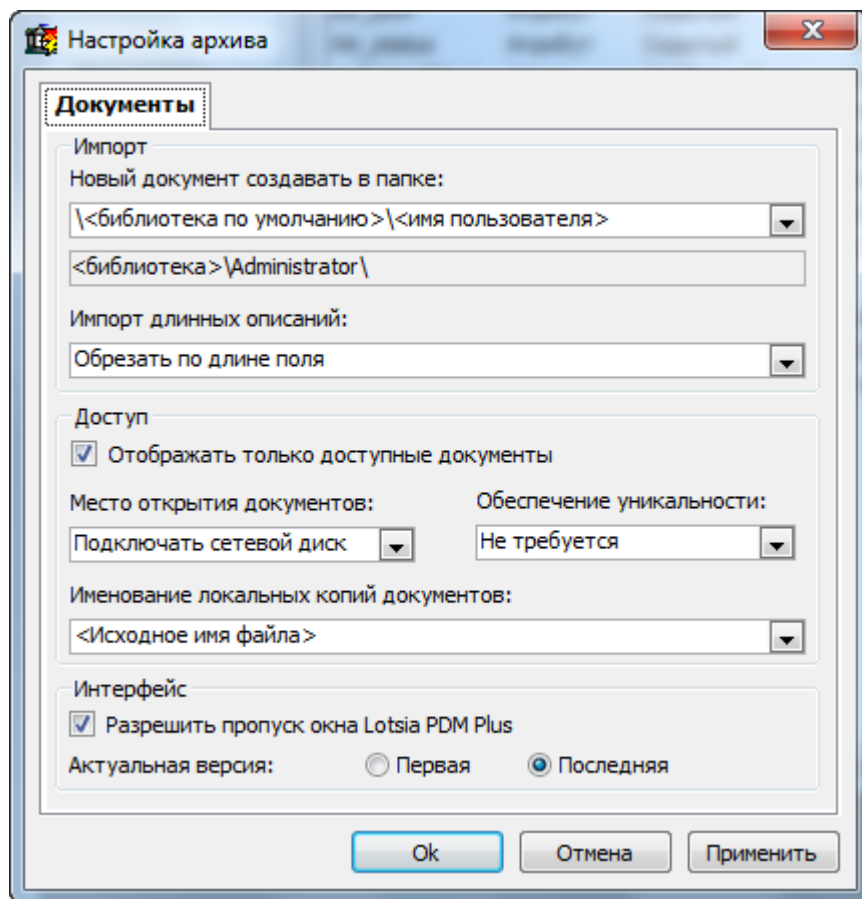


Рисунок 179 Окно «Настройка архива»

Значения указанных настроек выбираются, исходя из особенностей используемых приложений и требований к использованию программы. Назначение указанных настроек соответствуют их названиям. Некоторые пояснения:

- флажок «Отображать только доступные документы» включается, если требуется, чтобы пользователь не видел на вкладке «Документы архива» строк документов, к которым у него нет права доступа;

- настройка «Именованние локальных копий документов» относится к именам файлов захваченных документов;
- флажок «Разрешить пропуск окна Lotsia PDM PLUS» включает или отключает возможность не открывать окно выбора документа Lotsia PDM PLUS. Если флажок включен, при нажатой клавише <Shift> при сохранении и открытии документов в интегрированных приложениях, открывается сразу окно выбора файла данного приложения;
- Переключателем «Актуальная версия» можно установить умолчание для определения актуальной версии. По умолчанию актуальная версия последняя. Актуальность версии используется, например, при выделении документа на вкладке «Документы архива» – актуальная версия будет выделена по умолчанию, она же по умолчанию будет отображаться в окне предпросмотра.

Для задания настроек архива для профилей пользователей и отдельных пользователей используются соответствующие пункты профилей и настроек или настроек пользователя – см. п. 10.3.9.

Имеется возможность наследовать на документы архива не только наследуемые и унаследованные права с объекта, в который импортирован данный документ, но и непосредственные права, установленные для данного объекта. Для наследования непосредственных прав с объекта на документы архива данного объекта, в общих параметрах БД следует включить флажок «Наследовать все права на документы» (Рисунок 180).

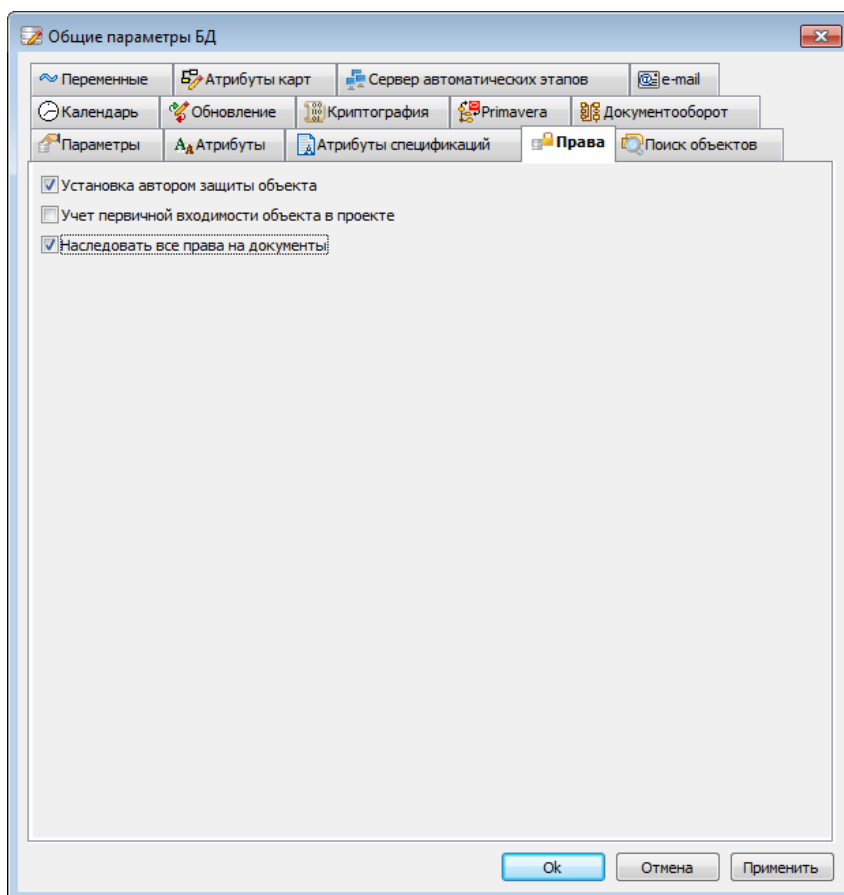


Рисунок 180 Установка наследования непосредственных прав с объекта на документы архива

Полную информацию по данным настройкам можно получить, воспользовавшись контекстной справкой или документацией по системе Lotsia PDM PLUS.

19.6 Импорт документов в объект

Процедура импорта документа (документов) в объект предназначена для прикрепления документов из файловой или почтовой системы в уже существующий объект. В процессе импорта происходит копирование (перемещение, регистрация) документа (документов) в архив программы. Импорт в защищенный архив может быть осуществлен только с контролем доступа (то есть, с заданием прав доступа к документу).

В Lotsia PDM PLUS одному объекту может принадлежать один или несколько документов архива.

Импорт документов в существующий объект в окне проекта на вкладке «Документы архива» можно производить различными способами, например:

С помощью пункта «Импорт» контекстного меню вкладки «Документы архива» окна проекта, либо нажав соответствующую кнопку на данной вкладке. В открывшемся окне «Импорт документов» (Рисунок 181) следует сформировать список импортируемых документов.

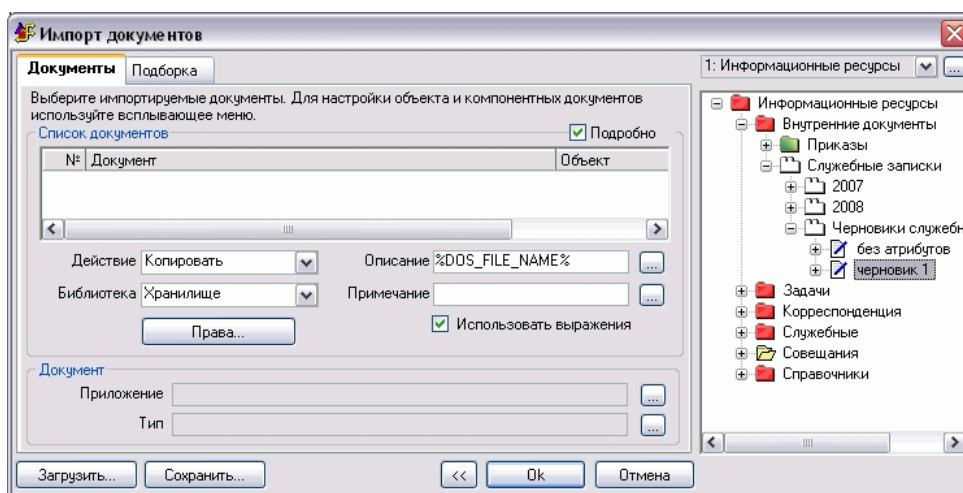


Рисунок 181. Окно импорта документов, включен флажок «Подробно»

Список выбранных для импорта документов отобразится в окне импорта.

В поле «Действие» (Рисунок 181) может быть выбрано одно из трех значений «Копировать», «Переместить», «Регистрировать». Выбор из первых двух значений, соответственно, Обеспечивает копирование или перемещение импортируемого файла в архив. Имя файла для документа архива в обоих случаях присваивается системой автоматически, пользователь может сформировать описание и примечание для документа архива. Если требуется сохранить для документа архива исходное имя файла, импортируемый документ следует предварительно поместить в хранилище, используя средства ОС. Далее документ импортируется в архив, при этом в поле «Действие» используется значение «Регистрация». В результате импортированный в архив документ сохраняет исходное имя файла.

Для выполнения импорта следует нажать «Ok»

Можно так же выполнять импорт, перетаскив мышкой файл (файлы) из окна проводника. Программа откроет окно импорта документа, содержащее перенесенные файлы.

Возможен пакетный импорт с использованием текстового файла импорта. Файл импорта должен содержать колонку с полными именами всех импортируемых файлов. Возможен так же одновременный импорт атрибутов. Импорт может осуществляться как в существующие объекты, так и созданием новых объектов.

19.7 Импорт структуры папок из файловой системы

Имеется возможность импорта дерева папок файловой системы с входящими в них файлами. В результате создается объектная структура с документами архива (Рисунок 182).

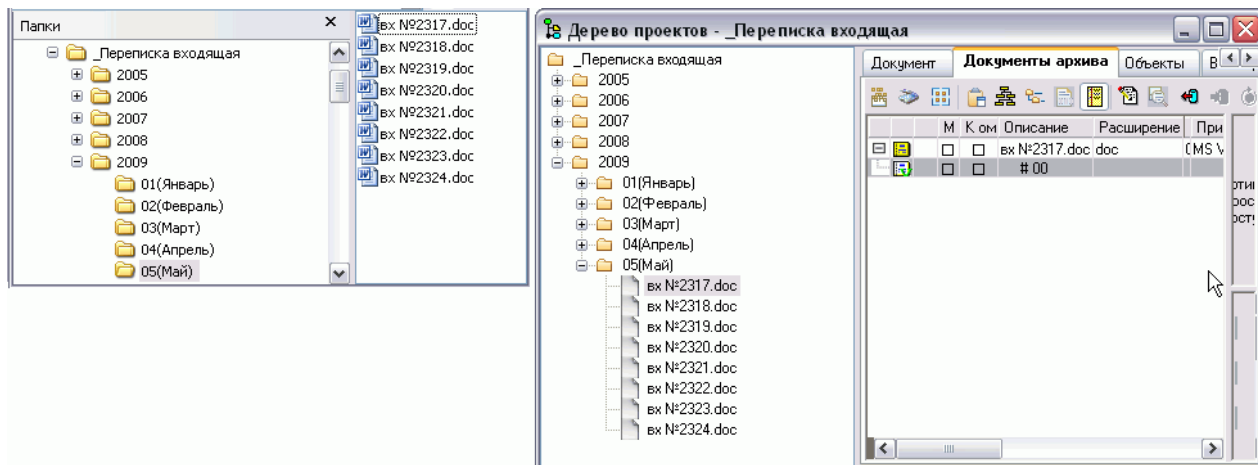


Рисунок 182 Исходная структура папок и результат её импорта в Lotsia PDM. Пример

Для осуществления указанного импорта необходимо выполнить следующее настройки:

- добавить приложение EXPLORER (Главное меню → Администрирование → Архив → Настройка приложений → кнопка «Создать») и выполнить требуемые настройки для данного приложения (Рисунок 183);

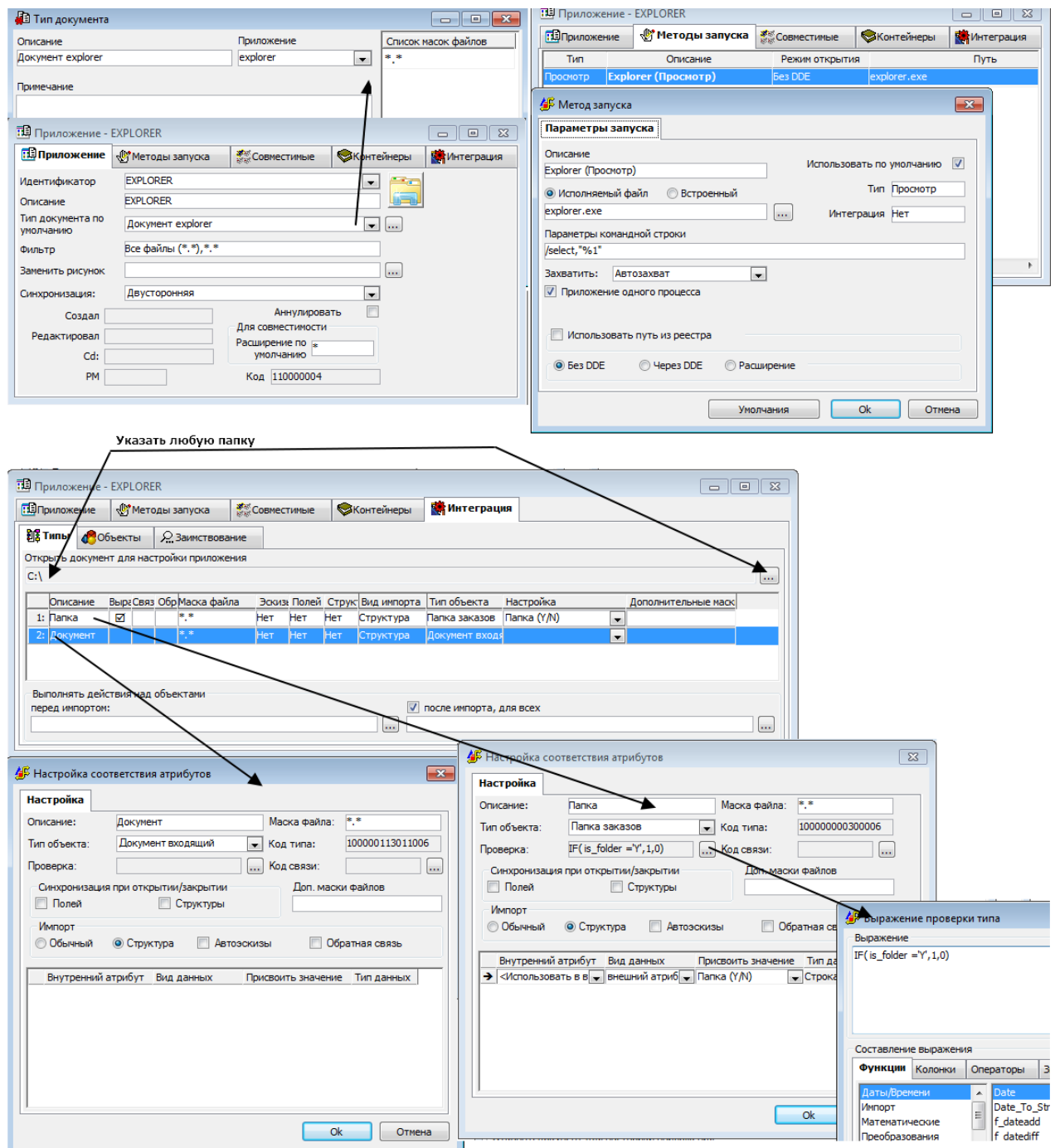


Рисунок 183 Выполнение настройки для приложения EXPLORER. Пример

- указать импортируемую структуру папок. Это можно сделать путем перетаскивания (применяя метод «drag-and-drop») корневой папки из окна проводника Windows в окно Lotsia PDM PLUS, либо с использованием инструмента «Импорт структуры» (Главное меню → Инструменты → Импорт структуры (интеграция) → в открывшемся окне выбрать «Импорт структуры папок», нажать «Ok» → в открывшемся окне указать требуемую папку, нажать «Ok»);
- для осуществления импорта нажать «Ok» в открывшемся окне «Импорт структуры» (предварительно выполнив, если это требуется, настройки в данном окне).

19.8 Резюме

Lotsia PDM PLUS обеспечивает возможность упорядоченного хранения файлов («Документов архива» или просто «Документов»).

Документы хранятся связанными с объектами. Доступ к документам регулируется средствами Lotsia PDM PLUS. Открытие документов архива в различных приложениях осуществляется из Lotsia PDM PLUS.

Документ может иметь произвольное, от 0 до 99, количество версий.

Понятие «Архив документов» включает в себя, с одной стороны, библиотеку, которую Администратор настраивает через интерфейс Lotsia PDM PLUS, а с другой стороны – сопоставляемое данной библиотеке хранилище документов, настройка которого зависит от используемой сетевой ОС. Библиотека может быть с полным доступом либо защищенной, защищенной библиотеке должно соответствовать защищенное хранилище, для библиотеки с полным доступом может использоваться незащищенное хранилище.

Количество библиотек может быть произвольным.

Для обеспечения возможности работы с документами, необходимо зарегистрировать в программе требуемые типы документов и приложения, используемые для их открытия, выполнить настройку интеграции с отдельными приложениями.

Импорт документов в Архив осуществляется различными способами – с вкладки «Документы архива», в пакетном режиме: с созданием списка документов или с использованием файла импорта, а также с помощью действий над объектами.

Имеется возможность импорта дерева папок файловой системы с входящими в них файлами, с созданием объектной структуры с документами архива.

19.9 Архив. Учебное задание

Создайте в учебной настройке библиотеку с полным доступом, указав в качестве пути любой доступный каталог.

Настройте типы документов – в учебных целях будет достаточно настроить тип документа – MS Word, настройте приложение MS Word.

Выберите объект документа в окне проекта. В правой части окна проекта перейдите на вкладку «Документы архива», выполните импорт произвольного файла MS Word в архив. Выделите строку импортированного документа. Откройте контекстное меню документа архива.

Вид контекстного меню (см. рисунок 184) документа зависит от заданных методов запуска приложения. Пункты меню «Открыть», «Печать», «Просмотр», «Совместимые приложения» будут доступны только в том случае, если созданы соответствующие методы запуска для приложения, работающего с данным документом.

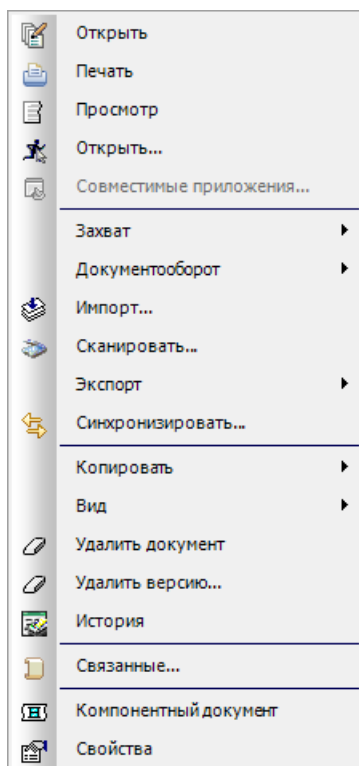


Рисунок 184 Контекстное меню документа архива

Пункт меню «Открыть ...» аналогичен пункту «Открыть», за исключением того, что в первом случае после выбора версии документа будут предложены все возможные типы запуска приложения и методы открытия с помощью других приложений (если таковые заданы). Если же метод только один, то документ будет открыт без дополнительного запроса. Выполните необходимые настройки приложений и откройте документ архива различными способами.

Пункт меню «Совместимые приложения» (активен при наличии совместимых приложений) позволяет после выбора версии открыть список совместимых приложений (и имеющихся методов запуска для них), с помощью которых может быть открыт данный документ. Задайте совместимые приложения, откройте документ архива в совместимом приложении.

Откройте документ архива, создайте новый документ, новую версию.

Используя контекстное меню выделенного на вкладке «Документы архива» документа (версии):

- откройте историю документа;
- удалите версию документа и документ целиком.

Выполнив создание библиотеки с полным доступом в учебной настройке, осуществив простейшие настройки приложений, выполнив импорт документов в архив и поработав с ними, вы сможете, используя контекстную справку и документацию по системе, выполнить настройку защищенного архива и настройку приложений с учетом ваших требований.

19.10 Архив документов. Вопросы для самопроверки

1. Как создать защищенное хранилище документов на сервере с ОС Windows Server 20003?
2. Как подключить хранилище к библиотеке?

3. Как импортировать документ в архив с сохранением исходного имени файла?
4. Что такое «захват» документа архива?
5. Как обеспечить захват документов с восстановлением исходного имени файла?

20 Создание и настройка отчетов

Lotsia PDM содержит встроенный генератор отчетов.

Отчет представляет собой настраиваемую табличную форму, каждая строка данных которой соответствуют найденному при выполнении отчета объекту (далее – «потомок»), а поля (колонки) содержат атрибутивную или другую информацию о потомке или другом объекте. В качестве «другого» объекта может использоваться выделенный объект, либо объект, связанный с найденным потомком (например, родитель данного потомка – см. рисунок 192).

Принципиально, что количество строк данных в отчете равно количеству найденных потомков (дополнительно, в зависимости от настроек отчета, может быть добавлена строка для выделенного объекта). Строки по одному и тому же объекту-потомку могут встречаться в отчете один или несколько раз – последний случай имеет место, например, при выполнении отчета по дереву проектов, где данный потомок встречается несколько раз (имеет множественную входимость).

Поиск потомков может выполняться как по связям – с использованием выделенного объекта (объектов), так и по различным критериям – без использования выделенного объекта и связей.

Результат выполнения отчета может передаваться в следующий отчет, где, в том числе, результаты выполненного отчета могут использоваться в качестве критериев поиска новых потомков.

В зависимости от выбранного стиля отчета, поля в строке могут располагаться либо как в обычной таблице – колонками, либо, при использовании стиля отчета «Свободный» – произвольным образом.

Комбинированные отчеты представляют собой несколько отдельных отчетов, собранных в общей отчетной форме.

Настроенные отчеты отображаются в виде структурированного списка (дерева). Для создания или редактирования отчета используется пункт Главного меню «Отчеты» → «Настройка отчетов» → для создания отчета в контекстном меню выбрать «Добавить отчет...», либо выбрать в дереве требуемый отчет для редактирования.

20.1 Общий порядок создания отчета

Для создания отчета, в дереве отчетов следует выделить папку, в которую будет добавляться новый отчет. В контекстном меню выбрать «Добавить отчет...».

На экране появится окно «Новый отчет» (Рисунок 185).

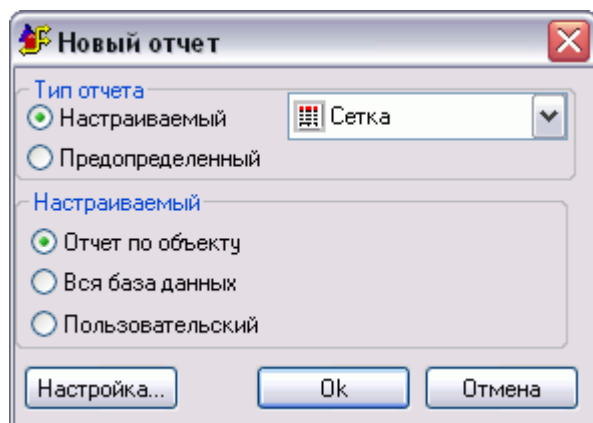


Рисунок 185 Окно «Новый отчет»

Тип отчета «Предопределенный» используется для определения отчетов типа «Спецификация» и «Групповая спецификация (тип А)»).

При выборе типа отчета «Настраиваемый» следует указать метод его формирования – «Отчет по объекту», «Вся база данных» или «Пользовательский». Смысл данного выбора следующий:

«Отчет по объекту» – отчет формируется по выделенному (либо переданному в отчет в качестве аргумента) объекту и связанным с ним объектам, в том числе и потомкам неограниченного уровня вложенности. При поиске потомков с использованием горизонтальных связей – глубина поиска объектов в любом случае – один уровень от выделенного объекта.

«Вся база данных» – отчет выполняется без использования выделенного объекта, информационные объекты ищутся в базе данных, может использоваться предварительный отбор объектов согласно условиям, что убыстряет выполнение отчета.

«Пользовательский» – поиск объектов осуществляется SQL-запросом, формируемым пользователем при настройке отчета.

Тип настраиваемого отчета может быть одним из следующих:

«Сетка» – таблица с видимыми линиями сетки. Строка таблицы соответствует найденному согласно настройкам отчета объекту (потомку), столбец (поле таблицы) – отдельной характеристике потомка или другого, соответствующего данному потомку объекта.

«Свободный» – принцип построения отчета похож на таблицу, но колонки (поля) в отчете располагаются при их добавлении в отчет, по умолчанию, горизонтально в области данных (друг под другом). Пользователю предоставляется возможность вручную размещать колонки в желаемых местах области данных отчета. Другими словами, результат выполнения отчета – также таблица, каждая строка которой соответствует найденному потомку, но представляет собой полосу, в которой размещены поля (колонки) отчета в заданном пользователем месте.

«Таблица» – повторяет стиль «Сетка», но без видимых линий;

«Комбинированный» – представляет собой отдельный отчет, состоящий из нескольких независимых, отдельно настроенных отчетов, включенных при настройке комбинированного отчета в данный отчет;

«Кросстаб» – позволяет формировать отчеты с динамическим количеством столбцов и итогами по строкам и столбцам. Специфика данного стиля отчетов налагает ряд ограничений при настройке отчета;

«Иерархический свободный» и «Иерархический с сеткой» – отличаются тем, что позволяют разворачивать/сворачивать группы. В отчетах данных стилей автоматически создается пустая первая группа, в которой потребуется указать колонку для группировки и в заголовке группы помещается индикатор узла – кнопка разворачивания/сворачивания группы. Индикатор узла автоматически создается в заголовке каждой группы. Стили «Иерархический свободный» и «Иерархический с сеткой» повторяют стили «Свободный» и «Сетка» соответственно, с учетом изложенной особенности иерархического стиля.

После нажатия «Ок» в окне «Новый отчет» (Рисунок 185) откроется окно настройки отчета (Рисунок 186).

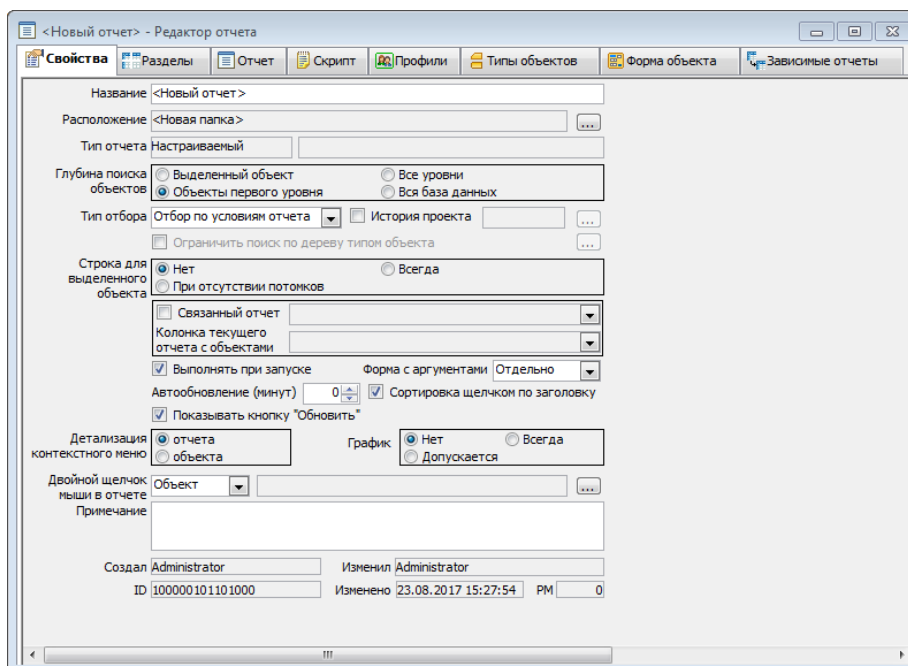


Рисунок 186 Окно «Отчет» вкладка «Свойства»

Вкладка «Свойства» окна отчета (см. рисунок 186) позволяет задать название отчета, глубину поиска объектов, тип отбора – возможны два варианта:

- «Отбор по условиям отчета» – производится выбор объектов, удовлетворяющих условиям отбора, заданным на вкладке «Разделы», согласно заданной глубине поиска;
- «Все объекты» – обработке подвергаются все объекты по типу и направлению связи, указанному на вкладке «Разделы». Если разделов в отчете нет – используется тип связи «Дерево проекта», направление – исходящая (то есть, «вниз» по дереву) согласно заданной глубине поиска. Если на вкладке «Разделы» заданы условия отбора по типам объектов – они игнорируются.

После включения флажка «связанный отчет» возможно указание связанного отчета. Это одна из возможностей, позволяющая по результатам выборки одного отчета формировать другой.

Строки отчета формируются по найденным потомкам, условия поиска которых задаются на вкладке «Разделы» (для пользовательских отчетов данная вкладка не используется, строки пользовательского отчета соответствуют строкам, полученным в результате выполнения SQL-запроса), с учетом условий дополнительного отбора (условия дополнительного отбора задаются на вкладке «Отчет» после нажатия кнопки «SQL», расположенной на инструментальной панели).

При выполнении отчета можно получить строки потомков исходя из текущего состояния связей между объектами, либо, при включении флажка «История проекта», данные, отражающие состояние связей на указанную дату и время.

При включенном флажке «Ограничить поиск по дереву типом объекта», поиск объектов по ветке дерева связей завершается, если найден дочерний объект (потомок) заданного типа. Для задания ограничительного списка типов объектов включите флажок и нажмите кнопку выбора в правой части строки с флажком. В открывшемся диалоге щелкните правой кнопкой мыши и во всплывшем меню выберите пункт «Добавить». Выберите и добавьте требуемые типы объектов. Для удаления типа объектов из списка используйте пункт меню «Удалить».

В зависимости от положения переключателя «Строка для выделенного объекта отчета обязательна», в отчет может добавляться дополнительная строка для выделенного (текущего) объекта. Если выбрано:

- «Нет» – в отчет попадает только информация о потомках, согласно условиям отчета.
- «При отсутствии потомков» – если потомки не найдены, формируется строка, в которой заполнены имеющиеся в отчете колонки с информацией по текущему объекту.
- «Всегда» – вышеуказанная строка добавляется в отчет и в случае наличия потомков.

Вышеуказанный переключатель доступен для отчетов по объекту, если установлена глубина поиска «Первый уровень» или «Все уровни».

В зависимости от положения переключателя «График», результат выполнения отчета может быть представлен графически.

Значение поля «Форма с аргументами» выбирается из списка. Выбранное значение определяет местоположение формы с аргументами. Флажок «Выполнять при запуске» включает/отключает автоматическое выполнение отчета при его запуске. Выключать флажок имеет смысл, когда от пользователя требуется ввод значений аргументов перед выполнением отчета.

В поле «Автообновление (минут)» можно указать интервал времени в минутах для автоматического обновления выполненного отчета. Следует учитывать, что отчеты обновляются не в фоновом режиме.

В поле «Примечание» рекомендуется вводить информацию о назначении и особенностях данного отчета.

Переключатель «График» устанавливается в требуемое положение, в зависимости от применения данного отчета.

Флажок «Выполнять при запуске» следует отключить, если перед началом выполнения отчета требуются какие-либо действия пользователя – например – ввод значений аргументов.

В поле «Двойной щелчок мыши в отчете» можно выбрать одно из значений:

- «Нет». То есть, при двойном щелчке мыши по строке выполненного отчета ничего происходить не будет;
- «Объект», «Действие», «Отчет» – будет открыт объект, либо выполнено соответствующее действие или отчет. В качестве ID открываемого объекта по умолчанию используется значение колонки с именем «isobject_id». Можно также задать выражение, возвращающее ID открываемого объекта (*Рисунок 187*). Аналогично настраивается выполнение действия или отчета.

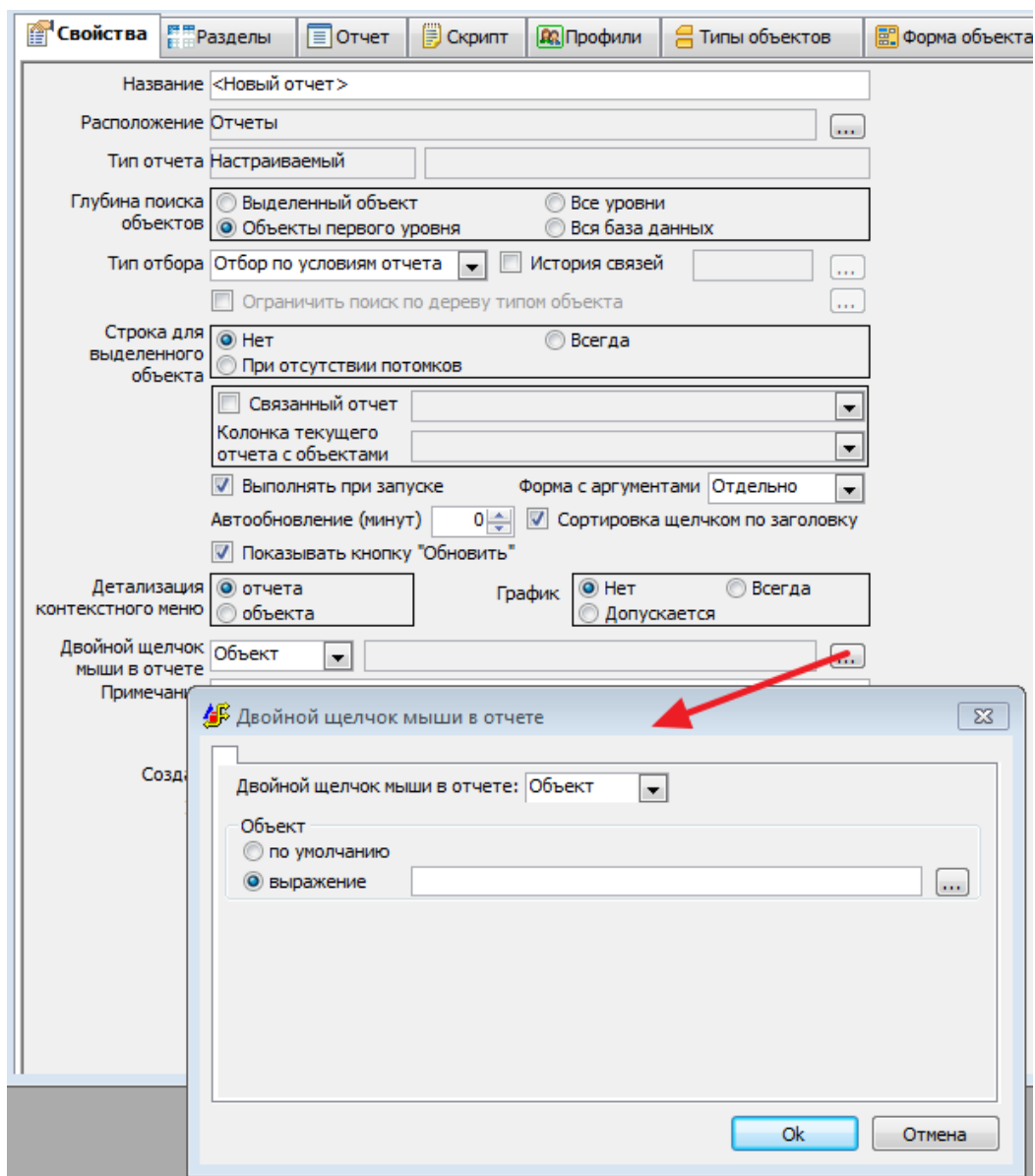


Рисунок 187. Указание обработки двойного щелчка мыши

- при наличии в отчете колонки с указанным в выражении в поле справа именем, содержащей ID объекта, будет открыто окно с соответствующим объектом;
- будут, соответственно, выполнены действие или отчет, указанные в поле справа от данного поля. Отдельно может быть обработан двойной щелчок мыши по каждому полю отчета.

Переключатель «Детализация контекстного меню» меняет вид контекстного меню пользователя для выполненного отчета (Рисунок 188).

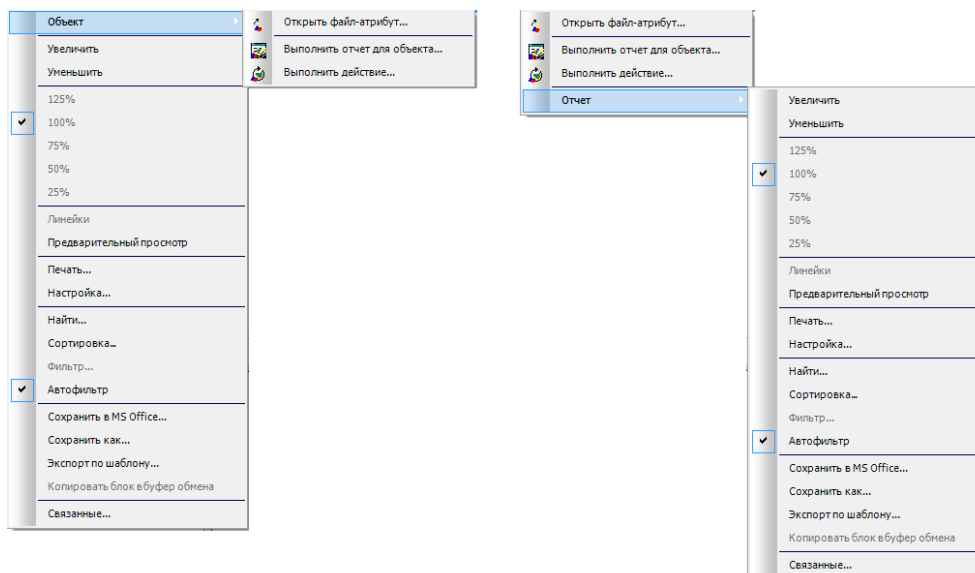


Рисунок 188 Вид контекстного меню выполненного отчета в зависимости от положения переключателя «Детализация контекстного меню»; слева – детализация контекстного меню отчета, справа – детализация контекстного меню объекта

Об изменении расположения отчета в папке отчетов см. п. 20.2

Вкладка «Разделы» (см. рисунок 189) предоставляет возможность задавать способы поиска потомков – тип связи и направление поиска по данной связи. Для каждой добавленной связи (раздела) отдельно задаются типы отбираемых объектов (потомков). У отчетов с типом отбора «Все объекты» разделы могут отсутствовать. Не используются разделы и для «Пользовательских» отчетов. Для каждого раздела задается тип и направление связи, по которому будет производиться поиск объектов в разделе относительно выделенного объекта (объектов) при формировании отчета (область № 1). Для отчетов по всей БД тип и направление связи, указанный для раздела, не имеет значения. Чтобы изменить тип связи для данного раздела, дважды щелкните по полю «Связь» требуемого раздела (область № 1). После выбора типа связи в открывшемся окне можно изменить направление поиска по выбранной связи (если выбрана направленная или подчиненная связь), изменить название раздела и его номер. При дальнейшей настройке, в строку отчета можно будет включить номер раздела (либо его название или идентификатор), и, при условии, что по данным параметрам выполняется сортировка строк отчета и их группировка, разделы в отчете будут фактически представлены в качестве таковых.

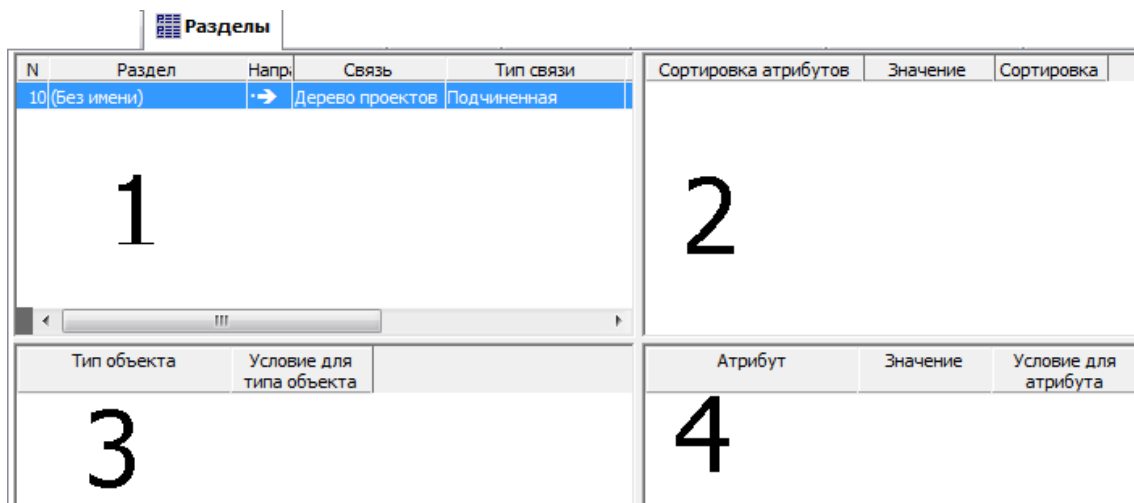


Рисунок 189 Окно «Отчет» вкладка «Разделы»

В области № 2 задается условие сортировки атрибутов для predetermined отчетов типа «Спецификация». Для настраиваемых отчетов данная область не используется.

В области № 3 задается список типов объектов для отбора в отчет по выделенному в области №1 условию поиска (то есть, для данного раздела).

В области № 4 задаются условия отбора по значению атрибута для типа объекта, который выделен в списке типов (области № 3).

В области № 3 в столбце «Условия для типа объекта» задается логический оператор для списка значений атрибута в области № 4.

Вкладка «Отчет» (см. рисунок 190) предназначена для добавления в отчет колонок, а так же позволяет настроить и просмотреть внешний вид и содержание отчета.

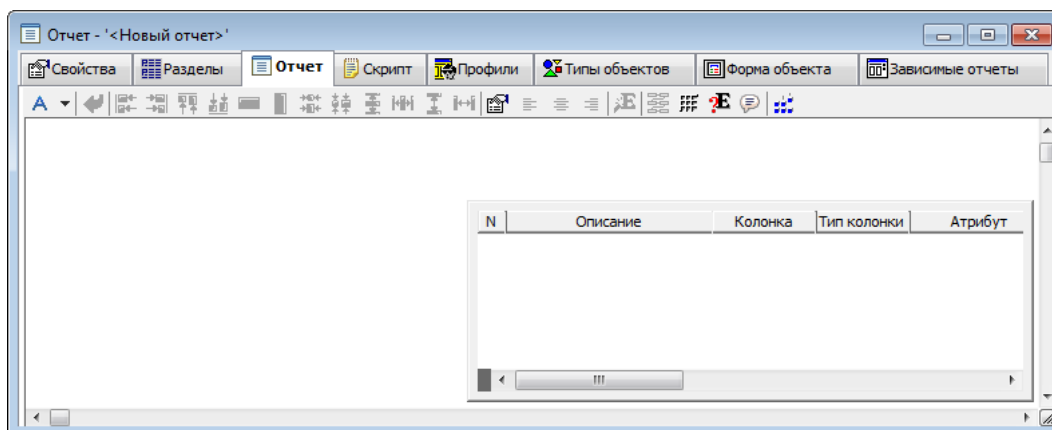


Рисунок 190 Окно «Отчет». Вкладка «Отчет»

Верхняя часть окна вкладки «Отчет» – «Область отчета», представляет собой макет создаваемого отчета. Вложенное окно, расположенное в нижней правой части окна отчета – «Область колонок» отображает список колонок, добавленных в отчет.

Добавление в отчет колонок во многом аналогично созданию колонок форм для типов объектов – см. также раздел 10.2.1.1

Чтобы начать настройку отчета, следует в контекстном меню области колонок выбрать пункт «Добавить». Таким образом, предоставляется возможность выбора свойств колонки отчета, то есть, задания её содержания (см. рисунок 191).

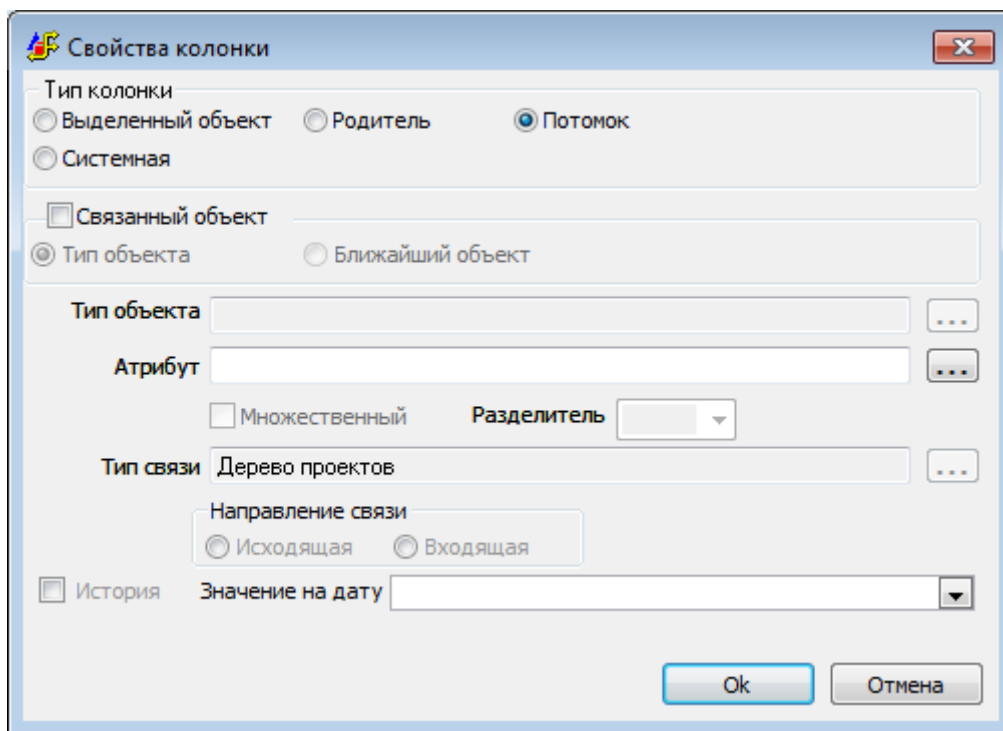


Рисунок 191 Окно выбора свойств колонки отчета

В верхней части окна «Свойства колонки» находится область «Тип колонки», где с помощью переключателя можно выбрать уровень объекта, информация о котором будет помещаться в создаваемую колонку. Переключатели «Родитель» и «Выделенный объект» имеют смысл для отчетов по объекту. В отчетах под «Родителем» подразумевается родитель первого уровня для данного потомка, с учетом направления поиска. То есть, например, при поиске потомков вверх по связи «Дерево проектов», в окне дерева связей потомки окажутся выше выделенного объекта, а «Родители» потомков – ниже найденных потомков – см. рисунок 192. Вверху – настройка раздела для поиска по дереву проекта вверх, внизу – иллюстрация реализации данного поиска.



Рисунок 192 Потомки при поиске по дереву связей вверх. Пример

При любом направлении поиска, родителем для потомков первого уровня будет выделенный объект.

Для выбора атрибута, необходимо нажать кнопку «...» напротив поля «Атрибут» в нижней части окна «Свойства колонки» (см. рисунок 191), а затем выбрать необходимый атрибут или другой параметр (на вкладке «Объект» открывшегося окна доступны описание объекта и некоторые другие параметры).

Значения множественных атрибутов включаются в отчет в виде общей строки, полученной сцеплением значений. Для множественных атрибутов становится активным флажок «Множественный». При включении данного флажка можно указать разделитель, который будет добавлен в строку между отдельными значениями данного атрибута.

Настройка внешнего вида отчета производится аналогично настройке формы. В отчетах имеется возможность указать способ обработки двойного щелчка по полю отчета. Для этого в окне свойств выберите пункт «Двойной щелчок» и далее выберите один из вариантов: «Объект», «Действие», «Отчет» или «Нет».

Настройка обработки двойного щелчка по полю аналогична настройке двойного щелчка по строке отчета.

Отметим, что фактически возможности по обработке выполненного отчета несколько шире – возможно использование скриптов, которые могут выполняться автоматически по следующим событиям «После заполнения данных», «Щелчок мыши» или «Двойной щелчок мыши». В качестве аргументов в функцию скрипта могут быть переданы номер текущей строки и имя выделенного поля выполненного отчета, в процессе выполнения скрипта могут быть считаны и изменены значения полей отчета, изменены данные в БД, выполнены действия, отчеты и т.д.

20.2 Перемещение и копирование отчетов

Имеющийся отчет можно скопировать. Для копирования отчета следует открыть дерево отчетов, выделить требуемый отчет, и при нажатой клавише <Ctrl> перетащить отчет в другую папку. Также для выделенного в дереве отчета можно использовать пункты контекстного меню «Копировать», «Вставить», либо – соответствующие быстрые клавиши <Ctrl>+<C> и <Ctrl>+<V>.

Имеющийся отчет может быть перемещен из одной папки в другую. Для этого, на вкладке «Свойства» окна настройки отчета, в поле «Расположение» следует указать требуемую папку, либо перетащить отчет в требуемую папку в дереве отчетов.

20.3 Использование отчета в качестве рабочего (интерфейсного) окна конечного пользователя

Возможность запуска действий и скриптов по щелчку мыши в строке и отдельных полях отчета, а также использование кнопок в отчете, обеспечивает возможность применения отчета в качестве рабочего окна конечного пользователя программы.

После выполнения действия, например, по щелчку по кнопке в форме отчета, отображаемые в отчете данные можно автоматически обновить. Для этого можно использовать либо переключатель «Обновить отчет после выполнения действия» либо, вернуть в соответствующие колонки отчета значения переменных из выполненного действия (Рисунок 193).

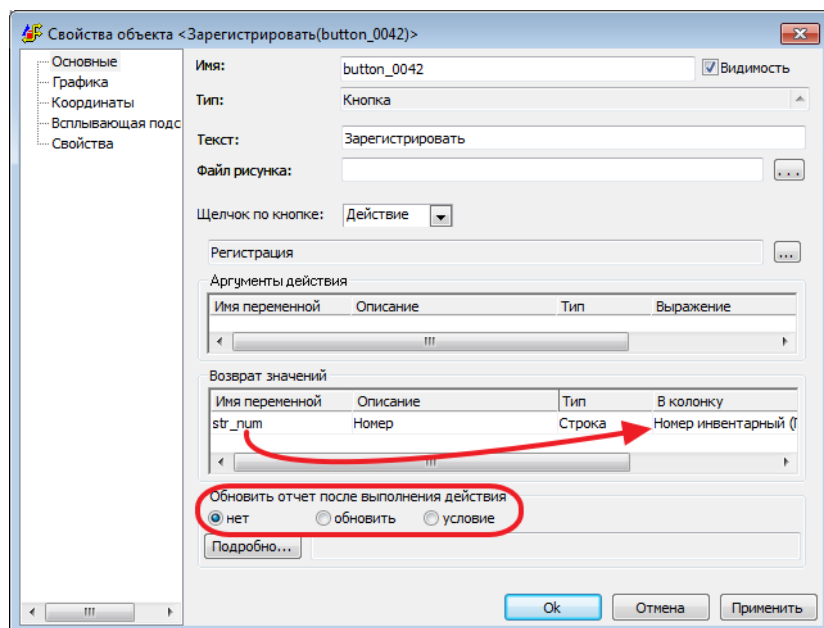


Рисунок 193 Задание свойств кнопки в отчете. Пример

Подобные отчеты могут открываться как в отдельном окне, так и использоваться в качестве форм (вкладок) объектов (см. п. 20.4).

20.4 Использование отчета в качестве формы (вкладки) для объекта в дереве связей

Отчет может быть добавлен в список форм, которые могут подключаться к типам объектов при настройке профилей (см. п. 10.3.5), после чего будут отображаться в дереве связей как дополнительные вкладки. Отчетная форма может использоваться также в качестве внедренной формы (см. п. 10.2.1.1).

Для добавления отчета в список форм, на вкладке «Форма объекта» окна настройки отчетов нажмите «Создать» и укажите наименование формы и имя вкладки. Форма будет создана (см. рисунок 194).

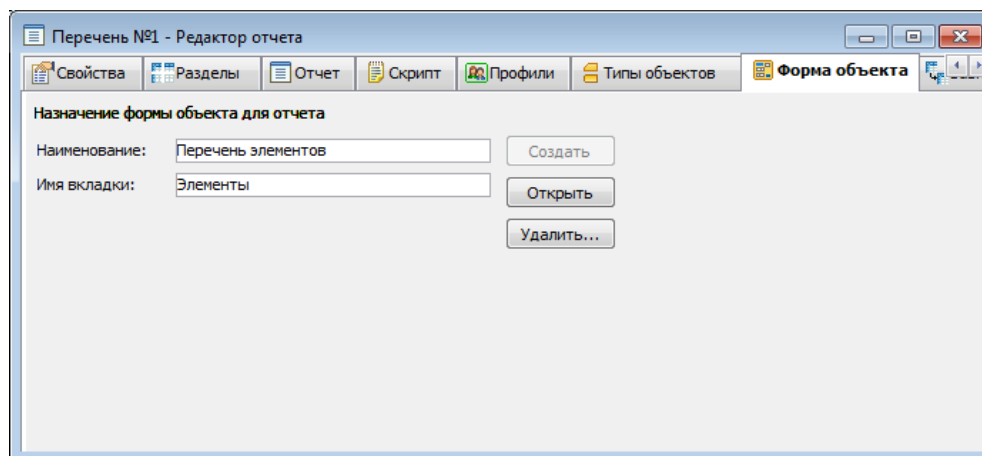


Рисунок 194 Отчет добавлен в список форм для объектов

20.5 Примеры создания отчетов

Сформируем отчет типа «Список элементов» (в зависимости от настройки это может быть «Опись...», «Книга регистрации...», «Журнал учета...» или что-то подобное) применительно к созданному учебному примеру настройки. В зависимости от Вашей настройки, названия отчета, колонок и т.д. могут отличаться от описываемых.

Подобные отчеты часто используются в совершенно различных предметных областях. В данном случае рассмотрим пример учебной настройки, посвященной инвентарному учету компьютеров, расположенных в помещениях, на разных этажах здания – см. рисунок 195. Выделенные подчеркиванием слова соответствуют типам объектов настройки. При проведении занятий, отчет сформируем по конкретной созданной учебной настройке, при самостоятельной работе можно последовательно повторить описанные ниже примеры.

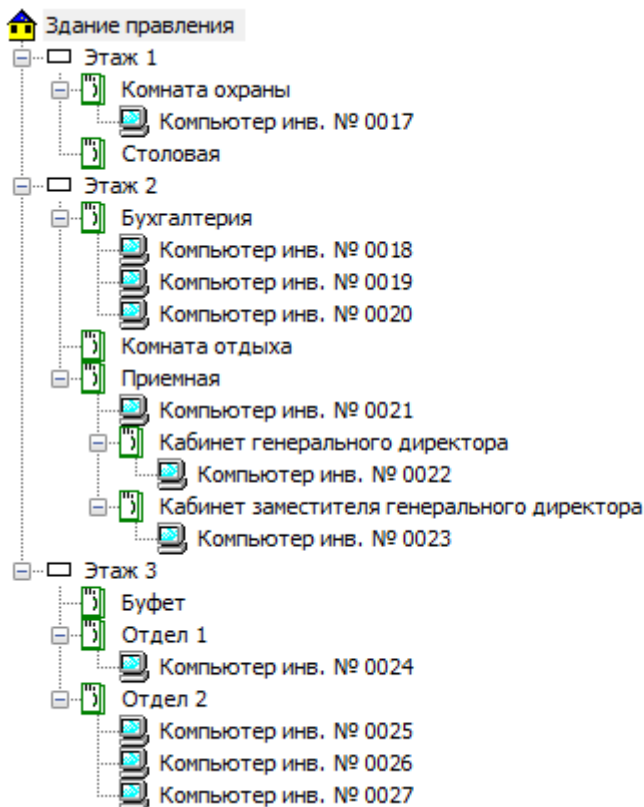


Рисунок 195 Учебный проект «Учет компьютеров»

Пусть имеются следующие присвоенные атрибуты объектов у потомков: «Номер инвентарный» – (строковое значение, формат – порядковый номер, дополненный впереди нулями до стандартной длины, например, 4 символа: 0001, 0002 и т.д.), «Дата постановки на инвентарный учет» – (дата, формат dd.mm.yyyy, задан для атрибута).

Все объекты имеют описания, соответствующие названиям объектов в дереве (см. рисунок 195).

20.5.1 Последовательность создания отчета

Для создания отчета:

Выберите: «Главное меню» → «Отчеты» → «Настройка отчетов».

В открывшемся окне папок отчетов в контекстном меню выберите «Добавить папку...», создайте папку «Учебные». В контекстном меню созданной папки выберите «Добавить отчет...», в открывшемся окне «Новый отчет» установите: тип отчета «Настраиваемый», «Сетка», «Отчет по объекту» нажмите «Ok».

На вкладке «Свойства» окна настраиваемого отчета в поле «Название» введите: «Журнал учета компьютеров», сохраните отчет (нажмите <Ctrl> + <S>).

Далее укажите:

Глубина поиска объектов – «Все уровни»

Тип отбора – «Отбор по условиям отчета»

Строка для выделенного объекта обязательна – «Нет»

Сохраните отчет.

Перейдите на вкладку «Разделы».

В контекстном меню верхней левой части окна выберите «Добавить». По умолчанию будет добавлен новый раздел (то есть, задано условие поиска потомков) «<Без имени>», направление поиска вниз («→») по связи «Дерево проектов». Название раздела, если требуется, можно изменить ручным вводом, направление поиска и тип связи – двойным щелчком по полю «связь».

В нашем учебном отчете будет 1 раздел, изменять его название, связь и направление поиска нет необходимости.

В контекстном меню нижней левой части окна выберите «Добавить», в открывшемся окне выберите добавляемые типы объектов – потомков, для нашего учебного отчета – «Компьютер». Более на вкладке «Разделы» ничего не меняем.

Сохраните отчет.


Перейдите на вкладку «Отчет».

Информация о потомках может включаться в отчет добавлением колонок соответствующих атрибутов (или других полей, например – описания объекта). Мы добавим в отчет описание потомка, его инвентарный номер и дату постановки на инвентарный учет.

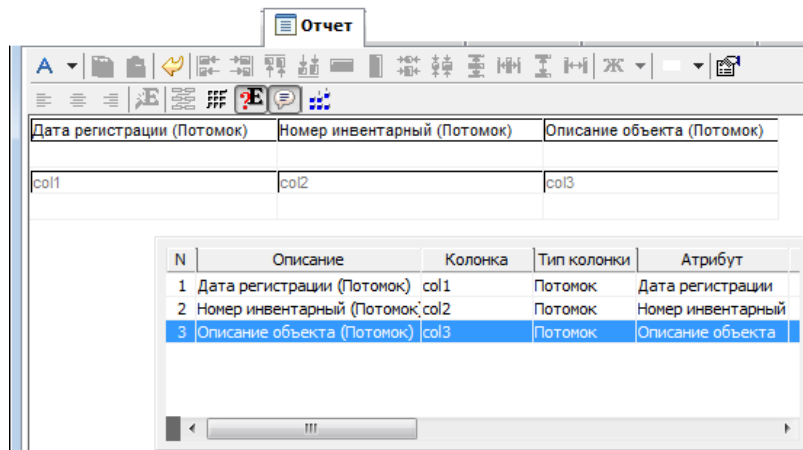
В контекстном меню области колонок (см. рисунок 190) выберите «Добавить».

В открывшемся окне (см. рисунок 191) укажите:

тип колонки – «Потомок»;

нажав кнопку  правее поля «Атрибут», на вкладке «Выбор» открывшегося окна выберите атрибуты «Дата регистрации» и «Номер инвентарный». После нажатия «Ок» в окне выбора атрибутов и в окне свойств колонки, добавленные колонки отобразятся в области колонок и в области отчета.

Аналогично добавьте в отчет колонку с описанием потомка (отличие: в окне выбора атрибутов перейдите на вкладку «Объект» и там выберите «Описание объекта»). В результате вкладка «Отчет» будет выглядеть приблизительно так – см. рисунок 196.



Дата регистрации (Потомок)	Номер инвентарный (Потомок)	Описание объекта (Потомок)
col1	col2	col3

N	Описание	Колонка	Тип колонки	Атрибут
1	Дата регистрации (Потомок)	col1	Потомок	Дата регистрации
2	Номер инвентарный (Потомок)	col2	Потомок	Номер инвентарный
3	Описание объекта (Потомок)	col3	Потомок	Описание объекта

Рисунок 196 Добавленные колонки отчета

Названия колонок отчета, их графическое оформление и положение можно настраивать заданием свойств одного или нескольких выделенных объектов (аналогично настройке форм, выбрав в контекстном меню выделенного объекта «Свойства объекта»).

Формат отображения данных задается в пункте «Вид» свойств объекта. Задайте правильный формат отображения даты – см. рисунок 197

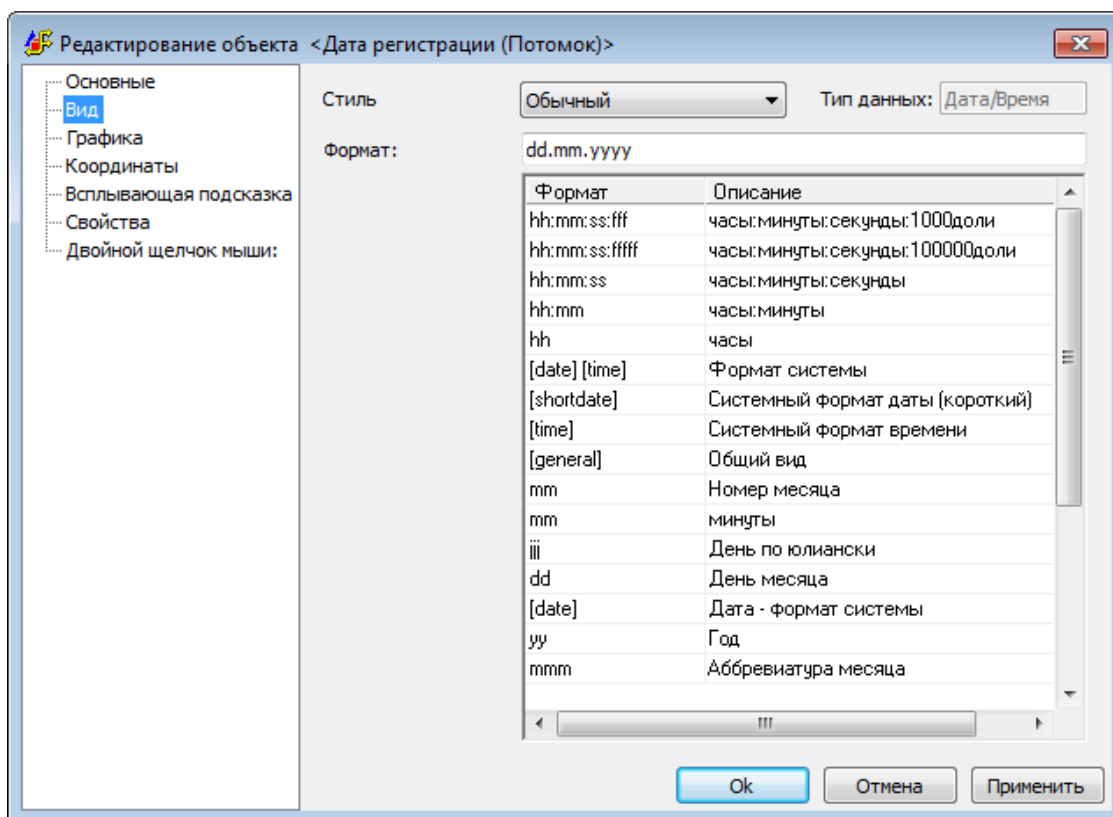


Рисунок 197 Задание формата отображения даты

При выполнении настроек, необходимо различать области отчета. Изначально в отчете имеются области:

- Заголовок отчета;
- Область данных;
- Область итогов;
- Нижний колонтитул;

Поместив курсор в область отчета (не в область колонок, не выделяя объекты в области отчета), из контекстного меню выберите «Свойства областей». Чтобы нагляднее отобразить области на экране, задайте различные цвета для областей отчета, для области нижнего колонтитула установите высоту около 80 единиц.

Сохраните отчет.

В области итогов создайте вычисляемое поле для подсчета количества потомков, его выражение может быть, например, следующим:

'Всего компьютеров: '+ Count(col2 for All)+' шт.'

где:

Count(col2 for All) – функция, определяющая количество заполненных строк в колонке (в данном случае – col2) для всего отчета (то есть, for All).

В кавычках указаны постоянные строковые значения.

Для перемещения полей отчета пользуйтесь мышью или клавиатурой. Можно также редактировать свойства объектов формы отчета – колонок (используя пункт «Координаты»), задавая значения координат, ширины и высоты. При использовании клавиатуры, клавиши со стрелками перемещают выделенный объект, при нажатой клавише <Shift> стрелками изменяется размер выделенного объекта, при нажатой клавише <Ctrl> указанные операции осуществляется ускоренно.

Подробнее об оформлении отчета – см. документы «Редактор форм» и «Руководство пользователя Lotsia PDM», раздел «Настройка отчетов».

В контекстном меню области заголовка отчета выберите «Выбрать» → «Все объекты». Нажмите клавишу со стрелкой, направленной вниз, переместите заголовки колонок ниже, освобождая место для общего заголовка отчета. Продолжайте нажимать ту же клавишу, но при нажатой клавише «Shift», увеличивая тем самым высоту заголовков колонок. Обратите внимание, что размер области заголовков при данных манипуляциях автоматически увеличивается.

Для отображения номера текущей страницы отчета и их общего количества, в области нижнего колонтитула создайте вычисляемое поле и введите в него следующее выражение:

'Стр. '+Page()+' из '+ PageCount()

Сохраните отчет.

В области заголовка отчета создайте вычисляемое поле, введите туда заголовок для отчета, например выражение такого вида:

'Журнал инвентарного учета компьютеров'

Для заголовка можно использовать и текстовое поле, но применение вычисляемого поля позволит нам в дальнейшем усовершенствовать отчет.

Чтобы визуализировать положение заголовка, используя контекстное меню данного вычисляемого поля («Свойства объекта» → «Графика») настройте рамку данного вычисляемого поля (например, «Тень»). Увеличьте размер шрифта.

Поместите поле с заголовком на передний план («Свойства объекта» → «Координаты» → Область «Передний план» см. рисунок 198), после чего, используя клавиатуру, установите требуемое положение и размер заголовка.

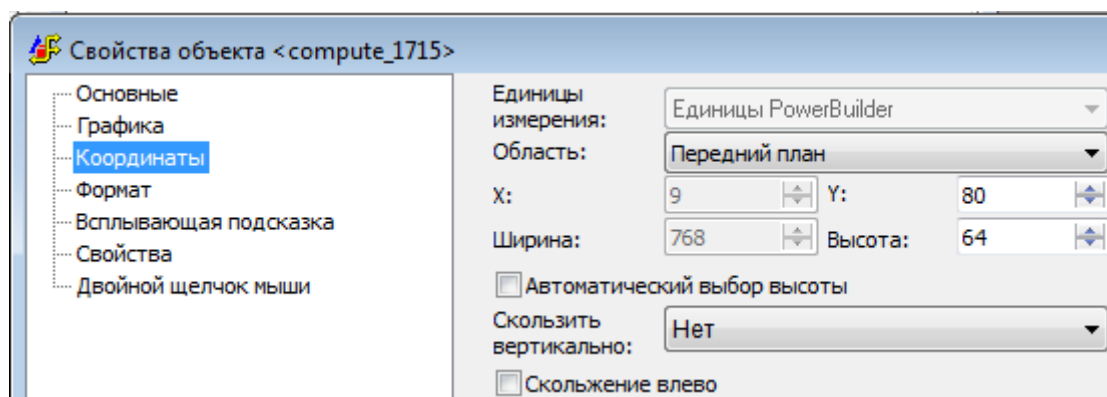


Рисунок 198 Перемещение объекта на передний план

В результате описанных манипуляций отчет примет примерно следующий вид – см. рисунок 199.

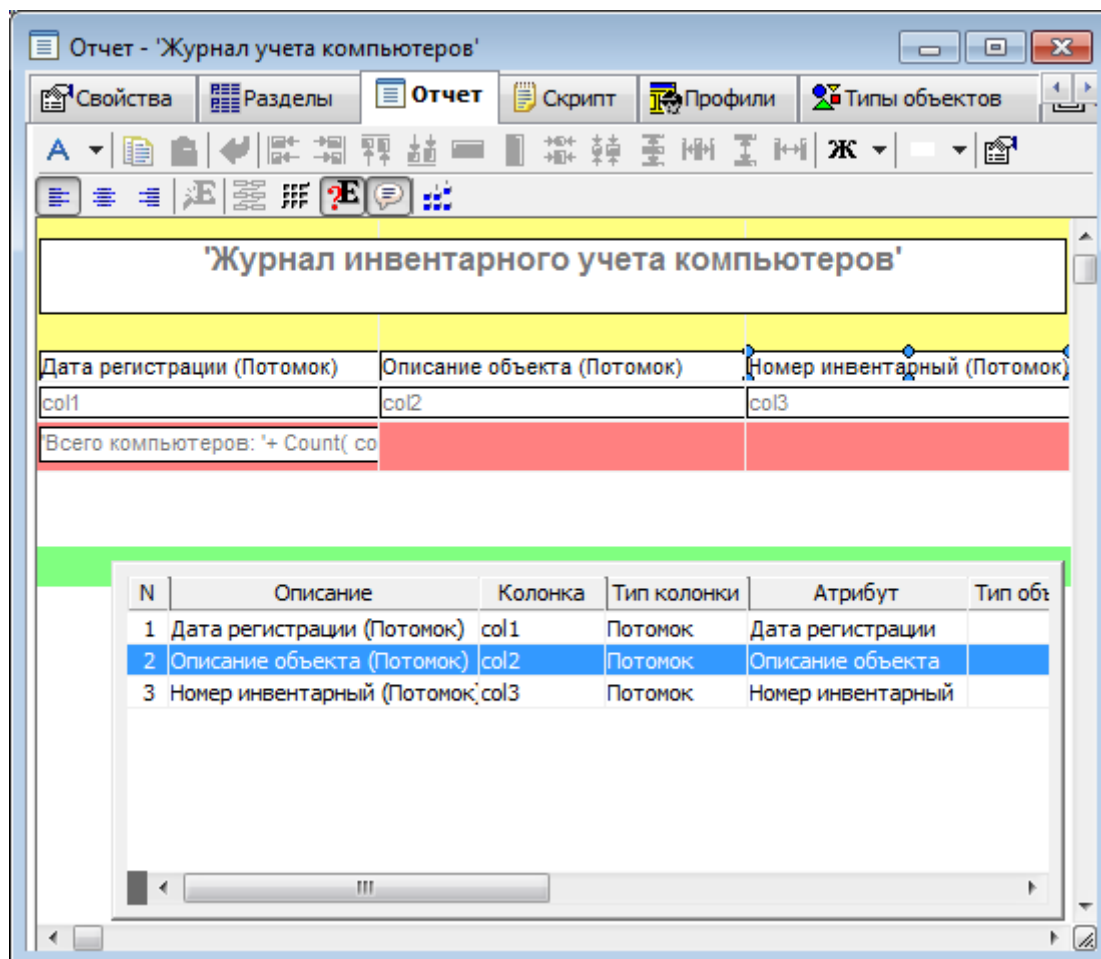


Рисунок 199 Отчет с выделенными областями

Выполните созданный отчет, запустив его из контекстного меню корневого объекта Вашего проекта (например, объект «Здание правления», см. рисунок 195). Результат выполнения может быть приблизительно таким – см. рисунок 200

Отчет 'Журнал учета компьютеров'

Отчет Аргументы

Журнал инвентарного учета компьютеров

Дата регистрации (Потомок)	Описание объекта (Потомок)	Номер инвентарный (Потомок)
03.10.2016	Компьютер №1.арх/16	1.арх/16
03.10.2016	Компьютер №5.арх/16	5.арх/16
03.10.2016	Компьютер №4.арх/16	4.арх/16
03.10.2016	Компьютер №3.арх/16	3.арх/16
03.10.2016	Компьютер №11.арх/16	11.арх/16
03.10.2016	Компьютер №2.арх/16	2.арх/16
26.10.2016	Компьютер №10.арх/16	10.арх/16
26.10.2016	Компьютер №6.арх/16	6.арх/16
26.10.2016	Компьютер №9.арх/16	9.арх/16
26.10.2016	Компьютер №7.арх/16	7.арх/16
26.10.2016	Компьютер №8.арх/16	8.арх/16
Всего компьютеров: 11 шт.		


Обновить

Рисунок 200 Выполненный отчет

Заголовки колонок и другие элементы отчета можно настроить по собственному усмотрению. При выполнении таких настроек, не закрывая окна выполненного отчета, перейдите в окно настройки отчета, выполните необходимые настройки, сохраните отчет.

Вернитесь в окно выполненного отчета, обновите его.

20.5.2 Сортировка строк отчета

Отметим, что сортировать можно как выполненный отчет, так и отчет при его настройке – см. «Руководство пользователя Lotsia PDM», разделы «Настройка отчетов» и «Выполнение отчетов». Для задания колонок и направлений сортировки нажмите  на панели инструментов, перенесите колонки, по которым осуществляется сортировка, в правую половину окна «Сортировка», установив их там в нужной последовательности и снимите или установите флажки направления сортировки.

Сортировка выполняется в алфавитном порядке, независимо от формата колонки сортировки. Для учебного отчета мы преднамеренно использовали данные, имеющие формат, обеспечивающий правильную сортировку.

В реальном отчете, для обеспечения правильной сортировки, например, по инвентарному номеру, необходимо выполнить дополнительную настройку.

Например, для определенности, пусть инвентарный номер имеет следующий формат:

X.B/YY, где:

X – порядковый номер, число, количество разрядов различное

. – разделитель

В – буквенный индекс, может быть фиксированный или различаться, в том числе быть переменной длины

/ – разделитель

YY – последние 2 цифры года даты регистрации

Сортировку необходимо выполнить в порядке возрастания номеров, то есть

Например, так: 2.ке/15, 17.в/15, 18.тмп/15, 19.W14/15, 21.цмк/15

а не: 17.в/15, 18.тмп/15, 19.W14/15, 2.ке/15, 21.цмк/15.

В подобных случаях используется следующий типовой прием – в области данных создается вычисляемое поле, где обрабатывается содержимое колонки (колонок) по которым необходимо выполнить сортировку, с целью приведения данных к числовому виду, либо к строкам вида, соответственно: 0002, 0017, 0018, 0019, 0021. По указанному вычисляемому полю и производится сортировка строк отчета (для удобства использования дайте данному вычисляемому полю название, например, «Порядковый номер» или «№» – см. п. 10.2.7), в готовом отчете данное поле можно не показывать (скрыв его с помощью задания соответствующих свойств, или ширину колонки равной 0). В выражения подобных вычисляемых полей обычно используют строковые функции, например, в рассмотренном случае выражение может быть таким:

number(Mid(col3 , 1 , Pos(col3 , '.')-1)), или таким:

string(number(Mid(col3 , 1 , Pos(col3 , '.')-1)), '0000') где:

col3 – колонка «Номер инвентарный»;

Pos(col3 , '.') – возвращает номер позиции разделителя «.»;

Mid – строковая функция, получает из строки подстроку заданной длины, начиная с указанной позиции;

Number – функция, преобразующая строку в число;

String(<первый аргумент>,<второй аргумент>) – функция, преобразующая число (первый аргумент) в строку требуемого формата (второй аргумент)

Измените данные (или используйте фактически имеющиеся значения атрибутов) в Вашей настройке, и выполните сортировку строк рассмотренным выше способом.

20.5.3 Нумерация строк отчета:

Для нумерации строк отчета можно использовать вычисляемое поле, содержащее функцию GetRow(), которая возвращает порядковый номер строки, или использовать вычисляемое поле, аналогичное примененному нами для сортировки, например:

number(Mid(col3 , 1 , Pos(col3 , '.')-1))


Добавьте в отчет нумерацию строк.

20.5.4 Группировка строк отчета

Использование группировки подразумевает то, что предварительно строки отчета были отсортированы по той же колонке отчета, по которой выполняется группировка.

Для рассмотренного учебного отчета сгруппируем строки по датам постановки на инвентарный учет.

Предварительно задайте сортировку строк, указав колонки сортировки в следующем порядке: «Дата регистрации», «Порядковый номер» (см. раздел «Сортировка строк отчета»). Сортировку задайте по возрастанию.

Для задания группировки нажмите кнопку  инструментальной панели.

В открывшемся окне «Группировка» (см. рисунок 201) задайте параметры группы – перетащите в правую часть окна колонку, по которой выполняется группировка – «Дата постановки на инвентарный учет», укажите высоту около 150 единиц для добавляемых областей отчета, а так же для визуализации этих областей пометьте их различными цветами.

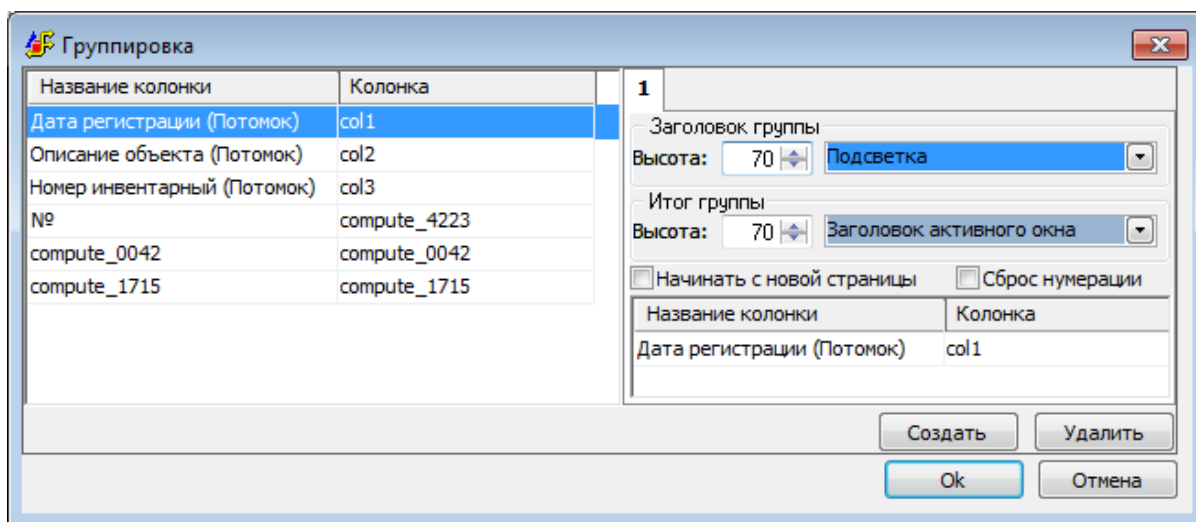


Рисунок 201 Задание параметров группы

Нажмите «ОК», сохраните отчет.

В области заголовка группы (см. рисунок 202) создайте вычисляемое поле, введя в него выражение:

Col1 где:

Col1 – колонка «Дата регистрации»

В области итогов группы (см. рисунок 202) так же создайте вычисляемое поле, введя в него выражение:

'За '+' date (col1) +' присвоено'+ '~r~n'+ Count(col3 for group 1)+ ' инв. ном.' где:

Col1 – колонка «Дата регистрации»

'~r~n' – специальная строка, обеспечивает жесткий перенос текста

col3 – колонка «Номер инвентарный»

Count(col3 for group 1) – функция, определяющая количество строк с непустыми колонками col3 в группе.

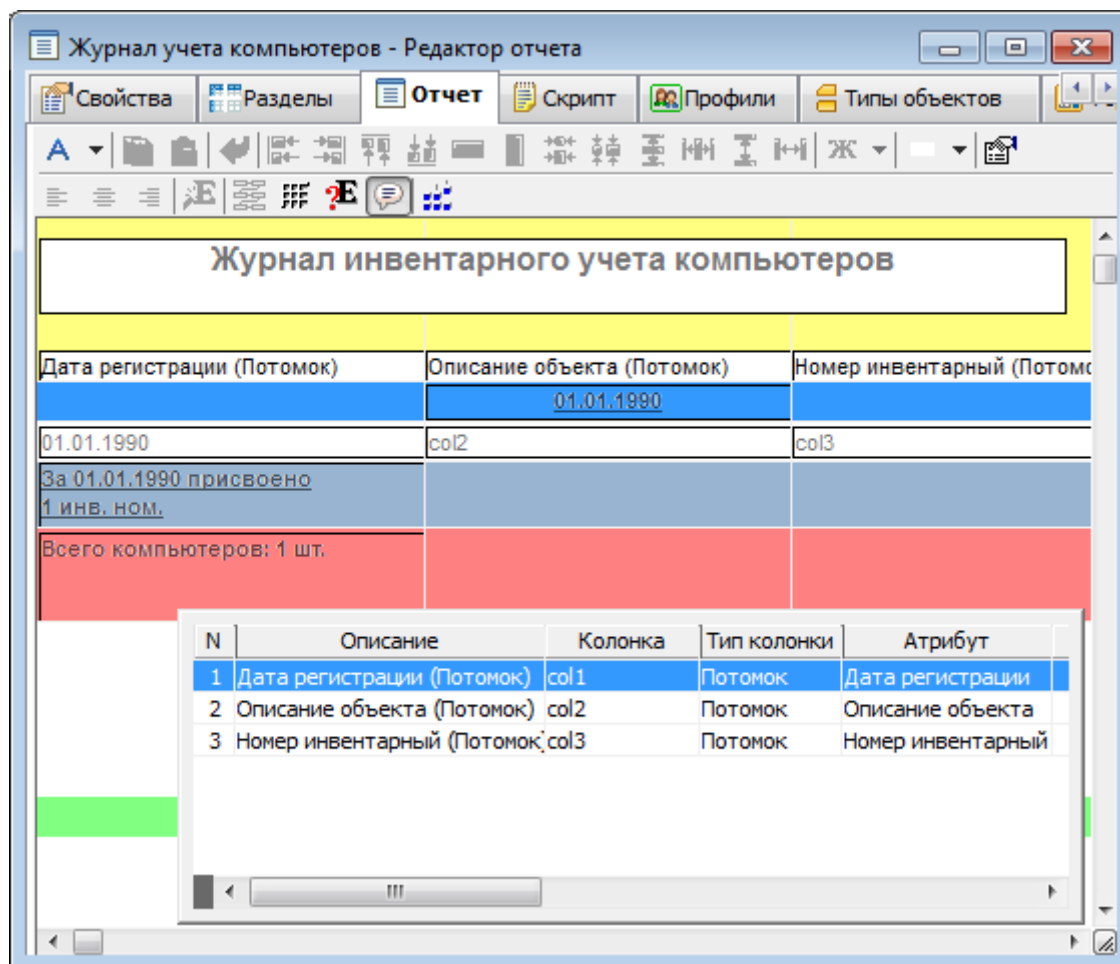


Рисунок 202 Настроенный отчет с группировкой. Пример

Сохраните отчет.

После выполнения данный отчет может выглядеть приблизительно следующим образом – см. рисунок 203:

Отчет 'Журнал учета компьютеров'

Отчет Аргументы

Журнал инвентарного учета компьютеров

Дата регистрации (Потомок)	Описание объекта (Потомок)	Номер инвентарный (Потомок)
<u>03.10.2016</u>		
03.10.2016	Компьютер №1.apx/16	1.apx/16
03.10.2016	Компьютер №2.apx/16	2.apx/16
03.10.2016	Компьютер №3.apx/16	3.apx/16
03.10.2016	Компьютер №4.apx/16	4.apx/16
03.10.2016	Компьютер №5.apx/16	5.apx/16
03.10.2016	Компьютер №11.apx/16	11.apx/16
За 03.10.2016 присвоено 6 инв. ном.		
<u>26.10.2016</u>		
26.10.2016	Компьютер №6.apx/16	6.apx/16
26.10.2016	Компьютер №7.apx/16	7.apx/16
26.10.2016	Компьютер №8.apx/16	8.apx/16
26.10.2016	Компьютер №9.apx/16	9.apx/16
26.10.2016	Компьютер №10.apx/16	10.apx/16
За 26.10.2016 присвоено 5 инв. ном.		
Всего компьютеров: 11 шт.		

Обновить

Рисунок 203 Фрагмент выполненного отчета с группировкой. Пример

Отметим, что в качестве колонки для группировки можно использовать и различные системные колонки. Например, при использовании системной колонки «Раздел отчета (номер)», группировка строк отчета может выполняться в соответствии с разделами отчета, заданными на вкладке «Раздел».

20.5.5 Обработка повторяющихся строк отчетов

В рассматриваемом выше примере (модели данных) отсутствуют повторяющиеся строки отчета (любой из компьютеров может быть расположен только в одном помещении). Подобная единичная входимость объектов учета в реальной настройке часто не соблюдается.

Для иллюстрации множественной входимости, рассмотрим в качестве объектов учета, например, объекты документов – актов проверки противопожарного состояния помещений. Будем считать, что такие акты могут оформляться по результатам разовых проверок нескольких произвольно выбранных помещений здания.

Тип объекта учета – «Акт проверки», внешний вид дерева проекта – см. рисунок

Атрибуты объектов учета, используемые в отчете – «Регистрационный номер» и «Дата регистрации». В остальном модель данных аналогична рассмотренной выше.

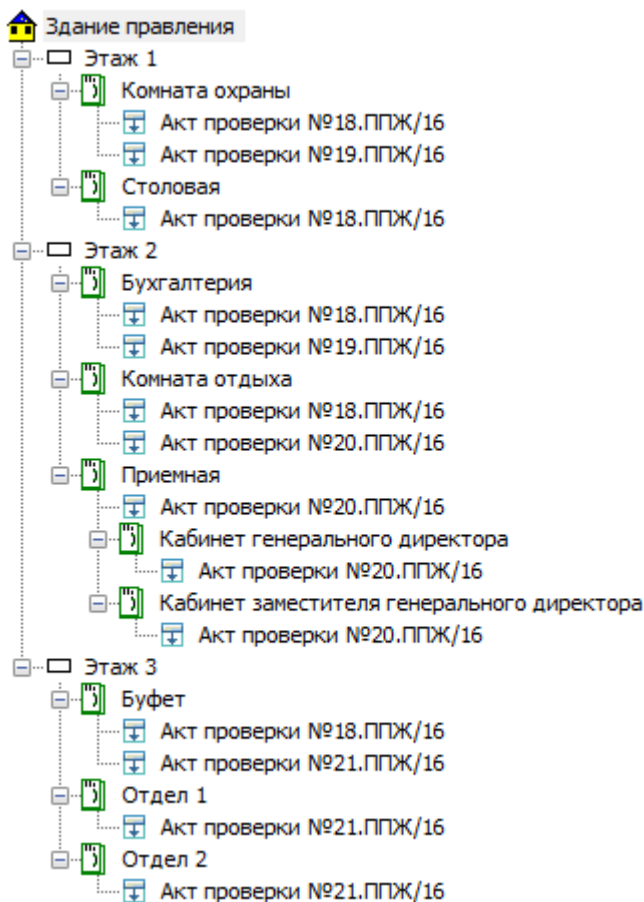


Рисунок 204 Учебный проект «Учет актов проверки противопожарного состояния помещений»

Создадим учебный отчет «Книга регистрации актов проверок», аналогичный рассмотренному выше отчету «Журнал регистрации компьютеров».

Для создания нового отчета можно выполнить копирование отчета «Журнал регистрации компьютеров» (о копировании отчетов см. «Руководство пользователя Lotsia PDM»), и далее отредактировать его элементы:

- На вкладке «Разделы» укажите тип объекта – «Акт проверки».
- На вкладке «Отчет» измените атрибуты в свойствах колонок (выделите строку с колонкой в области колонок, в контекстном меню выберите «Свойства», в окне свойств укажите требуемый атрибут).
- В области отчета отредактируйте содержимое текстовых полей с названиями колонок, а так же соответственно измените строки в вычисляемых полях.

Сохраните отчет. В результате отчет может выглядеть примерно так – см. рисунок 205:

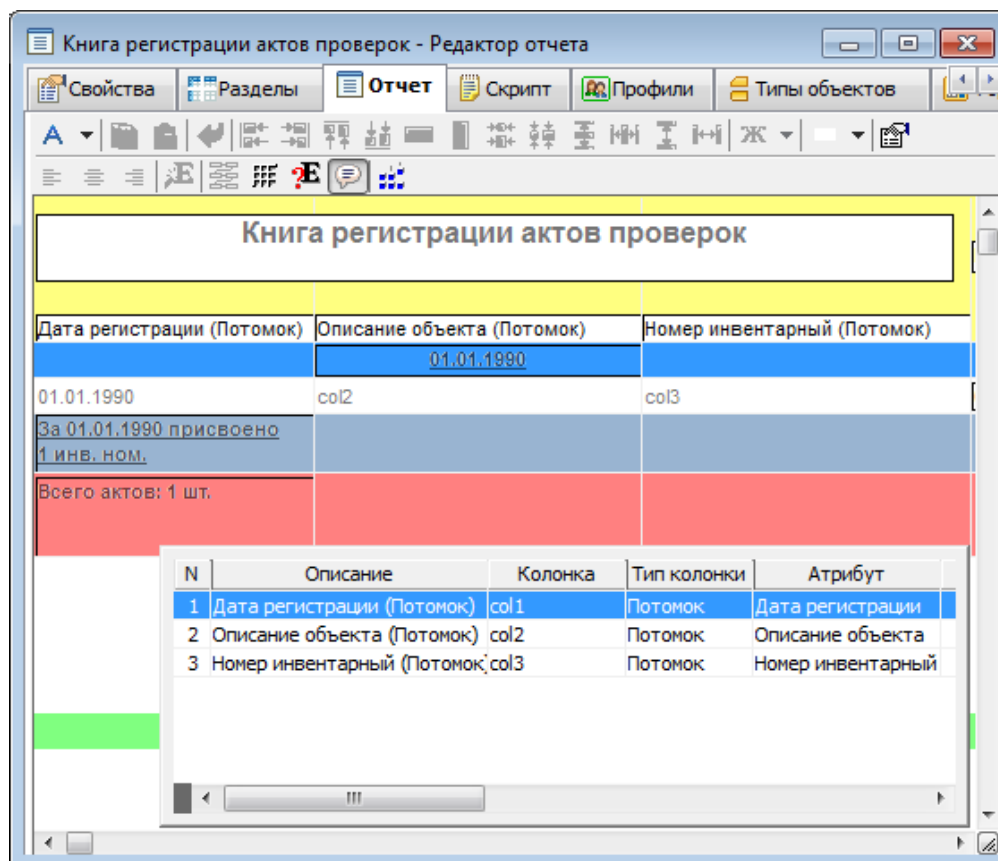


Рисунок 205 Настроенный отчет «Книга регистрации актов проверок»

Сортировку и группировку отчета следует сохранить как в исходном отчете.

Выполнив данный отчет по объекту «Здание правления» убедимся, что строки потомков в отчете повторяются, и количество актов определяется неверно – см. рисунок 206.

Для того, чтобы потомок использовался для определения итогов отчета или группы единожды, следует использовать опцию Distinct в функциях, определяющих итоги отчета (группы). Например, выражение вычисляемого поля для определения количества потомков в группе может иметь следующий вид:

'За '+ date (col1) +' присвоено'+ '~r~n'+ Count(col3 for group 1 distinct col2)+ ' инв. ном.' ' где:

Col1 – колонка «Дата Регистрации»

'~r~n' – специальная строка, обеспечивает жесткий перенос текста

col3 – колонка «Номер инвентарный»

col2 – колонка, по которой контролируется уникальность объекта (в данном случае все объекты имеют различные описания, поэтому можно использовать колонку «Описание объекта», что и сделано). Для того чтобы быть абсолютно уверенным в правильности подведения итогов по уникальным объектам, рекомендуется добавлять в отчет системную колонку «Объект(код)», в которой будет отображаться ID объекта – потомка. Ввиду того, что уникальность ID контролируется системой автоматически, использование этой колонки для контроля уникальности предпочтительно в большинстве случаев.

Отметим, что если данная колонка присутствует в отчете (хотя бы в области колонок, в области отчета её можно не показывать), то при двойном щелчке левой клавишей мыши по строке потомка в выполненном отчете, в отдельном окне будет открыт объект с данным ID.

Добавьте колонку «Объект(код)» в отчет, используйте опцию Distinct в выражения вычисляемых полей, определяющих итоги по группе и отчету в целом.

Сохраните отчет, обновите выполненный отчет, убедитесь, что теперь он выполняется верно – см. рисунок 207.

Отчет 'Книга регистрации актов проверок'


Отчет Аргументы

Книга регистрации актов проверок		
Дата регистрации (Потомок)	Описание объекта (Потомок)	Номер инвентарный (Потомок)
	01.08.2016	
01.08.2016	Акт проверки №18.ППЖ/16	18.ППЖ/16
	Акт проверки №19.ППЖ/16	19.ППЖ/16
За 01.08.2016 присвоено 2 инв. ном.		
	03.11.2016	
03.11.2016	Акт проверки №20.ППЖ/16	20.ППЖ/16
За 03.11.2016 присвоено 1 инв. ном.		
	16.11.2016	
16.11.2016	Акт проверки №21.ППЖ/16	21.ППЖ/16
За 16.11.2016 присвоено 1 инв. ном.		
Всего актов: 4 шт.		

Обновить

Рисунок 207 Выполненный отчет, в котором применено подавление повторяющихся значений и использовано определение итогов по уникальным значениям

Удалить из отчета повторяющиеся строки можно также с помощью фильтрации.

Для задания условия фильтрации нажмите кнопку  инструментальной панели. Выражение фильтра может быть следующим:

`isobject_id <> isobject_id [-1] or GetRow()=1`

Первая часть выражения (до `or`) сравнивает значение `isobject_id` (`isobject_id` – имя колонки «Объект(код)») в данной строке отчета и в предыдущей. Строка войдет в отчет в случае, если значения не повторяются.

Для первой строки любого отчета данное условие не выполняется, так как для первой строки значения `isobject_id [-1]` не существует

Чтобы первая строка автоматически не исчезала из отчета, в выражение фильтра добавлена вторая часть – функция GetRow() возвращает номер текущей строки.

Строки должны быть отсортированы по той же самой колонке, в данном случае по колонке «Объект(код)».

Использование фильтрации, в отличие от подавления повторяющихся значений, влияет на итоги по отчету или группе.

Отметим, что имеется возможность фильтровать как выполненный, так и настраиваемый отчет. В некоторых случаях, например, при использовании в выражении фильтра значений, содержащихся в различных строках отчета (как в приведенном выражении), результат, полученный при фильтрации выполненного отчета, может отличаться от того, который будет получен при использовании аналогичного фильтра при настройке отчета. В подобных случаях следует использовать фильтрацию при настройке отчета.

Отфильтруйте повторяющиеся строки в Вашем отчете, используя фильтр при настройке отчета.

20.5.6 Использование аргументов отчета

Отчет, в котором глубина поиска объектов не «Вся база данных», имеет обязательный аргумент – «Объект» (это текущий объект, относительно которого в соответствии с заданным направлением поиска ищутся потомки).

В выполненном отчете значение аргумента можно изменить. После обновления отчета он будет сформирован с использованием текущих значений аргумента.

Помимо обязательного аргумента, в отчете можно использовать дополнительные аргументы.

Для учебного отчета в качестве дополнительных аргументов можно использовать даты начала и окончания периода. Если значения таких аргументов использовать при фильтрации строк отчета, можно формировать отчет за заданный период.

Для добавления в отчет аргументов перейдите на вкладку «Отчет» окна настройки отчета. В верхнем меню «Правка» выберите «Аргументы». В контекстном меню правой части открывшегося окна выберите «Добавить».

Добавьте 2 аргумента – начало и конец периода – см. рисунок 208.

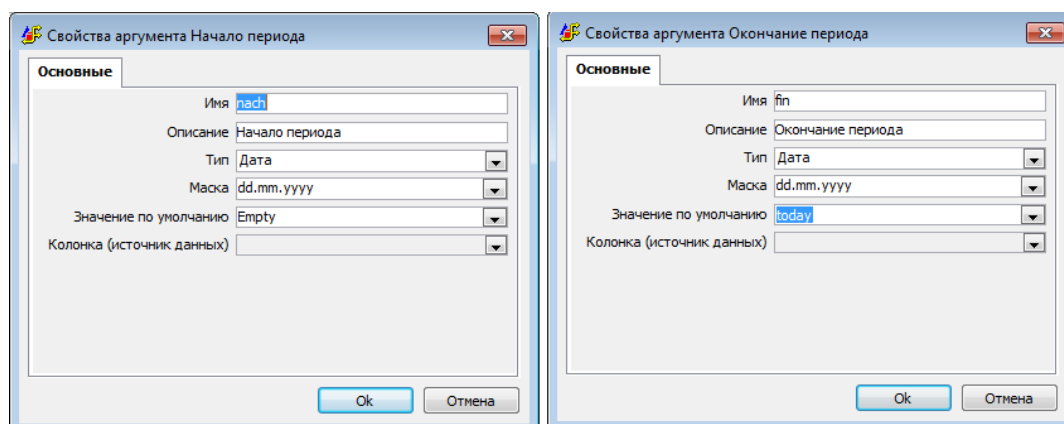


Рисунок 208 Свойства аргументов отчета

Сохраните отчет.

При необходимости, форма для ввода аргументов может быть дополнительно настроена. Использования всех возможностей настройки формы с аргументами, находясь

на вкладке «Отчет», верхнем меню аналогично настройке атрибутивных форм – см. п. 10.2.1 и форм действий над объектами – см. п. 17.3.1.1, имеются некоторые дополнительные возможности – 20.5.6.1.

20.5.6.1 Особенности настройки форм аргументов отчетов

Сама форма с аргументами может располагаться как непосредственно в окне отчета (слева, справа, сверху или внизу), так и на отдельной вкладке – «Отдельно» – в зависимости от настройки (Рисунок 209).

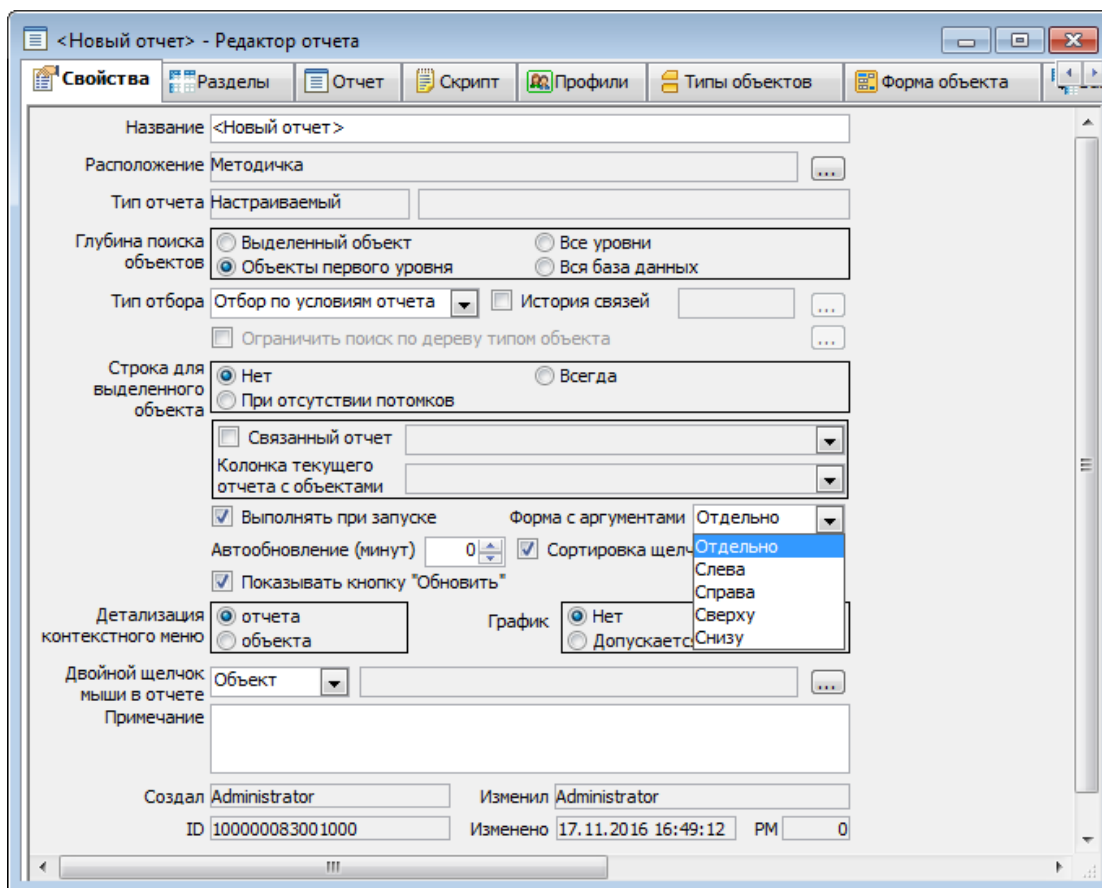


Рисунок 209 Указание расположения формы с аргументами отчета

В формах аргументов отчета имеется дополнительная (по сравнению с атрибутивными формами и формами шагов действий) возможность для формирования выпадающих списков возможных значений аргументов. Помимо фиксированного выпадающего списка, список значений может формироваться путем выполнения SQL-запроса или пользовательского отчета (который представляет собой также SQL-запрос). Для создания такого выпадающего списка, в свойствах соответствующего объекта формы выберите пункт «Редактирование», в поле «Стиль» выберите «Выпадающее окно данных», при выборе в поле «Тип» значения «Пользовательский запрос», по щелчку в поле «Окно данных» будет открыто окно для ввода SQL-запроса (Рисунок 210).

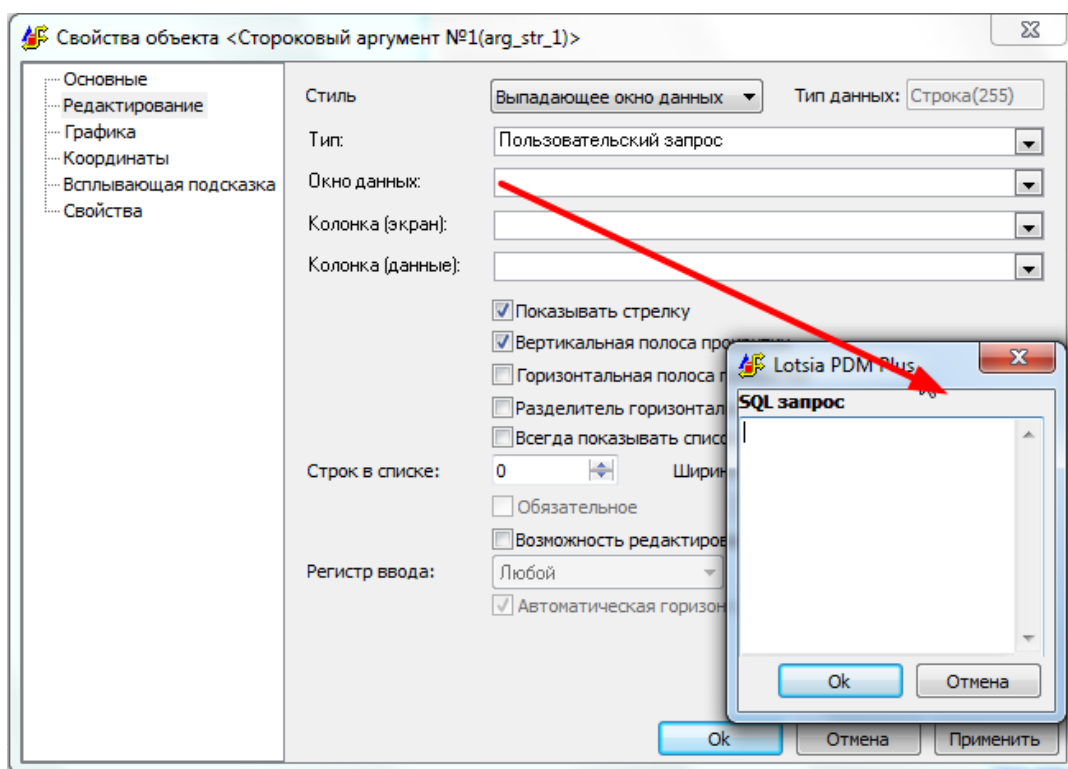


Рисунок 210 Настройка выпадающего списка для аргумента отчета

После ввода запроса в полях «Колонка (экран)» и «Колонка (данные)» следует указать соответствующие колонки, возвращаемые SQL-запросом. Можно для отображения на экране («Колонка (экран)») и для присвоения переменной («Колонка (данные)») использовать одну и ту же, или различные колонки запроса. Тип данных колонки, возвращаемой SQL-запросом, и указанной в поле («Колонка (данные)»), должен совпадать с типом значения переменной.

Добавленные аргументы можно использовать в вычисляемых полях отчета, в условиях предварительного отбора, а также в выражении фильтра.

Дополните выражение фильтра следующим:

and col2 >= nach and col2 <= finish , где col2 – «Дата регистрации».

Выражение вычисляемого поля заголовка так же отредактируйте:

'Книга регистрации актов проверок'~r~n~n+' за период с '+' nach '+' по '+' finish

Сохраните отчет.

Выполните отчет, указывая различные даты начала периода, убедитесь, что попадающие в отчет строки соответствуют заданному периоду.

20.5.7 Добавление в отчет информации из связанных с потомками объектов

Имеется возможность добавлять в отчет колонки, информация в которых извлекается не только из текущего объекта и потомков, но и из объектов, связанных с потомками связью заданного типа. Процедура добавления таких колонок описана в «Руководстве пользователя Lotsia PDM». Отметим, что практическое применение данная возможность имеет только в том случае, если у каждого потомка имеются единственный связанный объект. Если же связанных объектов у потомков произвольное количество, данный инструмент не применим.

Для того чтобы сформировать отчет, содержащий в каждой строке информацию из различных связанных объектов, рекомендуется пользоваться несколькими отчетами, при

выполнении которых результат выполнения одного отчета используется в качестве источника для присвоения значения аргументам следующего отчета (см. раздел 20.5.8).

20.5.8 Формирование строк отчета с использованием информации из различных объектов БД. Создание отчета с источником данных

В качестве модели данных продолжим использовать учебный проект «Учет компьютеров» – см. рисунок 195, добавив в него информацию о сотрудниках, обслуживающих каждый компьютер (тип объекта – «Сотрудник»). Считаем, что за каждым компьютером закреплен один или несколько сотрудников отдела вычислительной техники см. рисунок 211.

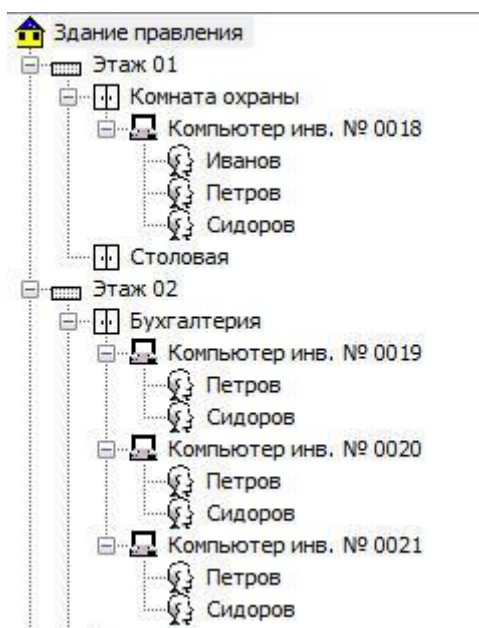


Рисунок 211 Учебный проект «Обслуживание компьютеров». Фрагмент

Для того чтобы получить сведения о компьютерах и обслуживающем персонале, достаточно сформировать один отчет. Рекомендуется создать такой отчет самостоятельно, согласно следующему описанию:

- название отчета – «Обслуживание компьютеров»;
- глубина поиска – все уровни;
- отбор по условиям отчета;
- переключатель «Строка для выделенного объекта отчета обязательна» установите в положение «Нет»;
- направление поиска – вниз по связи «Дерево проектов», в отчет включаются потомки типа «Сотрудник»;
- отчет состоит из двух колонок: описания родительского объекта (1) и описания потомка (2) (под родительским объектом понимается родитель потомка, в нашем случае это объекты типа «Компьютер»);
- сортировка и подавление повторяющихся значений – по колонке (1).
- Результат выполнения отчета по объекту «Здание правления» – см. рисунок 212.

Описание объекта (Родитель)	Описание объекта (Потомок)
0018	Сидоров
	Петров
	Иванов
0019	Петров
	Сидоров
0020	Петров
	Сидоров
0021	Петров
	Сидоров

Рисунок 212 Результат выполнения отчета «Обслуживание компьютеров». Фрагмент

Для того чтобы включить в отчет информацию о связанных с компьютерами объектах верхнего уровня (то есть, о помещениях), создадим два отчета.

Будем использовать отчет с источником данных, состоящий, соответственно, из отчета-источника и отчета-приемника. Последовательность создания отчетов произвольная.

Создайте отчет-приемник – «Обслуживание компьютеров по помещениям». Данный отчет будет выполняться по выделенному объекту – помещению, с глубиной поиска «Все уровни», отбор по условиям отчета, переключатель «Строка для выделенного объекта отчета обязательна» установите в положение «Нет», направлением поиска – вниз по связи «Дерево проектов», в отчет включаются потомки типа «Сотрудник». Добавьте в отчет следующие колонки (в учебных целях, для наглядности, сохраним автоматические названия колонок, дополнив их наименованием их содержимого):

1. «Описание объекта (Выделенный объект) – Помещение»;
2. «Описание объекта (Родитель) – Компьютер» (под родительским объектом понимается родитель потомка, то есть, «Компьютер»)
3. «Описание объекта (Потомок) – Сотрудник»

Отсортируйте строки отчета по первой и второй колонкам, подавите в тех же колонках повторяющиеся значения.

Выполните данный отчет, используя в качестве текущего объект помещения, убедитесь в правильной автономной работе отчета (см. рисунок 213).

Описание объекта (Выделенный объект) Помещение	Описание объекта (Родитель) Компьютер	Описание объекта (Потомок) Сотрудник
Бухгалтерия	0019	Петров
		Сидоров
	0020	Сидоров
		Петров
	0021	Петров
		Сидоров

Рисунок 213 Отчет «Обслуживание компьютеров по помещениям», выполненный по объекту «Бухгалтерия»

Однако, в случае если текущее помещение включает в себя другие помещения, (например, Приемная включает в себя еще и кабинеты руководителей – см. рисунок 195) данный отчет будет неверно отображать название помещения, для компьютеров, расположенных в кабинетах руководства. Чтобы избежать указанной ситуации, ограничим глубину поиска потомков (объектов типа «Сотрудник»). Нужные нам потомки располагаются на втором уровне от выделенного объекта, на вкладке «Свойства» такая глубина поиска установлена быть не может. Для того чтобы задать поиск потомков на требуемую глубину, добавим в отчет системную колонку «Уровень», и выполним фильтрацию отчета, выражение фильтра будет следующим:

$level = 2$

Сохраните отчет с указанным фильтром.

Чтобы сформировать общий отчет по всем помещениям здания, создайте отчет – источник «Помещения здания». Данный отчет предназначен для нахождения всех объектов типа «Помещение», входящих в выделенный объект (например, типа «Здание», или «Этаж»).

Глубина поиска – «Все уровни», отбор по условиям отчета, переключатель «Строка для выделенного объекта отчета обязательна» установите в положение «Нет», направление поиска – вниз по связи «Дерево проектов», в отчет включаются потомки типа «Помещение».

Добавьте в отчет системную колонку «Объект (Код)». После выполнения соответствующей настройки, данная колонка будет передаваться в отчет приемник, и найденные потомки данного отчета будут использоваться в отчете – приемнике в качестве текущих объектов. Данная колонка несет основную функциональную нагрузку, она может быть единственной в данном отчете.

Так как информация в колонке «Объект (Код)» для восприятия человеком не очень удобна, добавьте в отчет колонку, содержащую информацию, пригодную для визуальной идентификации найденного объекта-потомка, например, его описание.

Добавьте так же колонку с информацией о выделенном объекте, например, его описание. После выполнения соответствующей настройки, данная колонка может так же быть передана в отчет приемник.

Выполните данный отчет по объекту «Здание правления», убедитесь в его правильной автономной работе – см. рисунок 214.

Объект (код)	Описание объекта (Потомок)	Описание объекта (Выделенный объект)
100000031503044	Комната охраны	Здание правления
100000031603044	Столовая	Здание правления
100000031703044	Бухгалтерия	Здание правления
100000031803044	Комната отдыха	Здание правления
100000031903044	Приемная	Здание правления
100000032003044	Буфет	Здание правления
100000032103044	Отдел №1	Здание правления
100000032303044	Отдел №2	Здание правления
100000033003044	Кабинет Генерального директора	Здание правления
100000034003044	Кабинет зам. Генерального директора	Здание правления

Рисунок 214 Отчет «Помещения здания», автономно выполненный по объекту «Здание правления»

Убедившись, что оба отчета автономно работают правильно, подключите отчет «Помещения здания» в качестве источника данных к отчету «Обслуживание компьютеров по помещениям»:

Откройте окно настройки отчета «Обслуживание компьютеров по помещениям» на вкладке «Отчет»;

В верхнем меню «Правка» выберите «Аргументы»;

В открывшемся окне «Аргументы» установите флажок «Источник данных», в открывшемся окне «Выбрать отчет» укажите отчет «Помещения здания»;

В окне «Аргументы» выделите строку с аргументом «Объект» и дважды щелкните по ней. В открывшемся окне «Свойства аргумента Объект» в поле «Колонка (источник данных)» выберите «Объект (код)». Сохраните отчет, выполните его по объекту «Здание правления», убедитесь в правильной работе отчета.

Добавьте аргумент отчета «desc_a_obj» (см. так же Использование аргументов отчета) – см. рисунок 215. Данный аргумент можно использовать в вычисляемых полях отчета. Добавьте в область данных отчета вычисляемое поле, сделайте колонку данного вычисляемого поля первой в отчете. В выражении вычисляемого поля укажите колонку – аргумент desc_a_obj, подавите повторяющиеся значения в добавленной колонке (порядок подавления повторяющихся значений важен – см. рисунок 216).

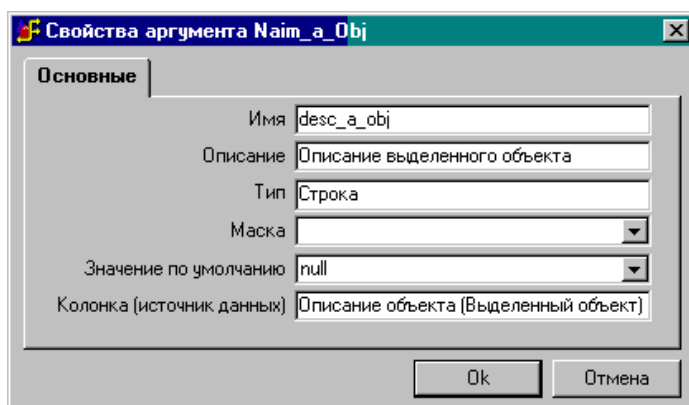


Рисунок 215 Добавление аргумента для передачи из отчета источника описания выделенного объекта

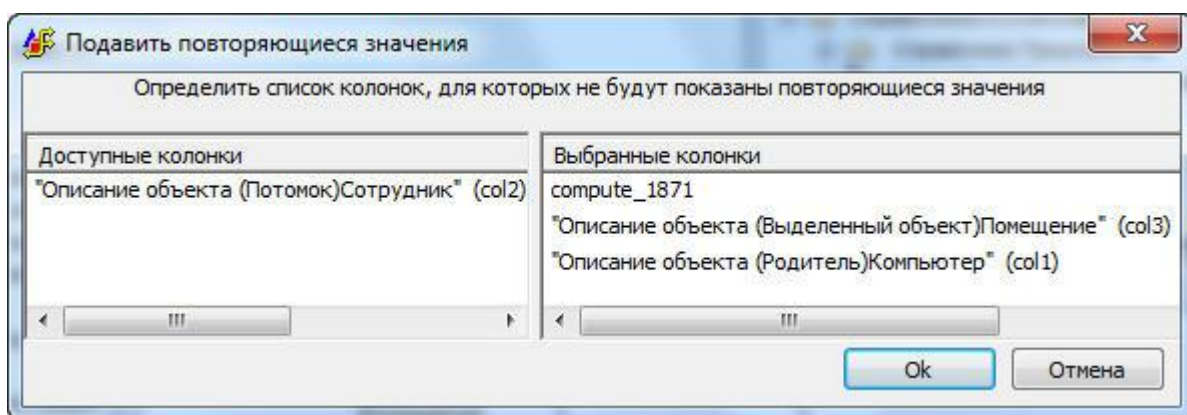


Рисунок 216 Последовательность указания списка колонок для подавления повторяющихся значений

Сохраните отчет, проверьте правильность его работы – см. рисунок 217

	Описание объекта (Выделенный объект) Помещение	Описание объекта (Родитель) Компьютер	Описание объекта (Потомок) Сотрудник
Здание правления	Бухгалтерия	0019	Сидоров
			Петров
		0020	Петров
			Сидоров
		0021	Петров
			Сидоров
	Кабинет Генерального директора	0023	Иванов
			Сидоров
	Кабинет зам. Генерального директора	0024	Сидоров
	Комната охраны	0018	Сидоров
			Иванов
			Петров
	Отдел №1	0025	Иванов
			Петров
	Отдел №2	0026	Иванов
			Сидоров
		0027	Иванов
			Сидоров
		0028	Иванов
	Приемная		Сидоров
		0022	Петров
			Иванов

Рисунок 217 Отчет «Обслуживание компьютеров по помещениям», выполненный по объекту «Здание правления», с использованием отчета «Помещения здания» в качестве источника данных

При использовании отчета с вычисляемыми полями в качестве источника, значения вычисляемых полей можно передавать в отчеты – приемники, аналогично колонкам отчета.

Чтобы использовать в отчете с источником данных аргументы, значения которых будут задаваться пользователем – см. Использование аргументов отчета, следует:

- требуемые аргументы добавлять вначале в отчет – источник;
- в отчете-источнике создать вычисляемое поле, в выражении которого указать колонку данного аргумента;
- в отчете приемнике создать соответствующий аргумент, указав в поле «Колонка (источник данных)» колонку вышеуказанного вычисляемого поля отчета – источника;
- после выполнения вышеперечисленного, значения аргумента можно использовать в выражениях вычисляемых полей или условий фильтрации отчета – приемника.

При передаче аргументов из отчета-источника в отчет-приемник с использованием вычисляемых полей необходимо так же учитывать, что формат аргумента отчета приемника должен соответствовать формату данных вычисляемого поля отчета-источника. Формат данных вычисляемого поля может принимать одно из следующих значений:

- Дата/время (даже если фактически задавалось значение в формате «Дата»)
- Строка
- Число

Один и тот же отчет – источник могут использовать различные отчеты – приемники, отчет – приемник, в свою очередь, может использоваться в качестве отчета – источника для следующего отчета – приемника, и. д., без ограничения глубины вложенности отчетов.

20.6 Резюме

Lotsia PDM PLUS содержит встроенный генератор отчетов. Настроенные отчеты отображаются в виде структурированного списка (дерева), для открытия которого используется пункт Главного меню «Отчеты» → «Настройка отчетов».

При создании отчета выбирается тип отчета: настраиваемый или предопределенный, для настраиваемых отчетов указывается стиль отчета, метод формирования отчета: «По объекту» (то есть, по выделенному и связанным объектам), «Вся база данных» (то есть, без использования выделенного объекта), «Пользовательский» (то есть, поиск выполняется настраиваемым SQL-запросом).

В ходе дальнейшей настройки для отчета по объекту определяется глубина поиска и порядок отображения строки для выделенного объекта.

Далее создаются разделы отчета (один или более), где определяется порядок поиска данных отчета.

Далее определяются колонки отчета, добавляются текстовые и вычисляемые поля, настраиваются условия дополнительного отбора объектов, сортировка, группировка, фильтрация строк отчета, а также его внешний вид.

При необходимости в отчет могут добавляться аргументы, которым присваиваются значения до начала построения отчета. Аргументы могут использоваться в вычисляемых полях отчета, условиях фильтрации и предварительного отбора элементов.

Результат выполнения отчета, путем выполнения соответствующей настройки может быть передан в следующий отчет, где данные первого отчета будут использованы в

качестве аргументов. Далее результат выполнения отчета можно передать в очередной отчет, без ограничения глубины вложенности отчетов.

Возможно построение комбинированных отчетов, когда в одном отчете объединено несколько различных отчетов.

Возможно представление результатов отчета в графическом виде, а также экспорт отчета по шаблону в MS Word или MS Excel.

20.7 Отчеты. Учебное задание

Выполните настройку отчетов в созданной учебной настройке, пользуясь рассмотренными выше примерами (п. 20.5). Настройте отчет «Журнал регистрации документов», в котором, аналогично рассмотренным примерам, постройте перечень зарегистрированных документов. Добавьте группировку документов по дате регистрации, определите количество зарегистрированных документов (общее и за отдельную дату). Добавьте в отчет аргументы «Начало периода» и «Окончание периода», сформируйте журнал регистрации за определенный значениями аргументов период времени.

20.8 Отчеты. Вопросы для самопроверки

1. Как добавить в отчет колонку с описанием потомка?
2. Как добавить в отчет со стилем «Сетка» широкий заголовок?
3. Как отсортировать строки отчета, как изменить сортировку?
4. Как добавить группировку, как поменять цвет области заголовка группы?
5. Как получить итоги по группе и общие?
6. Как добавить аргумент в отчет?
7. Как подключить отчет-источник, как передать из отчета – источника колонку в отчет приемник?

21 Бизнес – правила

Бизнес-правила предназначены для контроля определенных операций, выполняемых пользователем в программе Lotsia PDM.

Применять бизнес-правила следует в обоснованных случаях, так как использование избыточного количества бизнес-правил усложняет администрирование системы, приводит к замедлению работы программы. При настройке бизнес-правил рекомендуется контролировать взаимоисключение и «пересечение» различных бизнес-правил.

Бизнес-правила являются дополнительным инструментом, позволяющим в ряде случаев оперативно решать некоторые частные задачи, решение которых в более развернутом виде, например, с использованием прав доступа, действий и предопределенной маршрутизации процессов на данный момент, по каким-либо причинам не используется.

Бизнес-правила имеют ряд ограничений. Например, не контролируется процесс удаления объекта из проекта, когда происходит автоматическое удаление атрибутов проекта, для обеспечения такого контроля может быть использовано задание прав доступа.

Отключить, добавить или настроить бизнес-правило может Администратор, либо пользователь, которому установлена привилегия «Управление бизнес-правилами», и в Главном меню которого имеется пункт «Бизнес-правила».

Бизнес-правила разрешают или запрещают такие операции пользователя, как:

- Добавление объекта в проект
- Удаление объекта из проекта
- Добавление к объекту атрибута объекта (атрибута проекта)
- Удаление атрибута объекта (атрибута проекта)
- Изменение атрибута объекта (атрибута проекта).

Бизнес-правила бывают 3х типов:

1. Действие:

например, «Запрет изменения атрибутов у утвержденного проекта»;

«Запрет удаления объектов из утвержденного проекта»

2. Соответствие:

например, «В утвержденном изделии все объекты КД должны быть утверждены»,
«Обозначение ОКД должно совпадать с обозначением изделия»;

«Правило подписей: Руководитель проекта может подписаться только после подписи исполнителя»

3. Значение:

например, «Контроль даты окончания разработки объекта»;

«Контроль изменения количества данных объектов».

Каждое правило может иметь привязку к типам объектов (и атрибутам). Под привязкой правила, например, к типу объекта, понимается задание и проверка условий правила относительно объектов данного типа.

Открытие редактора бизнес-правил производится выбором пункта «Бизнес-правила» Главного меню («Администрирование» → «Структура данных» → «Бизнес-правила»).

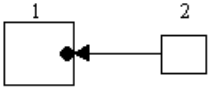
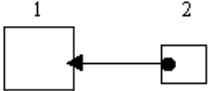
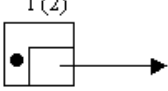
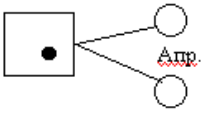
21.1 Бизнес-правило типа «Действие»

Бизнес-правило типа «Действие» предназначено для контроля следующих действий

- Добавление объекта в проект
- Удаление объекта из проекта
- Добавление (изменение, удаление) атрибута объекта
- Добавление (изменение, удаление) атрибута проекта.

К моменту проверки правила действие пользователя считается системой несовершенным.

Бизнес-правило типа «Действие» должно иметь привязку к типу объекта (типам объекта). При добавлении объекта к объекту проверка правила осуществляется относительно объекта, к которому добавляется объект. При удалении объекта правило проверяется относительно объекта, из которого удаляется объект. Атрибуты добавляемого объекта в правиле данного типа использоваться не могут. При удалении объекта атрибуты удаляемого объекта могут быть использованы в правиле. Добавление объекта в проект может осуществляться в результате заимствования из другого проекта. Для действия «Заимствование» проверка производится относительно объекта, который добавляется.

Схема правила	Действие	Привязка	Когда проверять	Проверка правила: действие совершено?
	Добавление в проект	К типу объекта 1	См. выражение*	Нет
	Заимствование	К типу объекта 2	См. выражение*	Нет
	Удаление из проекта	К типу объекта 1	См. выражение*	Нет
	Изменение атрибута объекта (изменение атрибута проекта)	К типу объекта 1	Добавление, изменение, удаление атрибута; См. выражение*	Нет

Обозначения в таблице:

1(2) - объекты

• - привязка

➔ - удаление объекта

➔ - добавление объекта

Аоб., Апр. - атрибуты объекта (проекта)

* - выражение задается в условии правила.

Пример бизнес-правила типа «Действие»:

Тип бизнес-правила	Вид действия	Привязка к типу объекта	Когда проверять?	Что проверять?
Действие	Удаление из проекта	Указать типы проверяемых родительских объектов	Выражение отсутствует (проверять всегда)	Значение атрибута (напр. Статус <> «Утвержден»)

Данное правило позволяет изменять состав объектов заданных типов, если атрибут «Статус» этих объектов не равен «Утвержден» (и, соответственно, не позволяет изменять утвержденные проекты).

21.2 Бизнес-правило типа «Соответствие»

Бизнес-правило типа «Соответствие» проверяет соответствие значений атрибутов одного объекта или различных объектов (в этом случае объекты должны быть на одну ступень «выше» или «ниже» проверяемого объекта, то есть, объекты, в которые входит проверяемый объект, или объекты, которые входят в проверяемый объект) между собой и предназначено для контроля следующих действий:

- добавление объекта в проект;
- удаление объекта из проекта;
- изменение атрибута объекта;
- изменение атрибута проекта.

Привязка осуществляется к типам объектов и атрибуту объекта или атрибуту проекта. Правило будет проверяться при любом изменении значения атрибутов, которые используются в правиле, только у объектов определенных типов.

При добавлении объекта к данному объекту в правиле данного типа можно использовать атрибуты данного объекта и атрибуты добавляемого объекта (атрибуты будут браться из дерева объекта, для которого задана привязка). Программа будет проверять истинность правила, учитывая атрибуты добавляемого объекта, так как на время проверки предполагается, что добавление объекта уже произошло.

При удалении объекта из проекта атрибуты удаляемого объекта в правиле использоваться не могут (программа считает, что на время проверки удаление объекта уже произошло).

Таким образом, для действий, связанных с изменением состава проекта (изменением атрибутов), при проверке бизнес-правила «Соответствие», в отличие от бизнес-правила «Действие», программа временно (на период проверки бизнес-правила) считает, что изменение состава проекта (изменение атрибутов) уже произошло.

Пример бизнес-правила типа «Соответствие»

«Башня завершена, если завершены входящие в нее объекты».

Завершенный объект имеет атрибут «Статус объекта» со значением «Завершен». При присвоении данному атрибуту башни значения «Завершен» должна происходить проверка атрибута у цилиндров и конусов (потомков объекта типа «Башня»).

Тип бизнес-правила	Привязка к атрибуту	Привязка к типу объекта	Когда проверять?	Что проверять?
Соответствие	Атрибут объекта	Тип «Башня»	Статус объекта = «Завершен» (для башни)	Статус объекта = «Завершен» (для цилиндров и конусов)

Параметры и условия правила задаются на вкладках «Объекты» (см. рисунок 218), «Свойства» (см. рисунок 219), «Условие...» (см. рисунок 220).

Редактор бизнес-правил - Башня завершена, если завершены все входящие в нее объекты

Привязка | **Свойства** | Условие 1

Название: Башня завершена, если завершены все входящие в нее объекты

☒ Активно Действует с: 06.08.2001 16:50 Действует по: 31.12.2099 00:00

Тип: Соответствие Для атрибутов: Только объекта

Номер	Описание	
1	Не все входящие объекты завершены	<input checked="" type="checkbox"/>

Рисунок 218 Окно редактора правил на вкладке «Свойства». Бизнес-правило типа «Соответствие»

Редактор бизнес правил - Башня завершена, если ...

Привязка | **Свойства** | Условие 1

Тип объекта	Вид	Мнемо
Башня	Изделия	Бн

Рисунок 219 Окно редактора правил на вкладке «Привязка». Бизнес-правило типа «Соответствие»

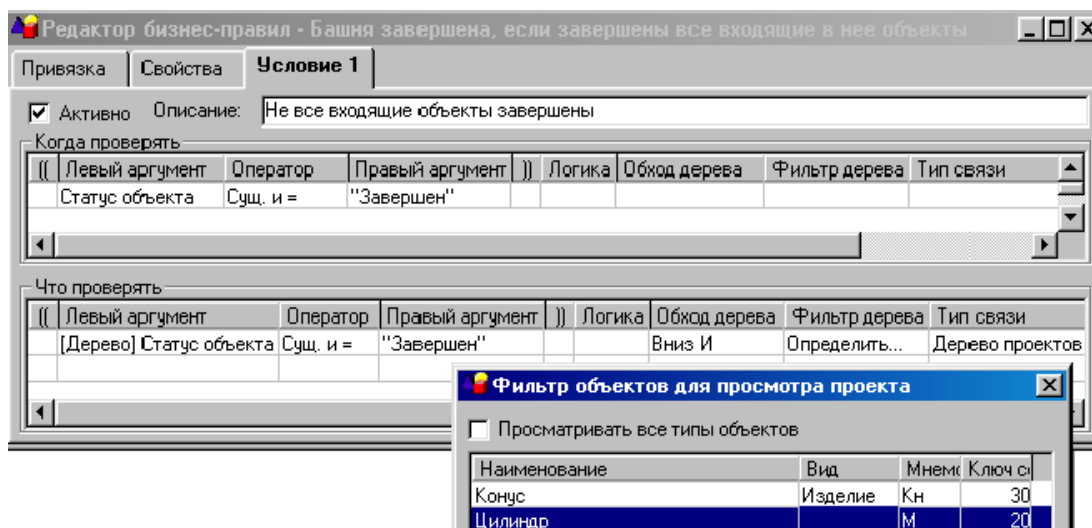


Рисунок 220 Окно редактора правил на вкладке «Условие...». Бизнес-правило типа «Соответствие»

21.3 Бизнес-правило типа «Значение»

Бизнес-правило типа «Значение» применяют, когда необходимо контролировать изменение различных атрибутов объектов (привязка задается к типам объектов и их атрибутам) или изменение конкретного атрибута объекта (привязка к типу объекта и к конкретному атрибуту), при этом изменение состава проекта не происходит. Правило предназначено для контроля таких действий, как:

- Добавление (изменение, удаление) атрибутов объекта и проекта;
- Добавление (изменение, удаление) атрибута только проекта.

Правило проверяется в случае изменения атрибута у объекта определенного типа.

В случае, когда правило составлено для конкретного атрибута в условиях правила, возможно использование нового значения атрибута. Возможно также указать, при каких действиях над атрибутом (добавление, изменение, удаление) будет проверяться данное условие бизнес-правила.

Пример бизнес-правила типа «Значение»:

«Дата завершения Главной башни не превышает 20.04.01»

Данное правило отнесено к правилу типа «Значение», поскольку необходимо выполнить контроль установки (добавление и изменение) и удаления атрибута объекта при неизменном его составе.

Тип бизнес-правила	Привязка к атрибуту	Привязка к типу объекта	Когда проверять?	Что проверять?
Значение	Дата завершения Объекта	«Башня»	При добавлении, изменении, удалении атрибута; «Наименование»= «Главная Башня»	Новое значение атрибута существует и ≤ 20.04.01

Параметры правила заданы на вкладках окна редактора правил (Рисунок 221, Рисунок 222, Рисунок 223).

Редактор бизнес правил - Контроль даты завершения Главной башни

Привязка | **Свойства** | Условие 1

Название: Контроль даты завершения Главной башни

☒ Активно Действует с: 06.08.2001 16:54 Действует по: 31.12.2099 00:00

Тип: Значение Для атрибутов: Объекта и проекта

Использовать новое значение атрибута: Дата завершения объекта

Номер	Описание				
1	Дата завершения объекта не превышает 20.04.01	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Рисунок 221 Пример бизнес-правила типа «Значение». Основные параметры

Редактор бизнес правил - Контроль даты завершения Главной башни

Привязка | Свойства | **Условие 1**

	Атрибут	Тип объекта	Вид
<input checked="" type="checkbox"/>	Дата завершения объекта	Башня	Изделие
<input type="checkbox"/>	Дата завершения объекта	Здание	Изделие
<input type="checkbox"/>	Дата завершения объекта	Стена	Изделие
<input type="checkbox"/>	Дата завершения объекта	Строение	Изделие

Рисунок 222 Пример бизнес-правила типа «Значение». Привязка

Редактор бизнес правил - Контроль даты завершения Главной башни

Привязка | Свойства | **Условие 1**

☒ Активно Описание: Дата завершения объекта не превышает 20.04.01

Когда проверять
☒ Добавление ☒ Изменение ☒ Удаление

[[Левый аргумент	Оператор	Правый аргумент]]	Логика	Обход дерева	Фильтр
<input checked="" type="checkbox"/>	Наименование	=	"Главная башня"				

Что проверять

[[Левый аргумент	Оператор	Правый аргумент]]	Логика	Обход дерева
	<Новое значение>	Сущ. и <=	DateTime("2001-04-20 00:00:00.000000")			

Рисунок 223 Пример бизнес-правила типа «Значение». Условие

21.4 Резюме

Бизнес-правила предназначены для контроля определенных операций, выполняемых пользователем Lotsia PDM.

Бизнес-правила бывают 3х типов:

- Действие – например, «Запрет изменения атрибутов у утвержденного проекта»; «Запрет удаления объектов из утвержденного проекта»;
- Соответствие – например, «В утвержденном изделии все дочерние изделия должны быть утверждены»;
- Значение – например, «Контроль даты окончания разработки объекта».

Каждое правило может иметь привязку к типам объектов (и атрибутам).

Бизнес-правило типа «Действие» предназначено для контроля следующих действий:

- Добавление и удаление объекта в проект (из проекта);
- Добавление (изменение, удаление) атрибута объекта или атрибута проекта.
- Данное правило привязывается к типам объектов.

Бизнес-правило типа «Соответствие» проверяет соответствие значений атрибутов одного объекта или различных объектов и предназначено для контроля следующих действий:

- Добавление и удаление объекта в проект (из проекта);
- Изменение атрибута объекта или атрибута проекта.
- Привязка осуществляется к типам объектов и атрибуту объекта или атрибуту проекта.

Бизнес-правило типа «Значение» предназначено для контроля таких действий, как добавление (изменение, удаление) атрибутов объекта и проекта или только атрибута проекта. Правило проверяется в случае изменения атрибута у объекта определенного типа.

22 Документооборот. Общая информация

Программа Lotsia Workflow (модуль документооборота, входящий в Lotsia PDM PLUS) предназначена для управления документооборотом и бизнес-процессами.

Lotsia Workflow можно использоваться как автономно, так и совместно с системой Lotsia PDM, системой автоматизации управления предприятием Lotsia ERP. Это позволяет построить законченную систему автоматизации документооборота и управления бизнес-процессами предприятия в соответствии с требованиями стандартов обеспечения качества ISO9000.

Под бизнес-процессом понимается совокупность взаимосвязанных действий участников процесса. Возможна предопределенная или свободная маршрутизация сообщений при осуществлении бизнес-процессов.

Использование свободной маршрутизации сообщений Lotsia Workflow позволяет пользователям обмениваться сообщениями, аналогично тому, как это делается при использовании, например, Outlook Express или другой почтовой программы.

Предопределенная маршрутизация предполагает предварительную настройку шаблона работы Workflow (другими словами – создание предопределенного процесса, или маршрута). При каждом запуске такой работы обеспечивается заранее заданная (то есть, предопределенная настройкой шаблона) последовательность обмена сообщениями между участниками процесса (они же – исполнители этапов работы).

Движение документов осуществляется прикреплением документов к сообщениям. В качестве документов могут использоваться файлы операционной системы, документы архива Lotsia PDM, объекты Lotsia PDM и другие сущности. Для предопределенных процессов возможные виды документов («Документы шаблона») определяются при настройке шаблона работы.

В соответствии с настройкой конкретного предопределенного процесса, отдельные этапы процесса могут выполняться в заданной последовательности или параллельно и независимо друг от друга. Параллельно выполняемые этапы и последовательности этапов могут динамически создаваться в требуемом количестве в зависимости от выполнения заданных условий.

22.1 Свободное сообщение

Свободное сообщение Lotsia Workflow (см. рисунок 224) содержит стандартный набор полей: тема, список получателей, приоритет, текст сообщения и др.

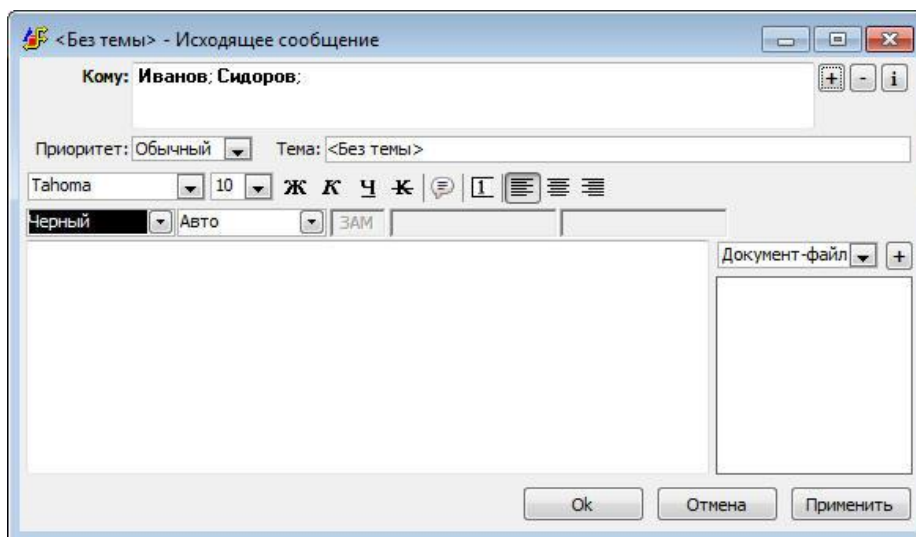


Рисунок 224 Окно свободного сообщения



Для создания нового сообщения выберите Главное меню → Документооборот → Новое сообщение.

В качестве получателей свободных сообщений и информационной рассылки предопределенных процессов Lotsia Workflow указываются контакты из адресной книги (АК) пользователя (о создании контактов см. раздел 11).

Имеются два вида АК: глобальные и пользовательские. Администратор настраивает глобальные АК. В глобальных АК используются контакты, сопоставленные пользователям.

Пользователь на основе глобальной АК (в соответствии с заданными правами на глобальные АК) формирует свою пользовательскую АК. Он может удалять книги, контакты, менять их местоположение, добавлять новые контакты с E-mail, не сопоставленные пользователям.

Выбор получателя свободного сообщения происходит из пользовательской АК, выбор адресатов информационной рассылки – из глобальной АК.

Для указания получателей следует нажать  в верхней части окна сообщения и выбрать адресатов из дерева или из списка. Кнопка  служит для удаления выделенного адресата.

Для отправки сообщения следует нажать «Ok».

К свободному сообщению могут быть прикреплены документы. Виды документов, которые могут прикрепляться к свободному сообщению, определяются в настройках рабочего места (Главное меню (либо верхнее меню) → «Настройки» → «Настройки рабочего места» → вкладка «Документы» – см. рисунок 225).

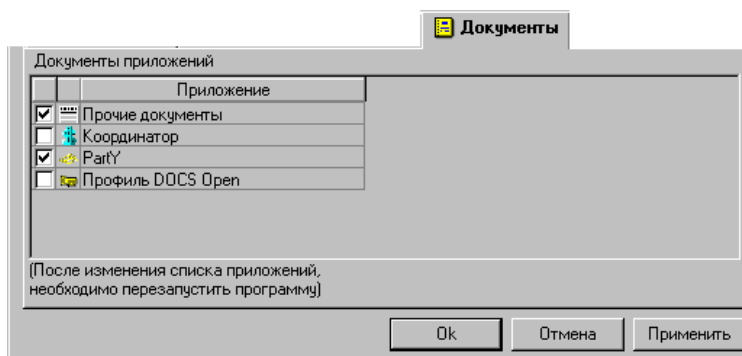



Рисунок 225 Определение возможных видов документов для прикрепления к сообщениям

Для обеспечения возможности использования в качестве документов файлов операционной системы и сущностей Lotsia PDM, следует установить флажки так, как показано на рисунке (см. рисунок 225). Документы к сообщению можно присоединить, используя кнопку  рядом с полем выбора вида документа, предварительно указав вид документа, либо используя контекстное меню области документов (см. рисунок 226).

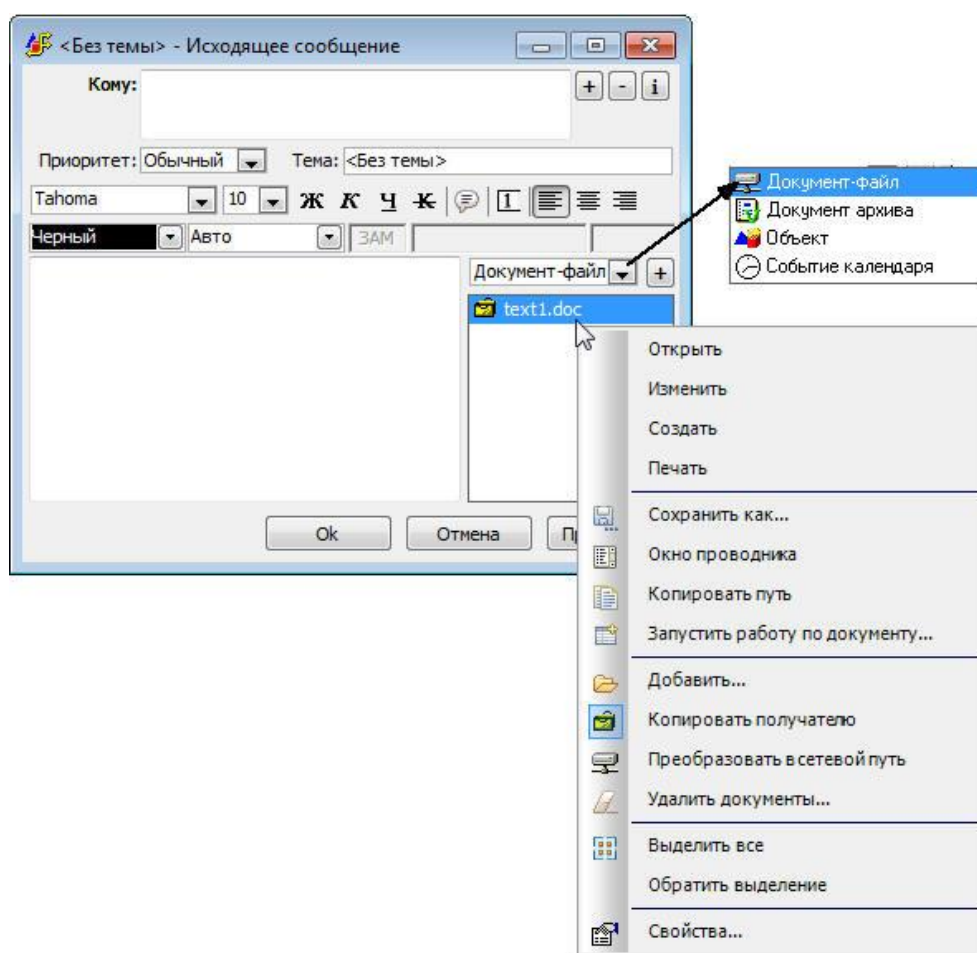


Рисунок 226 Добавление документа к сообщению

Используя метод «drag-and-drop», документ-файл можно присоединить к сообщению, перетаскивая файл на окно сообщения из окна проводника Windows.

При добавлении документа файловой системы возможна пересылка самого документа (флажок «Копировать получателю» в контекстном меню), либо отправка ссылки на документ. Если документ находится в папке, к которой открыт общий доступ,

ссылка может быть автоматически преобразована в сетевой путь (пункт контекстного меню «Преобразовать в сетевой путь», см. рисунок 226).

22.2 Добавление и настройка почтовых папок

Возможно создание дерева почтовых папок для помещения туда соответствующих входящих или исходящих сообщений. Создание и настройку почтовых папок может осуществлять как пользователь – непосредственно настройкой дерева своих почтовых папок, так и Администратор – путем создания шаблонов почтовых папок для профилей пользователей. Процедуры настройки почтовых папок пользователем и Администратором не различаются.

Для добавления дочерней папки следует выделить в окне «Задания и сообщения» (либо в окне «Профили и настройки» для требуемого профиля открыть пункт «Сообщения» → «Шаблоны папок») требуемую родительскую папку и в контекстном меню выбрать «Создать папку...». В открывшемся окне на вкладке «Основные» ввести описание и выбрать требуемые параметры создаваемой папки. На вкладке «Условия фильтрации почты» задать требуемые условия фильтрации (Рисунок 227).

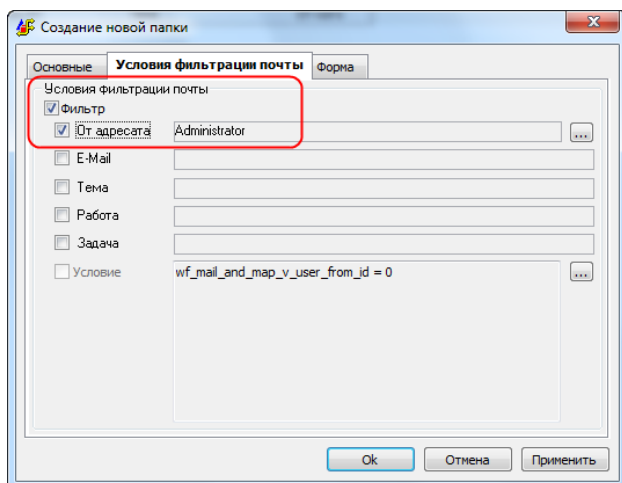


Рисунок 227 Пример задания условий фильтрации почты. В данную папку будут помещаться сообщения, отправленные администратором

Раскладка почты в папки может выполняться принудительно (для выделенной в дереве папки выбрать пункт контекстного меню «Разложить сообщения»), либо автоматически, после отправки/получения сообщений (при условии выполнении соответствующих настроек профиля, либо настроек пользователя – см. п. 10.3.11).

Возможно создание виртуальных папок. Сообщение в такой папке показывается до тех пор, пока удовлетворяется условие, заданное для данной папки – например, срок, прошедший с даты получения сообщения (Рисунок 228).

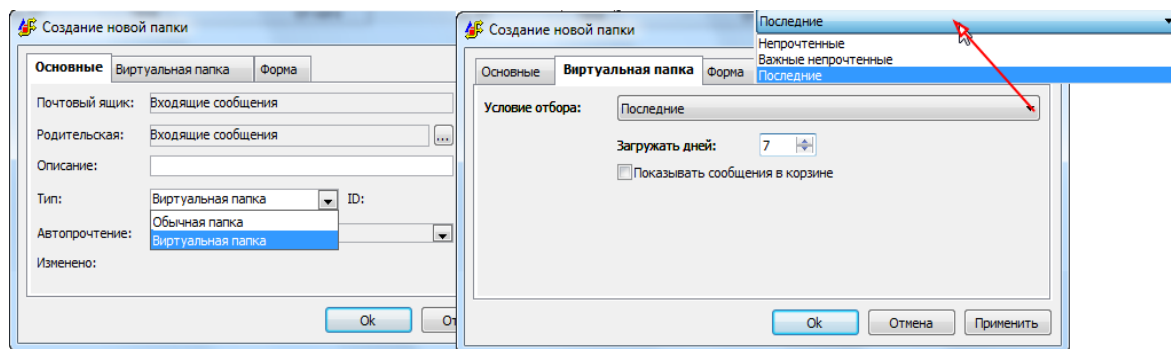


Рисунок 228 Пример настройки виртуальной папки

На хранение сообщения в обычной папке настройка виртуальных папок не влияет.

Для почтовых папок могут быть настроены формы – то есть, определен список отображаемых в форме колонок (Рисунок 229).

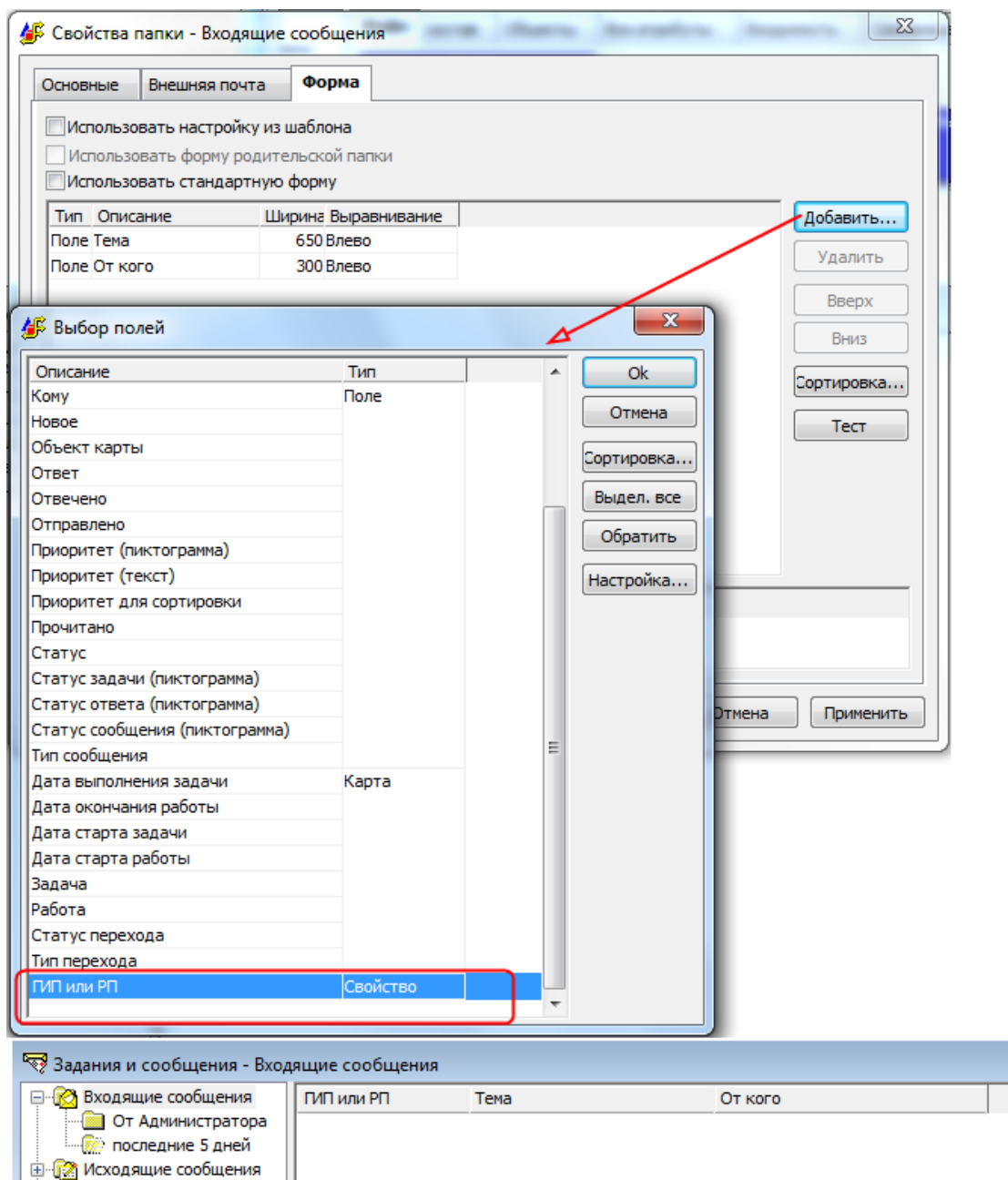


Рисунок 229 Пример и результат настройки формы почтовой папки

Полная подробная информация по работе с сообщениями содержится в документации пользователя Lotsia Workflow и контекстной справке по программе.

22.3 Временная фильтрация сообщений

Имеется возможность временной фильтрации сообщений. Фильтр включается щелчком левой кнопки мыши на ячейке с требуемым значением при нажатой и удерживаемой клавише «Alt». Отключается временный фильтр клавишей «Esc» или при обновлении списка сообщений. Таким образом, можно, например, быстро просмотреть список сообщений, имеющих одинаковую тему или сообщения с вложениями. При фильтрации по теме программа отсекает признаки ответа или пересылки, например, такие как «Re:», «FW:», «Fwd:». Поэтому, если вы хотите переслать сообщение другому адресату, добавив в тему свой текст и сохранив возможность быстрой фильтрации, добавляйте свой текст внутри признака ответа или пересылки, например, «Re[От Ивана]:»

22.4 Предопределенный процесс

Предопределенный процесс (работа Lotsia Workflow, маршрут) – бизнес-процесс (или его часть), выполняемый по заранее заданному маршруту (шаблону).

На практике это может быть формализованный (исчерпывающе описанный, с рассмотрением всех возможных вариантов выполнения), многократно повторяющийся процесс, выполняемый одними и теми же (либо одинаковым образом назначаемыми) подразделениями и/или исполнителями. Как правило, большинство таких процессов описано (предписано) директивными документами предприятия. Примером таких процессов могут служить:

- согласование комплекта конструкторских документов;
- регистрация и обработка входящей и исходящей корреспонденции;
- внесение изменения в конструкторскую или проектную документацию;
- согласование командировки, отпуска и др.;
- проведение голосования по какому-либо вопросу и т.п.

В Lotsia Workflow имеется возможность модульного конструирования сложных процессов с включением ранее созданных процессов как составных элементов в новый процесс (см. п. 23.18.2). Таким образом, можно вначале осуществлять автоматизацию часто повторяющихся наиболее простых (элементарных) процессов, с последующим их использованием в более сложных маршрутах.

Предопределенный процесс состоит из этапов и переходов между ними. Графическое изображение (блок-схема, выполненная встроенными средствами Lotsia Workflow) процесса называется картой. Карта процесса, не запущенного на выполнение, называется также шаблоном. При каждом запуске работы на выполнение, автоматически создается карта процесса (работы). Указанная карта представляет собой копию шаблона данной работы. Таким образом, каждому запущенному на выполнение экземпляру процесса соответствует собственная карта работы.


Настройка нового процесса осуществляется путем создания его карты – шаблона. После сохранения шаблона, соответствующий процесс (работа) попадает в перечень возможных работ и становится доступной для запуска оттуда всем пользователям, которые заданы в качестве возможных исполнителей начального этапа работы. (Для запуска процесса с помощью действия пользователь не обязательно должен быть явно указан в качестве исполнителя первого этапа).




Исполнитель – участник бизнес-процесса из числа пользователей Lotsia Workflow, выполняющий этап процесса. Исполнитель получает сообщение («Задачу») этапа процесса, выполняет полученную задачу. Исполнители задаются для каждого этапа предопределенного процесса.


Сообщение предопределенного процесса (задача, задание, предопределенное сообщение), в отличие от свободного сообщения имеет настраиваемую форму «Задание», соответствующую этапу процесса, с которой работает исполнитель данного этапа.

22.5 Работа с сообщениями. Учебное задание

После создания пользователей, адресных книг и выполнения необходимых настроек (см. п. 11) отправьте свободное сообщение нескольким пользователям, в том числе себе. Прочтите полученное сообщение.

Для того, чтобы узнать, прочтено ли адресатами Ваше сообщение, перейдите к отправленным сообщениям (Главное меню → «Задания и сообщения»), откройте отправленное Вами сообщение, посмотрите его статус используя кнопку .

Ознакомьтесь с функцией «Ответить отправителю», «Переслать сообщение». (Используются соответствующие пункты контекстного меню или кнопки    инструментальной панели).

Посмотрите историю переписки – используйте соответствующий пункт контекстного меню или кнопку  инструментальной панели.

Откройте окно глобального поиска сообщений (используется пункт «Найти...» контекстного меню окна «Задания и сообщения», либо сочетание клавиш <Ctrl+F>) выполните поиск сообщения.

Выполните настройку возможных видов документов для прикрепления к сообщениям, откройте окно нового сообщения, убедитесь, что список возможных видов прикрепляемых документов изменился в соответствии со сделанными настройками.

Прикрепите к сообщению документ-файл, ознакомьтесь с возможностью преобразования ссылки на прикрепленный документ в сетевой путь.

Настройте в окне «Задания и сообщения» несколько папок для помещения туда сообщений от конкретных пользователей.

Ознакомьтесь с настройками профилей пользователей и пользовательскими настройками, относящимися к работе с сообщениями (см. п. 10.3.11).

22.6 Резюме

Модуль документооборота, входящий в Lotsia PDM PLUS (Lotsia Workflow) предназначен для управления документооборотом и бизнес-процессами.

Под бизнес-процессом понимается совокупность взаимосвязанных действий участников процесса. Возможна предопределенная или свободная маршрутизация сообщений при осуществлении бизнес-процессов.

Свободное сообщение содержит стандартный набор полей: тема, список получателей, приоритет, текст сообщения и др., может быть отправлено различными способами. К сообщению могут присоединяться документы.

Предопределенный процесс (работа Lotsia Workflow, маршрут) – бизнес-процесс (или его часть), выполняемый по заранее заданному маршруту (шаблону). Предопределенный процесс состоит из этапов и переходов между ними.

Блок-схема процесса называется картой. Карта процесса, не запущенного на выполнение, называется также шаблоном. При каждом запуске работы на выполнение, автоматически создается карта процесса (работы). Указанная карта представляет собой копию шаблона данной работы. Таким образом, каждому запущенному на выполнение экземпляру процесса соответствует собственная карта работы.

22.7 Работа с сообщениями. Вопросы для самопроверки

1. Какие способы отправлять сообщения имеются?
2. Как отправить сообщение группе пользователей Lotsia PDM PLUS?
3. Как узнать, кто из адресатов прочел сообщение, а кто – нет?

4. Как настроить возможность присоединять к сообщению только документы-файлы и информационные объекты Lotsia PDM PLUS?
5. Как создать дополнительную подпапку для сообщений и обеспечить автоматическое попадание туда сообщений, удовлетворяющих заданным условиям?


23 Создание predetermined processes

Согласование документов в том или ином виде встречается в большинстве случаев использования Workflow. Для определенности, рассмотрение последовательности создания predetermined processes будем вести, в основном, на примере реализации возможных подобных процессов.


Ввиду многообразия инструментальных средств Workflow, используемых при настройке каждого процесса, описание порядка настройки конкретных процессов ведется по мере рассмотрения указанных функциональных средств. Сначала рассмотрим настройку простейшего процесса, а далее – его постепенное преобразование и дополнение.

23.1 Создание элементов шаблона

Для создания шаблона используйте пункт Главного меню «Администрирование» → «Шаблоны и карты работ» → «Шаблоны» → кнопка «Создать» в открывшемся окне.

Для реализации рассматриваемого далее примера, в открывшемся окне с новым шаблоном требуется добавить 3 задачи. Для добавления задач используется пункт контекстного меню «Объект» → «Задача» или кнопка  инструментальной панели.

Соедините задачи переходами. Для добавления перехода щелкните по задаче, откуда будет выходить переход. Перетащите, удерживая левую клавишу мыши, появившийся в центре задачи маркер на задачу, в которую должен входить создаваемый переход.

Сохраните полученный шаблон (кнопка  инструментальной панели). В открывшемся окне «Свойства карты» введите описание шаблона («Согласование»). Результат – см. рисунок 230

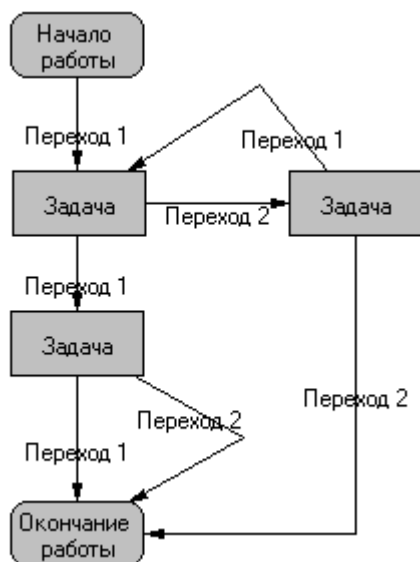


Рисунок 230 Сохраненный шаблон

Чтобы шаблон был более наглядным, измените описания элементов шаблона. Для этого дважды щелкните по каждому элементу, и в окне его свойств введите описание. Результат – см. рисунок 231

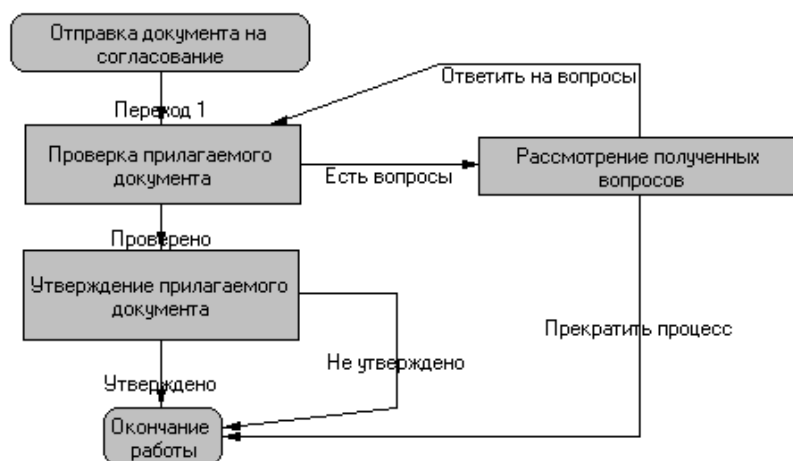


Рисунок 231 Шаблон «Согласование документа»

Для открытия окна «Свойства карты» дважды щелкните по свободному полю окна шаблона (при этом не должно быть выделено ни одного из объектов карты, двойной щелчок по выделенному элементу карты открывает окно свойств данного элемента).

23.2 Определение возможных исполнителей этапов

Откройте вкладку «Исполнители» окна свойств карты, задайте основных исполнителей этапов (см. рисунок 232).

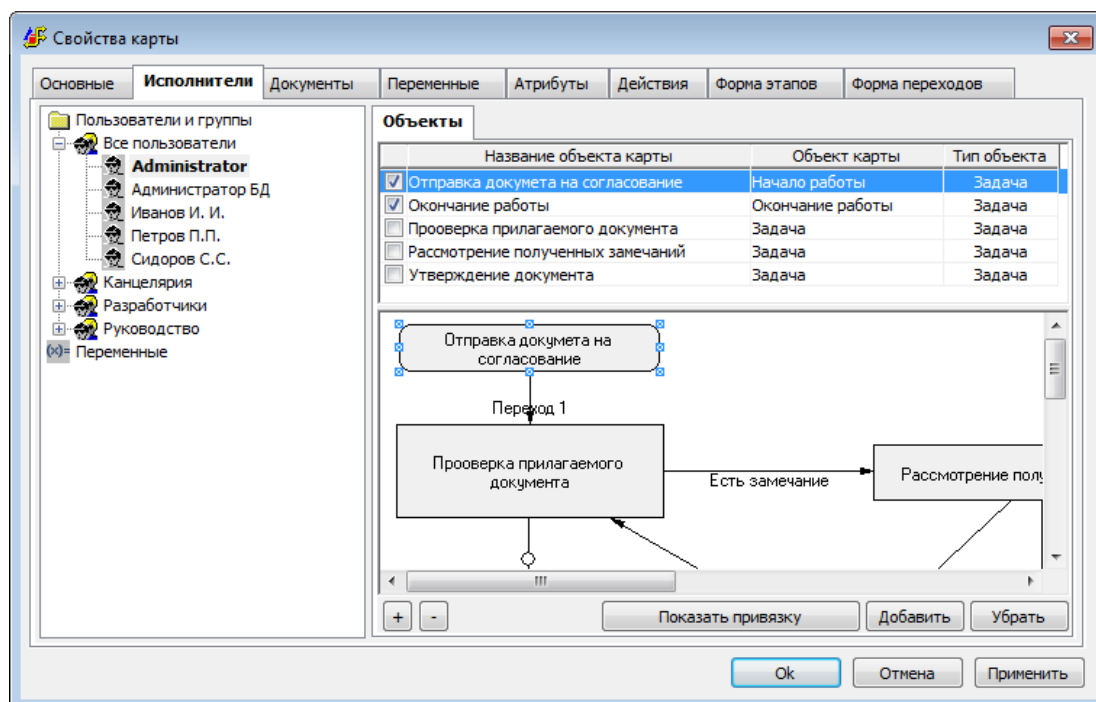




Рисунок 232 Задание исполнителей этапов

В левой части окна выделите пользователя, в правой верхней части установите флажок напротив требуемого этапа. В правой нижней части окна можно выделить очередной этап работы, пользуясь графическим представлением шаблона. Кнопки   позволяют изменять масштаб изображения.

23.2.1 Дополнительная информация по назначению исполнителей

Исполнителей можно также задавать в свойствах каждого отдельного этапа, на вкладке «Исполнители» (Рисунок 233).

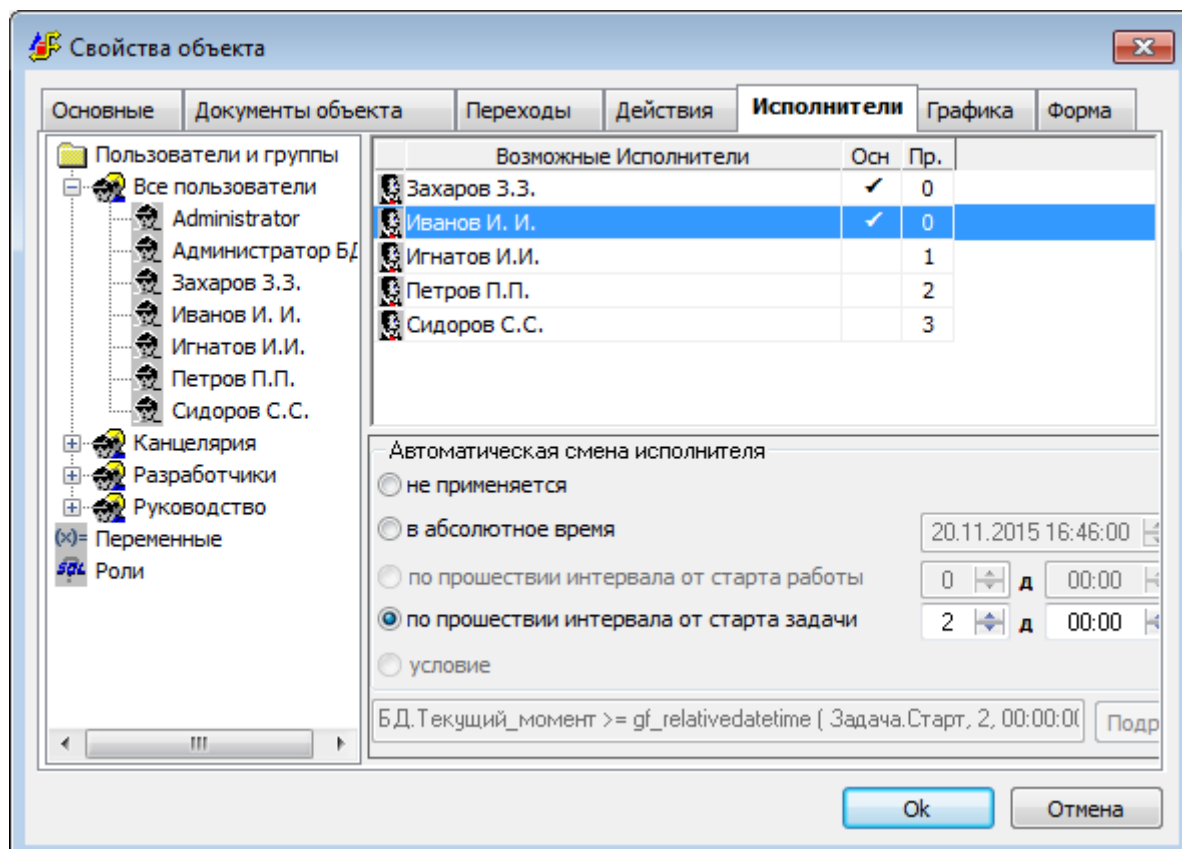


Рисунок 233 Свойства этапа работы, вкладка "Исполнители", включена автоматическая смена исполнителей

Добавление и удаление исполнителей через свойства объекта карты осуществляется методом «drag-and-drop» – исполнители из списка переносятся в список «Возможные Исполнители» и обратно.

Возможные исполнители этапа могут быть основными (иначе говоря, «иметь приоритет равный нулю») или не основными, (соответственно – «иметь приоритет отличный от нуля») – см. рисунок 233, правая верхняя часть формы, колонки «Осн» и «Пр.». Смена приоритета осуществляется двойным щелчком левой клавишей мыши по строке возможного исполнителя.

При выполнении работы (см. п. 23.10), если не использовать автоматическую смену исполнителей и не назначать исполнителей в форме перехода, сообщение очередного этапа работы направляется всем основным исполнителям этапа. Если этапы работы выполняются последовательно (то есть, параллельное выполнение нескольких этапов не используется), то после открытия и принятия к исполнению сообщения одним из исполнителей, у остальных исполнителей данное сообщение аннулируется.

Возможна автоматическая смена исполнителей этапа. Автоматическая смена может выполняться либо по прошествии заданного интервала от старта задачи – настройка показана на рисунке (см. рисунок 233), – либо в абсолютное время. Применение автоматической смены исполнителя означает, что при непринятии к исполнению задачи ни одним из основных исполнителей, при наступлении условия смены исполнителей у всех основных исполнителей задача аннулируется и перенаправляется всем не основным исполнителям. После открытия и принятия к исполнению задачи любым из не основных исполнителей, у остальных задача также аннулируется.

Откройте вкладку «Исполнители» второго этапа, добавьте несколько возможных исполнителей, основных и не основных, практически ознакомьтесь с настройкой автоматической смены исполнителей.

Автоматическая смена исполнителей выполняется только при запущенном сервере автоматических этапов (см. п. 23.12).

Для назначения исполнителей могут быть использованы переменные, значения которых могут определяться различными способами, в том числе с использованием действий (см. пп. 23.15.1, 23.16.1)

23.3 Отключение вкладок «Сообщение» и «Документы». Работа без открытия окна сообщения. Хранение сообщения начала работы

При настройке свойств задачи, флажки вкладок в окне задачи (Рисунок 234) позволяют отключить неиспользуемые вкладки в сообщении данного этапа работы. Управлять вкладками можно и в динамическом режиме, с помощью действий над переменными шаблона (см. п. 23.16.3). При отключенных флажках вкладок в окне задачи включить показ соответствующих вкладок выполнением действий невозможно.

В свойствах начального этапа работы по умолчанию включен флажок «Автоматически помещать в корзину» (Рисунок 234).

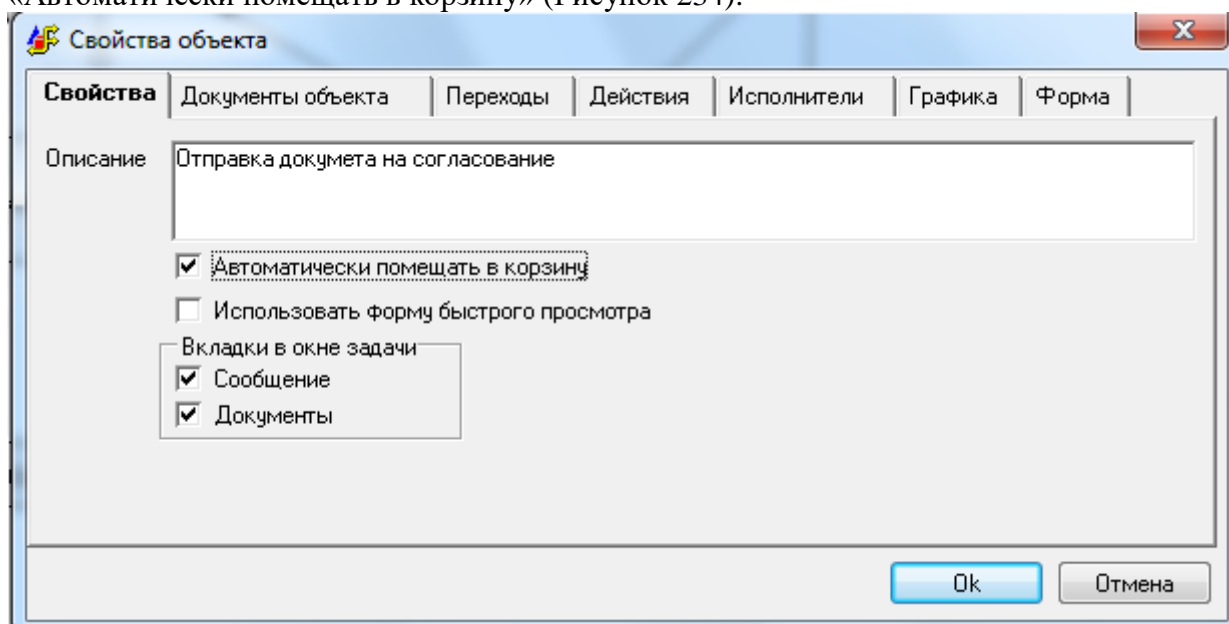


Рисунок 234 Окно свойств начального этапа работы

Это означает, что при запуске работы, сообщение, относящееся к её началу, будет автоматически помещено в корзину. Если данный флажок снять, сообщение будет оставаться в папке «Входящие» – иногда это требуется, чтобы обеспечить инициатору работы более удобный доступ к заполненной им ранее форме.

Флажок «Использовать форму быстрого просмотра», при включении, позволяет пользователю работать с полученной задачей непосредственно в окне «Задания и сообщения», без открытия сообщения задачи в отдельном окне.

23.4 Создание документов шаблона. Привязки документов шаблона к этапам работы

На вкладке «Документы» окна свойств карты задайте документы шаблона. Нажимая на кнопку «Документ» Добавьте «Объект» и «Документ-файл». Документы – файлы могут быть Разделяемыми или локальными. В первом случае передается ссылка на файл, во втором – файл копируется получателю.

Добавленный документ – объект назовите «Объект_документа», (либо оставьте предопределенное название «Объект»). Теперь при запуске работы по данному шаблону, к

сообщению может быть присоединен файл из файловой системы, а также выбранный объект Lotsia PDM.

Добавлять и удалять документы в сообщение задачи, в соответствии с настроенными документами шаблона, можно как вручную, так и с помощью действий (см. п. 23.17). Возможность работы пользователя с документами шаблона вручную может быть ограничена путем установки соответствующих привязок для каждого документа шаблона и этапа работы – вкладка «Объекты» для выделенного документа шаблона (вкладка «Документы» окна свойств карты или шаблона) (Рисунок 235).

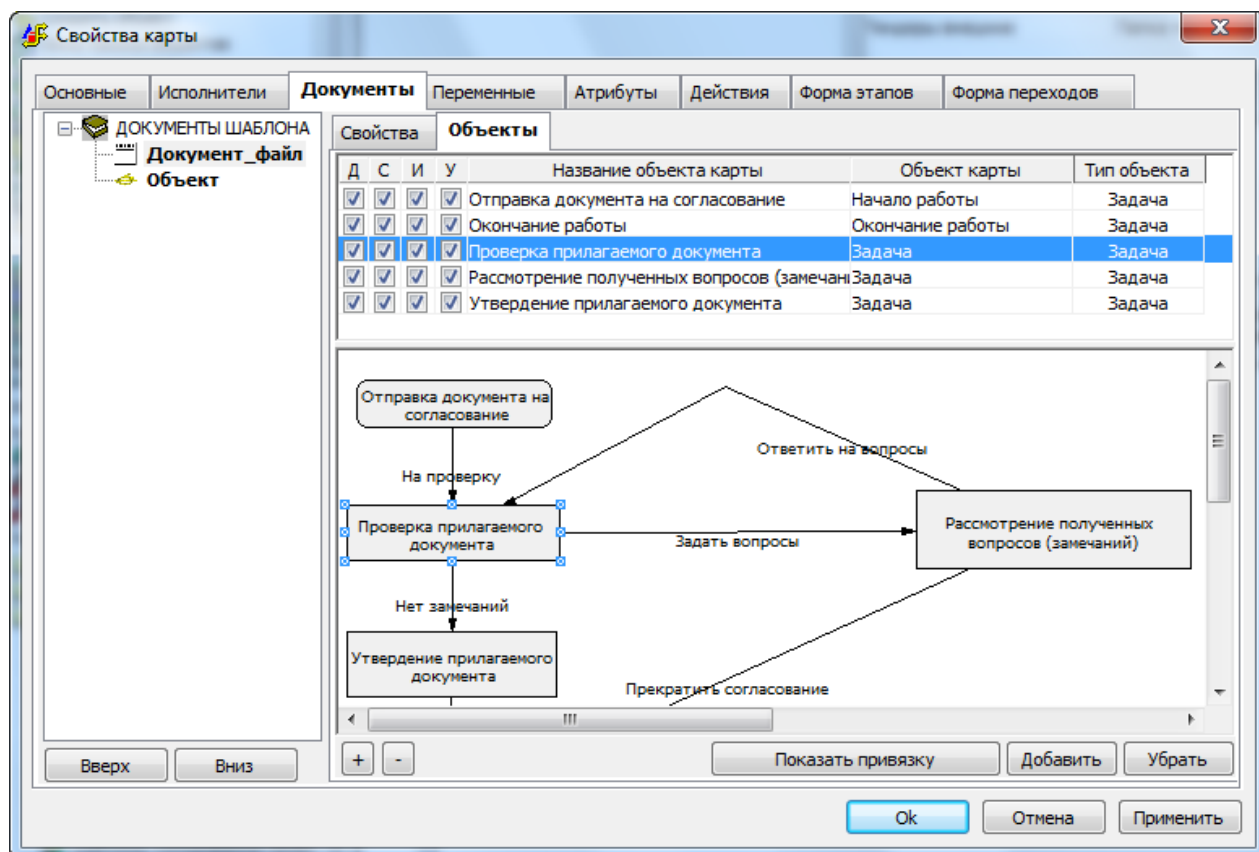


Рисунок 235 Окно свойств шаблона работы. Привязки документа шаблона к этапам работы

При отсутствии флажка «Д» (доступ), исполнитель этапа не будет иметь доступа к документу задачи (на действия, работающие с документами данного шаблона, привязки не влияют).

23.5 Формы этапов работы и переходов

Формы в шаблоне работ могут быть настроены для каждого этапа и каждого перехода. Для облегчения работы путем однократного выполнения общих однотипных настроек и обеспечения возможности настройки форм работы в едином стиле, можно вначале провести настройку форм «По умолчанию», а в дальнейшем выполнить дополнительную индивидуальную настройку формы для каждого элемента шаблона.

Перейдите на вкладки «Форма этапов» и «Форма переходов», настройте формы «По умолчанию». Настройки форм (левая часть окна) на данных вкладках выполняются аналогично редактированию атрибутивных форм (см. п. 10.2) или форм для этапов действий над объектами (п. 17.3.1) (имеющиеся дополнительные возможности и особенности пока не рассматриваем). Результат – см. рисунок 236.

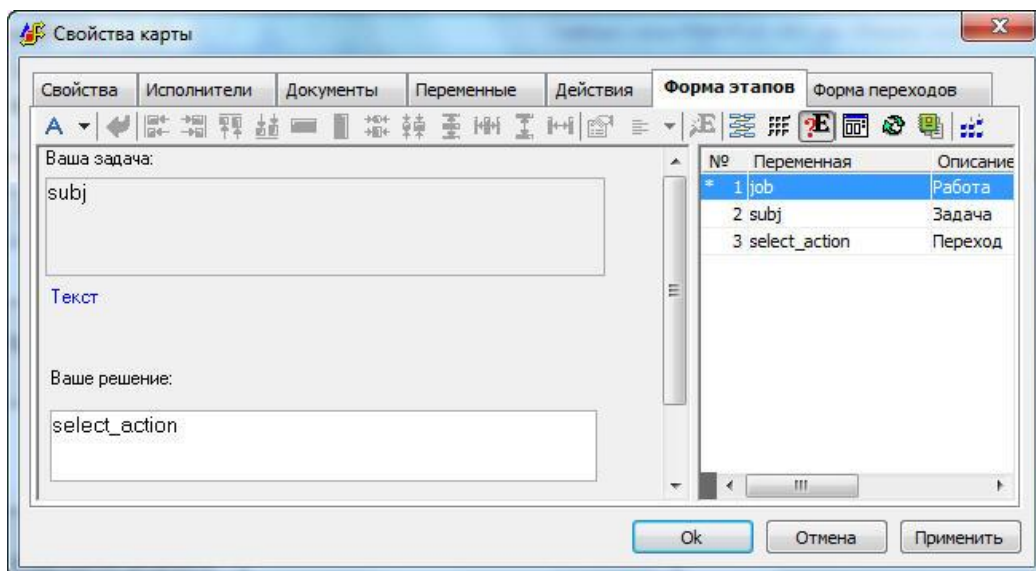


Рисунок 236 Пример настройки формы по умолчанию для этапов

При настройке данной формы «по умолчанию» выполнено следующее:

- удалена из формы предопределенная колонка «Job» и текстовое поле «Работа»;
- изменены размеры и расположение элементов;
- добавлено текстовое поле, изменен цвет шрифта для него;
- в предопределенном текстовом поле вместо «Переход:» введен текст «Укажите:».

Выполните собственные настройки аналогично.

Для форм переходов оставьте только поле сообщения и его текстовый заголовок. Нажмите «ОК» и сохраните шаблон. Настройте формы каждого элемента шаблона, пользуясь формами по умолчанию (в окне свойств каждого элемента шаблона открывайте вкладку «Форма»).

Для задач в текстовое поле введите инструкции исполнителям, там, где исходящий переход единственный, удалите предопределенную колонку выбора перехода (см. рисунок 237).

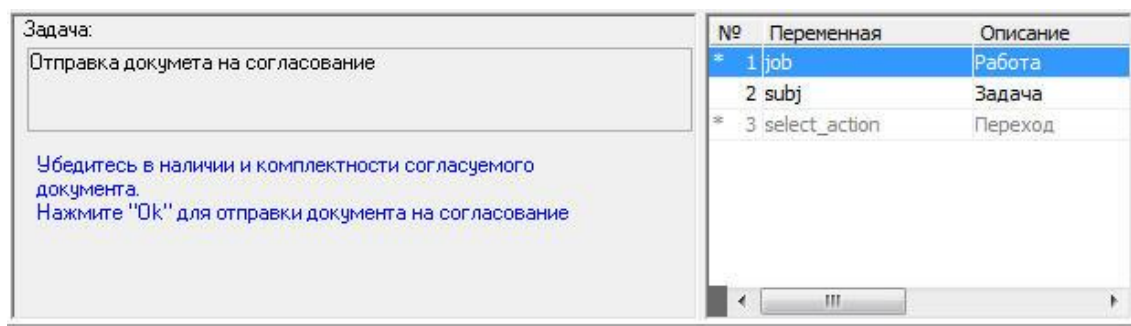


Рисунок 237. Возможная настройка простейшей формы первого этапа данной работы

В случае, если на данном этапе производится выбор перехода, в форме этапа установите стиль списка выбора поля перехода в положение «Переключатель» (Рисунок 238).

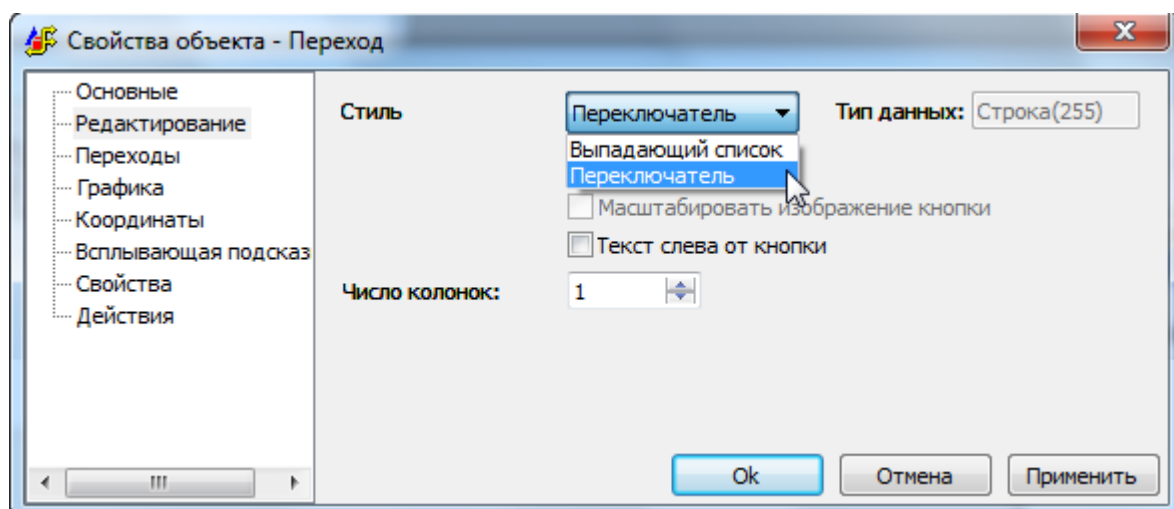






Рисунок 238. Задание стиля списка выбора перехода

После корректировки положения и размеров полей форма этапа примет законченный вид (см. рисунок 239).

№	Переменная	Описание
*	1 job	Работа
	2 subj	Задача
	3 select_action	Переход

Рисунок 239. Вид формы этапа при установленном в положение «Список переходов» выборе перехода

Аналогичным образом проведите настройку форм остальных задач.

При настройке форм удобно заимствовать ранее созданные формы – используйте кнопку , а для быстрой настройки размеров выделенных элементов . Для выравнивания и других действий используйте инструментальную панель окна редактора форм.  формы – кнопка 

23.5.1 Формы переходов

Помимо форм этапов работы могут использоваться и формы переходов. Исполнитель этапа работает с формой выполняемого исходящего перехода. Настройка форм переходов аналогична настройке форм задач, но при настройке формы перехода, в окне редактора форм видима и фактически настраивается нижняя часть формы. В верхней, не настраиваемой части формы, может располагаться поле со списком исполнителей следующего этапа работы. Управление показом списка исполнителей осуществляется

установкой/снятием флажка «Показывать список исполнителей» на вкладке «Свойства» в окне свойств перехода (Рисунок 240).

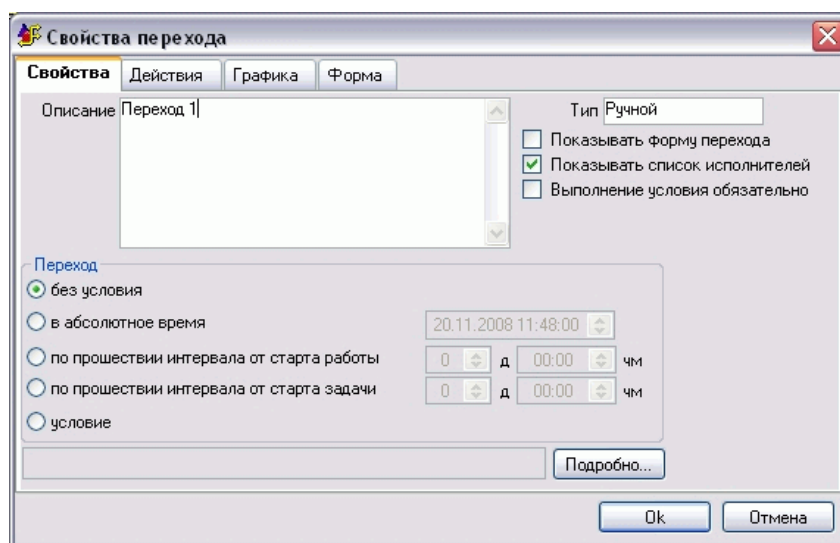


Рисунок 240 Задание свойств перехода

При неустановленном флажке «Показывать форму перехода», форма перехода открываться не будет. Заметим, что в большом количестве случаев использовать формы переходов нет необходимости. В рассматриваемом учебном шаблоне можно включить показ формы перехода от первого этапа к следующему, а также формы переходов «Есть вопросы» и «Ответить на вопросы». При настройке указанных форм оставьте в нижней части формы только поле для ввода сообщений и текстовое поле с его заголовком (Рисунок 241).

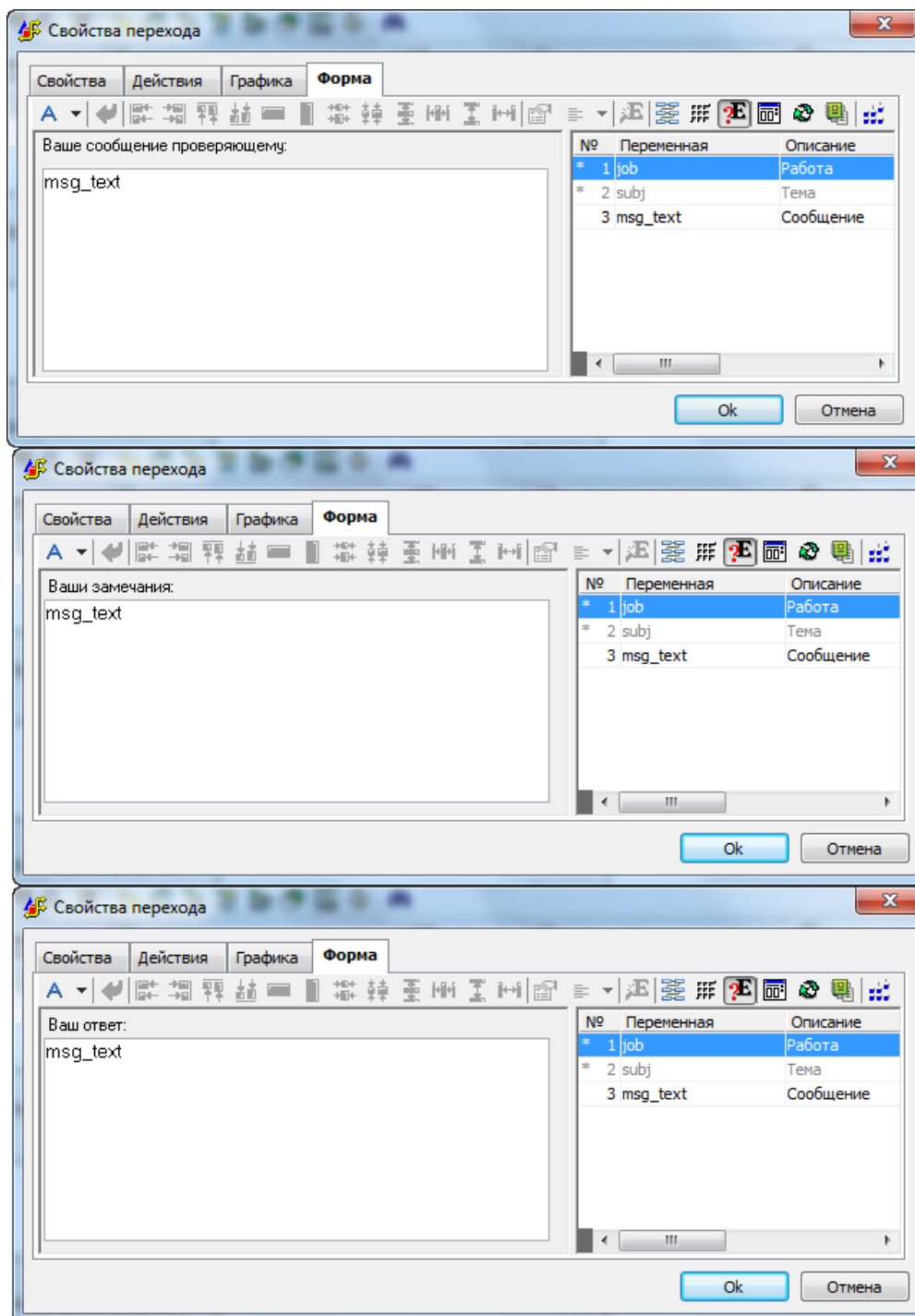


Рисунок 241 Настройка форм переходов шаблона

В ознакомительных целях оставьте показ списка исполнителей для первого перехода включенным.

Для остальных переходов показ формы включать не будем.

23.6 Просмотр формы задачи без открытия входящего сообщения в отдельном окне

С predetermined сообщениями можно работать и без открытия входящего сообщения в отдельном окне. Чтобы форма выделенной задачи была доступна в окне «Задания и сообщения» необходимо в настройках свойств соответствующего этапа работы включить флажок «Использовать форму быстрого просмотра» (см. рисунок 234, Рисунок 242, Рисунок 247).

23.7 Автоматическое завершение работы

В нашем примере этап «Окончание работы» смысловой нагрузки не несет, однако удалить данный элемент из шаблона нельзя, он обязательный. Чтобы отключить показ сообщения об окончании работы, в свойствах данного этапа включите флажок «Автоматически завершать работу» (Рисунок 242).

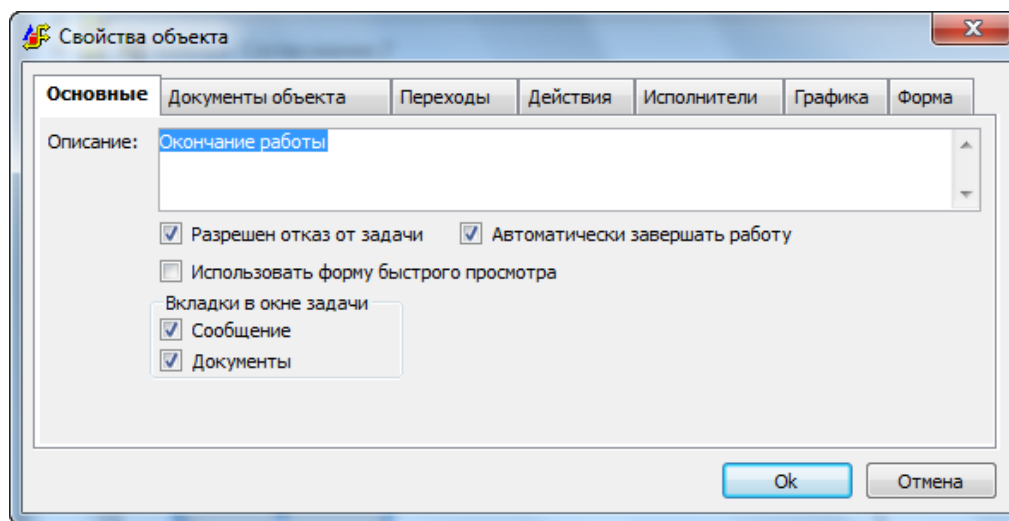


Рисунок 242 Отключение показа сообщения этапа окончания работы

23.8 Проверка шаблона

Проверьте шаблон, используя пункт «Проверка карты» контекстного меню шаблона. При отсутствии этапов, не соединенных входящими и исходящим переходами, а также этапов, не имеющих исполнителей, проверка карты должна выполняться успешно (в противном случае будет отображен список ошибок).

23.9 Различные способы запуска работы

Запустите работу по шаблону из Перечня возможных работ. Откройте вкладку «Документы», присоедините объект согласуемого документа.

При запуске работы по выделенному в дереве связей объекту (пункт верхнего меню «Объект» → «Запустить работу») согласуемый документ присоединяется к работе автоматически.

В атрибутивной форме для объекта документа создайте кнопку «На согласование» (Главное меню → «Администрирование» → «Настройка интерфейса» → «Настройка форм» → открыть либо создать требуемую форму типа «Атрибуты» → в окне редактора форм контекстное меню левой части окна → «Создать» → «Кнопку»).

Создайте действие для запуска данной работы (Главное меню → «Администрирование» → «Структура данных» → «Редактор действий» → «Создать»). Действие может состоять из одной строки: `WndWorkStart (Shabl_1 , a_Object)`, где `Shabl_1` – переменная типа «Шаблон работы», в качестве значения по умолчанию для данной переменной указать созданный шаблон работы).

Сопоставьте его указанной кнопке (при настройке атрибутивной формы, в свойствах объекта кнопки указать требуемое действие). Теперь вы можете запускать данную работу по нажатию кнопки в форме объекта документа. Отметим, что с помощью действия над объектами работа может быть запущена и без открытия окна начала работы, для чего используется функция `WorkStart`.

Также как и отправка свободных сообщений, запуск работ может осуществляться по выделенным в окне подборки или окне поиска объектам документов, по выделенным в списке сообщений сообщениям, по выделенному документу архива и др. Об использовании данных методов вы можете прочесть в документации по системе Lotsia PDM PLUS. Отметим, что для того, чтобы пользователь смог запустить работу, ему во многих случаях требуется выбирать работу в перечне возможных работ. Чтобы работа отображалась в указанном перечне, данный пользователь должен входить в список возможных исполнителей первого этапа данной работы. В случае запуска работы с помощью действия, когда шаблон работы выбирать не требуется, данного ограничения нет.

23.10 Выполнение работы

После запуска работы исполнитель видит форму первого этапа работы (Рисунок 243).

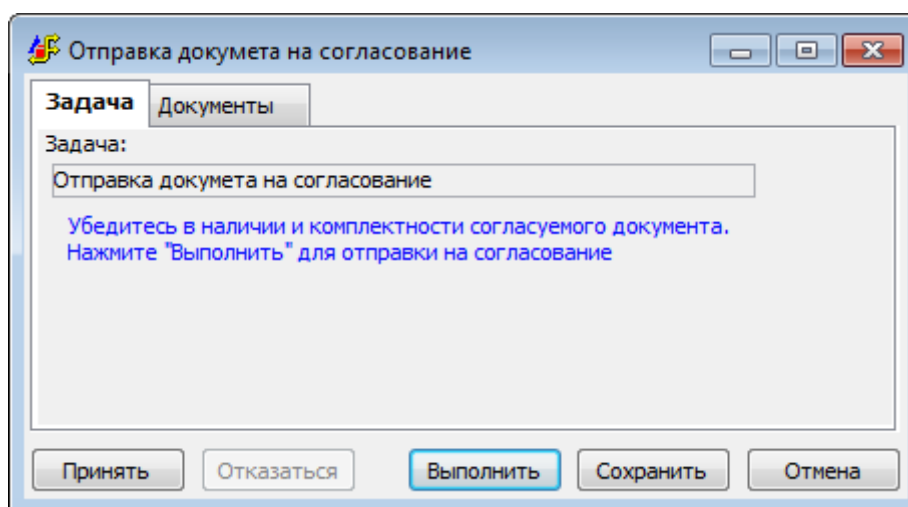


Рисунок 243 Начало работы

После присоединения к сообщению документа и нажатия «Выполнить», открывается форма выполняемого исходящего перехода (Рисунок 244).

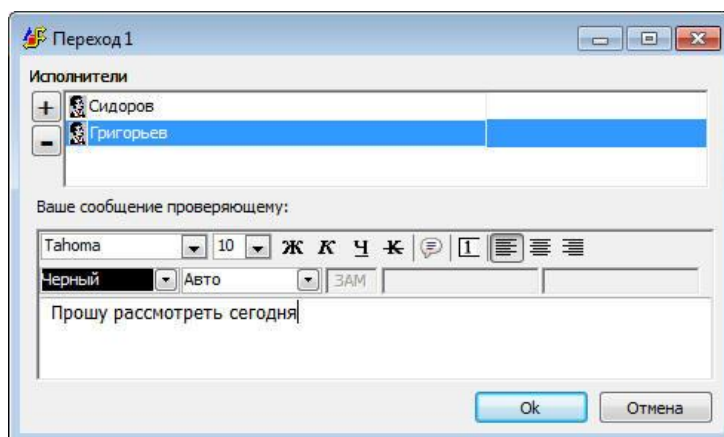




Рисунок 244 Форма перехода со списком исполнителей

В данной форме пользователь может ввести сообщение для исполнителя следующего этапа работы, а также изменить список исполнителей следующего этапа, которым будет направлено предопределенное сообщение. Если список не изменять, сообщение направляется основным исполнителям (см. п. 23.2.1). При нажатии кнопки  выделенный исполнитель (или исполнители) будет удален из списка, а при нажатии  будет открыт полный список всех возможных исполнителей следующего этапа, в котором

присутствуют как основные (то есть, имеющие приоритет «По умолчанию») исполнители, так и не основные (то есть, с приоритетом отличным от нуля) (Рисунок 245).

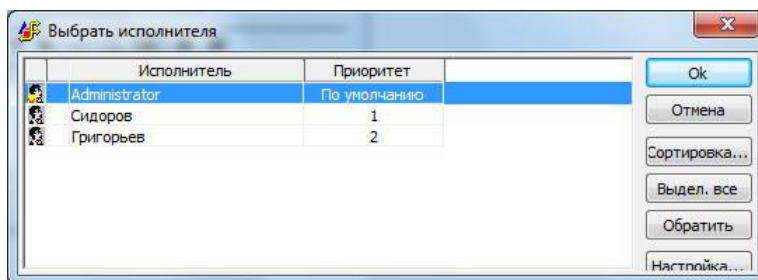


Рисунок 245 Выбор из полного списка возможных исполнителей

Выбранные в данном списке исполнители попадают в список исполнителей следующего этапа работы, им направляется предопределенное сообщение.

Исполнитель очередного этапа (в нашем случае это проверяющий) получает и открывает соответствующее предопределенное сообщение (Рисунок 246).

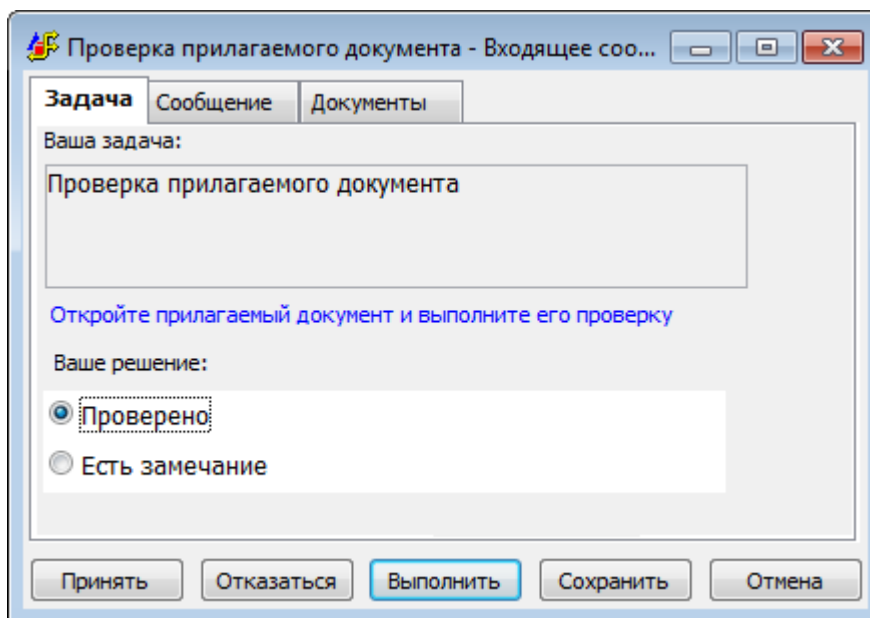


Рисунок 246 Вкладка «Задача» предопределенного сообщения исполнителя второго этапа работы

Ознакомившись с заданием, текстом сообщения (на вкладке «Сообщение») и открыв и ознакомившись с прилагаемым документом (на вкладке «Документы»), исполнитель этапа устанавливает переключатель Проверено/Есть вопросы в требуемое положение. При этом определяется следующий выполняемый переход.

Если выбрано «Проверено», по нажатию «Выполнить» выполняется соответствующий переход к задаче «Утверждение прилагаемого документа» (Рисунок 231), при выборе «Есть вопросы» открывается форма соответствующего перехода, где можно ввести текст вопросов (см. рисунок 241), после чего будет выполнен переход к задаче «Рассмотрение полученных вопросов». Дальнейшая последовательность выполнения данного процесса очевидна.

23.11 Отказ от выполнения задачи

Исполнитель этапа может отказаться от его выполнения, нажав кнопку «Отказаться» в форме задачи (Рисунок 246) и указать в открывшемся окне причину отказа. При отказе от выполнения всех исполнителей, которым была направлена задача,

процесс возвращается на один шаг назад. Исполнитель предыдущего этапа получает соответствующее сообщение с указанием причин отказа. Задание может быть направлено повторно тем же самым или другим исполнителям (из возможных исполнителей этапа).

Чтобы запретить исполнителю отказ от выполнения задачи, при настройке шаблона работы, в свойствах объекта соответствующего этапа следует снять флажок «Разрешен отказ от задачи» (см. рисунок 247)

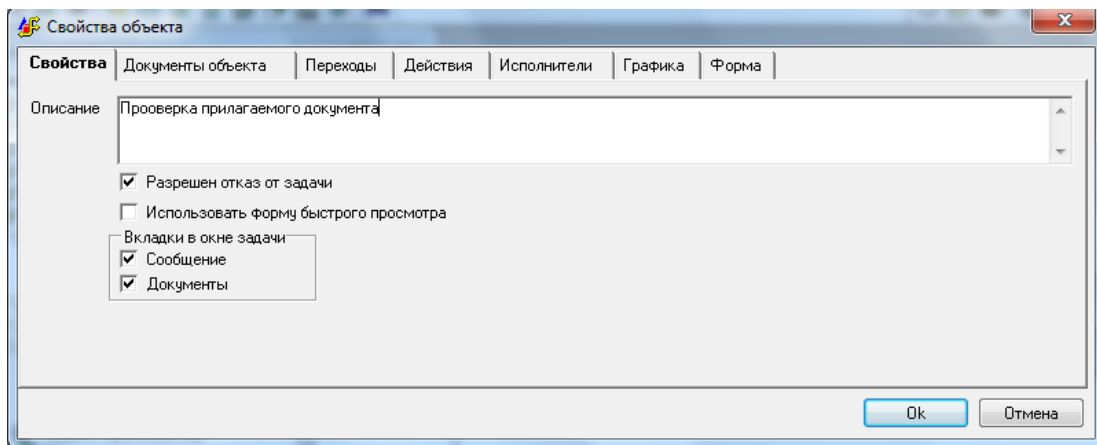


Рисунок 247 Окно свойств объекта шаблона работы, вкладка «Свойства»

Управление кнопкой «Отказаться» возможно также путем выполнения действий над переменными шаблона (см. п. 23.16.3).

23.12 Сервер автоматических этапов

Для обеспечения выполнения автоматических элементов работы используется сервер автоматических этапов, который функционирует в автономном режиме и не требует вмешательства пользователя.

Командная строка для запуска сервера автоматических этапов в качестве приложения выглядит следующим образом:

partyp.exe -uid=<Login> -pwd=<password> -auto , где:

-uid=<Login> – параметр для задания имени регистрации, вместо <Login> указать имя регистрации для подключения к БД. Можно не использовать, а ввести в окне регистрации

-pwd=<password> – параметр для задания пароля, вместо <password> указать пароль для подключения к БД. Можно не использовать, а ввести в окне регистрации

-auto – обязательный параметр, показывающий, что производится запуск Lotsia Workflow для работы в режиме сервера автоматических этапов

Пример запуска программы:

partyp -uid=dba -pwd=sql -auto

Сервер автоматических этапов не занимает дополнительную клиентскую лицензию при подключении к электронному ключу, но при запуске сервера автоматических этапов на каждом отдельном компьютере, на котором в данный момент не запущено клиентской программы Lotsia PDM PLUS, емкость электронного ключа уменьшается на одну лицензию. Данное замечание следует учитывать при определении требуемой физической емкости приобретаемого ключа аппаратной защиты.

После запуска сервера открывается окно сервера автоматических этапов (Рисунок 248).

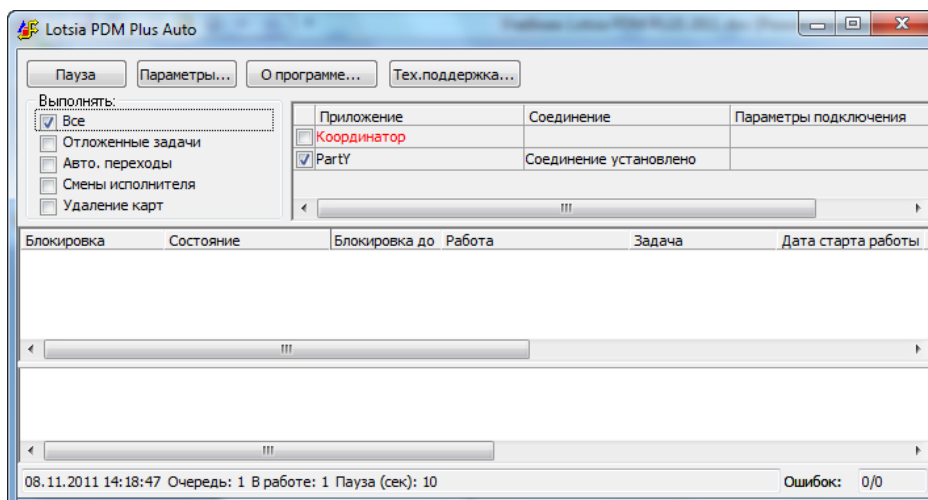




Рисунок 248 Окно сервера автоматических этапов

В области System Tray появится пиктограмма . Такой вид пиктограммы говорит о том, что сервер занят. Например, происходит загрузка или проверка условий. Для открытия Окна сервера автоматических этапов дождитесь, пока пиктограмма примет вид  и дважды щелкните на ней левой кнопкой мыши. Для запуска сервера автоматических этапов в минимизированном виде необходимо выполнить соответствующие установки в ярлыке или команде, запускающей сервер.

Включенный флажок «Все» обеспечивает автоматическое выполнение автоматических переходов (см. п. 23.14), выполнение серверных действий, удаление карт и автоматическую смену исполнителей (см. п. 23.2.1).

Автоматическое выполнение указанных автоматических элементов можно включать/отключать и по отдельности (флажок «Отложенные задачи» управляет выполнением серверных действий).

Выполнение серверных действий предполагает запуск сервера автоматических этапов от имени пользователя, имеющего необходимые права и приоритеты. Данный пользователь (лично, либо получив от группы, в которую входит) должен иметь возможность выполнять функционал, заложенный в серверных действиях. Обычно это следующие возможности для обрабатываемых объектов и проектов: создавать и разрывать связи, удалять и устанавливать атрибуты, устанавливать права доступа и т.д., а также административную привилегию «Запуск сервера автоматических этапов» (Рисунок 249) (Главное меню → Администрирование → Пользователи → ...). На практике в качестве такого пользователя чаще всего выступает администратор, что обеспечивает возможность гарантированного выполнения любого действия над любым объектом без дополнительных настроек.

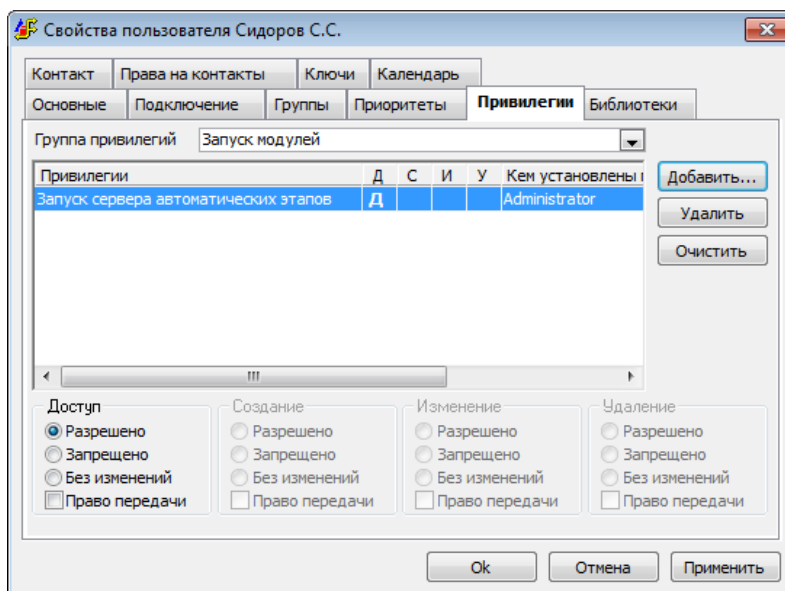


Рисунок 249 Пользователю установлена привилегия «Запуск сервера автоматических этапов». Пример

По нажатию кнопки «Параметры...» (Рисунок 248) открывается окно «Настройка сервера» (см. рисунок 250), где на вкладке «Сервер автоматических этапов» могут быть установлены дополнительные параметры сервера автоматических этапов. Аналогичная вкладка может быть открыта при определении общих параметров БД (Главное меню → Администрирование → Общие параметры БД → вкладка «Сервер автоматических этапов»).

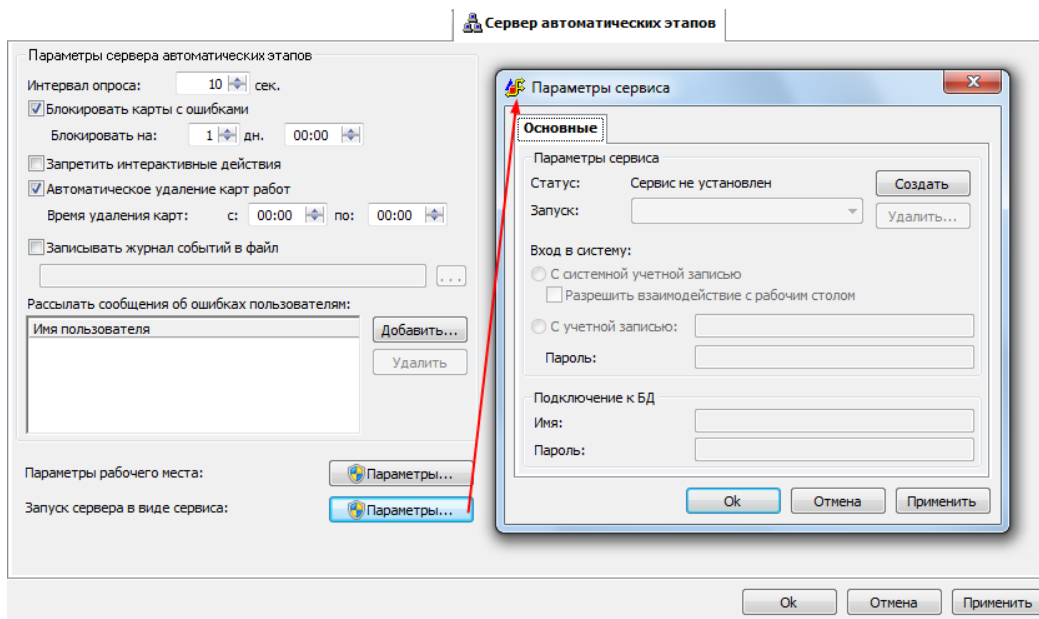


Рисунок 250 Настройка сервера автоматических этапов

При установке флажка «Блокировать карты с ошибками» будет автоматически устанавливаться блокировка карт, при выполнении которых произошли ошибки. В списке текущих задач сервера автоматических этапов для каждой карты можно вручную установить флажок «Блокировка». Заблокированные карты также не будут обрабатываться сервером автоматических этапов.

Имеется возможность ведения журнала событий. Флажок «Записывать журнал событий в файл» управляет ведением журнала событий. При включении данного флажка

становится возможным в поле, расположенным под данным флажком, указать файл журнала событий.

Включенный флажок «Запретить интерактивные действия» не позволяет выполнять на сервере автоматических этапов действия над объектами, которые требуют от пользователя ввода данных в формы – будет возникать ошибка выполнения таких действий.

В поле «Интервал опроса» устанавливается регулярность выполнения отмеченных флажками операций.

Для автоматической рассылки сообщений об ошибках на сервере автоматических этапов необходимо указать получателей данных сообщений. Пользователи – получатели сообщений об ошибках, указываются в поле «Рассылать сообщения об ошибках пользователям:» (см. рисунок 250). При рассылке автоматических сообщений рекомендуется автоматически блокировать карты с ошибками (включать флажок «Блокировать карты с ошибками»), в противном случае будет рассылаться большое количество однотипных сообщений.

Сервер автоматических этапов может быть запущен в виде службы (сервиса). Для этого (см. рисунок 250) следует в области «Параметры сервиса» нажать «Параметры...». Далее, в открывшемся окне «Параметры сервиса» выполнить требуемые настройки. Данные настройки также можно выполнить при определении общих параметров БД (Главное меню → Администрирование → Общие параметры БД → вкладка «Сервер автоматических этапов»).

23.13 Проверка условий выполнения переходов

При запуске работы по созданному шаблону «Согласование документа» (Рисунок 231), очевидно, не имеет смысла выполнять переход ко второму этапу работы до тех пор, пока согласуемый документ не присоединен к сообщению. Таким образом, наличие документа целесообразно сделать условием выполнения перехода к этапу проверки документа.

В окне свойств перехода, на вкладке «Свойства» (Рисунок 251) могут быть заданы условия выполнения перехода.

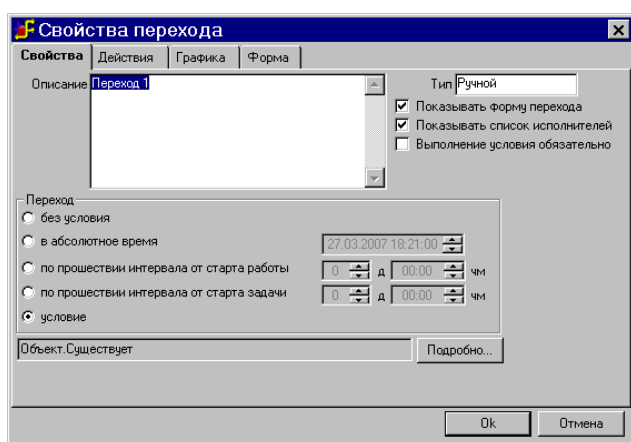


Рисунок 251 Окно свойств перехода, вкладка «Свойства» с заданным условием

При включенном флажке «Выполнение условия обязательно» переход нельзя будет выполнить при невыполнении условия, если флажок отключен, то переход можно будет выполнять, но в любом случае будет выдаваться сообщение о невыполнении условия перехода.

При определении выполнения переходов, помимо задания времени выполнения перехода, можно использовать выражения. Для открытия редактора выражений нажмите кнопку «Подробно...».

В выражениях для задания условий переходов могут быть использованы различные функции, а помимо или вместе с ними – информация о документах работы, в том числе значения атрибутов и др. На вкладке «Атрибуты» редактора выражений выберите ранее созданный Вами документ (см. п. 23.4), который является объектом Lotsia PDM PLUS, а в дереве «Атрибуты:», – «Существует (лог)» (Рисунок 252).

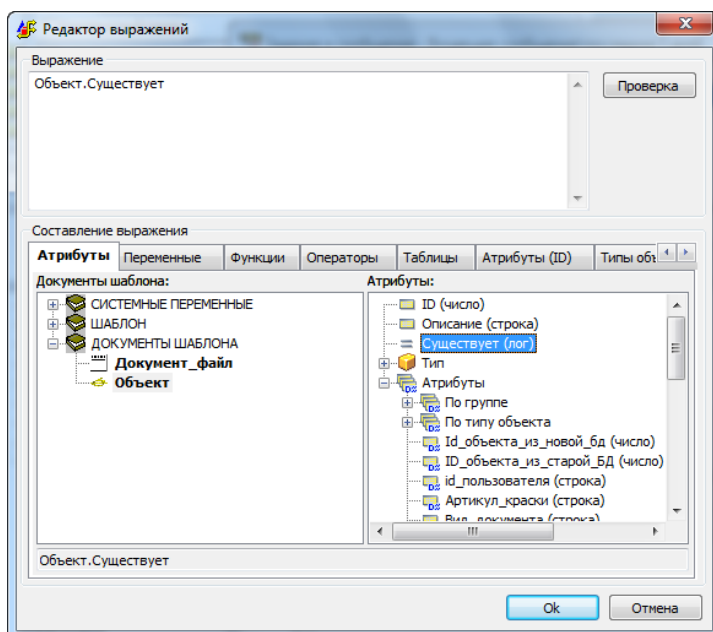


Рисунок 252 Задание условия – наличие документа

Нажмите «Ok» для сохранения выражения, «Ok» для сохранения свойств перехода, сохраните шаблон. Для проверки работы условия запустите работу из перечня возможных работ, без присоединения документа. При попытке выполнения перехода будет выдано соответствующее сообщение об отсутствии документа работы.

23.14 Автоматические переходы

В ряде случаев требуется, чтобы выполнение переходов выполнялось без вмешательства пользователя, то есть, автоматически. Для автоматического выполнения в свойствах перехода, на вкладке «Свойства» в поле «Тип» следует установить «Автоматический». (Рисунок 253)

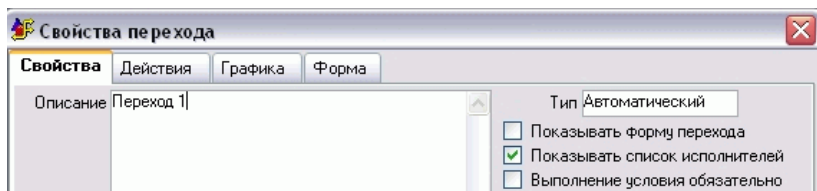


Рисунок 253 Указан тип перехода – «Автоматический»

Для автоматических переходов включение флажков «Показывать форму перехода» и «Показывать список исполнителей» игнорируется.

Для учебного шаблона согласования документа автоматический переход можно использовать, например, для возврата процесса на начало работы при отсутствии проверяющих (непринятия к исполнению второго этапа работы) в течение заданного

времени (Рисунок 254). Для того чтобы автоматические переходы выполнялись, необходимо, чтобы работал сервер автоматических этапов (см. п. 23.12)

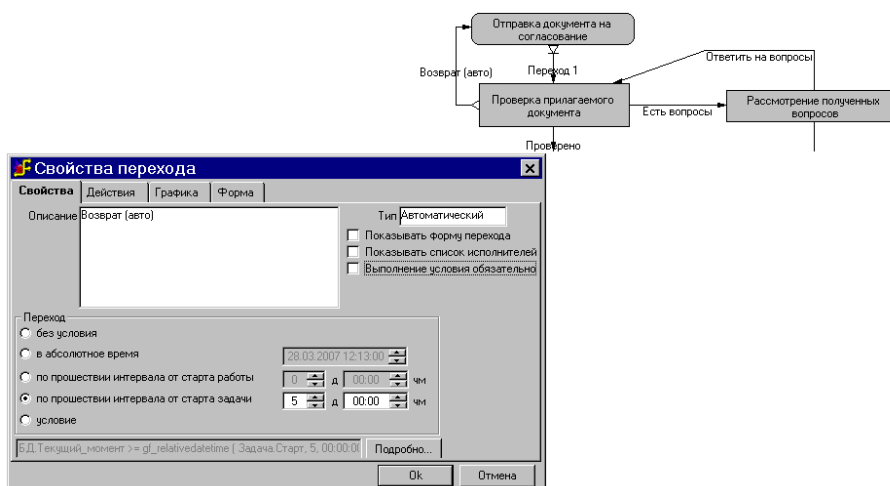


Рисунок 254 Добавлен переход «Возврат (авто)»

Чтобы проверить работу автоматического перехода, установите условие его выполнения – например, 1 минута от старта задачи, убедитесь, что автоматический переход выполняется.

23.15 Переменные Lotsia Workflow. Общая информация

В предопределенных процессах Lotsia Workflow всегда используются переменные (даже в нашем простейшем процессе, такие сущности как сообщение, задача и др. – предопределенные переменные шаблона). Рассмотрим возможности применения переменных более подробно.

В работах Lotsia Workflow могут непосредственно использоваться переменные данного шаблона работы. Для присвоения значений указанным переменным могут использоваться выражения, содержащие:

- переменные шаблона
- системные переменные
- атрибуты карт работ
- информацию об объектах Lotsia PDM (значения атрибутов и др.)
- информация об исполнителе этапов и др.

Переменные шаблона могут быть:

- предопределенными (создаются автоматически при создании шаблона и его элементов, имеют стандартные имена; доступный набор таких переменных несколько различается для различных этапов работы и переходов);
- пользовательскими.

Переменные шаблона доступны только в рамках работы, запущенной по данному шаблону. Напомним, что при каждом запуске работы, шаблон копируется, таким образом, создается отдельный экземпляр карты для каждого запущенного процесса, с собственным набором переменных шаблона.

Системные переменные и атрибуты карт работ создаются вне шаблона и после создания становятся доступны для всех работ. (Создаются: Главное меню →

Администрирование → Общие параметры БД → Вкладка «Переменные» (или, соответственно, «Атрибуты карт») → пункт «Добавить» контекстного меню).

Назначение системных переменных – хранение информации, общей для всех выполняемых работ, где данная переменная используется. Текущее значение системной переменной – общее для всех карт работ.

Основное предназначение и отличие атрибутов карт работ – возможность их использования в окне «Задания и сообщения», аналогично предопределенным колонкам «Тема», «Отправлено» и другим свойствам сообщений. По атрибутам карт работ возможна сортировка и фильтрация сообщений. Текущее значение атрибута карты работ – собственное для каждой карты (каждой запущенной работы).

Использование системных переменных, атрибутов карт работ и информации из объектов Lotsia PDM возможно в выражениях действий (см. 23.16), либо при задании условий (выражений) выполнения переходов или выполнения действий.

Считывание значений системных переменных и атрибутов карт работ при выполнении работ может осуществляться действиями над объектами или действиями Workflow (другое название – «Действия над переменными»), либо при проверке условий (вычислении выражений) при выполнении переходов и действий.

Присвоение значений системным переменным и атрибутам карт работ осуществляется действиями над переменными. Для атрибутов карт значения также могут быть заданы при настройке шаблона работы. (Окно «Свойства карты» → вкладка «Атрибуты»).

Все изменения информации из объектов Lotsia PDM выполняются только действиями над объектами (или ручным вводом при редактировании атрибутивной информации объектов).

Значения переменным шаблона Workflow могут присваиваться:

- ручным вводом (или выбором значений) в формах;
- при выполнении действий над переменными;
- при выполнении действий над объектами;
- по умолчанию.

Переменные шаблона Workflow, с точки зрения присвоения им значений, можно разделить на:

- глобальные;
- локальные

Глобальные переменные имеют единственное текущее значение для всей данной работы (карты работы) в целом. При изменении значения глобальной переменной, её значения одновременно меняется во всех задачах данной работы, где она используется, в том числе и в выполненных задачах. Локальные переменные имеют отдельное значение для каждого этапа данной работы. Отметим, что атрибуты карт работ, как и следует из их названия, относятся к карте в целом, то есть, по данному критерию соответствуют глобальным переменным.

Каждый шаблон содержит набор предопределенных переменных. Их удаление из шаблона невозможно. Можно создавать и удалять пользовательские переменные шаблона.

Список всех пользовательских переменных данного шаблона отображается в окне свойств карты, на вкладке «Переменные». На этой же вкладке, с помощью контекстного меню могут быть созданы новые пользовательские переменные данной карты. Создание

пользовательских переменных шаблона (карты) возможно не только с данной вкладки, но и при настройке форм этапов и переходов, а также при создании действий над переменными.

Для создания пользовательской переменной используется контекстное меню. Для вкладки «Переменные» – пункт «Добавить», для остальных случаев – «Создать переменную». В первом случае переменная добавляется только в список переменных шаблона, в остальных случаях – в указанный список и, соответственно, в редактируемую форму или действие.

Переменные шаблона и системные переменные могут иметь различный тип данных – число, строка, дата и время, пользователь и др., а также различный тип значения – одиночное либо массив. Список доступных переменных различных случаях их использования может различаться.

Далее рассмотрим несколько типичных случаев использования переменных.

23.15.1 Определение исполнителей с использованием переменных

При настройке шаблона работы указание исполнителей в явном виде (то есть, прямое указание пользователя или группы) для каждого этапа не всегда возможно. Например, проверяющие и утверждающие документ сотрудники (то есть, исполнители соответствующих этапов процесса) для учебного процесса согласования (Рисунок 231) будут различными, в зависимости от инициатора работы (то есть, возможного исполнителя первого этапа) и от согласуемого документа.

Кроме того, в качестве инициатора работы могут выступать различные пользователи, а отвечать на вопросы должен, скорее всего, именно тот единственный пользователь, который запустил данную конкретную работу, отдельно указывать его при задании вопроса было бы не правильно.

Для создания переменных, соответствующих проверяющему и утверждающему, откройте форму первого начала работы (Рисунок 237). В контекстном меню правой части окна выберите «Создать переменную»... Последовательно создайте 2 переменных: User_prover и User_utv. (Локальные, тип значения – «одиночное», тип данных – «Пользователь»). Созданные переменные будут добавлены непосредственно в форму данного этапа работы.

Если указанные переменные были созданы заранее (при настройке другого элемента шаблона, либо на вкладке «Переменные» окна свойств карты), для их добавления в форму используйте пункт «Добавить переменную» контекстного меню правой части окна редактора формы.

После выполнения соответствующих настроек графики элементов формы, добавления и редактирования текстовых полей, форма примет следующий вид – см. рисунок 255.

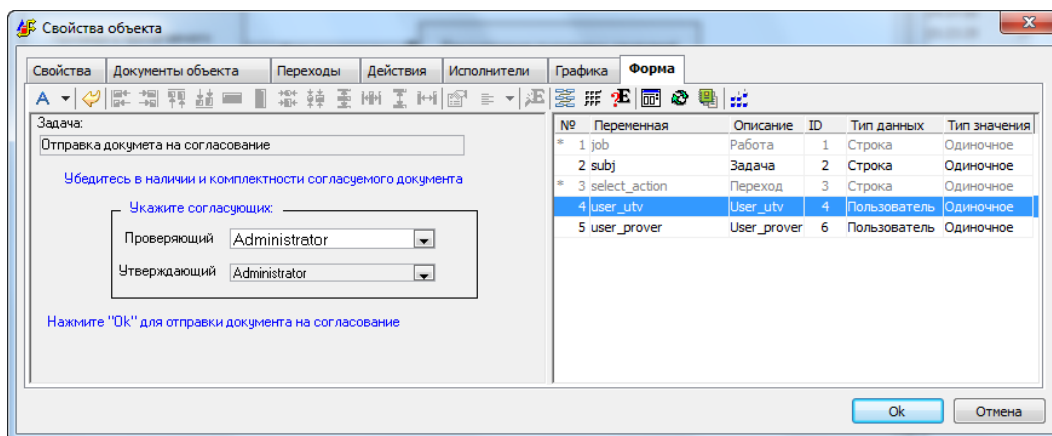


Рисунок 255 Настройка формы начала работы. Добавлены переменные для указания исполнителей последующих этапов

Для переменных типа «Пользователь» можно ограничивать область допустимых значений – то есть, указывать ограничительный список, из которого можно будет осуществлять выбор исполнителей каждого этапа. Если этого не сделать, исполнители будут выбираться из полного списка пользователей. Ограничительный список для переменной типа «Пользователь» (для данной формы) можно указать в свойствах соответствующего объекта формы, на вкладке «Редактирование».

Для того, чтобы данные переменные использовались как исполнители, следует указать их в качестве исполнителей соответствующих этапов работы (Рисунок 256).

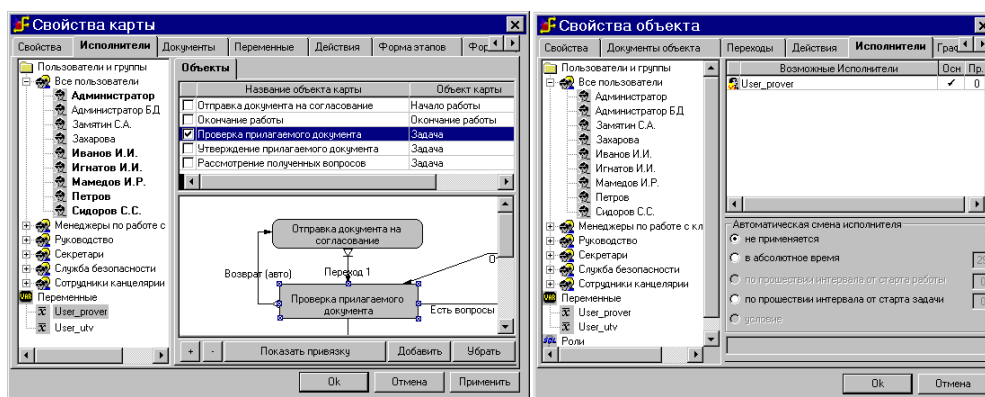


Рисунок 256 Задание исполнителей с использованием переменных различными способами – через свойства карты и свойства объекта этапа

В рассмотренном случае присваивать значения переменным в форме задачи будет инициатор работы.

Записать текущего исполнителя задачи в соответствующую переменную шаблона (либо в системную переменную) можно путем выполнения действия над переменными (см. п. 23.16.1).

23.15.2 Дополнительные (пользовательские) кнопки в формах задач Workflow

При настройке форм задач могут создаваться пользовательские кнопки, по нажатию на которые будут выполняться соответствующие действия (см. п. 23.16). Кнопки в Workflow имеют следующие особенности:

- Может использоваться обычная кнопка – аналогичная кнопкам в атрибутивных формах Lotsia PDM (такие кнопки рекомендуется применять). Кроме того, действия могут выполняться по щелчку по полю переменной. В

качестве такой переменной может быть использована пользовательская переменная Workflow. Объект переменной в форме может быть настроен таким образом, что щелчок по нему будет имитировать нажатие кнопки;

- Выполнение действий в Workflow имеет ряд особенностей – в том числе, на одно событие может приходиться несколько действий, на выполнение действий могут накладываться условия, действия могут выполняться однократно или многократно – циклически – до тех пор, пока выполняется условие;

Несколько замечаний, относящихся к выполнению действий по щелчку по полю переменной:

- Для того чтобы пользовательская переменная в форме внешне выглядела как кнопка и «нажималась», необходимо выполнение соответствующих настроек: чтобы кнопка имела рамку «3D-выпуклая» (дважды щелкнуть по объекту формы → свойства объекта формы → вкладка «Графика», поле «Рамка»), а также установить для данной переменной порядок обхода равный 0 (контекстное меню левой части окна настройки формы, пункт «Задать порядок обхода»). На событие «Щелчок по полю переменной» (см. п. 23.16), назначаются действия;
- Возможно, путем использования соответствующего графического оформления, сделать поле переменной похожим, например, на гиперссылку и т.п., при этом по щелчку в данном поле нажатие кнопки не имитируется;
- Отдельного свойства, предназначенного для отображения текста (названия кнопки) у переменной, нет. Однако, у переменной есть значение, которое обычно и используется в качестве текста (названия кнопки). Отметим, что в отличие от свойства «Текст» кнопки, значение в поле переменной может отображаться в несколько строк.

23.16 Действия в шаблонах работ. Общий порядок использования

В работах Lotsia Workflow применяются действия следующих видов:

- над объектами (см. п. 16);
- над переменными (они же «Действия Workflow»)
- Скрипты (в текущей версии программы поддерживаются языки VBScript и Jscript; написание и использование скриптов в данном руководстве не рассматривается).

В работах Lotsia Workflow действия могут выполняться:

- как клиентские;
- как серверные.

Клиентское действие выполняется при наступлении соответствующего события, от имени текущего исполнителя этапа работы.

Серверные действия (другое название – «Отложенные задачи») выполняются на сервере автоматических этапов (см. п. 23.12), от имени того пользователя, под которым запущен сервер автоматических этапов. Смысл использования серверных действий – обеспечить выполнение таких операций, на которые у исполнителя этапа нет достаточных прав. Например, при согласовании документа, согласующие, как правило, не имеют прав на изменение согласуемого документа. Однако, сам факт согласования – фамилию согласующего, возможно, его должность и дату согласования – при положительном

результате согласования требуется записать в соответствующие атрибуты объекта документа.

Данная задача может решаться следующим образом: исполнитель этапа определяет в форме задачи направление дальнейшего прохождения процесса (выбирает выполняемый переход: согласовано/не согласовано). При выборе «Согласовано», после выполнения задачи, имя текущего пользователя и другая необходимая информация записывается в переменные Workflow (используются действия Workflow). Далее выполняется переход к скрытой (см. п. 23.16.4) автоматической (то есть, исходящий переход из такой задачи – автоматический) задаче. Данная задача выполняется, автоматически выполняются и действия, присоединенные к данной задаче. При выполнении действий над объектами происходит изменение соответствующей атрибутивной информации и т.д. Выполнение действий происходит от имени пользователя, от имени которого запущен сервер автоматических этапов. Указанный пользователь должен иметь достаточно прав для выполнения указанных действий – обычно, для гарантированного наличия прав и приоритета, в качестве такого пользователя указывается администратор.

Для того чтобы не создавать подобные «лишние» этапы работ, действия в Workflow можно делать серверными, по функциональности и порядку выполнения это аналогично использованию рассмотренных выше скрытых автоматических задач. Выполнение серверных действий происходит только по событиям «После выполнения задачи» или «Перед выполнением перехода», то есть, когда работа пользователя с этапом работы завершена. Все действия, выполняемые после серверного действия, не зависимо от того, серверные они или клиентские, выполняются на сервере автоматических этапов, то есть, как серверные.

Действия над объектами при выполнении из процессов Workflow (см. п. 23.16.5) реализуют функциональность, которая в них была заложена при создании. Например: выполняют создание объектов, связей, атрибутов, открытие проектов, импорт документов и т. д. В качестве исходных значений могут быть использованы переменные Workflow, а результат выполнения действия может быть передан в Workflow – таким образом, использование действий над объектами возможно и для присвоения значений переменным Workflow.

Действия над переменными выполняют присвоение значения переменным шаблона и/или системным переменным. При присвоении значений используются выражения, куда могут входить переменные, константы и различные функции.

Результатом выполнения любой функции является возвращаемое данной функцией значение. Однако, отдельные функции, помимо возврата значений, осуществляют дополнительные операции, ради которых они и запускаются. Например, может быть изменен внешний вид формы, в том числе осуществляться управление отображением стандартных кнопок и вкладок окна задачи, может быть скрыта задача процесса (см. п.23.16.4), внедрен форму SQL-запрос и т. д.

В этом случае переменная Workflow, которой в ходе выполнения подобного действия присваивается значение, играет только вспомогательную роль: обеспечение выполнения выражения, содержащего соответствующие функции.

Действие в Workflow присоединяются к этапам, либо к переходам работы.

Действия в Workflow выполняются по различным событиям, а именно:

- перед открытием окна задачи;
- после открытия окна задачи;
- после нажатия кнопки формы;

- щелчок по полю переменной;
- после изменения значения переменной;
- после открытия окна формы массива;
- перед закрытием окна формы массива;
- после перехода к другой строке формы;
- после вставки строки в форму;
- перед закрытием окна задачи;
- после закрытия окна задачи;
- после выполнения задачи.

События «Перед закрытием окна задачи» и «После закрытия окна задачи» происходят только при закрытии окна без выполнения перехода (по кнопке «Отмена»).

Для событий, связанных с переменными или формами указывается и соответствующая переменная (или форма).

Для выполнения действия, присоединенного к переходу, указывается событие – «Перед выполнением перехода».

На каждое событие можно назначить выполнение нескольких действий в различной последовательности. Результат выполнения предыдущего действия может использоваться в следующем выполняемом действии. Особенности:

- серверные действия выполняются только «после выполнения задачи» или «перед выполнением перехода»;
- клиентские действия, расположенные в списке действий после серверных выполняются на сервере автоматических этапов, то есть, как серверные. Такие действия фактически являются серверными

В шаблонах работ также имеются встроенные (системные) действия: «Подготовка переходов» и «Ветвление переходов».

Действие «Подготовка переходов» используется только для события «После выполнения задачи». Указанное действие добавляется в процесс автоматически, при добавлении первого действия к событию «После выполнения задачи» (см. рисунок 262). Это действие производит подготовку и проверку (определение) следующего перехода. Результат выполнения действий, выполняемых до данного действия, учитывается при проверке условий выполнения переходов и определении выполняемого перехода. Действия, выполняемые после данного действия, не могут влиять на то, какой из переходов задачи будет выполнен. Следует учитывать, что в случае наличия у задачи автоматических переходов (вне зависимости, выполняются они или нет), действия по событию «После завершения задачи», расположенные перед действием «Подготовка переходов», начинают выполняться на сервере автоматических этапов (то есть, в любом случае они выполняются как серверные). Выполнение таких действий происходит после создания задачи (независимо от её выполнения и выполнения условий перехода). Указанные действия выполняются сервером автоматических этапов постоянно, до тех пор, пока какой-либо переход из данной задачи не будет выполнен. Управлять частотой выполнения таких действий можно путем установки предопределенных переменных «Задача.Интервал_проверки_ATS» или «Задача.Следующая_проверка_ATS». (Подробная информация содержится в контекстной справке и в документации по Lotsia PDM PLUS).

Для того чтобы действия задачи выполнялись только после выполнения задачи, достаточно действия по событию «После выполнения задачи» выполнять после действия «Подготовка переходов».

Действие «Ветвление переходов» используется только для события «Перед выполнением перехода». Данное действие добавляется в процесс автоматически, при добавлении первого действия по событию «Перед выполнением перехода». Это действие производит для перехода определение списка исполнителей и создает нужное число переходов, если для перехода установлен режим разветвления по исполнителям или повтора (см. п. 23.18.3, Рисунок 284).

Аналогично пользовательским переменным шаблона полный список действий отображается на вкладке «Действия» окна свойств карты, там же можно, пользуясь контекстным меню, создавать и удалять действия, изменить привязку действий к элементам шаблона (этапам или переходам, устанавливаются флажки в строке с названием требуемого элемента карты при выделенном в списке действии). На вкладке «Действия» окна свойств элемента карты отображаются действия, привязанные к данному элементу. Здесь также можно создавать новые действия (контекстное меню → Добавить), использовать существующие (контекстное меню → Выбрать...) и т.д. Удаленные на данной вкладке действия остаются в списке действий в свойствах карты и доступны для использования (контекстное меню → Выбрать...).

Одно и то же действие может использоваться в различных элементах карты многократно. В том числе, одно и то же действие может привязываться к одному и тому же элементу карты несколько раз. Одно и то же действие может выполняться по различным событиям (см. рисунок 270).

На выполнение действия может накладываться условие. Одно и то же действие не может иметь различных условий выполнения, требуется использовать различные действия.

Действия можно копировать. Для копирования действия используется кнопка «Копировать» в окне свойств действия.

Действия могут выполняться циклически, до тех пор, пока выполняется условие.

Рассмотрим создание различных действий Workflow на конкретных примерах

23.16.1 Запоминание текущего исполнителя задачи. Использование действий для назначения исполнителей

Возможных исполнителей у начального этапа работы может быть много. Однако при выполнении конкретного процесса бывает важно, чтобы определенные этапы выполнялись именно тем пользователем, который данный процесс фактически запустил. Например, для учебного шаблона (Рисунок 231), этап рассмотрения полученных вопросов должен выполнять тот, кто запустил данный процесс. Создайте действие над переменными, которое будет «запоминать» текущего исполнителя начала работы в соответствующую переменную шаблона:

Откройте вкладку «Действия» (Рисунок 257) окна свойств первого этапа созданного шаблона работы (Рисунок 231).

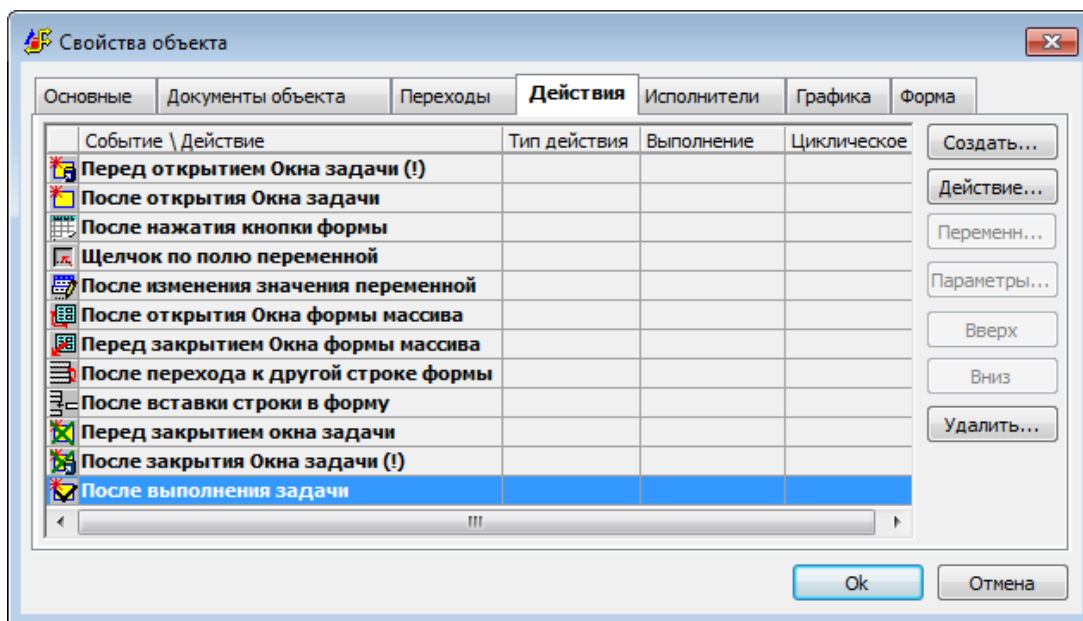


Рисунок 257. Вкладка «Действия» окна свойств этапа работы. Действия не добавлены

Выделите в списке событий «После выполнения задачи», нажмите «Создать...». Откроется окно «Свойства действия». Для создания переменной для исполнителя задачи, в контекстном меню данного окна (Рисунок 258) выберите «Создать переменную».

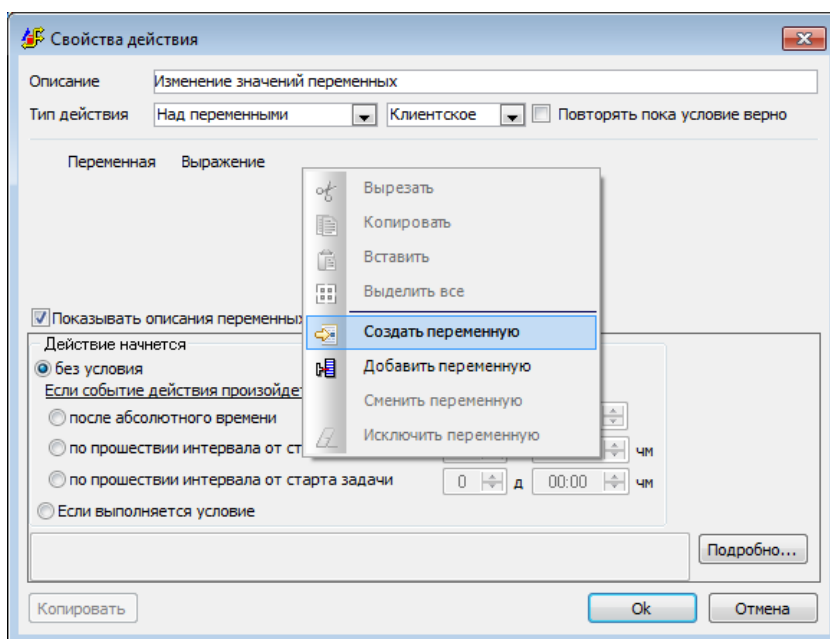


Рисунок 258 Контекстное меню окна «Свойства действия»

Создайте переменную: имя и описание переменной – «Initsiatr», тип – «Пользователь», тип значения – «Одиночное» (флажок «Массив» выключен), локальная. Созданная переменная будет добавлена в создаваемое действие – см. рисунок 259

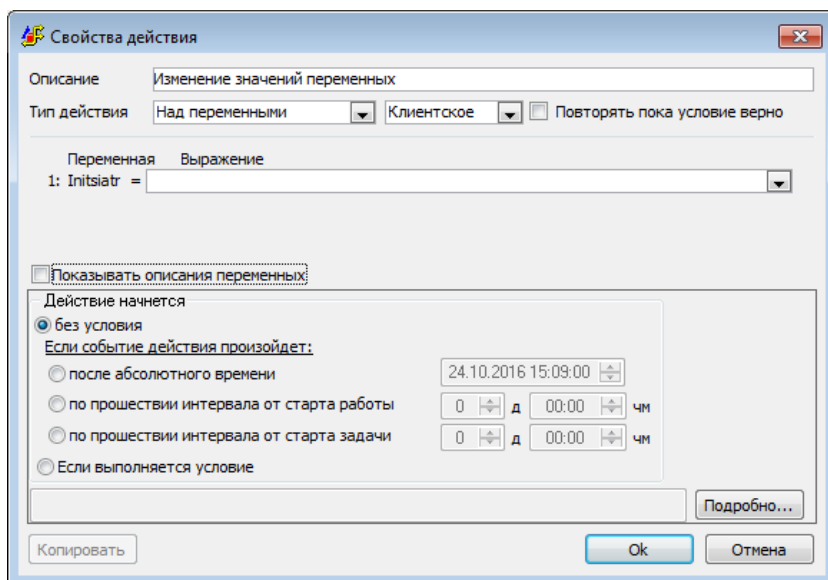


Рисунок 259 Добавлена переменная

В появившуюся строку переменной необходимо ввести выражение для присвоения значения переменной. Для открытия окна редактора выражений щелкните левой клавишей мыши в поле «Значение» переменной. В открывшемся окне редактора выражений введите выражение для получения идентификатора пользователя. Выражение может быть различным, например, можно использовать функцию `f_GetCurrentUserID()`, расположенную в группе Lotsia. Можно также использовать вкладку «Атрибуты» редактора выражений (Рисунок 260).

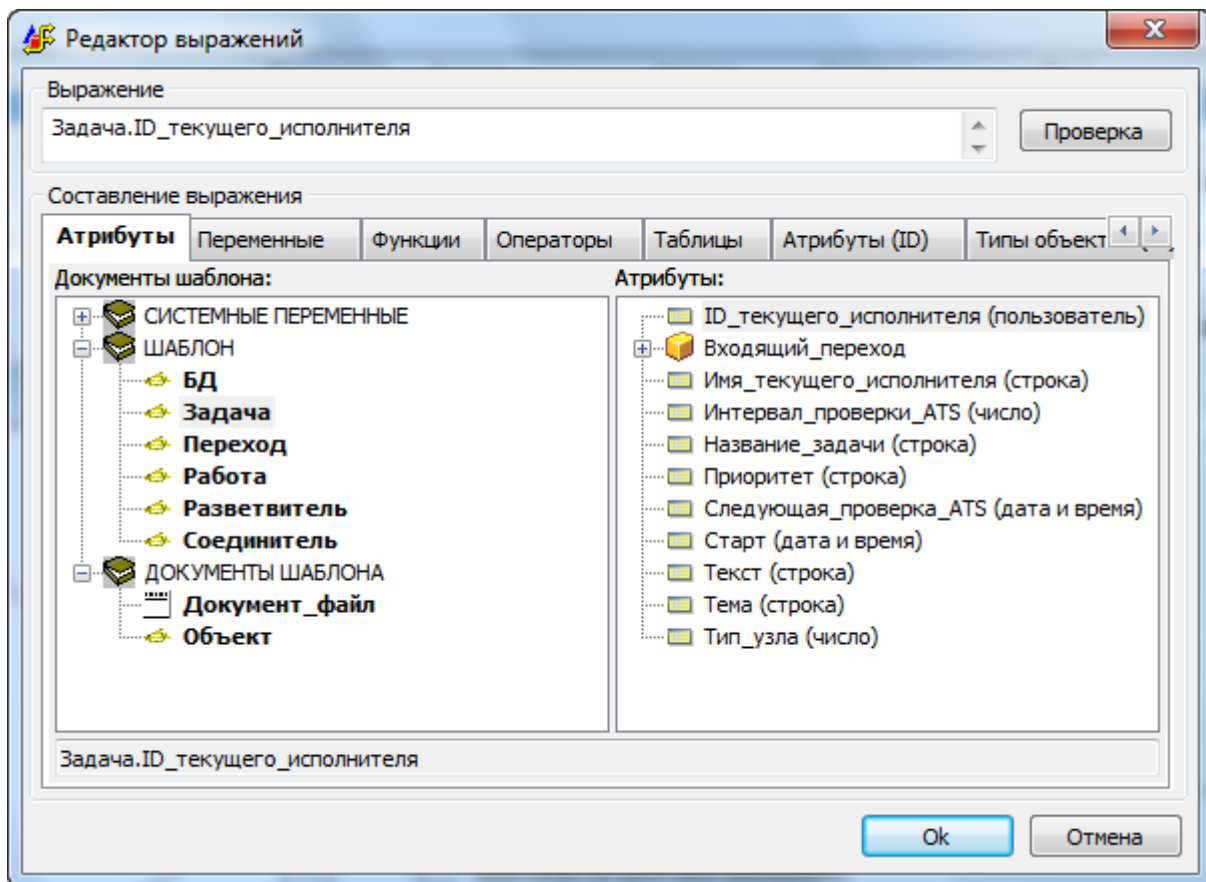


Рисунок 260 Задание выражения для получения идентификатора текущего пользователя (один из возможных вариантов)

После нажатия «Ok» в окне редактора выражений, окно редактора закроется, составленное выражение отобразится в поле «Значение» данной переменной. В поле «Описание» в окне свойств действия введите описание создаваемого действия над переменными (Рисунок 261).

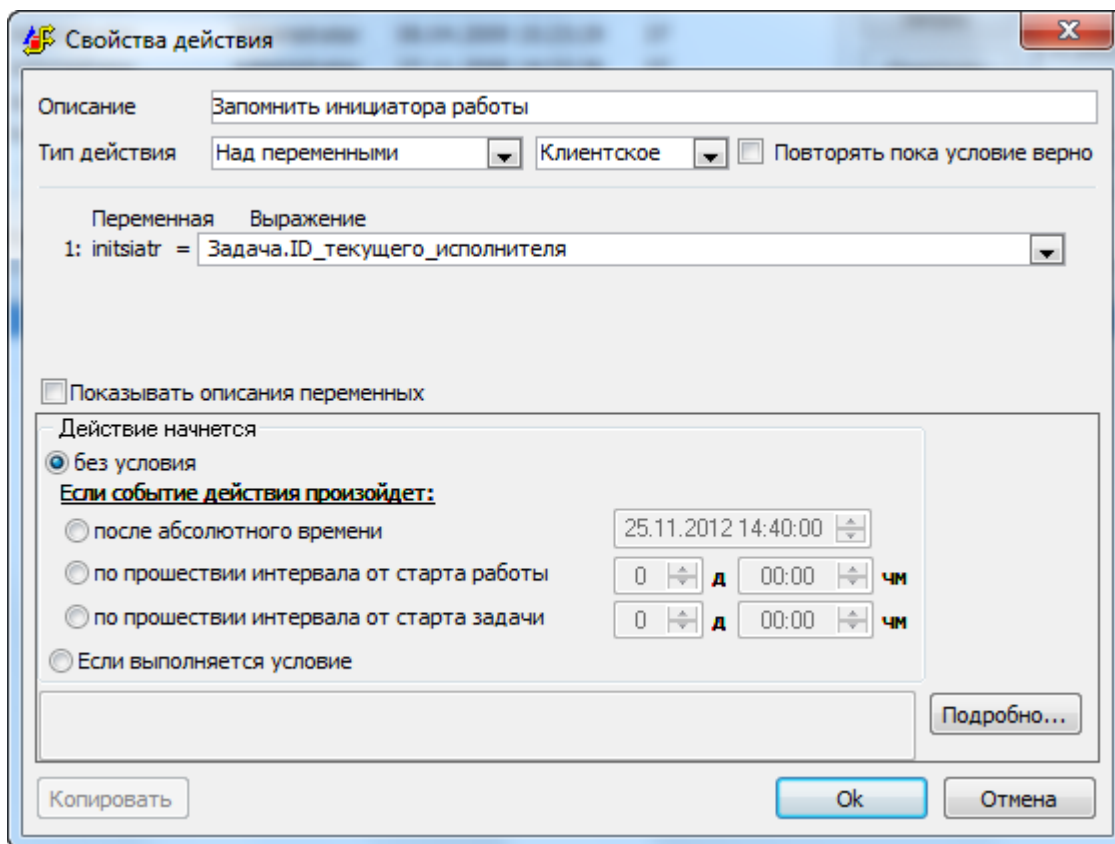


Рисунок 261 Введено описание действия над переменными

Нажмите «Ok» для сохранения действия. На вкладке «Действия» окна свойств объекта, под выбранным событием («После выполнения задачи») появится строчка созданного действия. (Рисунок 262).

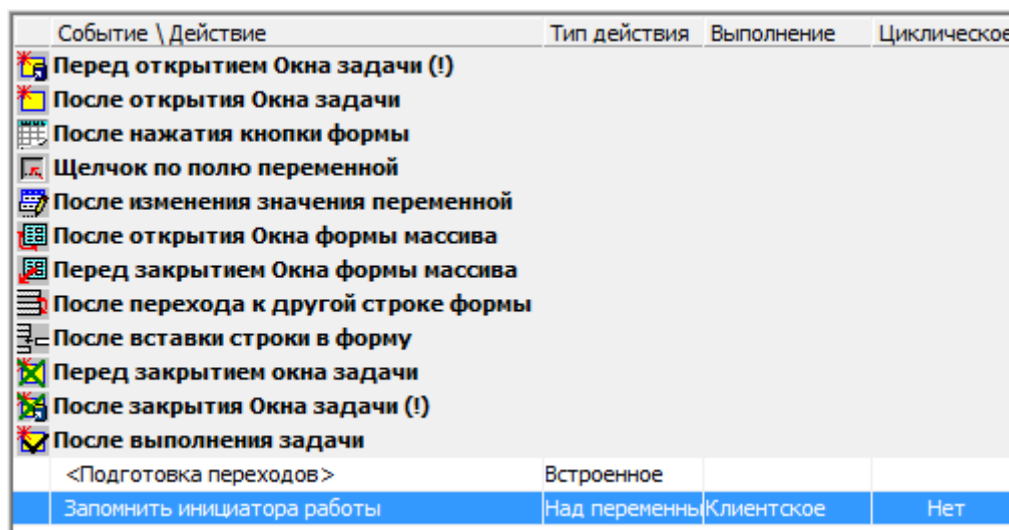


Рисунок 262 Действие добавлено

При добавлении первого действия к данному событию также автоматически добавляется встроенное действие «<Подготовка переходов>».

Сохраните объект карты, после чего сохраните шаблон для записи проведенных изменений в БД. Переменную `Initiatr` укажите в качестве исполнителя задачи «Рассмотрение полученных вопросов» (см. рисунок 256) учебного шаблона (Рисунок 231).

Обычно список исполнителей окончательно определяется после выполнения последнего действия на переходе, кроме двух случаев:

1. для данного перехода включена форма перехода с отображением исполнителей следующего этапа. В этом случае список исполнителей определяется сразу после закрытия формы;
2. для ветки установлено разветвление по исполнителям (см. п. 23.18.3, Рисунок 284) и не включена форма перехода и отображение списка исполнителей. В этом случае список исполнителей определяется встроенным действием «Ветвление переходов».

23.16.2 Формирование сообщения задачи

Использовать стандартные средства формы перехода для ввода сообщения не всегда удобно: в большинстве случаев формы задачи пользователю вполне достаточно и открывать еще и форму перехода нежелательно. Кроме того, сообщение может быть адресовано исполнителю не следующей, а более поздней задачи процесса, то есть, исполнитель следующей задачи данного сообщения получать не должен. Чтобы обеспечить формирование подобного сообщения, можно создать текстовую переменную в форме задачи Workflow и вводить туда сообщение (до 2000 символов).

Создайте в форме первого этапа работы строковую переменную «`Mes_1`» (в контекстном меню правой части окна редактирования формы выберите «Создать переменную»...). Имя и описание переменной – «`Mes_1`», тип – «Строка», тип значения – «Одиночное», вид – «Локальная». Значение по умолчанию – «'» (пустое). Задайте соответствующее графическое оформление формы (Рисунок 263).

The screenshot shows a window titled 'Настройка формы' (Form Configuration). On the left, the 'Задача:' (Task) section contains the text 'Отправка документа на согласование' (Document submission for approval). Below it, a blue instruction reads 'Убедитесь в наличии и комплектности согласуемого документа' (Ensure the document being approved is present and complete). A box labeled 'Укажите согласующих:' (Specify approvers:) contains two dropdown menus: 'Проверяющий' (Reviewer) and 'Утверждающий' (Approver), both currently set to 'Administrator'. Below this is a text area for 'Ваше сообщение проверяющему:' (Your message to the reviewer:). At the bottom, a blue instruction says 'Нажмите "Ok" для отправки документа на согласование' (Click 'Ok' to submit the document for approval). On the right, a table lists variables:

№	Переменная	Описание	ID	Тип данных	Тип значения
1	job	Работа	1	Строка	Одиночное
2	subj	Задача	2	Строка	Одиночное
3	select_action	Переход	3	Строка	Одиночное
4	user_utv	User_utv	4	Пользователь	Одиночное
5	mes_1	Mes_1	5	Строка	Одиночное
6	user_prover	User_prover	6	Пользователь	Одиночное

At the bottom right of the window are 'Ok' and 'Отмена' (Cancel) buttons.

Рисунок 263 Форма первого этапа после добавления строковой переменной для ввода сообщения. Пример

Конечно, данную строковую переменную можно в дальнейшем отобразить в форме соответствующей задачи получателя сообщения, однако, обычно удобнее отображать сообщение именно на вкладке «Сообщение».

Для формирования сообщения задачи можно использовать действие над переменными, прикрепленное к входящему переходу (либо к задаче, предшествующей задаче получателя сообщения).

Для создания действия, на вкладке «Действия» соответствующего объекта карты (для определенности, пусть в нашем случае это будет «Переход 1» к задаче «Проверка прилагаемого документа», Рисунок 231) нажмите «Создать...».

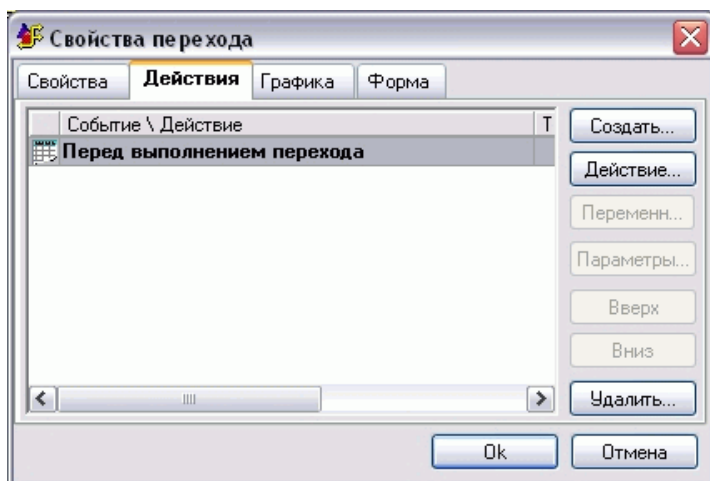


Рисунок 264 Вкладка «Действия» окна свойств перехода

В контекстном меню в открывшемся окне «Свойства действия» выберите «Добавить переменную». В открывшемся списке переменных выберите предопределенную переменную «Следующая задача. Текст», нажмите «Ok», поле переменной отобразится в окне свойств действия (Рисунок 265).

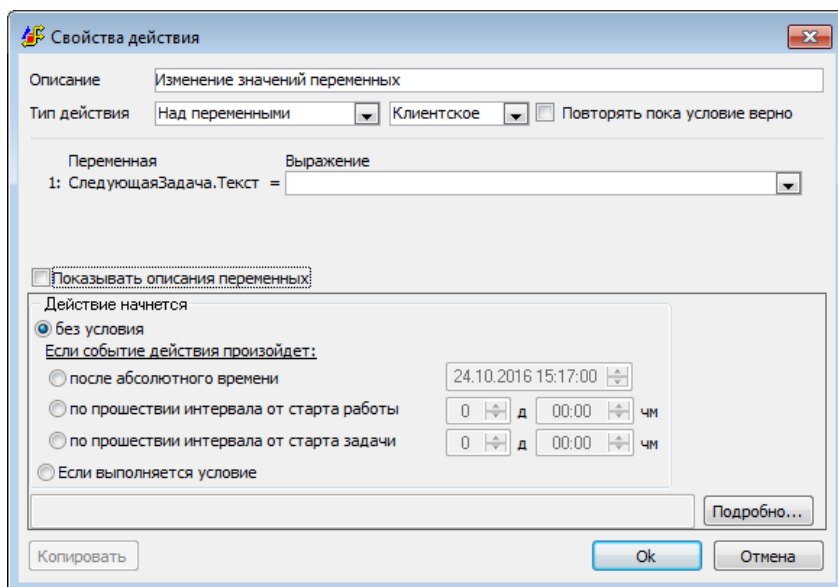


Рисунок 265 В действие добавлена переменная «Следующая задача. Текст»

В поле «Значение» переменной «Следующая задача. Текст» укажите переменную «Mes_1»: щелкните по полю, в открывшемся окне редактора выражений перейдите на вкладку «Переменные», выберите в списке требуемую переменную, нажмите «Ok» для закрытия редактора выражений. Введите описание понятное действия, например: «Формирование сообщения задачи». Нажмите «Ok» для сохранения действия, «Ok» для сохранения свойств перехода, после чего сохраните шаблон для записи проведенных изменений в БД. Отключите показ формы данного перехода.

Для проверки работы действия запустите работу по шаблону, назначьте исполнителей этапов, введите текст сообщения в созданное поле в форме первого этапа работы, убедитесь, что сообщение поступило исполнителю второго этапа.

Для того, чтобы переменная «Mes_1» могла использоваться на других этапах работы, после присвоения значения переменной «Следующая задача. Текст» следует очистить переменную. Откройте окно свойств действия, добавьте в действие переменную «Mes_1» после переменной «Следующая задача. Текст». Присвойте переменной «Mes_1» пустое значение – две одинарные кавычки, – Рисунок 266.

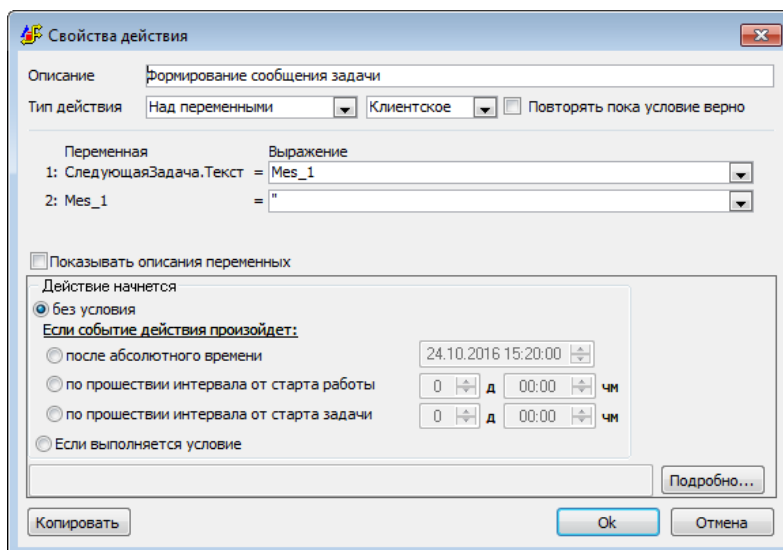


Рисунок 266 В действие добавлена еще одна переменная

При выполнении перехода «Проверено» (Рисунок 231), сообщение утверждающему в нашем процессе не формируется. Однако если на предыдущих этапах участники работы обменивались сообщениями, вкладка «Сообщение» сообщения задачи «Утверждение прилагаемого документа» будет, скорее всего, непустой. Чтобы удалить с данной вкладки весь текст, создайте аналогичное действие по очистке переменной «Следующая задача. Текст» на переходе «Проверено».

23.16.3 Управление стандартными кнопками и вкладками в форме задачи

С помощью действий над переменными возможно управление вкладками и кнопками в формах задач. Кнопки в форме задачи могут быть отключены, либо сделаны неактивными, а также сделаны видимыми либо активными вновь. Аналогично можно отключить не используемые в данной задаче вкладки, например – вкладку «Документы». Также с помощью действий можно осуществлять нажатие кнопок.

Отметим, что:

- управление вкладками с помощью действий возможно только в случае, если вкладки не отключены в настройках данного этапа работы (см. п. 23.3);
- неактивную кнопку нельзя использовать. Видимая, либо невидимая, но активная кнопка может быть «нажата» путем выполнения действия над переменной (используется функция `f_ButtonPost`).

В качестве примера, будем управлять активностью кнопок в форме задачи. Часто используется принудительное отключение стандартных кнопок в форме задачи при открытии задачи, чтобы не допустить возможности их случайного нажатия пользователем, например – нажатия кнопки «Ok» до заполнения обязательных полей формы, Включение указанных кнопок в этом случае осуществляется также принудительно, например, после подтверждения пользователем выполнения данной задачи.

В форме первого этапа работы (Рисунок 231) добавьте переменную FL_1 (Рисунок 267).

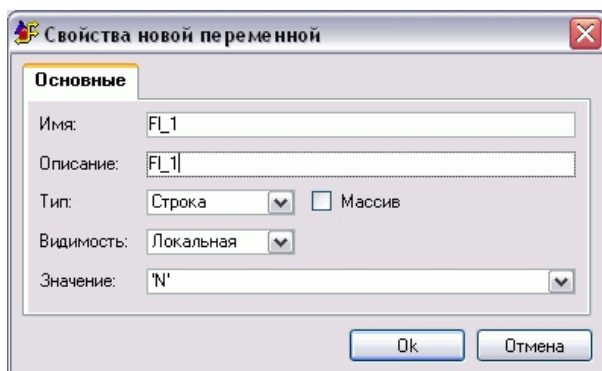


Рисунок 267 Свойства переменной FL_1

Данная переменная будет использоваться как флажок в форме задачи, для подтверждения пользователем выполнения задачи (Рисунок 268).

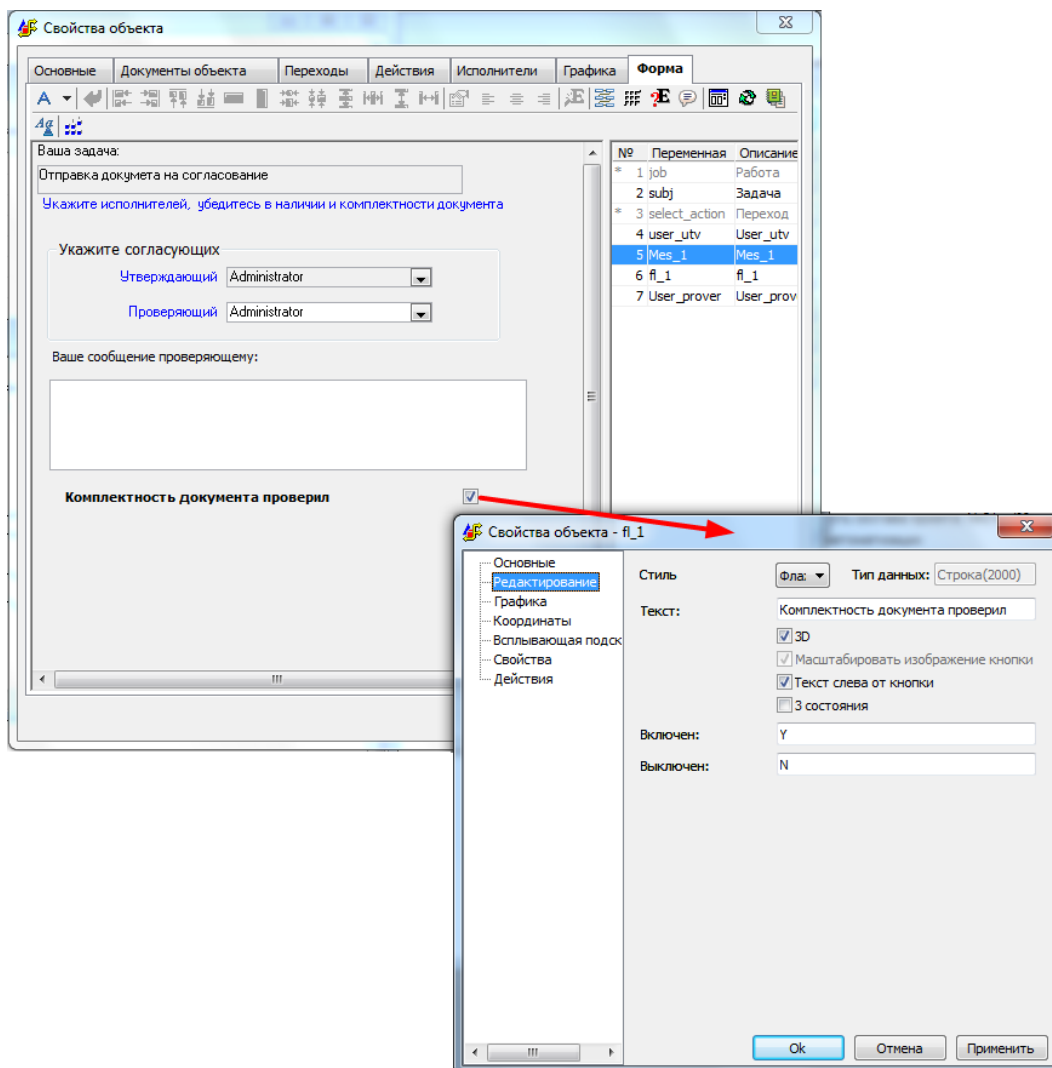


Рисунок 268 Добавление в форму флажка подтверждения выполнения задачи

Действие по управлению кнопками будет выполняться по двум событиям – после открытия окна задачи и после изменения значения переменной FL_1. Добавьте действие над переменными «Управление кнопками формы» по событию «После открытия окна задачи», для этого – на вкладке «Действия» окна свойств данной задачи, в списке событий выделите «После открытия Окна задачи», нажмите «Создать...».

Создайте в действии строковую переменную ls_tmp (имя и описание могут быть произвольные, тип – «Строка», тип значения – «Одиночное», вид и значение по умолчанию могут быть любыми).

Для управления кнопками используется функция f_ModifyWin (), расположенная в группе «Lotsia» редактора выражений. Сформируйте для поля «Значение» переменной ls_tmp следующее выражение:

If(Fl_1='Y' , f_ModifyWin (this , 'cb_All' , 'Enabled=1') , f_ModifyWin (this , 'cb_All' , 'Enabled=0'))

Смысл данного выражения следующий: в зависимости от текущего значения переменной Fl_1 (если Fl_1='Y' – флажок включен, см. рисунок 268) будет выполняться функция f_ModifyWin () с различным значением последнего аргумента: 'Enabled=0' – погасить (сделать неактивными), 'Enabled=1' включить (сделать активными) кнопки, указанные во втором аргументе (значение 'cb_All' означает, что управлять надо всеми кнопками, кроме кнопки «Отмена») (Рисунок 269).

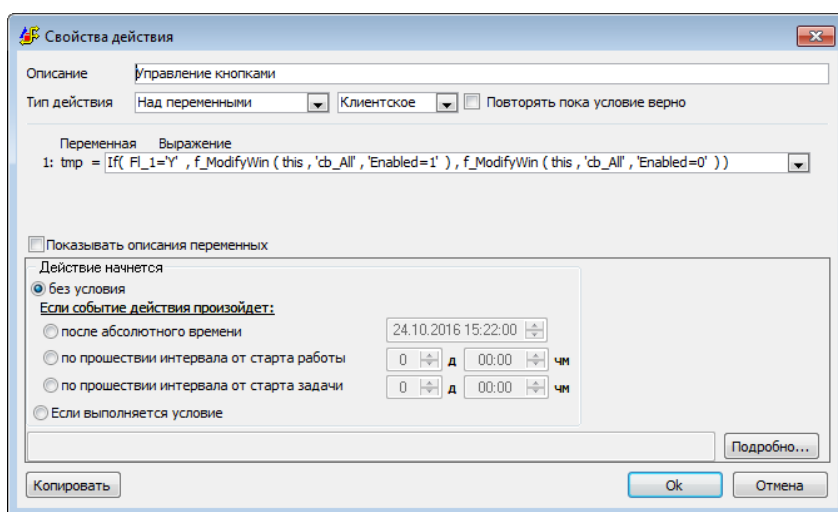


Рисунок 269 Свойства действия «Управление кнопками формы»

Если требуется скрыть predetermined кнопки, вместо 'Enable...' следует использовать инструкцию 'Visible=0' (невидима), 'Visible=1' (видима). Та же функция используется и для удаления неиспользуемых вкладок (см. описание синтаксиса данной функции в нижней части окна редактора выражений).

Ранее добавленное к этапу работы действие, методом drag-and-drop можно перенести в другое событие, а при нажатой клавише <Ctrl> происходит не перенос, а добавление действия в соответствующее событие (Рисунок 270).

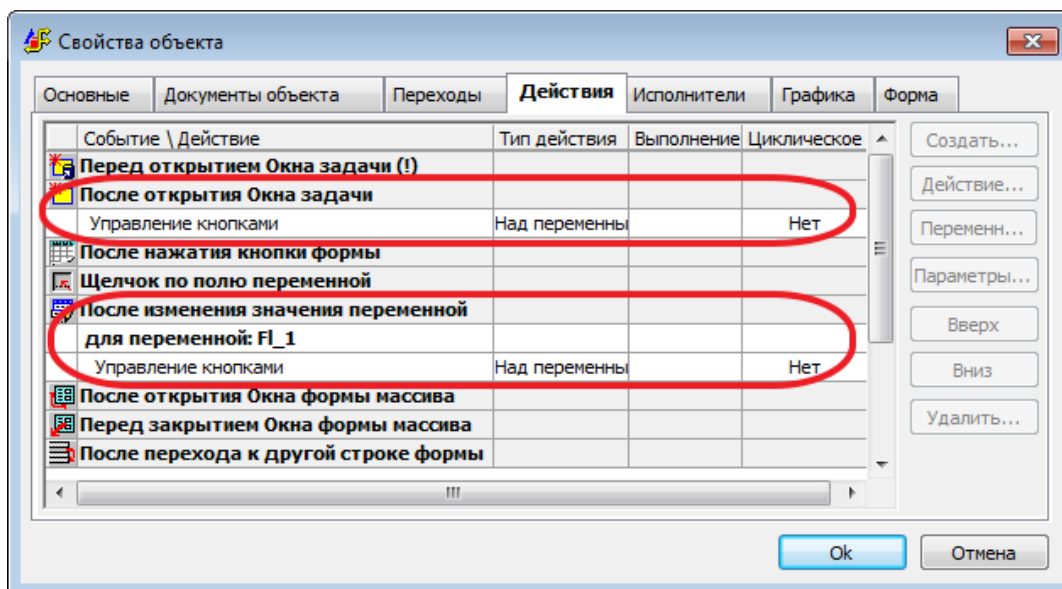


Рисунок 270 События выполнения действия

Данное действие будет отключать все стандартные кнопки при открытии окна задачи, включать кнопки при включении флажка и вновь гасить кнопки при выключении данного флажка.

Отметим, что:

- по изменению того же самого флажка удобно управлять отображением объектов формы (через задание выражений для свойств соответствующих объектов) – например, текст «Нажмите «Ok» для отправки на согласование» отображать при включенном флажке, а редактируемые поля отображать только при выключенном флажке и т.д.
- кнопка «Отказаться» может быть сделана неактивной в свойствах задачи (см. п. 23.11)
- отключение, перемещение, изменение внешнего вида и т.п. пользовательских кнопок осуществляется управлением их свойствами, аналогично полям формы.

23.16.4 Скрытие задач

При необходимости, задачи можно делать скрытыми. Скрытие задачи означает, что после выполнения входящего перехода, задача автоматически становится прочтенной и помещается в корзину исполнителя. При появлении скрытой задачи нотификатор не активизируется. Очевидно, что для дальнейшего выполнения процесса, переходы из скрытой задачи должны выполняться автоматически.

Для того чтобы скрыть следующую задачу, на необходимо выполнить специальное действие над переменной.

Для создания такого действия откройте окно свойств перехода или задачи, на вкладке «Действия», в контекстном меню выберите «Добавить».

В контекстном меню в открывшемся окне «Свойства действия» выберите «Добавить переменную» и в открывшемся списке переменных выберите предопределенную переменную «СледующаяЗадача.Скрытая» (Рисунок 271).

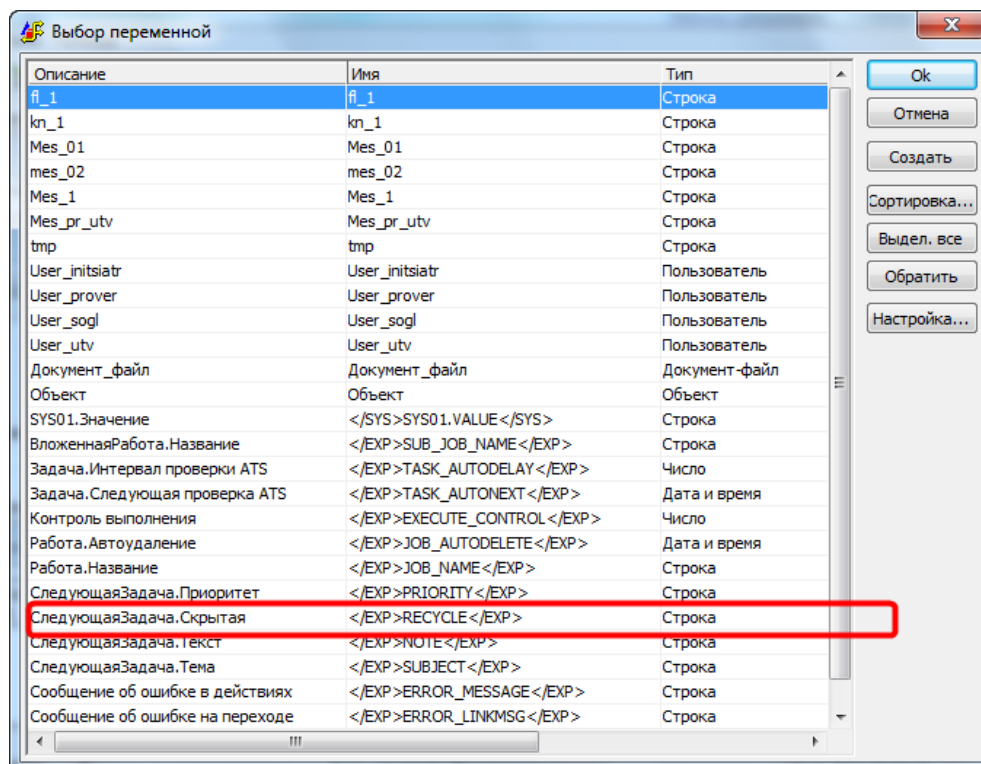


Рисунок 271 Выбор переменной «СледующаяЗадача.Скрытая»

Данной переменной следует присвоить значение 'Y', ввести описание действия (Рисунок 272).

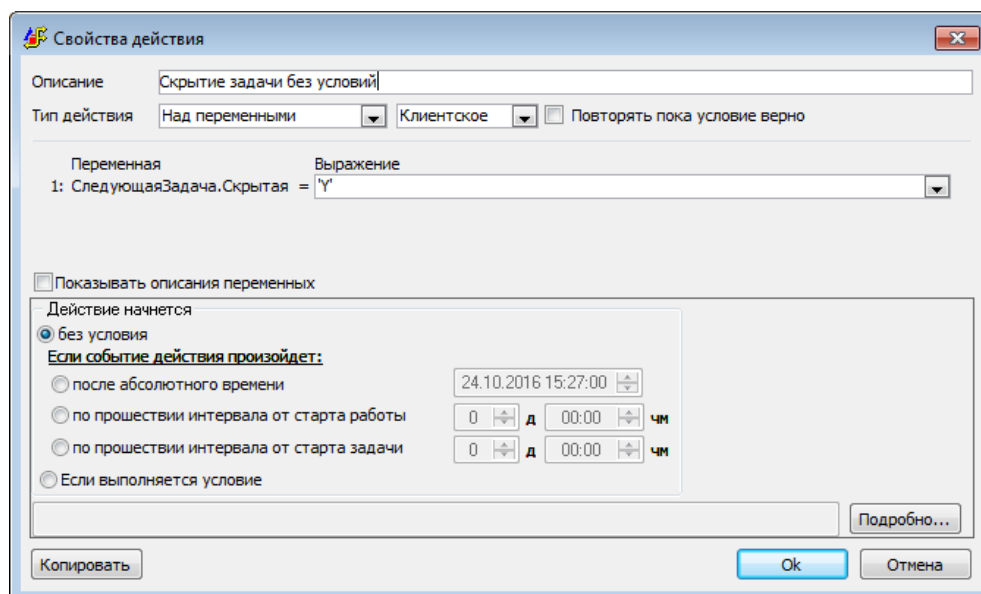


Рисунок 272 Создание действия по скрытию задачи

При выполнении данного действия следующая задача будет скрытой.

Отметим, что данное действие должно выполняться только по событиям «После выполнения задачи» или «Перед выполнением перехода» (соответственно – для задачи или перехода).

23.16.5 Использование действий над объектами в Workflow

В Workflow могут использоваться и действия над объектами (см. п. 16). В качестве аргументов действия могут использоваться переменные Workflow (см. п. 23.15) или

документы работы, возврат значений переменных из действий над объектами может осуществляться в переменные шаблона Workflow, либо в документы шаблона.

Далее рассмотрим пример использования действия над объектами в Workflow.

23.16.5.1 Регистрация согласуемого документа

Для учебного процесса «Согласование документа» (Рисунок 231) можно добавить возможность присвоения регистрационного номера документу непосредственно из формы начала работы. Используем ранее созданное или создадим действие над объектом «Присвоение регистрационного номера» (см. п. 17.11.5).

В форме первого этапа создайте кнопку, по нажатию на которую будет выполняться присвоение регистрационного номера объекту. В качестве такого объекта используйте объект, являющийся документом шаблона – «Объект» (см. п. 23.4). Кроме того, добавьте в форму первого этапа строковую переменную str_01, в поле которой будет отображаться значение присвоенного регистрационного номера (Рисунок 273).

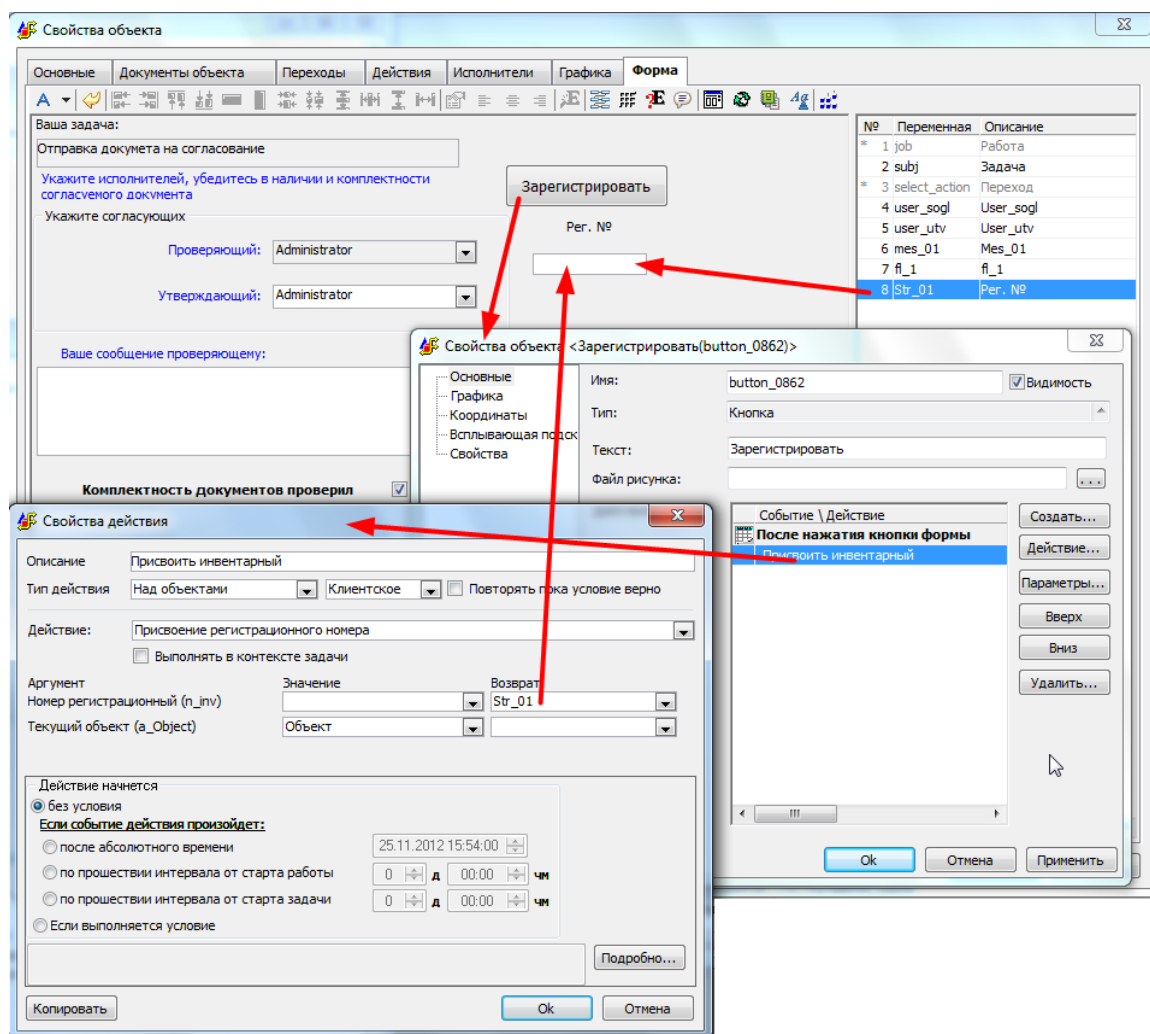


Рисунок 273 Добавлены кнопка, действие по ней, и поле для отображения номера

Откройте вкладку «Форма» в окне свойств объекта начала работы.

В свойствах кнопки, в пункте «Основные», нажмите «Создать».

В открывшемся окне «Свойства действия» выберите тип действия – «Над объектами», «Клиентское». В поле «Действие», выбором из открывшегося окна «Выбор действия» укажите требуемое действие – «Присвоение регистрационного номера».

В контекстном меню окна свойств действия выберите «Добавить переменную».

В открывшемся окне отображается список всех переменных действия «Присвоение регистрационного номера». Выберите из данного списка переменные с описаниями «Номер регистрационный» и «Текущий объект». Нажмите «Ok» для добавления в окно свойств действия соответствующих данным переменным аргументов.

В поле «Значение» аргумента «Текущий объект» укажите «Объект».

В поле «Возврат» аргумента «Номер регистрационный» укажите Вашу переменную – «str_01» (Рисунок 273).

Сохраните действие, форму и шаблон.

Для проверки работы действия:

- создайте объект документа;
- запустите по нему процесс «Согласование документа»
- в форме первого этапа нажмите кнопку, результат – выполнение регистрации документа и возврат присвоенного номера в форму задачи (Рисунок 274).

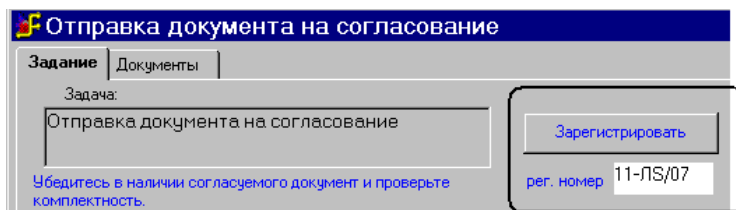


Рисунок 274 Результат выполнения действия

23.17 Использование действий при работе с документами шаблона

Для добавления/удаления документов шаблона можно использовать как действия над объектами, так и действия над переменными (последние использовать предпочтительнее, ввиду того что выполняются быстрее).

Работа действий над переменными с документами шаблона осуществляется также, как и с обычными переменными. Присваивать значения можно любым документам, независимо от их привязок к задачам (см. п. 23.4)

Присваиваемое значение должно содержать идентификатор (ID) документа соответствующего типа (объекта или версии документа) в строковом виде. Для удаления документа следует присвоить ему значение в виде пустой строки. Документам типа «Файл» в действиях над переменными можно присвоить только пустое значение. Документ, таким образом, обнуляется. После обнуления документа он не будет передаваться дальше по этапам работы. Если документ шаблона является массивом, то для его обнуления следует присвоить ему значение в виде пустого строкового массива.

Индексы (порядковые номера) элементов для документов-массивов определяются в момент загрузки (открытия Окна) задачи, элементы следуют в порядке их сортировки по описанию документов. Индексы элементов в базе данных не сохраняются.

При выполнении действий над объектами по документу шаблона типа «Массив», действие выполняется отдельно по каждому экземпляру документа (аналогично выполнению действий над выделенными в подборке объектами).


Для того чтобы действием над объектом обнулить документ шаблона, надо, чтобы действие вернуло в указанный документ пустое (Null) значение, соответствующее типу документа шаблона.

23.18 Добавление прочих элементов шаблона

23.18.1 Добавление информационной рассылки

Информационная рассылка – специальный этап работы, предназначенный только для передачи информационных сообщений. Из рассылки не может быть исходящих переходов. На задаче информационной рассылки не могут использоваться действия, переходы к информационной рассылке всегда автоматические, условия выполнения перехода и действия на переходах могут использоваться.

Адресаты информационной рассылки выбираются не из списка пользователей, а из глобальной адресной книги. Могут использоваться и переменные типа «Пользователь» (см. п. 23.15.1).

Добавьте в шаблон «Согласование документа 1» объект информационной рассылки. Для добавления информационной рассылки используется пункт контекстного меню «Объект» → «Информационная рассылка» или кнопка  инструментальной панели.

Рассылку требуется выполнить после выполнения проверки документа (Рисунок 275).

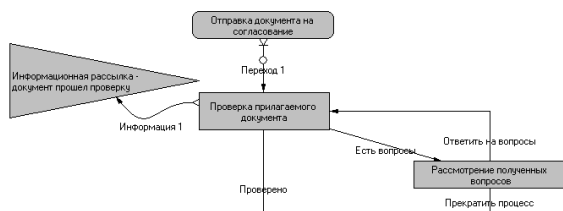


Рисунок 275 Добавлена информационная рассылка

Однако, в случае выполнения перехода «Есть вопросы», выполнение данной рассылки, очевидно, будет преждевременным. Для того чтобы выполнить рассылку только в случае успешного завершения этапа проверки документа, можно действовать двумя различными способами:

- 1) Задать соответствующие условия выполнения перехода «Информация 1».

Создать, на вкладке «Действия» объекта данной задачи действие над переменной, которое будет выполняться по изменению значения переменной `select_action`. Данное действие должно изменять переменную, значение которой используется в условиях перехода к объекту рассылки.

- 2) Добавить после выполнения перехода скрытую (см. 23.16.4) автоматическую задачу (то есть, задачу с исходящим автоматическим переходом), и рассылку выполнять из нее (Рисунок 276). Исполнителем задачи «Скрытая, авто» укажите переменную `User_prover` (см. п. 23.15.1).

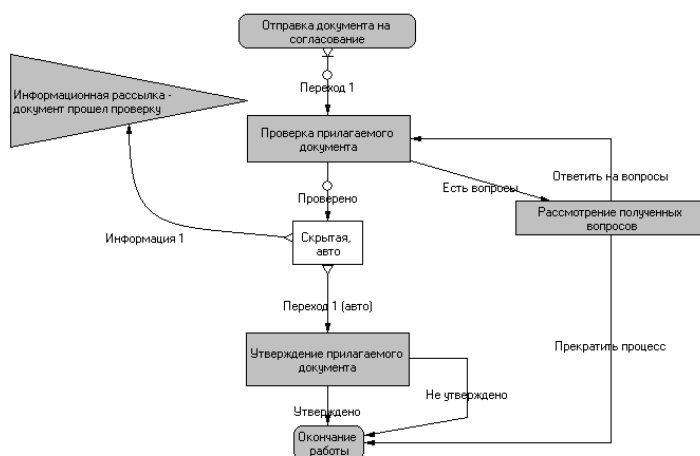


Рисунок 276 Добавлена информационная рассылка с использованием дополнительной скрытой задачи

Данная рассылка будет информировать инициатора работы о выполнении проверки документа. Укажите адресата рассылки (Рисунок 277) – переменная «Initsiatr» (см. п. 23.16.1).

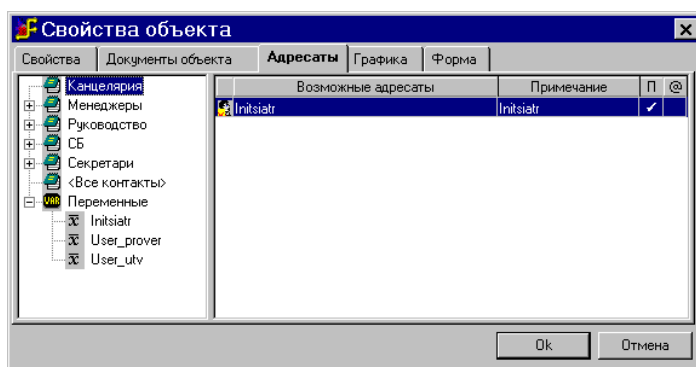


Рисунок 277 Указание адресата рассылки

Настройте (в произвольном виде) форму этапа рассылки. Запустите процесс на выполнение, убедитесь, что информационная рассылка поступает адресату. Для реального процесса, описание задачи «Скрытая, авто» лучше изменить на более информативное (оно будет отображаться у получателя рассылки).

23.18.2 Добавление вложенных и внешних работ


Система позволяет упростить создание сложных процессов путем использования для их построения ранее созданных процессов (шаблонов работ). Ранее созданные процессы могут включаться в разрабатываемый шаблон в качестве внешней работы или как вложенная работа. Добавленному объекту вложенной или внешней работы сопоставляется вложенная карта – имеющийся шаблон работы.

Выполнение внешней работы идет независимо от выполнения родительской. При выполнении вложенной работы родительский процесс не выполняется, после выполнения вложенной работы продолжается выполнение родительского процесса.

Вложенность процессов может быть многократной, глубина вложенности не ограничивается. Во вложенные и внешние работы могут передаваться текущие значения переменных и документы родительской работы.

Значения переменных и документы из вложенной работы могут возвращаться в родительскую работу после выполнения вложенной работы.

23.18.2.1 Добавление внешних работ

Для добавления объекта внешней работы используется пункт контекстного меню «Объект» → «Внешняя работа» или кнопка  инструментальной панели.

В ознакомительных целях создайте копию учебного шаблона «Согласование документа», добавьте в созданную копию шаблона объект внешней работы. Добавьте переход в объект внешней работы из одного из этапов шаблона (Рисунок 278).

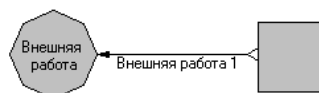


Рисунок 278 Добавлен объект внешней работы (фрагмент шаблона)

Откройте окно свойств перехода к внешней работе. Аналогично переходу к информационной рассылке данный переход может быть только автоматическим. В отличие от перехода к информационной рассылке, на данном переходе могут выполняться действия.

Выполните сопоставление объекту внешней работы имеющегося шаблона – вложенной карты (в ознакомительных целях можно использовать исходный шаблон «Согласование документа», либо другой), для этого:

- откройте окно свойств объекта внешней работы. На вкладке «Свойства» щелкните по полю «Шаблон». В открывшемся окне «Выбрать шаблон» выберите требуемый шаблон для использования в качестве вложенной карты (Рисунок 279);

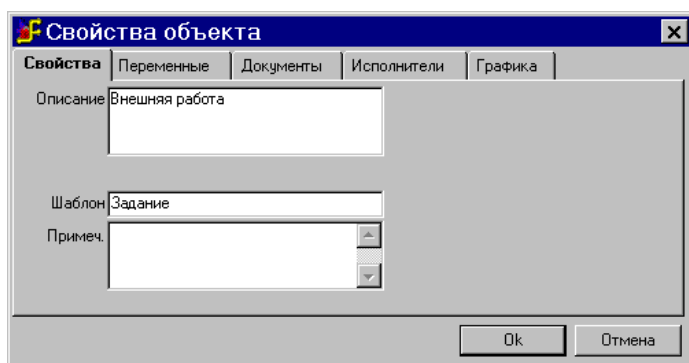


Рисунок 279 Объекту внешней работы сопоставлен шаблон «Задание»

- перейдите на вкладку «Переменные». Проведите сопоставление переменных родительской и вложенной карты, для чего методом «drag-and-drop» передвигайте требуемые строки из списка переменных родительской карты на соответствующие им строки переменных вложенной карты (сопоставляемые переменные должны быть одинакового типа) (Рисунок 280);

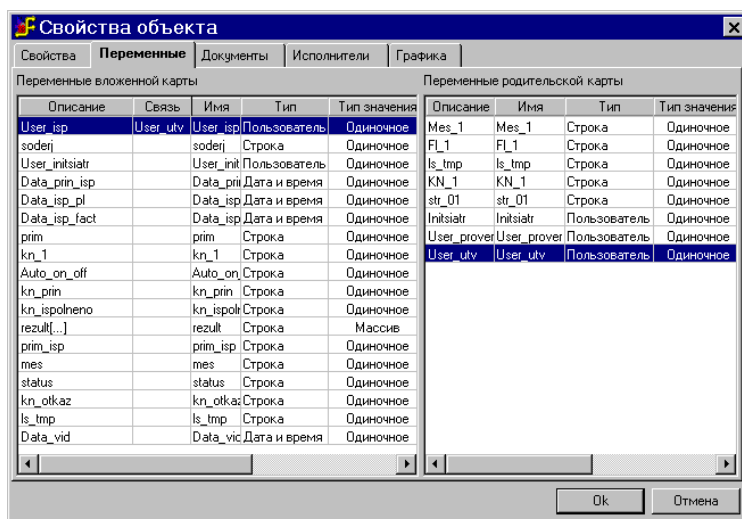


Рисунок 280 Сопоставление переменных родительской и вложенной карты


- перейдите на вкладку «Документы», аналогично сопоставлению переменных выполните сопоставление документов родительской и вложенной карт (сопоставляемые документы должны быть одинакового типа).

Действия к объекту внешней работы не присоединяются, формы для объекта внешней работы не используются – требуемые указанные элементы сопоставляются первому этапу вложенной карты.

Чтобы перейти к редактированию вложенной карты (шаблона), в контекстном меню объекта внешней работы выберите «Вложенный шаблон». Будет открыто окно редактирования соответствующего вложенного шаблона. Для возврата в родительский шаблон, в контекстном меню окна редактирования вложенного шаблона выберите «Родительский шаблон».

Исполнитель (то есть, возможные исполнители, определенные в карте вложенного процесса) первого этапа внешней работы игнорируется. Исполнителем первого этапа внешней работы будет исполнитель этапа внешней работы родительской карты.

23.18.2.2 Добавление вложенных работ

Для добавления объекта вложенной работы используется пункт контекстного меню «Объект» → «Вложенная работа» или кнопка  инструментальной панели.

В ознакомительных целях используйте созданную ранее (или создайте новую) копию учебного шаблона «Согласование документа», добавьте в созданную копию шаблона объект вложенной работы. Добавьте переход в объект вложенной работы из одного из этапов шаблона, а также переход из вложенной работы в один из этапов шаблона (Рисунок 281).

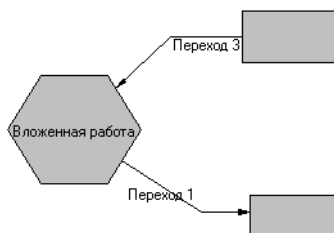


Рисунок 281 Добавлен объект вложенной работы (фрагмент шаблона)

Переходы, связанные с вложенной работой, не отличаются по набору свойств переходов между задачами работы – они могут быть ручными или автоматическими, содержать присоединенные действия и т.д.

Аналогично настройке внешних работ (см. п. 23.18.2.1), объекту вложенной работы ставится в соответствие шаблон работы. Задается соответствие переменных и документов родительской и вложенной карт.

После выполнения вложенного процесса, в родительский процесс из вложенного передаются значения сопоставленных переменных и документы, продолжается выполнение родительского процесса.

Проведите настройку соответствия переменных и документов родительской и вложенной работ.


23.18.2.2.1 *Начало и завершение вложенных работ*

Исполнитель начала вложенной работы игнорируется и заменяется исполнителем этапа вложенной работы родительского процесса (аналогично внешней работе).

Исполнителем этапа окончания вложенной работы остается исполнитель, определенный настройкой вложенного процесса, но форма его этапа заменяется формой объекта вложенной работы родительской карты. Форма последней задачи (окончания) карты вложенной работы игнорируется.

В учебных целях, для закрепления материала, создайте простейший шаблон с этапом вложенной работы (Начало работы → вложенная работа → завершение работы) и простейший шаблон для использования в качестве вложенного процесса (Начало работы → завершение работы). Сопоставьте вложенный процесс соответствующему этапу родительского, проведите назначение исполнителей и выполните минимальные настройки форм этапов процессов, чтобы обеспечить визуальную узнаваемость сообщений этапов. Запустите родительскую работу и проследите её выполнение до завершения.

23.18.3 Параллельное выполнение этапов работ (разветвление)

В Lotsia Workflow имеется возможность включать в шаблоны работ элементы, выполняемые параллельно. Для обеспечения параллельности выполнения этапов и переходов используется специальный элемент шаблона – разветвитель, который всегда имеет парный элемент – объединитель, вставляемый в шаблон автоматически при вставке разветвителя. Для добавления разветвителя и объединителя используется пункт контекстного меню «Объект» → «Разветвитель» или кнопка  инструментальной панели.

Между разветвителем и объединителем размещаются цепочки из этапов и переходов, выполняемые параллельно, допускается использование вложенных пар разветвлений/объединений, глубина вложенности не ограничивается. Параллельно выполняемые этапы работы не могут выходить за пределы разветвителя и объединителя. (Рисунок 282).

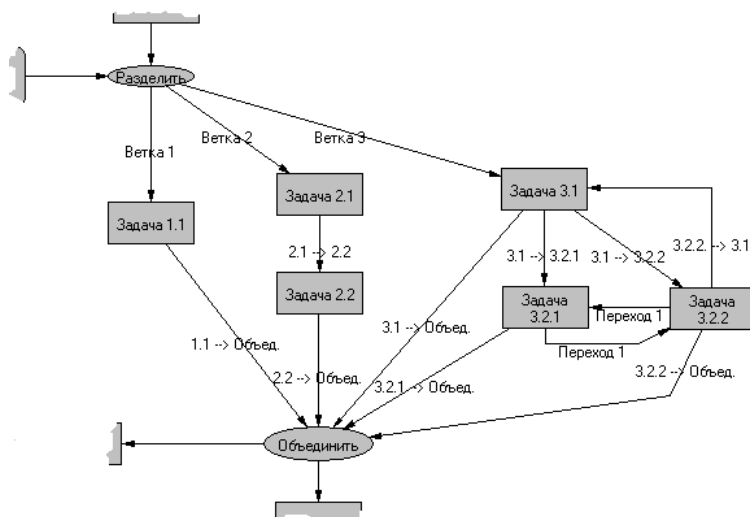


Рисунок 282 Возможный вид процесса с разветвлением. Фрагмент

Каждый исходящий переход из разветвителя называется веткой и выполняется параллельно. Ветки не могут соединяться, каждая ветка выполняется отдельно. Обратите внимание на отличие ветки от перехода: Ветка 1, Ветка 2, Ветка 3 выполняются параллельно и одновременно. Задача 1.1 и любая одна из Задач 2. ... и любая одна из Задач 3. ... могут также выполняться параллельно и одновременно.

Задача 2.1 и Задача 2.2 выполняются последовательно и не могут выполняться одновременно.

Задачи: 3.1, 3.2.1, 3.2.2 могут выполняться в различной последовательности, но одновременно они выполняться не могут. При объединении Ветки 3 может быть выполнен только один из переходов к объединителю.

Таким образом, независимо от количества возможных переходов в карте работы, количество выполненных входящих переходов в объединитель не может быть больше количества веток, вышедших из разветвителя.

23.18.3.1 Настройки разветвления

Объект разветвителя может использоваться как обычная задача, единственное отличие – невозможность использования в данной форме predetermined колонки «Переход» (переменной `select_action`): если в форме такой задачи выполнять выбор перехода стандартными средствами (например, см. рисунок 239), то выбор будет игнорироваться – выполняются одновременно все ветки (с учетом условий выполнения – см. п. 23.13).

Об использовании форм переходов для веток можно прочесть в документации по системе Lotsia PDM PLUS. Отметим, что для конечного пользователя такие понятия, как «ветка», «разветвить» и т.д. не всегда очевидны, поэтому использование конечным пользователем указанных форм рекомендуется исключить. Рекомендуется следующая настройка разветвления:

- задача разветвителя скрывается (см. п.23.16.4);
- все ветки делаются автоматическими (см. п. 23.14);
- если требуется обеспечить для пользователя возможность управления ветками, следует использовать условия выполнения перехода для веток. В качестве таких условий могут использоваться проверки значений пользовательских переменных, значения переменным могут присваиваться в форме одного из предыдущих этапов (Рисунок 283).

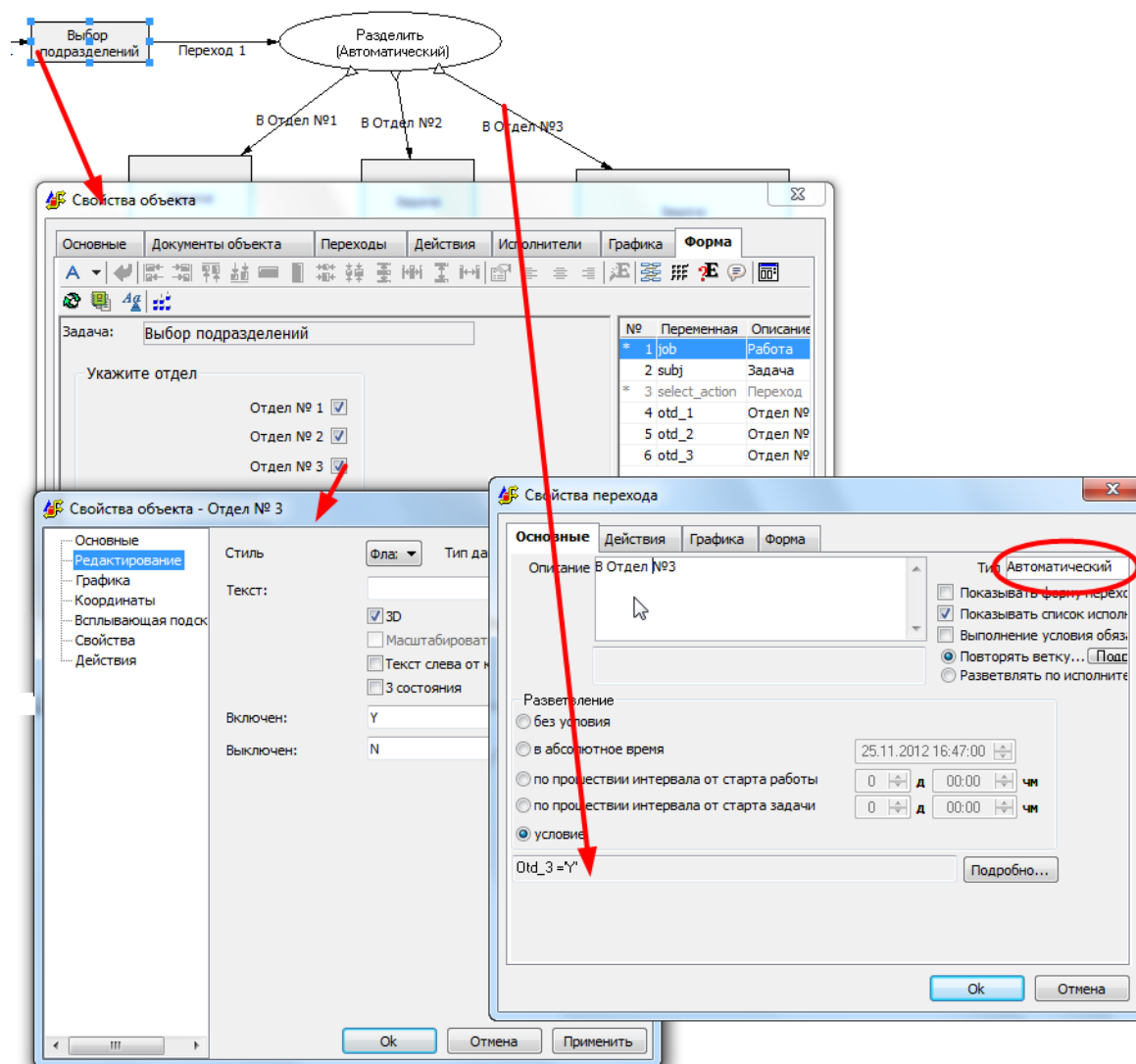


Рисунок 283 Пример возможной настройки управления разветвлением

Остальные ветки управляются аналогично.

Возможен также аналогичный вариант без скрытия этапа разветвителя, с выполнением аналогичных настроек в форме разветвителя и использованием ручных переходов, без показа форм переходов.

После выполнения разветвления процесс разделяется на параллельные работы.

Каждая ветка, в отличие от обычного перехода, имеет дополнительные настройки, позволяющие выполнять её более одного раза (Рисунок 284).

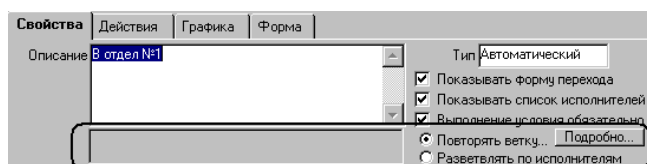


Рисунок 284 Настройки повтора ветки

Имеются следующие возможности настройки разветвления:

- при положении переключателя «Разветвлять по исполнителям» процесс по данной ветке разбивается на параллельные работы, количество которых

равно количеству основных исполнителей первой задачи после указанной ветки;

- при положении переключателя «Повторять ветку», процесс по данной ветке будет повторен столько раз, какое значение будет возвращено выражением, указанным в поле слева. Для ввода выражения используйте кнопку «Подробно» около переключателя. Если выражение не задано или возвращает нулевое значение, то ветка повторяется один раз.

Каждая ветка выполняется как отдельный процесс, до тех пор, пока из нее не будет выполнен переход на объект «Объединитель».

Создайте новый шаблон «Экспертиза документа» (Рисунок 285).

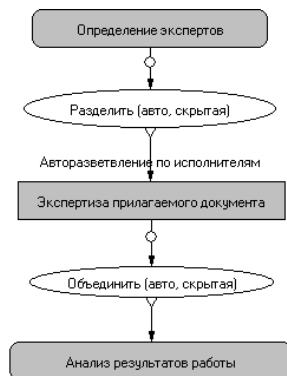


Рисунок 285 Шаблон «Экспертиза документа»

Создаваемый шаблон предполагает отправку объекта документа (объект содержит необходимые документы архива) экспертам, назначаемым на первом этапе согласования.

Предполагается, что работа экспертов между собой не связана, количество экспертов произвольное, список экспертов будет определяться на первом этапе.

В свойствах ветки «Авторастветвление по исполнителям» установите переключатель (Рисунок 284) в положение «Разветвлять по исполнителям». Сделайте данный переход и исходящий переход из объединителя автоматическими. На переходы, входящие в разделитель и объединитель добавьте действия по скрытию следующих задач (см. п.23.16.4).

Создайте документ шаблона (см. п. 23.4): добавьте документ – объект, назовите «Объект_документа_для_экспертизы»

23.18.3.2 Особенности настройки объединителя

В ходе выполнения параллельных работ локальные переменные, передаваемые на этап объединителя, могут иметь различные значения (см. п. 23.15). Кроме того, входящие сообщения могут иметь различный набор документов. Окончательный набор значений локальных переменных и документов, переданных на этап объединителя, определяется, так называемым «Главным сообщением». Какое из входящих сообщений считать Главным, определяется условием для Главного сообщения в свойствах объекта объединителя (Рисунок 286).

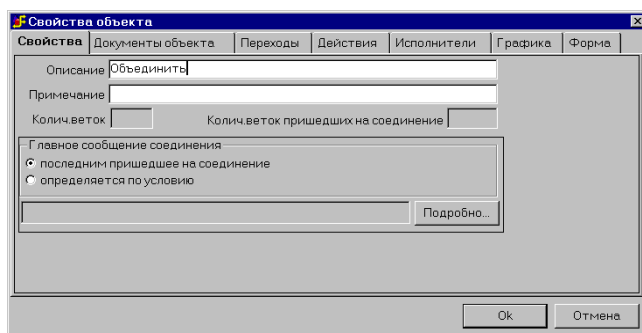


Рисунок 286 Свойства объекта «Объединитель»

По умолчанию, Главным является сообщение, пришедшее на этап объединения последним. В нашем шаблоне оставим настройку главного сообщения соединения по умолчанию.

Информация о значениях локальных переменных карты, поступившая из объединяемых параллельных веток, может отображаться на специальной вкладке сообщения объединителя – «Аудит». Об использовании данной вкладки можно прочитать в документации по системе Lotsia PDM PLUS.

На практике, в применении объединителя в качестве выполняемой пользователем (то есть, не скрытой) задачи обычно нет необходимости. Если объединитель не скрывать (см. п. 23.16.4), то сообщение задачи у исполнителя данного этапа будет появляться при выполнении каждого входящего в объединитель перехода. Обычно получение большого количества однотипных сообщений нежелательно, а требуется общая информация по итогам выполнения параллельных этапов после их полного завершения. Для последнего случая этап объединителя делается скрытым, переход из него, соответственно, автоматическим. Условие выполнения такого перехода обычно задается в виде равенства количества веток вышедших из разветвителя и пришедших в объединитель (Рисунок 287).



Рисунок 287 Задание свойств перехода, исходящего из объединителя

В шаблоне «Экспертиза документа» условие выполнения автоматического перехода из этапа объединителя задайте согласно рисунку (Рисунок 287).

23.19 Замечания по использованию локальных и глобальных переменных

При выполнении параллельных этапов важно, чтобы исполнители таких этапов могли работать независимо друг от друга, каждый со своими значениями переменных в форме. Например, в шаблоне «Экспертиза документа» для ввода экспертного заключения каждым экспертом, необходимо, чтобы переменные, которые будут для этой цели использоваться, были локальными. В противном случае, эксперты не смогут работать –

после обновления формы может отображаться совсем не та информация, которую ввел данный эксперт.

После завершения выполнения параллельных этапов важно сохранить результаты каждого из указанных этапов, а не только те, что будут переданы с Главным сообщением. Для сохранения нескольких значений локальных переменных, как правило, применяется запись указанных значений в соответствующие глобальные переменные (к текущему значению глобальной переменной дописывается очередное локальное значение). Указанная операция осуществляется действием, выполняемым по событию, происходящему не позднее, чем перед выполнением перехода, входящего в объединитель.

Отметим еще один важный момент: если два пользователя одновременно будут изменять одну и ту же глобальную переменную, в результате в данную переменную запишется лишь последнее изменение. Поэтому, если требуется именно дописать очередное значение к уже имеющемуся значению глобальной переменной, следует быть уверенным в том, что каждое очередное дописывание будет выполняться по завершении предыдущего обновления значения. Чтобы обеспечить гарантированное выполнение записи локальных значений в глобальные переменные последовательно, достаточно указанную операцию выполнять на сервере автоматических этапов. Данное замечание справедливо для всех глобальных переменных, как для массивов, так и для переменных с единичным типом значения, то же самое относится к изменению атрибутов и системных переменных.

В некоторых случаях исполнители параллельных этапов должны обмениваться информацией. Это достигается использованием глобальных переменных, например, на одной из веток присваивается значение такой переменной, на другой ветке данное значение может являться, например, условием выполнения автоматического перехода и появления соответствующего сообщения у пользователя. Использование системных переменных (либо атрибутивной информации объектов Lotsia PDM) позволяет обмениваться данными и между различными процессами Workflow.

В документе «Практические примеры и рекомендации» рассмотрены некоторые актуальные приемы работы с глобальными и локальными переменными, рекомендуется обращаться к данному документу при решении практических задач.

23.20 Массивы. Использование и некоторые приемы обработки

Переменные Workflow могут иметь тип значения «Одиночное» или «Массив». Последнее означает, что переменной соответствует список значений. Аналогично переменным с единичным типом значения, для локальных массивов список значений собственный для каждого этапа работы, для глобальных массивов – текущий список значений общий для всей работы в целом.

Внутри форм Workflow могут использоваться дополнительные (дочерние) формы. Для создания таких форм используется добавление в родительскую форму переменной с типом значения «Массив».

Таким образом, применение массивов в Workflow двоякое:




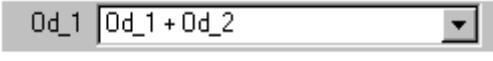
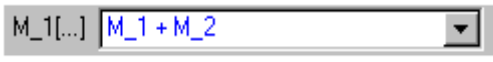

- хранение, отображение списков значений;
- создание дочерних форм

23.20.1 Использование массивов для хранения списков значений

Применение массивов в качестве переменных аналогично использованию переменных с одиночным типом значения, хотя и имеет особенности:

- массивы пользователей могут указываться в качестве исполнителей этапов аналогично переменным с типом значения «Одиночное» (Рисунок 256);
- присвоение значений переменным в действиях. Пусть Od_1 и Od_2 – строковые или числовые переменные с типом значения «Одиночное», M_1 и M_2 – массивы, тип может быть различным. Типы значений (Одиночное, массив) и типы (число, строка...) переменных должны соответствовать друг другу. Некоторые варианты присвоения значений приведены в таблице (Таблица 2):

Таблица 2 Некоторые возможные варианты присвоения значений переменным с различным типом данных

№	Аргумент и присваиваемое значение	Описание
1.		Переменной Od_1 присваивается значение Od_2
2.		Массиву M_1 присваивается список значений массива M_2.
3.		К имеющемуся списку (в конец списка) значений массива M_1 добавляется новый элемент, ему присваивается значение Od_1.
4.		Переменной Od_1 присваивается значение – к исходному значению Od_1 добавляется значение Od_2.
5.		К имеющемуся списку значений массива M_1 добавляется (в конец списка) список значений массива M_2. Случай присвоения массиву значения M_2 + Od_1 некорректен – такую строку необходимо разбить на две (№2 и №3)
6.		Первому элементу массива M_1 присваивается значение Od_1.

В списке переменных и аргументов действий описания переменных с типом данных «Массив» автоматически дополняются символами [...] (квадратные скобки с многоточием). При настройке действий над переменными с массивами, можно указать позицию в списке значений (индекс), в которую следует записать присваиваемое единичное значение. Для указания индекса, при настройке действия следует дважды щелкнуть левой клавишей мыши по описанию соответствующего аргумента, в открывшемся окне редактора выражений ввести выражение, возвращающее требуемое значение индекса (например, указать явно – см. строка №6, Таблица 2).

Правила заполнения массивов, в зависимости от указания индекса, следующие:

- если при присвоении значения элементу массива указывается положительный индекс несуществующего элемента, то в массив автоматически добавляются недостающие до этого индекса элементы;
- если используется значение 0 для индекса, то значение добавляется в конец массива;
- если значение индекса -1 – ничего не делается;
- в остальных случаях выдается сообщение об ошибке.

В редакторе выражений, используемом в настройке действий Workflow, имеется большое количество специфических функций для работы с массивами.

23.20.2 Создание форм массивов

Использование массивов для создания дочерних форм рассмотрим на практическом примере.

В работе «Экспертиза документа» (Рисунок 285) настройте форму первого этапа. Предполагается, что на данном этапе формируется список экспертов (в виде массива пользователей) и каждому эксперту дается собственная дополнительная информация (пояснения) (формируется соответствующий строковый массив).

В окне настройки формы этапа удалите лишние предопределенные элементы из правой части окна. В контекстном меню правой части окна выберите «Создать переменную». Создайте переменную Users: тип – «Пользователь», «Массив», вид – «Локальная».

Созданная переменная добавится в список переменных формы, в макете формы появится соответствующий объект массива и текстовое поле с его описанием. Добавленный объект массива, с точки зрения настройки формы этапа, представляет собой дочернюю форму.

Набор свойств объекта массива отличается от свойств колонок переменных с единичным типом значения:

- имеется пункт «Форма массива» для настройки соответствующей дочерней формы;
- пункт «Редактирование» специфическая (Рисунок 288).

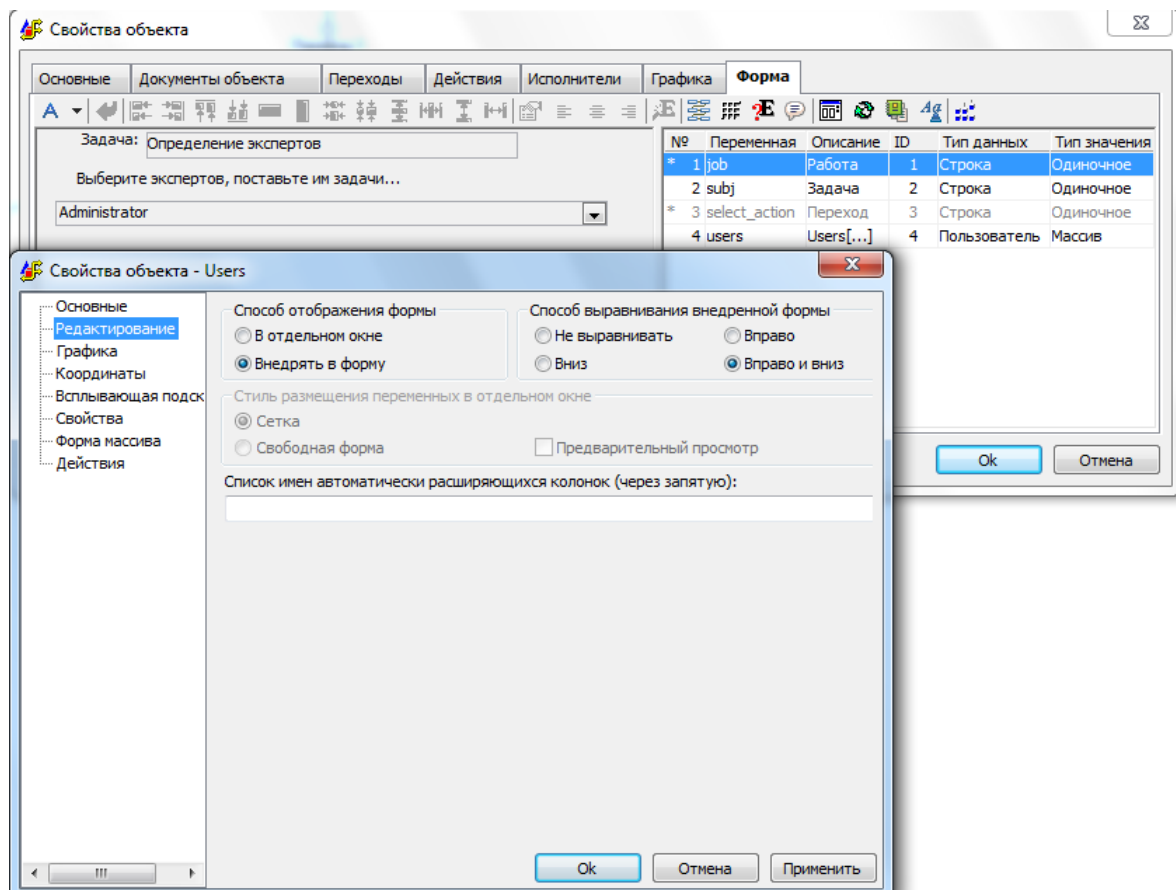


Рисунок 288 Начало настройки формы этапа. Открыто окно свойств объекта добавленного массива

Добавленная дочерняя форма может располагаться внутри родительской формы, что соответствует положению «Внедрять в форму» переключателя «Способ отображения массива» (Рисунок 288).

Форма массива также может отображаться в отдельном окне. При этом может использоваться два различных стиля размещения переменных в отдельном окне – «Сетка» и «Свободный». Стил «Сетка» используется и для внедренных массивов, он аналогичен одноименному стилю, используемому в отчетах. Стил «Свободный» соответствует настройке родительской формы этапа работы – в такую форму можно добавлять колонки, соответствующие переменным с одиночным типом значения, а также массивы. Массивы можно внедрять в форму, либо открывать в отдельном окне, в том числе используя стил «Свободный»... глубина вложенности таких дочерних форм не ограничивается. Рекурсивная входимость массивов для свободных форм не допускается. То есть: переменная типа массив, использованная для создания дочерней формы со стилем размещения переменных в окне «Свободная форма», не может использоваться внутри указанной свободной формы. Настройка форм массивов, открываемых в отдельном окне, не отличается, соответственно:

- для стили «Сетка» – от настройки форм внедренных массивов;
- для стили «Свободная форма» – от настройки форм задач.

В форме внедренного массива и в форме массива со стилем размещения переменных в окне – «Сетка», применение переменной, использованной для создания данной формы, не ограничивается.

Установите переключатель «Способ отображения массива» (см. рисунок 288) в положение «Внедрять в форму» (так установлено по умолчанию). Установите соседний

переключатель «Способ выравнивания внедренного массива» в положение «Вправо и вниз».

Для настройки формы массива выберите пункт «Форма массива» (Рисунок 289) окна свойств объекта массива.

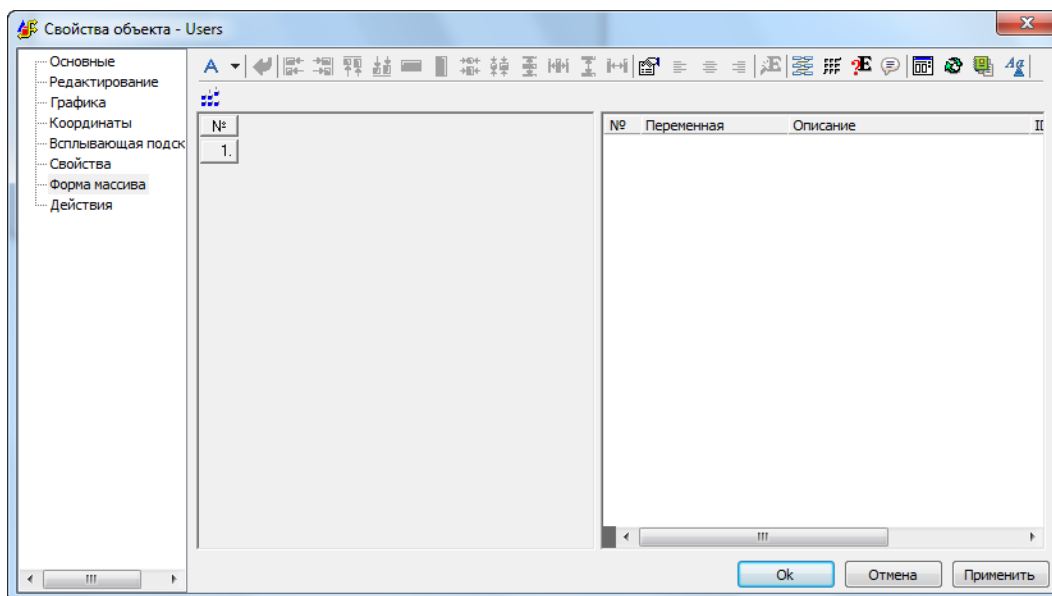


Рисунок 289 Вкладка «Форма массива». Начало настройки дочерней формы

Добавьте в форму массива колонку «Users_1» (контекстное меню правой части окна → «Добавить переменную»...). Создайте еще одну колонку – для ввода вопросов и дополнительной информации для каждого эксперта: контекстное меню правой части окна → «Создать переменную» → задайте свойства переменной: имя и описание – «zad_1», тип – «Строка», «Массив», вид – «Локальная» → нажмите «Ok» в окне свойств переменной. Созданная переменная будет добавлена в форму массива. После редактирования графики колонок и ввода текста названий колонок, вид формы может быть, например, следующим (Рисунок 290).

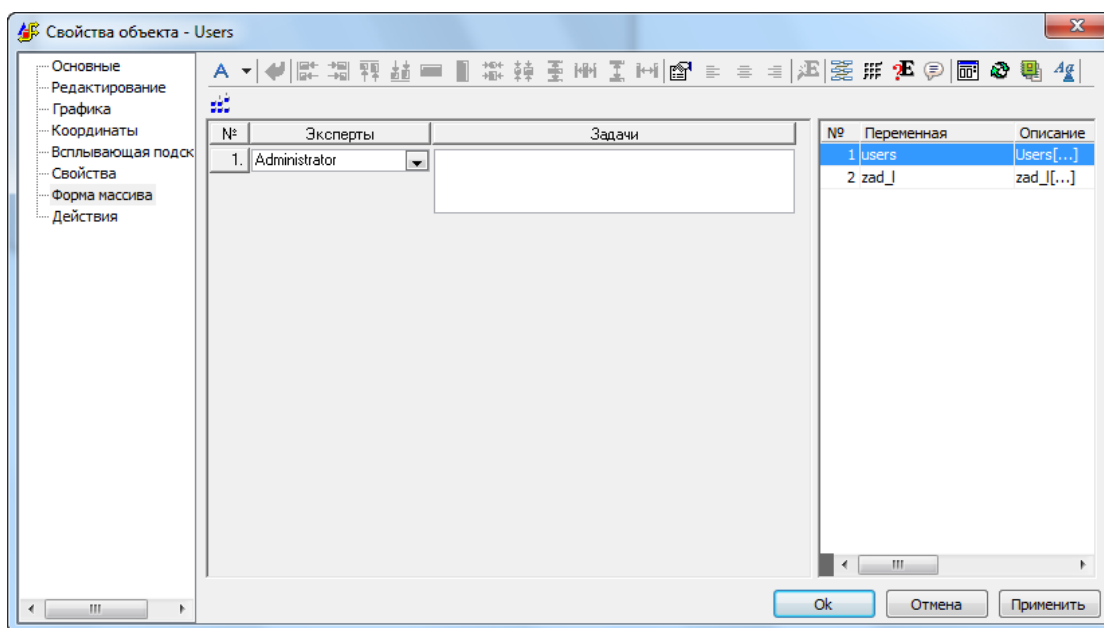


Рисунок 290 Настройка формы массива. Пример

Свойства дочерних форм стиля «Сетка» (в том числе и внедренных) определяются аналогично настройке отчетов, но имеется специфическая для массивов настройка

элементов формы. В контекстном меню левой части окна (Рисунок 290) выберите «Свойства областей», выберите пункт «Элементы массива» (Рисунок 291)

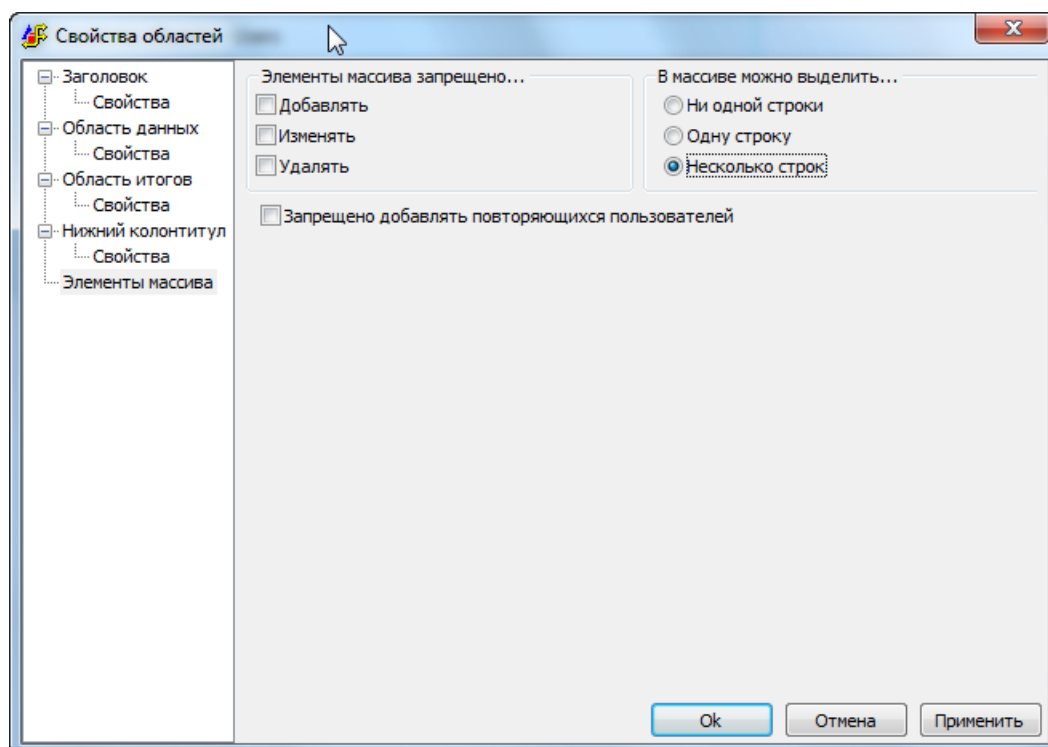


Рисунок 291 Специфические настройки форм массивов

Если для каждого эксперта следует запретить добавление более одной строки, следует включить флажок «Запрещено добавлять повторяющихся пользователей», если это не так (например, данный эксперт должен дать заключение по нескольким вопросам) – данный флажок устанавливать не следует – для определенности – добавление повторяющихся пользователей ограничивать не будем.

Установите переключатель «В массиве можно выделить...» в положение «Несколько строк» (при выделении нескольких строк можно будет удалять сразу несколько строк в форме массива).

Сохраните сделанные установки и форму массива. Закончите редактирование и сохраните форму начального этапа работы (Рисунок 292), сохраните шаблон.

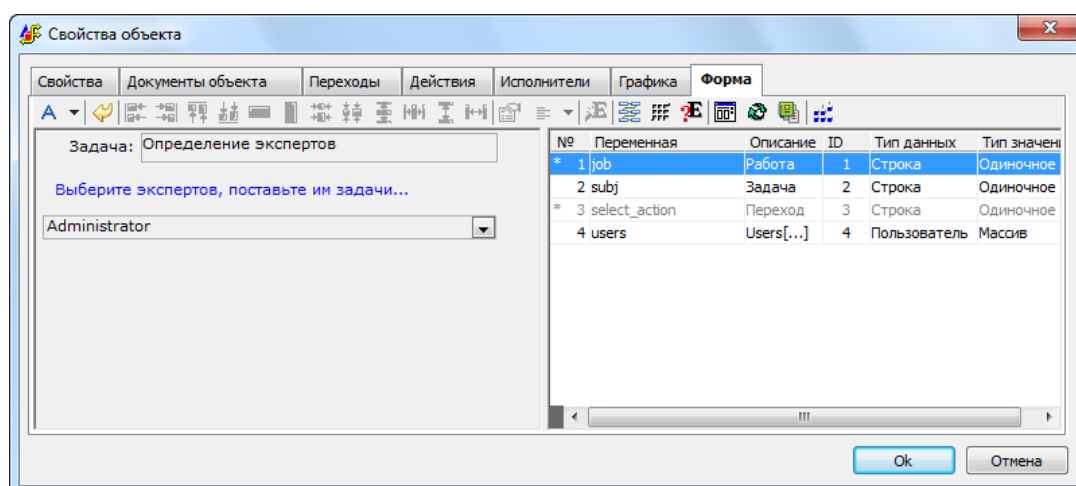


Рисунок 292 Окончание редактирования формы начального этапа

В качестве исполнителя задачи «Экспертиза прилагаемого документа» (Рисунок 285) укажите переменную Users, исполнителями окончания работы, разветвителя и объединителя может быть указан инициатор работы.

Чтобы увидеть результат настройки формы первого этапа, можно запустить работу на выполнение. В открывшемся окне формы первого этапа, в форме внедренного массива откройте контекстное меню и выберите «Добавить». Если в массиве есть колонка типа «Пользователь» (наш случай), то вначале будет открыто окно выбора пользователя (независимо от порядка следования колонок). После указания пользователя, в обе колонки формы добавляется по одному элементу. Текстовое поле будет пустым, в него можно ввести текст задания, вопросы и необходимую информацию для выбранного эксперта и продолжить добавление строк аналогичным образом, до окончательного формирования списка. Для удаления строк их необходимо выделить и выбрать в контекстном меню «Удалить». Запускать работу далее первого этапа до окончания настройки шаблона преждевременно, нажмите «Отмена» для завершения.

23.20.3 Разделение и объединение массивов. Пример настройки шаблона

Продолжите настройку шаблона. Отключите показ форм переходов.

На переход «Авторазветвление по исполнителям» необходимо добавить действие, которое бы обеспечивало передачу каждому эксперту только предназначенной для него дополнительной информации и вопросов. В шаблоне «экспертиза документа» используются массивы Users и zad_1. Позиция в списке значений (индекс) пользователя и индекс соответствующего данному пользователю значения в списке вопросов и дополнительной информации совпадают, поскольку добавлялись в общей форме. Каждому эксперту следует передать только значения из списка вопросов и дополнительной информации, расположенные напротив его фамилии, то есть, имеющие тот же индекс, что и его фамилия в списке. Достигнуть этого проще всего, удалив из локального массива zad_1 «чужие» элементы – то есть, те элементы, индекс которых не равен индексу данного эксперта в массиве Users (напомним, что массив zad_1 – локальный, на каждой задаче может иметь собственный набор значений).

При открытии окна задачи конкретным исполнителем, данного исполнителя можно записать в локальную переменную «Us_1», тип – «Пользователь», тип значения – «Одиночное». Создайте такую переменную.

К задаче «Экспертиза прилагаемого документа» (Рисунок 285) добавьте действие над переменными. В данном действии (Рисунок 293) присвоим переменной «Us_1» значение f_GetCurrentUserID().

Индексы удаляемых элементов запишем в локальный массив m_num_del, тип – «Число». Данную переменную добавьте в создаваемое действие над переменными и в поле «Значение» данной переменной введите выражение:

```
f_GetArrElInd( Users, '@ArrEL<>' + Us_1 )
```

f_GetArrElInd – функция, расположенная в группе Arrays редактора выражений. Ищет в массиве Users (первый аргумент – в скобках, до запятой) элементы по строке – шаблону (это второй аргумент – выражение в скобках после запятой).

@ArrEL – предопределенный аргумент, означающий элементы массива. Строка – шаблон указывается в кавычках, если используются переменные – они должны находиться в выражении второго аргумента вне кавычек. Если в выражении используются функции – они должны быть внутри кавычек. Если в аргументах таких функций необходимо использовать кавычки, то перед каждой такой кавычкой ставится символ «~» («тильда») без кавычек. Функция f_GetArrElInd возвращает числовой массив индексов найденных

элементов – в данном случае – индексов элементов массива пользователей, значения которых не соответствуют исполнителю очередной задачи.

Для удаления из локального массива `zad_1` элементов с индексами, содержащимися в массиве `m_num_del`, добавьте в действие переменную `zad_1`, в её поле «Значение» введите выражение:

`f_ExcludeArrEl(zad_1, m_num_del)`

`f_ExcludeArrEl` – функция, расположенная в группе «Массивы» редактора выражений. Возвращает массив того же типа, что и первый аргумент (`zad_1`), удаляет из массива – первого аргумента, – элементы, индексы которых указаны во втором аргументе (`m_num_del`).

Введите описание действия (Рисунок 293) и сохраните его, сохраните шаблон.

Рисунок 293 Действие, удаляющее элементы массива, не относящиеся к данному исполнителю задачи

Действие будем выполнять по событию «Перед открытием Окна задачи».

Настройте форму этапа «Экспертиза прилагаемого документа».

Для отображения вопросов и дополнительной информации для данного эксперта, в форму задачи добавьте массив `Users`, способ отображения – внедрять в форму. В дочернюю форму добавьте колонки:

- массив Users – для отображения дополнительной информации и вопросов для данного эксперта;
- создайте и добавьте в форму новый локальный строковый массив для отображения заключения эксперта по каждому вопросу – otv_1.

Настройте необходимые графические элементы и заголовки формы (Рисунок 294).

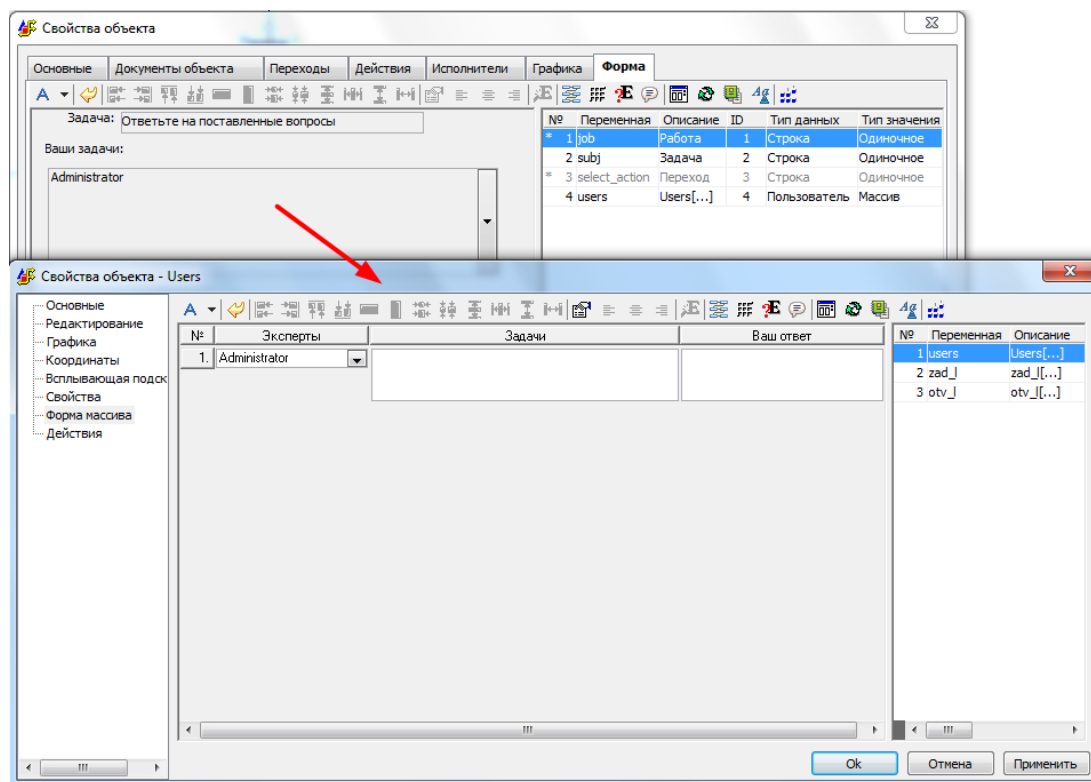


Рисунок 294 Настройка формы этапа «Экспертиза прилагаемого документа»

Для записи общих результатов согласования создайте соответствующие глобальные массивы:

Gl_Us – тип – «Пользователь», для записи экспертов;

Gl_zad, Gl_otv – типы – «Строка», для записи, соответственно, вопросов экспертам и их заключений по данным вопросам.

Для заполнения созданных массивов создайте действие над переменными (Рисунок 295), которое будет выполняться после завершения задачи «Экспертиза документа» (либо перед выполнением перехода к этапу «Объединить»). Отметим, что действие по формированию глобальных массивов в данном случае правильнее делать серверным (подробнее см. п. 23.19).

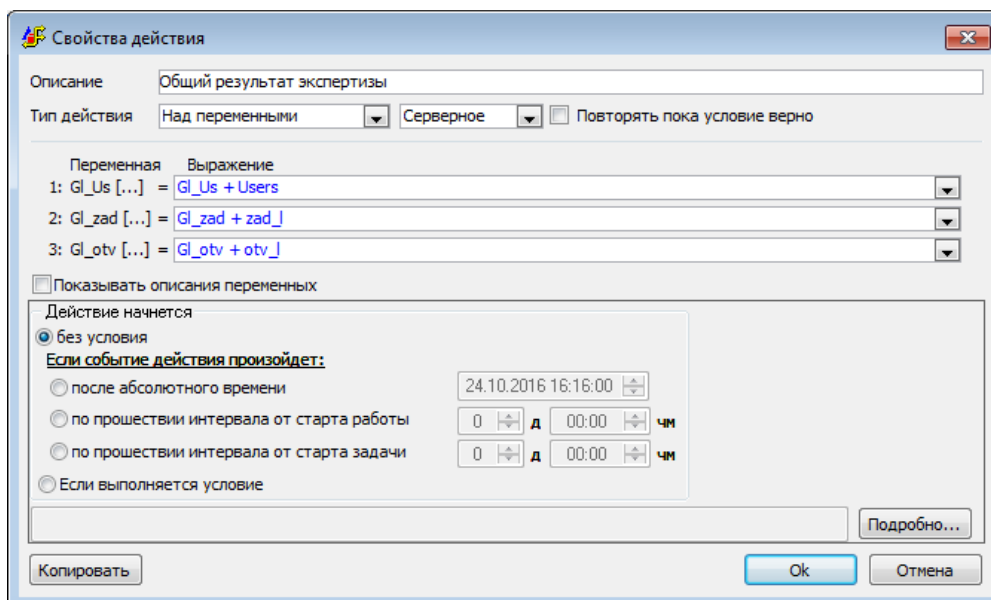


Рисунок 295 Действие по формированию общего результата экспертизы

Указанные глобальные массивы можно отобразить в форме окончания работы для просмотра результатов экспертизы. Выполните соответствующую настройку формы заключительного этапа.

23.20.4 Преобразование массивов в одиночное значение

В некоторых случаях может потребоваться преобразование массивов в одиночное строковое значение.

Для преобразования **одного** массива в строку с разделителями, рекомендуется использовать функцию `f_ArrayToString`, описание функции доступно в контекстной справке по Lotsia PDM PLUS. Действие не циклическое.

Если строку требуется сформировать из нескольких массивов, понадобится их циклическая обработка. В качестве примера рассмотрим такое действие над переменной Workflow. Результат выполнения такого действия можно использовать для отображения в сообщении задачи – добавив соответствующее действие над переменными – см. п. 23.16.2., либо для присвоения значения атрибуту объекта при выполнении действия над объектом – в качестве входящего значения аргументу действия над объектом передается указанная переменная Workflow.

Действие будет выполняться циклически – считывать из массивов `Gl_Us`, `Gl_zad`, `Gl_otv` очередные значения и записывать соответствующую информацию в строковую переменную «rez» с типом значения «Одиночное». Создайте такую переменную.

Создайте на переходе из этапа «Объединить» действие над переменными, добавьте переменную `rez`. В поле «Значение» данной переменной введите выражение:

```
rez +if (rez = " ','~r~n') +'Эксперт '+f_UserName(f_GetNArrEl( Gl_Us , f_GetCycleCounter(
this ) ))+', вoпpoc: '+f_GetSArrEl( Gl_zad , f_GetCycleCounter( this ) )+' Oтвeт:'+f_GetSArrEl(
Gl_otv , f_GetCycleCounter( this ) )
```

Данное выражение к имеющемуся значению `rez` дописывает значения очередных элементов массивов.

Функция `if (rez = " ,", '~r~n')` используется для того, чтобы не добавлять пустую строку при пустой `rez`, выражение `'~r~n'` означает вставку перехода на новую строку.

Функции `f_GetNArrEl()` и `f_GetSArrEl()` используются, соответственно, для извлечения из массива пользователей или из строковых массивов элементов, индекс которых указан во втором аргументе (в скобках после запятой). В качестве указанного второго аргумента используется функция `f_GetCycleCounter(this)`, которая возвращает номер текущего цикла действия.

Функция `f_UserName()` преобразует код пользователя, извлеченный из массива пользователей `Gl_Users_1` в имя пользователя.

Для того чтобы действие выполнялось циклически, в окне свойств действия включите флажок «Повторять пока условие верно». В качестве условия завершения цикла укажите следующее выражение:

`f_GetCycleCounter(this) <= f_GetArrUpperBound(Gl_Us)`

Данное условие обеспечит прекращение выполнения действия при номере очередного цикла действия больше, чем количество элементов в массиве экспертов (Рисунок 296).

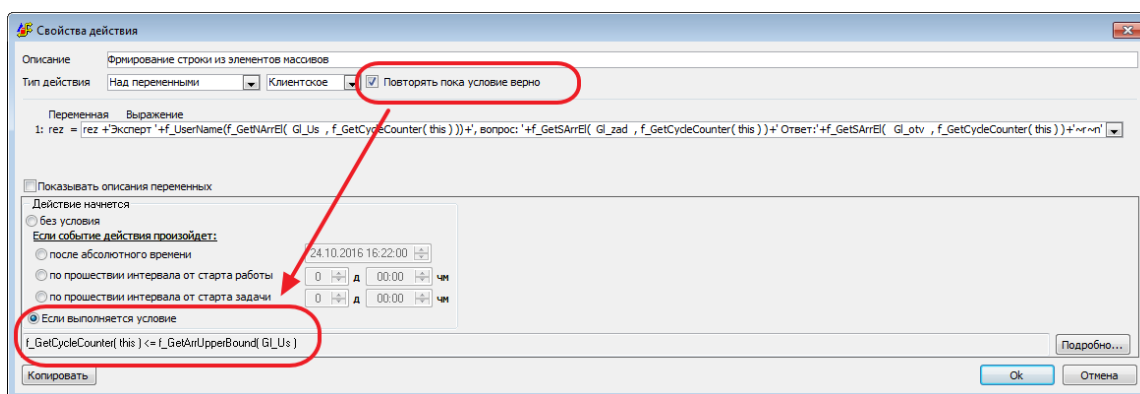


Рисунок 296 Указание условия циклического выполнения действия

Результат выполнения действия – сформированное значение переменной «`rez`», – передайте в текст следующей задачи (см. п. 23.16.2). Полученное значение переменной «`rez`» может быть также использовано, например, для присвоения значения атрибуту объекта документа, отправляемого на экспертизу. Для записи указанного значения в атрибут, следует создать и использовать в работе Workflow соответствующее действие над объектом.

23.20.5 Преобразование строки с разделителями в массив

В некоторых случаях требуется решить задачу, обратную рассмотренной (п. 23.20.4) – из строки с разделителями получить **один** массив. Рекомендуется использовать для этих целей функцию `f_ArrayToString`.

23.21 Резюме

Для создания шаблона используется пункт Главного меню «Администрирование» → «Шаблоны и карты работ» → «Шаблоны» → в открывшемся окне нажать «Создать».

В открывшемся окне настройки шаблона выполняется настройка шаблона в следующей последовательности:

Строится блок – схема будущего процесса – создаются этапы и переходы.

Созданные элементы шаблона снабжаются описаниями.

Определяются исполнители этапов работы, создаются документы шаблона (определяются возможные документы, которые могут передаваться в данную работу при её запуске).

Строятся формы этапов и переходов (при необходимости). Выполняется проверка шаблона.

После выполнения указанных настроек, работа по данному шаблону может быть запущена.

Исполнители работают с формами соответствующих этапов и формами исходящих переходов (если такие формы используются).

В работах могут использоваться автоматические элементы – смена исполнителей, автоматические переходы и серверные действия. Для обеспечения функционирования указанных элементов необходим запуск сервера автоматических этапов. Сервер автоматических этапов может запускаться как приложение, либо как служба (сервис).

На выполнение переходов могут накладываться условия. Условия задаются в виде выражений, используется редактор выражений.

В шаблонах работ могут использоваться переменные данного шаблона работ, а также системные переменные. Для чтения и присвоения значений переменным шаблона может использоваться ввод значений в поля форм, а также выполнение действий над объектами и действий над переменными.

Работа с системными переменными возможна только с помощью действий.

Чтение информации с документов шаблона может осуществляться как действиями над переменными, так и действиями над объектами. Изменения объектов Lotsia PDM выполняются только действиями над объектами (или ручным вводом при редактировании атрибутивной информации объектов или состава проектов).

Действия в работах Workflow выполняются по различным событиям, в том числе:

- Перед открытием окна задачи
- После открытия окна задачи
- После нажатия кнопки формы
- Щелчок по полю переменной
- После изменения значения переменной
- После открытия окна формы массива
- Перед закрытием окна формы массива
- После перехода к другой строке формы
- После вставки строки в форму
- Перед закрытием окна задачи
- После закрытия окна задачи
- После выполнения задачи
- Перед выполнением перехода

На каждое событие можно назначить выполнение нескольких действий в различной последовательности. Результат выполнения предыдущего действия может использоваться в следующем выполняемом действии.

С использованием действий возможно автоматическое запоминание и назначение исполнителей этапов работы, формирование сообщения задачи и других её свойств, в том числе изменение формы, скрытие и т.д.

Помимо задач, в шаблонах работ могут использоваться этапы других типов, в том числе: информационные рассылки, вложенные и внешние работы.

Информационная рассылка – специальный этап работы, предназначенный только для передачи информационных сообщений.

Добавленному объекту вложенной или внешней работы сопоставляется вложенная карта – имеющийся шаблон работы. Вложенность процессов может быть многократной, глубина вложенности не ограничивается.

В Lotsia Workflow имеется возможность включать в шаблоны работ элементы, выполняемые параллельно, для чего используется пара специальных элементов шаблона – «Разветвитель» и «Объединитель». Между разветвителем и объединителем размещаются цепочки из этапов и переходов, выполняемые параллельно.

Переменные Workflow могут иметь тип значения «Одиночное» или «Массив».

Применение массивов в Workflow двоякое – с одной стороны, переменная с типом значения «Массив» обеспечивает хранение и отображение списков значений, а с другой – может использоваться для создания дочерних (вложенных) форм. Дочерняя форма может располагаться внутри родительской формы (т.н. «Внедренная форма», или «Внедренный массив»), либо отображаться в отдельном окне. В последнем случае может использоваться два различных стиля размещения переменных – «Сетка» и «Свободный».

Присвоение значений переменным типа массив может выполняться как ручным вводом, так и действиями, с использованием специальных функций.

23.22 Шаблоны работ. Учебное задание

Пользуясь настоящим описанием, настройте в учебной БД шаблоны:

«Согласование документа» (Рисунок 231);

«Экспертиза документа» (Рисунок 285).

Настроенные Вами шаблоны могут полностью повторять описанные или отличаться. Цель – отладить созданный шаблон и добиться его правильной работы, получив соответствующие навыки работы с программой.

23.23 Шаблоны работ. Вопросы для самопроверки

1. Как создать новый шаблон работы, добавить задачу, переход, информационную рассылку, вложенную и внешнюю работы?
2. Как можно назначать исполнителей этапов работ?
3. Как настраивается форма этапа и перехода?
4. Как отключить показ формы перехода?
5. Как создать переменные шаблона, чем отличаются локальные и глобальные переменные?
6. Как создать и использовать системные переменные?
7. Как сделать задачу скрытой, отключить стандартные кнопки и вкладки в форме задачи?
8. Как создать кнопку в форме задачи?

9. Как добавить к элементу шаблона действие над переменными и действия над объектами?
10. Как использовать одно и то же действие для различных событий и элементов шаблона?
11. Как копировать действия шаблона?
12. Как обеспечить циклическое выполнение действия?
13. Как запустить и зачем нужен сервер автоматических этапов?
14. Как сделать действие серверным, где выполняются действия, расположенные после серверного?

24 Присвоение значений различным параметрам и переменным – имеющиеся возможности

Как для атрибутов, одиночных переменных (в Lotsia PDM и в Workflow), так и для массивов, значения могут присваиваться следующими способами:

- по умолчанию;
- ручным вводом;
- при выполнении действий.

Свойства объектов форм могут определяться с помощью выражений, заданных в полях соответствующих свойств (Рисунок 61).

Далее рассмотрим общий порядок и особенности присвоения значений для различных случаев.

24.1 Значения по умолчанию

В действиях над объектами, при создании переменной значение по умолчанию автоматически не присваивается (то есть, если специально не вводить значение – по умолчанию значение будет Null). В Workflow при создании единичных переменных значения по умолчанию автоматически создаются (то есть, даже если специально не вводить значение – по умолчанию значение будет не Null).

При создании переменных с типом значения «Массив», автоматического создания элементов массива не производится. Если требуется значение по умолчанию – его следует ввести в свойствах данной переменной. Для формирования списка значений следует щелкнуть по полю «Значение», в открывшемся окне «Массив значений» сформировать требуемый список значений.

24.2 Ручной ввод значений. Формирование и использование списков значений

Ручной ввод значений может осуществляться с использованием списков значений. Списки значений могут быть статическими (то есть, возможные значения переменной указываются при настройке данной формы), либо формироваться динамически, в соответствии с текущей (актуальной) информацией, содержащейся в БД. Формирование статического списка значений средствами редактора форм в большинстве случаев очевидно.

Для Lotsia PDM, динамически (то есть, с использованием актуальных значений, имеющихся в БД на момент использования данного списка, без его дополнительной настройки), списки могут формироваться в следующих случаях:

- присвоение значений в формах действий над объектами с использованием списка имеющихся значений атрибута (см. п. 17.3.2.1);
- присвоение значений в формах действий над объектами с использованием выражений для формирования списка возможных значений (см. п. 17.3.2.2);
- использование классификатора в атрибутивных формах и действиях над объектами (настройка выполняется в свойствах переменной, см. п. 17.3.4);
- отбор объектов в форме действия над объектами, выполняемый согласно заданным условиям отбора объектов (см. п. 17.3.2.3);

- использование настраиваемых окон выбора в формах действий над объектами (см. п. 17.3.1.3);
- использование шагов типа «Окно выбора» (см. п. 17.4)

При работе с отчетами, выпадающие списки для присвоения значений аргументам отчета могут формироваться с помощью пользовательского отчета или запроса (см. п. 20.5.6.1)

В Lotsia Workflow могут использоваться следующие собственные средства формирования списков значений пользовательских переменных:

- для переменных типа «Пользователь» при настройке формы, на вкладке «Редактирование» соответствующего объекта формы может быть задан ограничительный список, в том числе с использованием групп пользователей. При выборе пользователей доступен актуальный состав соответствующих групп. Для массивов возможен выбор сразу нескольких пользователей;
- сформировать список возможных значений переменной Workflow можно с помощью выполнения действия Workflow – см. п. 24.2.1. Действие в Workflow может выполняться по различным событиям (например, по изменению переменной, используемой в качестве аргумента SQL-запроса), обеспечивая соответствующее динамическое обновление списка возможных значений переменной;

Средства ввода Lotsia PDM, в том числе выбор объектов (и классификатор), можно использовать в Workflow, путем выполнения действий над объектами с возвратом полученных значений в переменные Workflow.

Возможно и использование средств Lotsia Workflow в Lotsia PDM. Для этого следует запустить работу и использовать форму её начального этапа в качестве окна ввода данных, с последующей передачей результата ввода в атрибуты и/или использование их при создании объектных структур.

24.2.1 Динамическое формирование списка возможных значений переменной для формы Workflow

Для формирования/обновления списка возможных значений переменной, необходимо до использования данного списка, но после открытия окна с формой задачи выполнить соответствующее действие над переменными. В данном действии используется функция `f_ModifyForm`, которая позволяет определить свойства формы, в том числе, значения списка для объектов заданной формы. Аргументом данной функции должна быть специальная строка, содержащая необходимые значения.

Сформировать такую строку можно функцией `f_ExecSQLSelect_3`, которая выполняет в текущей Базе Данных SQL-запрос и возвращает требуемую строку, составленную из значений колонок запроса, соединенных посредством символов-разделителей. Итоговая строка данной функции подготавливается для её использования в качестве выпадающего списка значений переменной формы. Она должна возвращать два поля – описание значения и само значение. Значения должны разделяться табулятором (`char(9)` или `'~t'`), а строки – символом `'/'`.

Пример выражения для формирования выпадающего списка сотрудников:

```
f_ModifyForm ( this , " , 'sotr_str.values=~"+ f_ExecSQLSelect_3( 'select
O_R.description,V_S.value from lsdbo.object_reference_view O_R, lsdbo.value_string_view
V_S, lsdbo.attrb_value_view A_V where A_V.Object_ID= O_R.id and
```

O_R.Type_ID=100000000700031 and A_V.Value_id= V_S.id and V_S.attrib_id =1000000004003031 and A_V.attrib_id =1000000004003031', '~t', '/', 'ddlb', 10)+ '~")

Данный список (см. рисунок 297) формируется для строковой переменной sotr_str, с использованием имеющихся в БД объектов типа «Сотрудник» (в данном примере – id типа объекта = 100000000700031). В выпадающем списке отображаются описания объектов сотрудников, которым сопоставлены пользователи БД, то есть, в данном примере отбираются объекты сотрудников, имеющие атрибут «id_пользователя» (в данном примере – id атрибута = 1000000004003031). Переменной присваивается значение данного атрибута соответствующего объекта сотрудника.

Для переменной sotr_str в форме задачи должен быть установлен стиль редактирования «Выпадающий список». В список значений в настройке свойств данного объекта формы необходимо добавить произвольное значение. По умолчанию в форме будет отображаться текущее значение данной переменной.



Рисунок 297 Справочник Предприятия и сформированный по нему в форме задачи выпадающий список сотрудников. Пример

24.3 Способы автоматической обработки значений

В predetermined processes possible use of actions over objects, as well as actions over variables. Sometimes, for one and the same purposes both types of actions are applicable, in this case one should take into account the following:

Actions over variables are performed, as a rule, faster, therefore, when this is possible, especially in forms, with which the user works, one should strive to use actions over variables (for example, when reading attribute information from a single object Lotsia PDM PLUS – document template).

In some cases the value of a variable or attribute does not need to be changed, but it is necessary to display in the form transformed data – for example, by the user identifier, recorded in the variable form to display in the form the name of the user, or by the object identifier – its description and so on. In such cases the simplest solution may be the use in the form of calculated fields.

When solving specific tasks, related to the processing of variables of various types, it is recommended to study the document «Practical recommendations and examples».

24.3.1 Использование действий над объектами

In Workflow it is recommended to use actions over objects only in those cases, when actions over variables are inapplicable, and namely:

- reading attribute information from a document template, which is a mass;

- выполнение операций, вызывающих изменения объектов Lotsia PDM: изменение атрибутивной информации, создание объектов, создание или разрыв связей между объектами, импорт документов в архив, синхронизация атрибутов с документом архива, изменение прав доступа и т.п.;
- очистка документа шаблона, добавление документов в документ шаблона выполняемого процесса;
- открытие окна документа (например, дерева проекта) из формы задачи.

В действиях над объектами обрабатываются одиночные значения (переменные имеют только одиночный тип значения). При необходимости автоматической обработки нескольких элементов в действии над объектом существуют следующие пути:

- запуск действий по нескольким выделенным объектам. Объекты могут выделяться в окне поиска, выполненном отчете или подборке. Если в Workflow в качестве документа шаблона используется массив объектов, действие над объектом, в качестве аргумента которого использован данный документ шаблона будет выполняться по нему многократно – аналогично подборке, без использования флажка «Повторять пока условие верно» (см. рисунок 296).
- запуск действия по массиву (переменной Workflow). В этом случае аргументом действия должен быть конкретный элемент (элементы) массива (массивов). При необходимости обработки массива целиком, в качестве аргумента указывается текущий элемент массива. Например, для строкового массива `Gl_zad` его очередной элемент извлекается выражением: `f_GetSArrEl(Gl_zad , f_GetCycleCounter(this))`. Действие выполняется циклически, флажок «Повторять пока условие верно» (см. рисунок 296) должен быть включен, условие выполнения задано, например, в виде: `f_GetArrUpperBound(Gl_zad) >= f_GetCycleCounter(this)`
- использование циклической обработки внутри действий над объектами (см. п. 17.6). Сформировать строку, содержащую идентификаторы объектов, можно, например, с использованием множественного выбора в окне выбора объекта или в окне выбора файла (см. п. 17.3.1.3, п. 17.3.2.4), либо с использованием функций редактора выражений – например – `f_ExecSQLSelect_3`.

24.3.2 Использование действий над переменными

Для считывания данных и присвоения значений переменным Workflow в большинстве случаев достаточно использования действий над переменными.

Для действий в Workflow возможно и циклическое выполнение – например, если необходимо обработать элементы массива по отдельности (см. п. 23.20.4).

В некоторых случаях вместо использования документов шаблона, удобно использовать переменные Workflow (в том числе, с типом значения «Массив») для хранения ID объектов. Отображать атрибутивную информацию в формах массивов можно с помощью выполнения SQL-запросов (см. п. 24.3.3). Вводить/изменять данные пользователь может в тех же формах, запись данных в объекты Lotsia PDM будет выполняться действиями над объектами (например, путем циклического, по элементам массива, выполнения серверного действия).

24.3.3 Использование SQL-запросов

Использование прямых SQL-запросов зачастую является оптимальным при решении конкретных прикладных задач. В Lotsia PDM PLUS имеются следующие возможности использования SQL-запросов:

24.3.3.1 Применение SQL-запросов в шаблонах работ

В Workflow имеется возможность заполнения формы массива результатами SQL-запроса. Для этого SQL-запрос внедряется в форму массива непосредственно (контекстное меню окна настройки формы массива → «Внедрить SQL- Select...») или при выполнении соответствующего действия. В указанном действии необходимо последовательно выполнить функции `f_EmbeddedSQLSelect ()`, `f_RetrieveForm ()`, `f_FormDataToArray ()`. Можно использовать единственную строковую переменную для возврата значения функцией и общее выражение вида:
`f_EmbeddedSQLSelect ()+f_RetrieveForm ()+f_FormDataToArray ()`.

Отметим, что и при использовании контекстного меню окна настройки формы массива для внедрения SQL в форму, для отображения результата выполнения SQL в форме, необходимо выполнение действия, содержащего функции `f_RetrieveForm ()` и `f_FormDataToArray ()`.

При использовании функции `f_EmbeddedSQLSelect ()` для внедрения в форму массива SQL, в форму массива предварительно должны быть добавлены колонки. Порядок следования колонок в списке переменных формы (в правой части окна настройки формы) и типы значений должны соответствовать колонкам SQL-запроса в функции `f_EmbeddedSQLSelect`. Для текущей версии программы необходимо выполнение еще одного условия: объекты колонок в форме должны присутствовать (если их показ пользователю не нужен – задайте в свойствах соответствующих объектов равные 0 ширину и видимость колонок, но не удаляйте объекты колонок из формы).

Для выполнения такого запроса окно с формой массива должно быть открыто – то есть, данный способ применяется, когда окно формы с SQL-запросом открывается пользователем. В этом случае действие может быть выполнено после открытия окна задачи или по другому событию при открытой форме.

Если же задача выполняется автоматически и требуется выполнить SQL-запрос, следует в действии (могут использоваться как действия над переменными, так и действия над объектами – см. п. 24.3.3.3) использовать функцию `f_ExecSQLSelect_3`. Результат выполнения запроса будет передан в одиночную строковую переменную. При необходимости, подобное значение может быть автоматически преобразовано в массив значений (см. п. 23.20.5). Подобным образом, может быть реализована, например, автоматическая рассылка сообщений найденным при выполнении SQL-запроса пользователям.

24.3.3.2 Использование SQL-запросов в выражениях

Имеется возможность использования SQL-запроса в различных выражениях: в вычисляемых полях, в свойствах объектов форм и отчетов, в действиях над объектами и переменными и др. В указанных выражениях используется функция `f_ExecSQLSelect_3`.

Таким образом, можно, например, управлять видимостью объектов формы в зависимости от наличия или отсутствия у текущего объекта родительских или дочерних объектов. В вычисляемом поле можно отобразить, например, информацию об одном или нескольких объектах, найденных в БД. Имитировать несколько строк можно, если в функции `f_ExecSQLSelect_3` указать в качестве разделителя строк выражение «~r~n».

Отдельный случай – использование SQL-запросов в отчетах для задания условий предварительного отбора объектов. Об использовании условий предварительного отбора прочтите в документации по Lotsia PDM PLUS, либо воспользуйтесь контекстной справкой.

Еще один случай возможного применения SQL-запросов – формирование списков возможных значений переменных. Для переменных действий над объектами такой список формируется настройкой свойств переменных (см. п. 17.3.2.2), для переменных Workflow – выполнением действий над переменными.

24.3.3.3 Использование SQL-запросов в действиях над объектами

Использование внутри действий над объектами SQL-запросов возможно. Для этого используется функция `f_ExecSQLSelect_3`, помещаемая внутрь выражений действия `Set()`, `GoTo()` и др. Напомним, что данная функция (имеется в списке функций редактора выражений), возвращает результат выполнения SQL-запроса в виде одной строки заданного формата – с разделителями.

Ограничением данного способа является возможное превышение результатом выполнения `f_ExecSQLSelect_3` максимальной длины строковой переменной (около 32000 символов, если значение переменной не используется в формах, 2000 символов в остальных случаях). Чтобы этого избежать, внутри SQL-запроса (аргумента `f_ExecSQLSelect_3`) рекомендуется использовать агрегатные функции (`min`, `max` и т.д.). Данный подход обеспечивает получение в результате выполнения запроса единственного (а значит гарантированно короткого) значения для обработки в действии.

24.3.3.4 Другие случаи использования SQL-запросов

Использование SQL-запросов возможно также в отчетах для задания условий предварительного отбора объектов. Об использовании условий предварительного отбора прочтите в документации по Lotsia PDM PLUS, либо воспользуйтесь контекстной справкой. Для отчетов типа «Пользовательский» – SQL-запрос основной инструмент настройки отчета.

Еще один случай возможного применения SQL-запросов – формирование списков возможных значений переменных. Для переменных действий над объектами такой список формируется настройкой свойств переменных – см. п. 17.3.2.2, для переменных Workflow – выполнением действий над переменными – см. п. 24.2.1.

25 Создание рабочих окон для конечного пользователя. Табличный интерфейс

Для того чтобы создать удобное рабочее окно для конечного пользователя, часто используются средства Workflow. Обычно создается шаблон работы, состоящий только из начала и окончания. Используется только форма начального этапа работы. Стандартные кнопки и вкладки отключаются. Создаются пользовательские переменные, кнопки, которые обеспечивают запуск необходимых действий.

Использование форм массивов оправдано и необходимо, если требуется обеспечить конечному пользователю возможность работы непосредственно в табличной форме для ввода или изменения данных. Заполнение и обновление полей формы в большинстве случаев осуществляется выполнением SQL-запроса с использованием `f_EmbeddedSQLSelect()`, либо путем ввода данных пользователем. Для сохранения результата следует использовать действия над объектами, которые будут сохранять введенные в окне начала работы данные в создаваемые объекты и присваиваемые им атрибуты. Действия могут выполняться после завершения работы с окном – например, после нажатия кнопки, либо перед закрытием окна. Возможно также выполнение действий при изменении значений переменных в окне – например, после ввода значения в колонку массива действие устанавливает атрибут соответствующего объекта.

В случае если требуется только получение необходимой табличной информации без её изменения, либо допускается выполнять изменение информации во всплывающих окнах (формах действий) в использовании средств Workflow нет необходимости, достаточно использовать отчет, в различных областях отчета – использовать кнопки (см. раздел 20.3). Применение SQL-запроса в пользовательском отчете расширяет возможности получения из БД требуемых данных.

Запуск отчета или работы (открытие формы первого этапа) с помощью действия над объектом, выполняемого по нажатию кнопки в форме (либо по щелчку на объекте формы, либо из контекстного меню), делает эти инструменты достаточно удобными для конечного пользователя. В некоторых случаях удобным является использование вкладки с отчетом, основное ограничение использования таких вкладок – время выполнения отчета не должно быть слишком большим.

Если потомки при выполнении отчета находятся на первом уровне от текущего объекта, может оказаться достаточно использования атрибутивной формы одного из следующих типов: «Входимость», «Дочерние объекты», «Связанные объекты». Ограничение для данных форм – невозможность использования в них кнопок. Возможно использование контекстных меню с добавленными пунктами «Действие».

26 Приложения

26.1 Файл импорта типов объектов

Пример таблицы для импорта типов объектов (см. Таблица 3).

Таблица 3 Импортируемые типы объектов

Код (ID) типа объекта	Наименование	Вид	Мнемо	Сортировка	Пиктограмма
1000000	Флигель	P	ФЛГ	0	wing.ico
1000001	Блок	P	Блк	30	block.ico
1000002	Окно	P	Окн	20	window.ico
1000003	Дверь	P	Двр	10	d.ico

Код (ID) типа объекта – цифровой уникальный код типа объекта. Рекомендуется использовать длину кода не короче 6 знаков. Максимальная длина кода 15 знаков;

Тип объекта – наименование типа объекта, которое высвечивается в списке типов объектов системы Lotsia PDM;

Вид – символ, соответствующий виду типа объекта. Возможны следующие латинские символы: P – изделие, D – основные документы (ОКД), N – неосновные документы (НКД), R – прочие;

Мнемо – не более 4 символов, представляющих собой сокращенное название типа объекта;

Сортировка – ключ сортировки, определяющий положение объекта в дереве связей;

Пиктограмма – имя файла пиктограммы для данного типа объекта. Можно указывать путь к данной пиктограмме относительно папки с пользовательскими пиктограммами.

26.2 Файл импорта атрибутов

Пример таблицы для импорта атрибутов (см. Таблица 4).

Таблица 4 Импортируемые атрибуты

Код (ID) атрибута	Атрибут	Группа	Тип данных	Связь	Редактирование	Множественность	История	Формат
10000003	Срок сдачи	Дополнительная	T	O	E	N	Y	dd.mm.yyyy
10000004	Материал	Стандартная	S	O	W	Y	N	

Код (ID) атрибута – цифровой уникальный код атрибута. Рекомендуется использовать длину кода не короче 6 знаков. Максимальная длина кода 15 знаков;

Атрибут – наименование атрибута, которое высвечивается в списке атрибутов системы Lotsia PDM;

Группа – название группы, к которой относится данный атрибут. При отсутствии в базе данных указанной группы, она автоматически создается;

Тип данных – символ, соответствующий типу атрибута. Возможны следующие латинские символы: N – числовой, S – строковый, T – дата/время;

Связь – символ, соответствующий типу связи атрибута: с объектом или проектом. Возможны следующие латинские символы: O – объект, L – проект;

Редактирование – символ, соответствующий типу редактирования значения атрибута. Возможны следующие латинские символы: E – свободный, X – классификатор, W – список;

Множественность – символ, соответствующий множественности данного атрибута. Возможны варианты: Y – да, N – нет;

История – символ, включающий опцию хранения истории значений данного атрибута. Возможны варианты: Y – да, N – нет;

Формат – формат значения атрибута в соответствии с общепринятыми типами форматов. Это поле не обязательно для заполнения.

26.3 Файл импорта списка возможных значений атрибутов

Данный файл может потребоваться, например, при редактировании значений путем выбора из списка (пример – см. Таблица 5).

Таблица 5 Импортируемые значения атрибутов

Код (ID) атрибута	Атрибут	Значение
10000004	Материал	Дерево
10000004	Материал	Стекло
10000004	Материал	Пластмасса

26.4 Файл импорта типов связей

Пример таблицы для импорта типов связей (см. Таблица 6):

Таблица 6 Импортируемые типы связей

Код (ID) типа связи	Связь	Тип
440000001	Состав	T
440000002	Подобные объекты	E
440000003	Схема сборки	D

Возможны следующие латинские символы, соответствующие типу связи: T – подчиненная, E – горизонтальная равнозначная, D – горизонтальная направленная.

26.5 Файл импорта входимости типов объектов по типам связей

При импорте структуры возможно задание входимости для типов объектов. Для горизонтальных связей правила входимости не определяются (ограничений на связывание любых пар объектов горизонтальными связями нет).

Пример таблицы входимости типов объектов (Таблица 7).

Таблица 7 Импортируемые соотношения входимости типов объектов

ID типа объекта-родителя	Тип объекта-родителя	ID типа объекта-потомка	Тип объекта-потомка	ID связи	Связь	Множественность
1000000	Флигель	1000001	Блок	1	Состав	0
1000000	Флигель	1000002	Окно	1	Состав	1
1000000	Флигель	1000003	Дверь	1	Состав	1

При использовании единственного типа связи колонка «ID связи» может остаться незаполненной. Колонка «Множественность» может принимать цифровые значения 0 (множественная входимость) или 1 (единичная входимость).

26.6 Файл импорта соотношения тип объекта – атрибут

Пример таблицы импорта соотношения тип объекта – атрибут (Таблица 8).

Таблица 8 Импортируемые соотношения тип объекта – атрибут

Код (ID) типа объекта	Тип объекта	Код (ID) атрибута	Атрибут	Обязательность	Значение «по умолчанию»
1000000	Флигель	-30	Наименование	Y	ввести значение
1000001	Блок	-30	Наименование	Y	ввести значение
1000002	Окно	-30	Наименование	Y	ввести значение
1000003	Дверь	-30	Наименование	Y	ввести значение
1000001	Блок	-29	Количество	Y	0
1000002	Окно	-29	Количество	Y	0
1000003	Дверь	-29	Количество	Y	0
1000001	Блок	680515049701014	Габариты	N	
1000002	Окно	680515049701014	Габариты	Y	0
1000003	Дверь	680515049701014	Габариты	Y	0
1000000	Флигель	10000003	Срок сдачи	N	
1000000	Флигель	10000004	Материал	Y	ввести значение

В колонке «Обязательность» используются латинские буквы Y или N. Y – означает обязательность данного атрибута для данного типа объекта, N –необязательность.

26.7 Примерная последовательность создания и развертывания прикладной системы на основе Lotsia PDM PLUS

Последовательность ваших действий для создания прикладных решений на основе системы Lotsia PDM PLUS может быть примерно следующей:

- Разработайте модель данных.
- Выделите основные потоки документооборота, подлежащие автоматизации.
- Определите, какую СУБД вы будете использовать.
- Инсталлируйте Lotsia PDM PLUS (см. разделы «Порядок установки программы», «Установка программы», «Установка аппаратной защиты» Общей части руководств).
- Создайте пустую БД (см. разделы «Создание пустой базы данных», «Инициализация базы данных» Общей части руководств).

Если планируется использование нескольких баз данных и модуля репликации для обмена данными между ними, обязательно, в каждой базе данных установите различные номера филиалов.

- Зарегистрируйте в программе основные элементы структуры БД (см. раздел «Настройка структуры данных» Руководства по администрированию и настройке программы Lotsia PDM):
 - Типы объектов.
 - Атрибуты.
 - Типы связей.
 - Задайте допустимые атрибуты для каждого типа объекта.
 - Задайте входимость объектов.
- Создайте необходимый набор средств, обеспечивающих работу пользователей в системе.
- Пользовательские формы (см. раздел «Настройка экранных форм» Руководства по администрированию и настройке программы Lotsia PDM).
- Сохраненные запросы (см. раздел «Запросы» Руководства пользователя программы Lotsia PDM)
- Действия (см. раздел «Редактор действий» Руководства по администрированию и настройке программы Lotsia PDM).
- Отчеты (см. раздел «Настройка отчетов» Руководства пользователя программы Lotsia PDM).
- Шаблоны для экспорта отчетов (см. раздел «Шаблоны экспорта отчетов» Руководства пользователя программы Lotsia PDM).
- Создайте необходимые профили пользователей (см. раздел «Регистрация пользователей. Создание профилей пользователей» Общей части руководств).
- Настройте Главное меню и панель инструментов для каждого профиля (см. разделы «Настройка Главного меню», «Настройка Панели приложения», «Добавление пользовательского меню в профиль» Общей части руководств).

- Включите в профили соответствующие пользовательские формы и действия (см. разделы «Назначение профилю форм», «Назначение профилю действий» Общей части руководств).
- Создайте пользователей и группы пользователей, задайте приоритеты, привилегии (см. разделы «Создание пользователя», «Создание группы пользователей», «Задание приоритета пользователю (группе)», «Предоставление административных привилегий группе (пользователю)» Общей части руководств).
- Создайте защищенные хранилища (см. раздел «Архив документов» Руководства по администрированию и настройке программы Lotsia PDM).
- Создайте необходимые библиотеки, задайте для них группы пользователей (См. раздел «Управление библиотеками документов» Руководства пользователя программы Lotsia PDM).
- Зарегистрируйте типы документов (см. раздел «Регистрация типов документов» Руководства по администрированию и настройке программы Lotsia PDM).
- Зарегистрируйте приложения для обработки документов и методы их запуска (см. раздел «Регистрация приложения, метода (-ов) его запуска и совместимых приложений» Руководства по администрированию и настройке программы Lotsia PDM).
- Проведите обучение пользователей.
- Организуйте первичное наполнение базы данных (см. разделы «Импорт объектов», «Расширенный импорт объектов», «Импорт документов» Руководства пользователя программы Lotsia PDM).
- Если требуется, создайте типовые документы и настройте синхронизацию атрибутов с полями этих документов (см. раздел «Синхронизация атрибутов объекта с полями документов» Руководства пользователя программы Lotsia PDM).
- Если это необходимо, создайте бизнес-правила, представления проектов (см. раздел «Бизнес-правила», Руководства по администрированию и настройке программы Lotsia PDM, раздел «Вид проекта (представление)» Руководства пользователя программы Lotsia PDM).
- Создайте Глобальные адресные книги, контакты, сопоставьте контакты пользователям, определите права контактов (см. раздел «Настройка глобальной адресной книги» Руководства по администрированию программы Lotsia Workflow).
- Определите и задайте, при необходимости, параметры групп пользователей (см. раздел «Создание и редактирование параметров групп пользователей» Общей части руководств).
- Создайте шаблоны типовых процессов (см. раздел «Создание и редактирование шаблона (карты) предопределенного процесса (работы)» Руководства пользователя программы Lotsia Workflow).