

# **БАЗОВЫЙ КУРС ПО 3SL CRADLE**

Авторы учебного пособия: Мадорская Ю.М., Тимофеев А.Н.

# СОДЕРЖАНИЕ

<b>Содержание . . . . .</b>	<b>2</b>
<b>Введение . . . . .</b>	<b>7</b>
Системное проектирование	8
Область применения 3SL Cradle .....	8
Современный уровень технологического развития компаний .....	16
Системная инженерия, системный подход и системное мышление .....	20
Поддержка принципов системного подхода в 3SL Cradle .....	21
Общая идея применяемого процесса проектирования .....	26
Информационная модель учебного проекта .....	29
Задачи и функции аналитика в учебном проекте .....	33
Шаги учебного проекта .....	41
3SL Cradle	43
Назначение.....	43
Поддерживаемые нотации моделирования.....	45
Архитектура.....	47
<b>Установка и подготовка работы . . . . .</b>	<b>50</b>
Как получить пробную версию 3SL Cradle?	51
Зарегистрироваться на сайте <a href="http://www.saturs.ru">www.saturs.ru</a> .....	51
Скачать дистрибутив .....	54
Установить Cradle .....	55
Проверить установку	57
Запустить DEMO-проект .....	57
Проверить работу DEMO-проекта, используя Панель фаз.....	58

<b>ТИПОВЫЕ ЗАДАЧИ . . . . .</b>	<b>67</b>
<b>Управление проектами Cradle</b>	<b>68</b>
Создать проект.....	68
Удалить проект.....	70
Структурировать проекты .....	71
Войти в проект .....	72
Загрузить схему проекта .....	74
<b>Загрузка исходных требований из спецификаций</b>	<b>75</b>
Понять принципы захвата документов.....	75
Загрузить с базовыми настройками Document Loader.....	80
Проверить загрузку исходных требований .....	90
Запустить отчет о содержании .....	96
Запустить отчет о покрытии .....	96
Отменить загрузку .....	97
Уточнить настройки загрузки и повторно загрузить документ .....	99
<b>Анализ и формализация исходных требований</b>	<b>110</b>
Познакомиться с моделью формализации исходных требований .....	110
Настроить представление для анализа исходных требований .....	112
Провести анализ, оценку и формализацию исходных требований .....	119
Изучить пример формализации .....	122
Организовать рабочее пространство для анализа исходных требований.....	124
Отразить ошибки в исходных требованиях .....	131
Зафиксировать вопросы по исходным требованиям .....	132
Изменить статус проанализированных исходных требований .....	140
Удалить или восстановить элементы.....	141
Разделить объемные требования на элементарные .....	143

Объединить дублирующиеся требования .....	150
Установить общие атрибуты для производных требований .....	152
Проставлять связи от уже созданных элементов .....	153
Зафиксировать конфликтующие требования .....	154
Реструктурировать формализованные требования .....	155
Завершить первичный анализ и формализацию исходных требований .....	156
<b>Оценка объема и сложности проекта</b>	<b>159</b>
Настроить метрики и опубликовать результаты расчетов .....	159
Запустить сохраненную метрику .....	171
Настроить панель фаз для быстрого вызова метрик .....	172
Настроить стартовую страницу для быстрого вызова метрик .....	174
Настроить панель индикаторов (KPI) .....	179
<b>Планирование и контроль проекта</b>	<b>190</b>
Три подхода к учету трудозатрат и управлению задачами .....	190
Настроить схему проекта .....	191
Подготовить очередь задач .....	193
Подготовить план работ .....	197
Использовать панель задач для отслеживания .....	203
Отчитываться по задачам в Workbench .....	204
Выгрузить задачи в MS Project .....	206
Провести планирование в MS Project .....	207
Обновить план в Workbench .....	209
Отчитываться по задачам .....	210
Настроить вычисляемое поле для определения длительности .....	211
Добавить метрики и показатели задач в KPI .....	212
<b>Взаимодействие с Заказчиком</b>	<b>216</b>

Настроить простой отчет для генерации повестки .....	216
Настроить корпоративный шаблон для генерации повестки .....	219
Опубликовать повестку рабочей встречи .....	230
Расширить шаблон для вывода связанных исходных требований .....	233
Настроить панель фаз для генерации повестки .....	235
Подготовиться к оценке области влияния.....	238
Провести встречу и зафиксировать результаты.....	239
Загрузить результаты встречи .....	241
Объединить ответы с вопросами.....	244
Оценить область влияния .....	246
Формализовать новые ИТ - результаты рабочей встречи .....	248
<b>Проектирование системы</b>	<b>250</b>
Понять задачи проектирования.....	250
Настроить матрицы трассировки.....	251
Вызвать сохраненные матрицы трассировки .....	259
Настроить диаграммы иерархий (HID).....	260
Дополнить панель фаз или стартовую страницу.....	265
Быстро посмотреть все связанные элементы .....	266
Повторно использовать требования из другого проекта .....	267
Разработать модели .....	268
Понять принципы моделирования в Cradle .....	268
<b>Специальные темы .. . . . .</b>	<b>274</b>
<b>Использование горячих клавиш</b>	<b>275</b>
Редактирование информации.....	275
Навигация по основному меню.....	275

Другие привязки .....	275
<b>Разработка Use Cases</b>	<b>277</b>
Немного о Use Cases.....	277
Технологии разработки Uses Cases в 3SL Cradle .....	278
<b>Смена парадигмы</b>	<b>281</b>

# **ВВЕДЕНИЕ**

# Системное проектирование

## Область применения 3SL Cradle

3SL Cradle - это инструмент, предназначенный для системного проектирования. Его можно отнести к классу CASE систем - Computer Aided Software Engineering с тем уточнением, что он создан для проектирования не только программного обеспечения, а для проектирования систем, независимо от их класса. То есть более правильно было бы дать следующую расшифровку - Computer Aided System Engineering.

Но можно ли создать инструмент, который подходил бы для проектирования систем любых классов? Ведь системы разных классов обладают совершенно разными свойствами, например, химическими, пространственными, электрическими, механическими.

Чтобы ответить на этот вопрос и понять область применения 3SL Cradle, посмотрим какие классы автоматизированных сред сейчас широко используются для проектирования и какие задачи они решают.

Первый, хорошо известный класс систем, - это CAD-системы - Computer Aided Design, системы автоматизированного проектирования пространственных свойств систем - геометрических форм. Создание чертежей - это одна из древнейших проектных дисциплин и одна из задач, которая была автоматизирована в первую очередь, пройдя развитие от электронного кульмана до трехмерного моделирования. Сегодня CAD-системы позволяют моделировать уже не только геометрические свойства объектов, но и визуальные свойства их материалов, такие как цвет, отражающие способности и др..

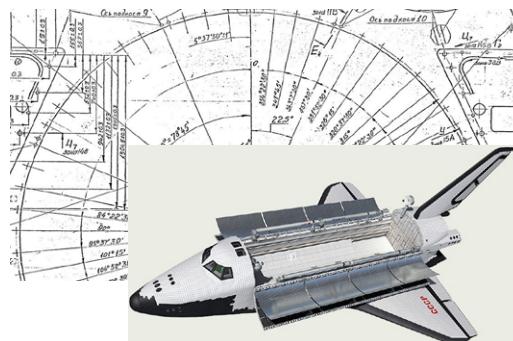
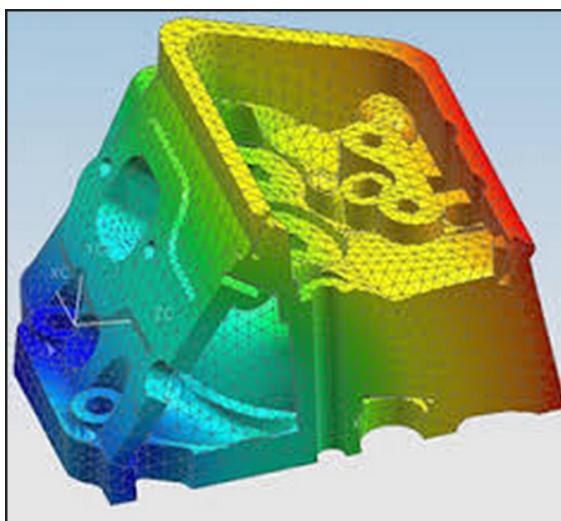


Рис. 2-1. Проектирование геометрических форм - от бумажных чертежей до трехмерного моделирования

Следующий класс систем - это CAE - Computer Aided Engineering. Системы этого класса предназначены для анализа и моделирования физических свойств объектов, таких как механические, температурные свойства, параметры движения и др., которые могут быть описаны и рассчитаны с использованием различных численных методов (конечных элементов и др.). Эти инструменты используются не только для описания и исследования характеристик объектов в статике, но и в динамике.

CAE-системы часто интегрируются с CAD-системами, т.к. многие физические свойства объектов зависят от их геометрических форм.



*Рис. 2-2. Анализ напряженности*

Еще один широко применяемый класс систем - EDA - Electronic Design Automation - системы автоматизации проектирования электронных приборов. Системы этого класса позволяют выполнять автоматическую расстановку элементов и разводку дорожек на печатных платах. В качестве входных данных EDA используют принципиальную схему устройства.

Эти расчеты также напрямую связаны с геометрией проектируемой системы.

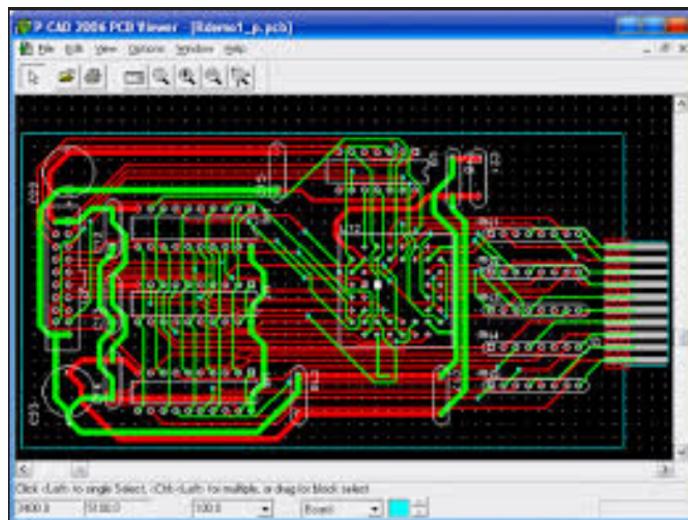


Рис. 2-3. Проектирование печатной платы

Для исследования других свойств систем, зависимости между которыми могут быть описаны с помощью математических уравнений, применяются специальные пакеты типа MatLab/MathCAD, позволяющие визуализировать поддающиеся формализации зависимости между различными свойствами системы.

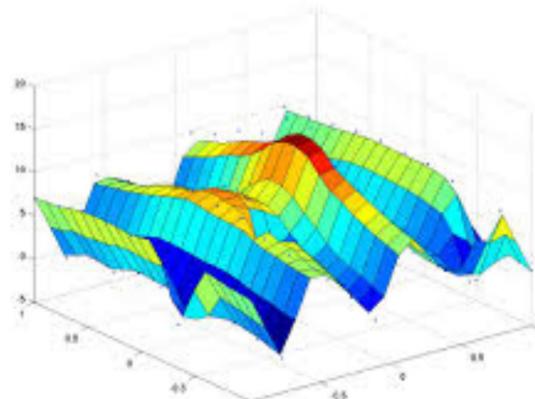


Рис. 2-4. Визуализация математических зависимостей

Важной особенностью рассмотренных четырех классов систем является то, что они помогают автоматически рассчитывать одни свойства системы на основании значений других свойств, используя заранее проработанную формализацию - алгоритмы расчета. При этом хорошо известно, что такие вычисления, используемые для оценки или проектирования системы, возможны для очень узкого круга хорошо формализуемых задач.

Эти задачи локализуются в разных компонентах системы, охватывая весьма

небольшой круг свойств, между которыми могут быть установлены математические зависимости. Эти зависимости не охватывают и принципиально не могут охватить все разнообразные свойства системы, которые должны быть заданы в ходе ее проектирования. Например, EDA система позволяет пересчитать размещение дорожки контакта при изменении электрической схемы, однако ни одна из рассмотренных выше систем не сможет показать как отразятся изменения в жесткости крыла на разводке печатной платы блока управления посадкой. Эти задачи лежат вне их «компетенции» и управление огромным объемом неформализуемых связей ложиться на плечи проектировщика, инженера-системотехника.

И пятый класс наиболее распространенных сред, без которых не обходится создания современных сложных систем - это RAD-среды - Rapid Application Development. Они используются при проектировании (разработке) компонентов программного обеспечения и позволяют визуально разрабатывать интерфейс системы, отражают структурные связи программных компонентов и облегчающие проверку работы программного компонента.

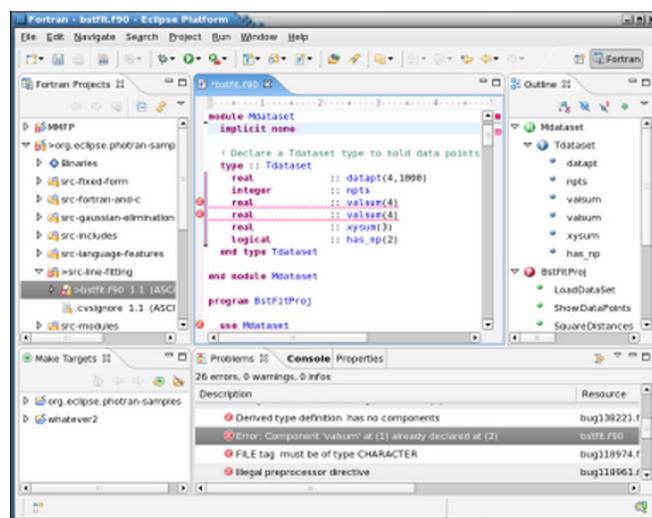


Рис. 2-5. Разработка программы

RAD-среды отличаются от CAD/CAE/EDA систем тем, что с их помощью компонент системы (программное обеспечение) не только проектируется, но и непосредственно создается. По сути эти среды ближе к станкам на которых вытаскивается продукт, нежели к системам проектирования.

Любой из перечисленных выше классов сред проектирования направлен на то, чтобы повысить качество и скорость проектирования, снизив затраты, путем:

- автоматизации рутинных задач, например, повторного использования ранее созданных элементов;
- автоматизации контроля ошибок проектирования за счет формализации правил контроля;
- визуализации моделируемых свойств для облегчения поиска тех ошибок, которые нельзя выявить автоматически, а также для оценки соответствия полученных результатов моделирования исходным требованиям;
- предсказания свойств компонентов системы на ранних стадиях проектирования на основе анализа моделей системы;
- автоматизации управления изменениями, для сокращения ошибок при внесении изменений и обеспечения возможности быстро проигрывать различные варианты «Если, то».

Без применения таких систем сегодня невозможно представить ни одно современное предприятие занимающееся проектированием, хотя надо отметить, что уровень освоения данных технологий может сильно различаться. Например, CAD систему могут использовать только как электронный кульман, не пользуясь мощью трехмерного моделирования.

Такая разница в применении технологий связана прежде всего с дефицитом хорошо подготовленных специалистов, которые готовы использовать «электронный микроскоп» по назначению, а не для забивания гвоздей. Но рынок - это сито естественного отбора и всегда выигрывают те, кто развивается быстрее, предоставляя продукт, удовлетворяющий поставленным задачам.

Скорость проектирования как самой системы, так и процессов ее выпуска и определяет скорость вывода продукта на рынок, а значит и конкурентоспособность компании. Поэтому успешные компании или те, которые рвутся в этот круг, вкладывают значительные ресурсы в развитие своей технологической базы и в повышение квалификации специалистов - основной силы проектных организаций.

Итак, перечисленные классы систем позволяют определить характеристики компонентов системы, непосредственно необходимые для производства системы. Это то, без чего просто невозможно будет выпустить современный продукт. Без чертежей невозможно выточить деталь, без схемы разводки невозможно

напечатать плату, без программного кода не заработает программное обеспечение.

Однако рынку (потенциальным пользователям, потребителям проектируемой системы) вовсе не интересны такие свойства продукта как, например, структура разводки печатной платы или напряженность боковины крышки сотового телефона, т.е. все то, что так тщательно проектируется в средах, рассмотренных выше.



*Рис. 2-6. Проектирование геометрии*

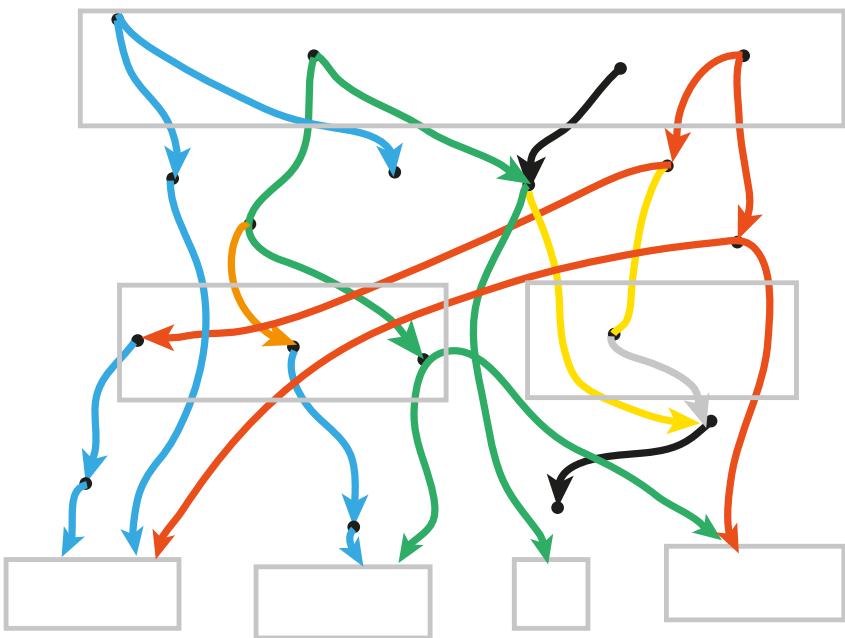
Потребителей интересуют совершенно другие свойства продуктов - свойства, которые чаще всего невозможно описать математически - это комплексные (эмурджентные) свойства системы, которые находят отражение в каждом проектируемом компоненте - эргономика, безопасность, срок службы, наконец функциональность.



*Рис. 2-7. Свойства системы, важные для пользователя*

Это свойства, которыми обладает система, но часто не обладает ни один из ее компонентов. Как, например, свойство самолета «Лететь по заданному курсу». Обладает ли отдельно взятое крыло самолета обладать таким свойством? Нет. Крыло самолета обладает такими свойствами как обтекаемость, вес и др., благодаря которым обеспечивается общее свойство самолета «Лететь по заданному курсу».

### От системных требований

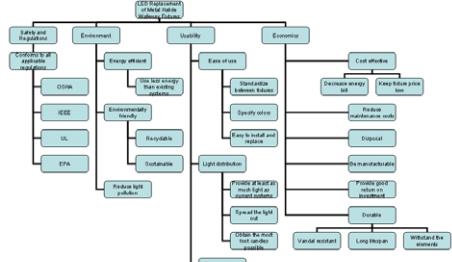


### К физическим и информационным свойствам элементарных компонентов

### Потребитель



### Системный инженер/аналитик/архитектор



### Инженер



Рис. 2-8. Декомпозиция общесистемных свойств до свойств конкретных компонентов при проектировании и их сложных взаимосвязи

Чтобы дойти от свойств системы, представляющих ценность для Заказчика и потребителя, до требований к каждому конкретному, элементарному компоненту, на основе которых будет выполнено детальное проектирование его физических или информационных свойств, необходимо провести огромную работу по проектированию - декомпозицию требований верхнего уровня, согласование всей сети требований, взаимосвязи между которыми далеко не однозначны. И несмотря на то, что связи между системными свойствами и свойствами компонентов, составляющих систему, нельзя описать математически, эти связи объективно существуют и являются предметом проектирования.

Рассмотренные выше CAD/CAE и другие инженерные системы позволяют управлять согласованностью лишь небольшого подмножества свойств, для которых возможно построение математических моделей, описывающих связи между ними. Большая же часть свойств проектируемой системы, а главное связей, не может быть описана математическими моделями. И именно эти свойства и связи являются тем самым kleem, связующими нитями, которые позволяют получить на выходе целостный востребованный продукт, а не просто набор отличных компонентов.

*Любая, идеально спроектированная часть системы идеальна лишь тогда, когда в соединении с остальными частями системы может обеспечить заданные системные свойства.*

Поэтому так важно автоматизировать и упростить работу системных архитекторов с описанием всех уровней требований, их декомпозицией, иерархическими и другими связями. Необходима автоматизация, которая аналогично предыдущим инструментам, позволила бы сократить объем рутинных операций по управлению требованиями и их связями и предоставила бы инструменты контроля непротиворечивости общей системы требований, позволяла бы снизить затраты на управление изменениями.

Это и есть область применения инструментов системной инженерии. Причем не ограниченная каким-либо классом проектируемой системы, т.к. этот слой проектирования необходим при проектировании любого класса систем.

## **Современный уровень технологического развития компаний**

На сегодня наиболее часто используемой системой автоматизированного проектирования системных свойств являются текстовые редакторы. Если сравнивать с проектированием геометрии, то по сути - это аналог электронного кульмана. Безусловно, это дает некоторый скачок производительности проектировщика - нет необходимости долго тереть резинкой бумагу, можно копировать повторно используемые элементы текста или быстро проводить замену.

Однако текстовый процессор находится даже ниже по уровню технологичности, чем электронный кульман, т.к. текст, в отличие от графической, пусть плоской модели, не позволяет контролировать ошибки проектирования и быстро оценивать объем изменений, что просто необходимо процессе проектирования - процессе итеративном по своей сути.

Таким образом, для управления наиболее важными свойствами системы, свойствами, которые определяют ее востребованность, жизнеспособность, конкурентоспособность, свойствами, которые интересуют Потребителя, на сегодня чаще всего используется наименее технологичный из всей цепочки инструмент, возлагая всю ответственность за достижение поставленных целей на системных аналитиков, архитекторов, лишая их столь необходимой технологической поддержки в процессе, от которого зависит успех всей компании.

Не странно ли? Действительно «с потребительской точки зрения» весьма странно, однако легко объяснимо. Если несколько ключевых факторов, которые определяют такое положение дел. О них ниже.

**Фактор 1. Прагматичная недальновидность или «короткие деньги».** В жестких условиях неопределенности люди чаще склонны выбирать альтернативы, которые приносят сиюминутный выигрыш, игнорируя дальнейшие перспективы развития.

Лучше всего этот подход иллюстрирует эта юмористическая зарисовка.

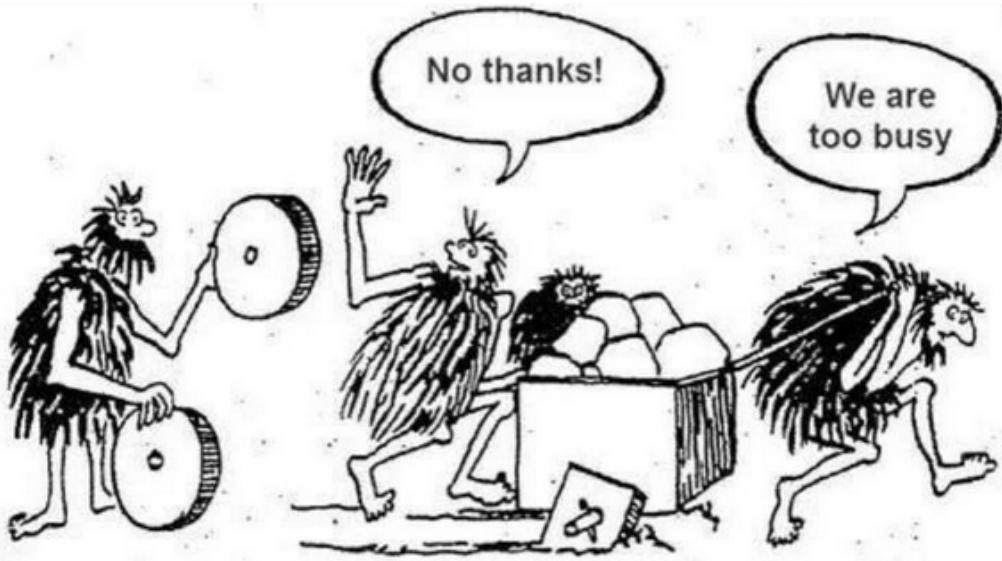


Рис. 2-9. «Нет спасибо, мы слишком заняты»

И хотя со стороны нам кажется, что предлагаемое специалистом слева решение очевидно лучше, это не столь очевидно тем, кто привык работать совершенно по-иному.

Для того, чтобы сменить текущую технологию работягам справа придется остановиться и потратить некоторое время. Более того, все рефлексы (управление) и мышцы (компоненты) уже отстроены под работу с квадратными колесами и переобучение также потребует времени и наконец, фактор определенности - на квадратных колесах они могут сегодня предсказать результат, а на технология на круглых колесах для них пока полна неопределенности.

Если же мы вообразим над ними того самого «прагматичного» менеджера, которому камни необходимы сегодня, то, наверное, более 50% компаний увидят близкую и родную картину.

Такая картина, безусловно, оправдана с точки зрения оперативного управления, однако лишает компанию возможности уверенно конкурировать в будущем, иногда не столь далеком, как кажется тем, кто сегодня перевозит камни на квадратных колесах, бороздя землю. Поскольку всегда найдется тот, кто имея сегодня более слабые позиции и находясь в поиске решений, которые ему помогут, с радостью схватится за круглое колесо и быстро выскочит вперед.

**Фактор 2. И дело опять в «математике».** Зависимости, между свойствами элементов системы, которые исследуются в системах типа CAD/CAE/EDA воз-

можно описать математически и использовать автоматические алгоритмы для быстрого их пересчета при внесении изменений, в то время как взаимодействие системных свойств, в которые укладываются и цели заинтересованных лиц и бюджет проекта, и требования к функциональности и безопасности системы и др., описать формулой невозможно, а потому, движимые математикой, CAD/CAE/EDA системы получают большее распространение, снимая затраты по проведению громоздких расчетов вручную.

**Фактор 3. Так исторически сложилось - проектирование «снизу вверх».** Такое положение дел оправдано и исторически, т.к. развитие систем шло «снизу» от более простых частей к более сложным многокомпонентным системам, прорабатывались отдельные свойства компонентов и лишь потом происходило их объединение в крупные программно-технические системы, обладающие сложным поведением. Поэтому и соответствующие инструменты имеют более широкое распространение.

**Фактор 4. Кадры решают все.** В дополнение ко всему, как уже говорилось ранее, переход на следующий уровень технологичности требует более высокой квалификации специалистов. И если в области геометрического проектирования, области, которая для нас наиболее проста, т.к. имеет легко воспринимаемые графические модели, переход от плоского к трехмерному моделированию осуществляется с трудом из-за недостатка подготовленных кадров, то еще большие проблемы в области системной инженерии, в которой визуализация страдает и необходимо развитое абстрактное мышление. Специалистов высокого уровня в этой области всегда было крайне мало, а тех, кто морально готов взбираться на следующую ступень технологического развития еще меньше, т.к. это требует систематизации и освоения новых приемов, прежде всего, мышления. А это часто дается с трудом, особенно «вошедшим в рабочую колею» специалистам.

Поэтому слабое развитие современных компаний в этой области легко объяснимо, однако также очевидны успехи тех, кому удается оседлать эту рабочую лошадку и эффективно использовать ее в своей проектной деятельности. Применение среды системного проектирования позволяет не на словах, а на практике сориентировать процесс создания и сопровождения систем на Заказчика и Потребителя - от сбора их исходных требований - до преобразования ко всем конечным свойствам частей продукта.

И потому сегодня во внедрении таких систем в процессы проектирования в наибольшей степени заинтересованы именно Заказчики и Потребители. Та-

ким заинтересованным заказчиком часто являются системные интеграторы, генеральные подрядчики. Это как раз яркий пример компаний, которых интересуют в большей степени системные свойства, которым необходимо получить заданные верхнеуровневые свойства системы - требуемую функциональность, надежность, эксплуатационные характеристики, собрав ее из разнородных компонентов, чьи свойства определены на более низком уровне.

При этом, чтобы успешно внедрить профессиональный инструментарий для системного проектирования, например, 3SL Cradle, необходимо освоить и адаптировать под свою организацию методы системной инженерии.

Поэтому в следующем разделе рассматриваются основные принципы системной инженерии, а их реализация на уровне конкретных процессов проектирования заложена в технологии выполнения учебного проекта, хотя и не высвечивается явным образом.

## **Системная инженерия, системный подход и системное мышление**

Системная инженерия - это аккумулированный опыт, приобретенный человечеством за тысячелетия создания различных систем. Системная инженерия включает методы, которые необходимы для создания «успешных» систем, то есть систем, соответствующих назначению и целям создания.

По сути, системная инженерия - это мультидисциплина, вбирающая в себя лучшие методы и практики, которые применимы в рамках того или иного этапа жизненного цикла системы, независимо от класса проектируемой системы.

И хотя внешне многие из этих методов могут сильно отличаться, они имеют общий скелет, что обусловлено ни чем иным, как характеристиками процесса мышления. Внешние же отличия связаны скорее с привычками и языковыми барьерами, выстраиваемыми различными школами.

На наш взгляд все методы системной инженерии базируются на системном подходе, ключевые принципы которого изложены ниже.

1. Исследуемый (проектируемый) объект рассматривается как система. То есть значительное внимание уделяется изучению внутрисистемных связей и связей с окружающим миром.
2. Организационно-техническая структура, осуществляющая проектирование, также является системой. Т.к. только система, может на выходе проектирования произвести систему.
3. Сам подход к исследованию и проектированию является систематизированным, организованным. То есть понятны связи между этапами и данными проектирования.

Реализация системного подхода невозможна без перехода к системному мышлению. Более подробно о критериях системного мышления вы можете прочитать в замечательной книге Ф. Капры “Паутина жизни”. Здесь мы приведем лишь несколько цитат, которые нам кажутся особенно важными.

*...В основе системного мышления лежат понятия связности, взаимоотношений и контекста...*

*...Другим ключевым критерием системного мышления служит способность перемещать фокус одного уровня системы на другой...*

*...Таким образом, системное мышление - это контекстуальное мышление...*

*...То, что мы называем частью, - это всего лишь паттерн в неделимой паутине взаимоотношений. Следовательно, переход от частей к целому можно также рассматривать как переход от объектов к взаимоотношениям...*

## **Поддержка принципов системного подхода в 3SL Cradle**

Процитированные свойства системного мышления позволяют легко сформулировать ключевые требования к среде проектирования, без которых невозможно реализовать системный подход:

- Функции работы со связями должны быть удобными и не трудоемкими.
- Должна быть обеспечена легкость переключения между уровнями декомпозиции описания системы.
- Среда должна позволять настраивать новые срезы и представления, не теряя фокус текущей задачи и не прерывая мыслительный процесс на длительное время.

3SL Cradle обладает всеми перечисленными фундаментальными свойствами, поэтому и позволяет значительно упростить работу системного инженера.

Cradle также поддерживает второй и третий принцип системного подхода.

В соответствии со вторым принципом, *организационно-техническая структура, осуществляющая проектирование, также должна являться системой*. Переводя на язык проектного управления это означает, что внутри команды должны существовать налаженные коммуникации, которые позволяют передавать весь объем информации о проектируемой системе, а значит и о ее связях без искажений.

Наша практика показывает, что связей, которые необходимо учитывать для создания успешных систем как правило в два-три раза больше, чем самих элементов. При документо-ориентированной технологии, отразить все эти связи просто невозможно, поэтому каждый раз читая техническое задание или любой другой проектный документ, участники проекта заново восстанавливают эти связи в своей голове. Часто почти случайным образом. И мы прекрасно знаем, что восстановленные образы практически никогда не совпадают, что порождает затем множество конфликтов и затрат на их согласование. Так как без согла-

сованной картины часто невозможно принять правильные проектные решения и двинуться дальше.

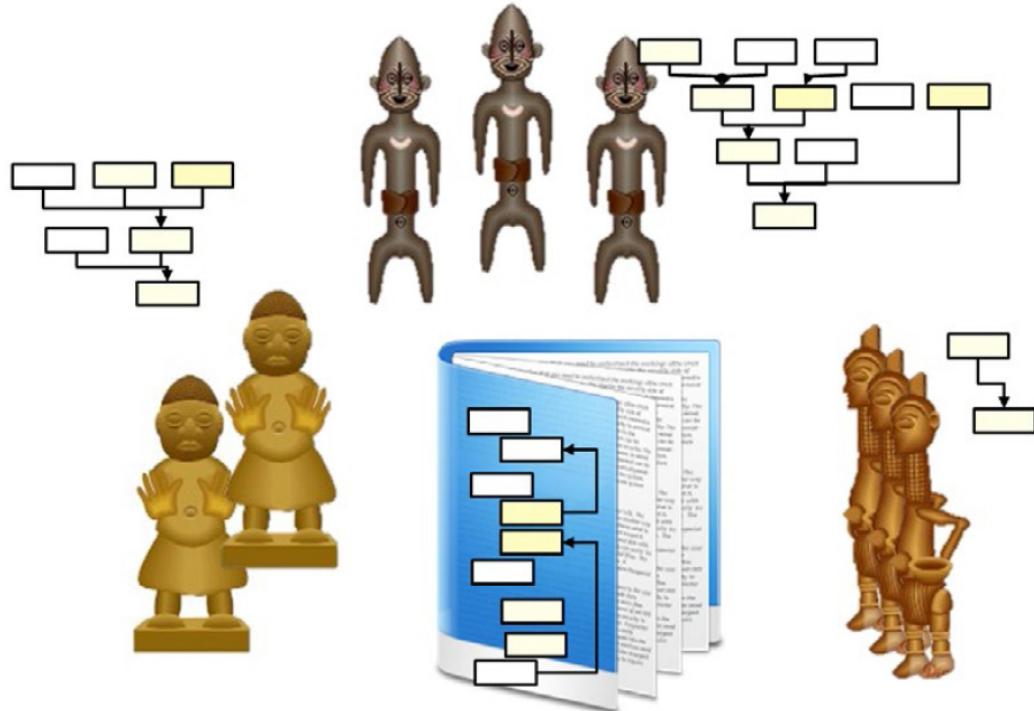


Рис. 2-4. Проблемы извлечения согласованных знаний из документов

Использование удобного инструмента, который позволяет без весомых затрат быстро фиксировать все необходимые связи помогает избежать таких проблем и превратить команду в целостную систему. Это также позволяет очень быстро отвечать на ежедневные вопросы по проекту и генерировать согласованные между собой документы, тратя минимум времени на управление изменениями. Примеры таких вопросов приведены на иллюстрации ниже.



Рис. 2-5. Подход к проектированию с использованием БД позволяет быстро отвечать на ежедневные вопросы и генерировать согласованные документы

Рассмотрим теперь, что стоит за третьим принципом «подход к исследованию и проектированию должен быть систематизированным, организованным». На практике это означает, что команда проекта должна хорошо представлять сами этапы проектирования и какие в рамках каждого этапа создаются проектные данные, например: истории пользователей, варианты использования, функциональные требования, структура систем и т.д.

Более того, чтобы обеспечить сквозную прослеживаемость - трассируемость требований, необходимо четко описать допустимые связи между этими проектными данными, их направление и семантику (смысл). Это называется моделью трассировки.

Чем здесь поможет Cradle? Он поможет тем, что будет форсировать создание именно такой трассируемой технологии проектирования и взаимодействия в команде. Почему? Просто потому, что по-другому работать в этом инструменте будет сложно. Безусловно, хаос можно создать в любой базе данных, но в отличие от документов хаос в базе данных сразу же бросается в глаза и быстро надоедает.

Таким образом, Cradle помогает реализовать на практике все три базовых принципа системного подхода. Он упрощает работу с системными связями, представляет общую среду для создания правильных коммуникаций в команде и форсирует систематизацию используемых методов проектирования.

Но необходимо понимать, что Cradle при этом вовсе не задает конкретные этапы и данные проектирования. Этот выбор должна сделать сама проектная команда с учетом своего опыта и принятого рабочего словаря. Однако важно помнить, что в ходе переложения ваших традиций на новый инструмент придется задать себе ряд важных вопросов, которые возможно приведут к пересмотру понятий и стоящих за ними процессов.

Построение такой выверенной технологии проектирования задача не одного дня и для того, чтобы упростить старт, можно начать с некоторого технологического прототипа и опробовать его на пилотном проекте. Далее скорректировать свое представление и настройки системы по мере необходимости. Приятная новость заключается в том, что при этом не надо будет что-либо программировать, все настройки выполняются с помощью простых форм.

Еще одно важное свойство Cradle - он не фиксирует жестко ваш поток работ по проектированию. Это крайне важно, поскольку процесс проектирования нелинеен и итеративен. В любой момент времени вы сможете ввести новое или изменить ранее созданное требование или проектное решение, которое родилось у вас в процессе взаимодействия с Заказчиком, командой или при анализе проекта системы в Cradle.

Действительно, никто и никогда в реальных проектах не выполняет в чистом виде проектирование сверху вниз (от общего к деталям) или снизу вверх (от деталей к системе в целом). Картина системы приобретает завершенный вид постепенно, при этом ее контуры могут прорисовываться сразу на нескольких уровнях. Это означает, что вы можете начать как с определения целей создания и назначения системы, так и с отдельных штрихов, описывающих, например, способы шифрования персональных данных, т.е. с любого требования нижнего уровня.

Важно, что с помощью удобных запросов и представлений вы всегда сможете проконтролировать, что у вас «голова и тело» системы не разъехались и конечные проектные решения полностью соответствуют требованиям верхнего уровня, т.е. требования Заказчика, пользователей системы полностью учтены при проектировании.

Чтобы упростить вам старт, мы разработали простейшую начальную технологию, включающие наиболее общие понятия, используемые при проектировании, такие как функциональные и нефункциональные требования, компоненты, а также исходные требования Заказчика.

Эта технология позволит вам освоиться в Cradle, а после перейти к обсуждению того, что именно вас не устраивает, корректировке исходной модели и доведению ее до соответствия корпоративным или личным задачам.

## Общая идея применяемого процесса проектирования

Важный аспект системного мышления - понимание контекста. Это позволяет правильно интерпретировать данные каждого следующего уровня проектирования, а также вовремя выявлять ограничения, накладываемые на решения. Поэтому выбранный процесс системного проектирования дает возможность уделять должное внимание анализу контекстных связей.

*Важно помнить, что каждый раз принимая проектные решения, вы “изменяете ландшафт” и создаете новый контекст на всех уровнях.*

При проектировании систем актуальна следующая картина вложенности контекстов:

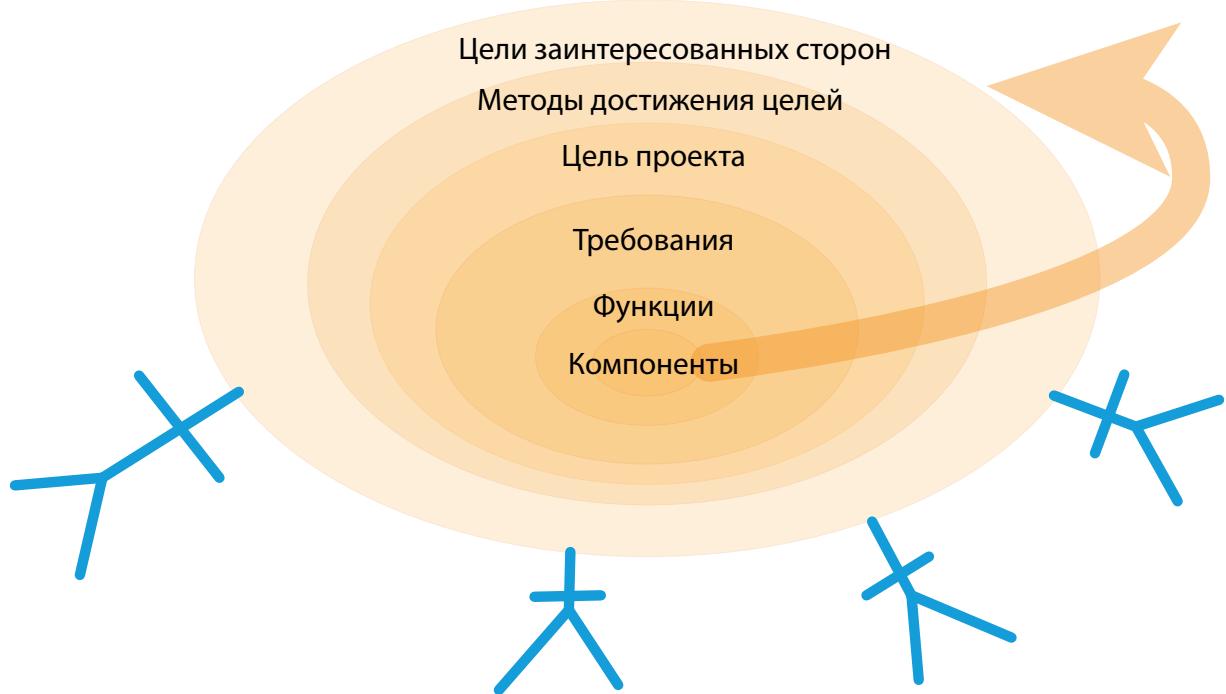


Рис. 2-6. Вложенность контекстов при проектировании

Любой процесс проектирования итеративен. Итерация за итерацией проводится уточнение образа будущей системы, прочерчивание границ и поиск решений, удовлетворяющих сформулированным требованиям. Исходные границы системы определяются уже тогда, когда вы очерчиваете круг заинтересованных сторон.

*Выделяя конкретных людей, которые будут «поставлять» исходные требования, вы тем самым определяете свойства будущей системы.*

Работая с заинтересованными лицами, необходимо выстроить систему целей, которые будут согласованы и будут иметь пересечение в области цели проекта. Если вам не удастся выстроить систему заинтересованных лиц и согласовать их цели, то требования следующего уровня согласовать между собой будет невозможно, т.к. не согласован их контекст. К сожалению, из-за пропуска этого этапа об этом часто узнают при сдаче проекта, который «внезапно» оказывается неуспешным.

Полезным приемом является определение границ системы не только в терминах «какой должна быть система», но и в терминах, «какой быть не должна». Границы системы - это те же требования к системе.

Цели и требования вместе образуют постановку задачи проектирования.

На основе целей, критериев их достижения и требований, проводится обобщение и поиск множества функций системы, которые позволили бы реализовать заданное множество требований. Например, если анализ исследуемых процессов показывает, что есть такие процессы как “Согласование договора”, “Согласование технического задания”, то резонно предположить, что в системе можно реализовать общую параметризованную функцию “Согласование документа” и это будет экономичнее, чем реализовывать несколько разных функций.

Таким образом, важный процесс, который должен происходить при переходе с одного уровня проектирования на другой - это обобщение и поиск (по-возможности) универсальных решений, устойчивых паттернов, наложение которых и формирует выявленное многообразие ситуаций.

Ошибки проектирования здесь как правило связаны с тем, что такое обобщение не производится и тогда рождается избыточность, которой впоследствии сложно управлять. Или обобщение производится на неполных данных, т.е. там где его быть не должно, и тогда рождаются усеченные малопригодные решения.

Вот так, переходя с одного уровня абстракции на другой, не забывая про контекст, проводится декомпозиция системы с целью дойти до точной спецификации отдельных компонентов.

Учет и фиксация связей, в ходе такой декомпозиции, т.е. *обеспечение трас-*

*сериемости требований*, способствуют прежде всего сохранению контекста при переходе на следующий уровень, т.е. возможности применять системное мышление, а это значительно повышает вероятность того, что спроектированные части будут обладать теми системными свойствами, которые были указаны в требованиях верхнего уровня.

Это общая идея процесса проектирования, основанного на системном мышлении. При всем многообразии наименований и флагов, выброшенных в области методологий проектирования программно-технических систем - варианты использования, бизнес-процессы, прецеденты, истории пользователей, бизнес-правила и т.д. и т.п., **базовых категорий**, с помощью которых может быть описан этот процесс не так много. Эти аспекты и их связи мы отразили в виде минимальной модели трассировки, которую вы найдете в следующем разделе.

## Информационная модель учебного проекта

В рамках этого учебного пособия и соответствующего ему проекта мы будем использовать упрощенную минимальную модель трассировки, которая обладает рядом полезных свойств. Выгоду от ее использования лучше всего оценить через практику.

Модель проиллюстрирована ниже и содержит четыре основных категории, которые используются для разработки и управления требованиями. Хотя Cradle позволяет создавать типы элементов на русском языке, в этом примере, чтобы легко отличать технологические настройки системы от ее содержания мы выбрали обозначение на английском языке.

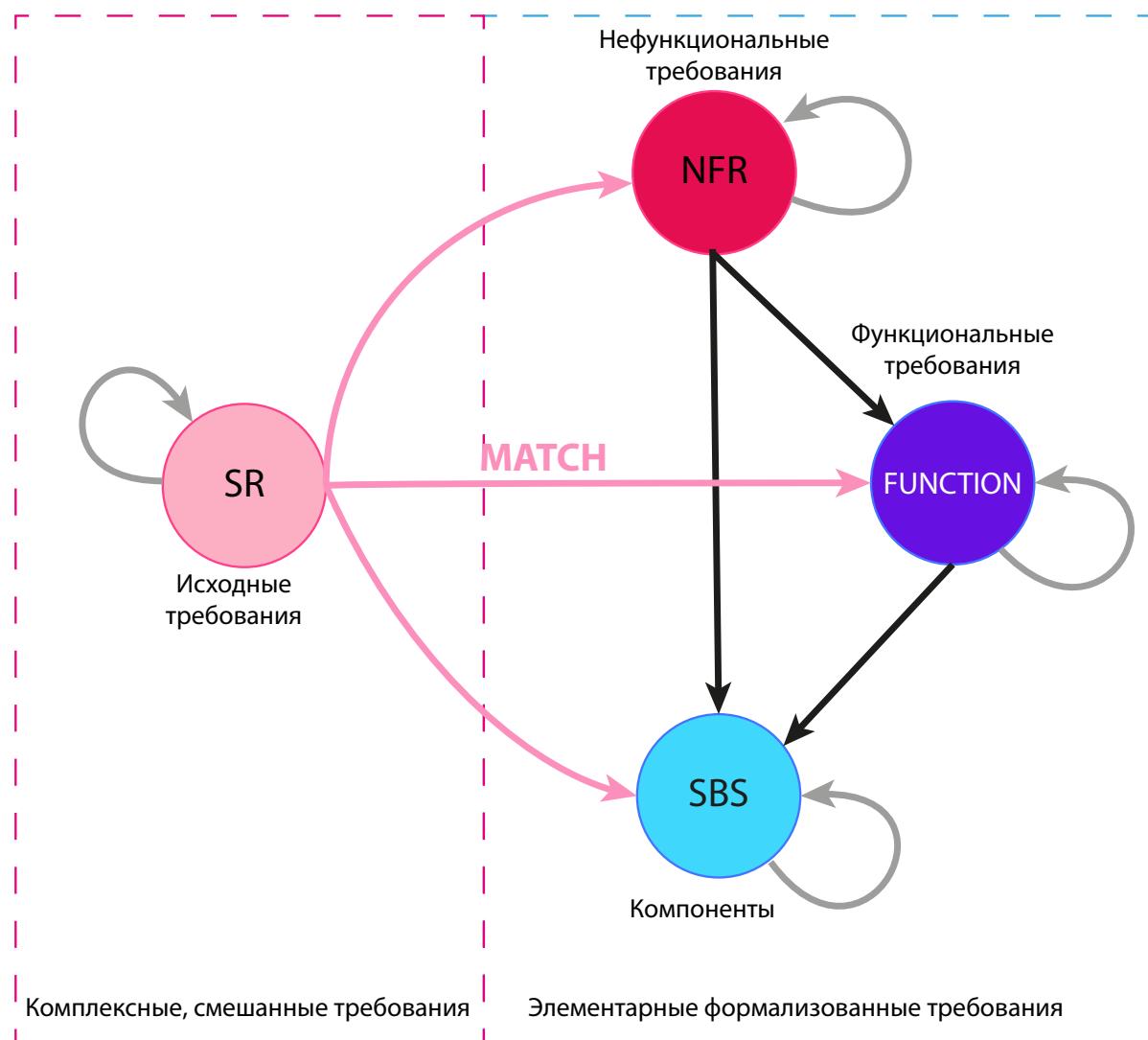
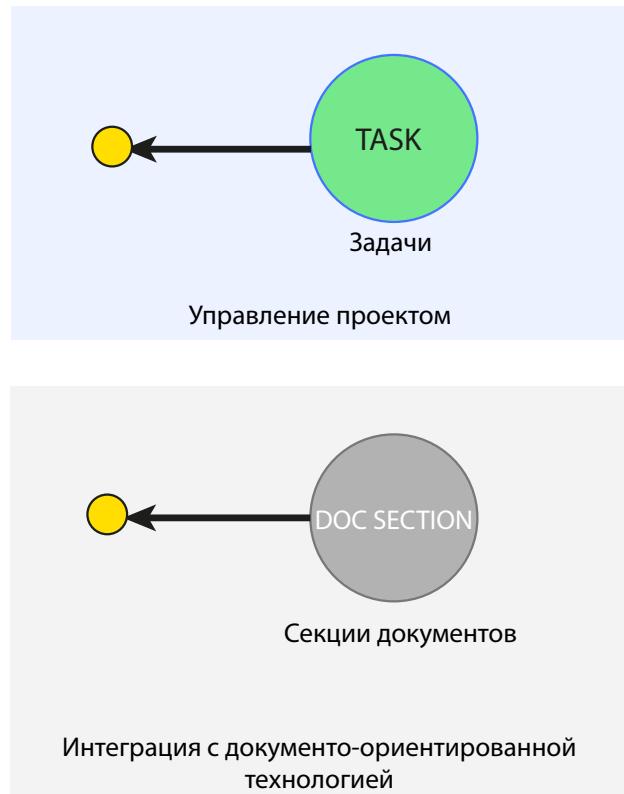


Рис. 2-7. Ядро модели трассировки учебного проекта

● - любой тип элемента



**CONFLICT**

Рис. 2-8. Технологическая часть модели трассировки учебного проекта

Рассмотрим подробнее элементы этой модели трассировки.

**SR (Source Requirements) - Исходные требования.** В эту «корзинку» складывается все, что получено извне, часто это плохо формализованные требования. В SR мы будем загружать документы, полученные от Заказчика, результаты проведения совещаний, записи бесед с Заказчиком - это исходные, необработанные проектные данные. Элементы SR могут содержать одно или много требований и как правило содержат ошибки типа неоднозначность.

Исходные требования (SR) могут быть структурированы любым удобным для вас образом, для этого используется связь SR-SR, отраженная на схеме се-рым цветом.

В этом стратовой модели трассировки мы не стали давать имена всем связям, что тоже допустимо в Cradle.

**FUNCTION** - это **функциональные требования** к системе. По сути, это уже результат проектирования. Каждый элемент этого типа может включать только одну функцию. Если необходимо показать, что функция состоит из других подфункций, то подфункции создаются как отдельные элементы этого же типа и связываются с главной функцией. Уровень вложенности не ограничен.

Если в исходных требованиях заказчика (SR), вы увидели явное указание на необходимость конкретной функции или из них это со всей очевидностью следует, то для обеспечения трассируемости, необходимо поставить связь типа MATCH от этих исходных требований к созданной функции. Этот тип связи «многие-ко-многим», т.к. из нескольких SR может следовать необходимость некоторой функции, а также из одного SR может следовать необходимость нескольких функций.

С помощью неименованной связи FUNCTION-FUNCTION, отображенной серым цветом, мы можем создавать любые иерархии и классификации функций, например, по роду выполняемых ими задач - Печать, Отправка оповещений и др.

**SBS** (System Breakdown Structure) - **Структурная Декомпозиция Системы** еще один стандартный термин в области системной инженерии. Элементы этого типа будут использоваться для отражения компонентного состава системы, включая компоненты программного обеспечения, компоненты «железа», а также роли.

Компоненты выполняют функции системы. Можно также сказать, что функции системы реализуются компонентами. Между функциями (FUNCTION) и компонентами (SBS) могут существовать отношения типа «многие-ко-многим». Для отражения этих связей также используется простая неименованная связь, отраженная на диаграмме черным цветом.

**NFR** (Non Functional Requirements) - **нефункциональные требования** - последний тип элементов, используемый для описания требований к системе, но первый по значимости. Сюда относятся все требования, не являющиеся функциями системы. Например, цели заинтересованных лиц, цели и назначение системы, бизнес-правила, условия эксплуатации, условия выполнения конкретных функций, требования к информационной модели системы и любые

другие свойства системы, за исключением ее функций, которые отражаются в FUNCTION и структуры, которая отражается в SBS.

Для управления проектом введен элемент **TASK**, который позволяет хранить данные о задачах проекта и их связях с требованиями любого уровня и типа.

**DOC SECTION** - исключительно технический элемент, необходимый для управления структурой и наименованиями разделов публикуемой из проекта документации, в данном случае - Технического задания.

Чтобы получить представление о других возможных моделях трассировки, рекомендуем познакомиться со статьей «Зоопарк трассировки»

## **Задачи и функции аналитика в учебном проекте**

В рамках данного учебного проекта мы будем разрабатывать требования к системе и выполнять ее верхнеуровневое проектирование на основе исходных данных от Заказчика - исходного документа с требованиями. На выходе мы получим техническое задание, которое описывает функции системы и требования к ним.

То, как мы будем проводить анализ и формализацию исходных данных, их оценку и дальнешее проектирование будет определяться методологией проектирования, основанной на рассмотренной в предыдущем разделе модели трасировки и схемой сборки документов.

Собирать требования и проектировать систему можно бесконечно, вплоть до самых атомов. Когда остановится - определяется требуемой степенью детализации проектирования. Этот показатель определяется для каждого конкретного проекта и в этом учебном примере мы не будем на нем останавливаться, оставляя его на ваш выбор. Вместе с ходом книги вы можете выполнить проектирование на верхнем уровне, а затем, уже освоив все базовые операции, продолжить самостоятельно и выполнить детальное проектирование.

Хотя в нашем учебном проекте все действия выполняет роль «Аналитик», в реальном процессе какая-то часть функций возможно будет делегирована руководителю проекта, другая - бизнес-аналитику, третья - системному архитектору. Но если аналитик понимает все смежные аспекты проекта, то проект только выигрывает от этого. Более того, столь мелкое разделение функций по ролям существует, как правило, только в очень крупных компаниях. В более компактных коллективах это бывает крайне редко.

В любом случае распределение ролей должно выбираться под конкретную команду, в зависимости от сильных и слабых сторон каждого ее участника.

В данном учебном проекте аналитик будет выполнять следующие функции:

- управлять проектом;
- взаимодействовать с заказчиком и фиксировать исходные требования;
- анализировать исходные требования и проектировать систему;
- тестировать проектные решения;

- выпускать техническое задание.

Все сказанное выше проиллюстрировано на контекстной диаграмме, которая раскрывается далее на следующей странице.

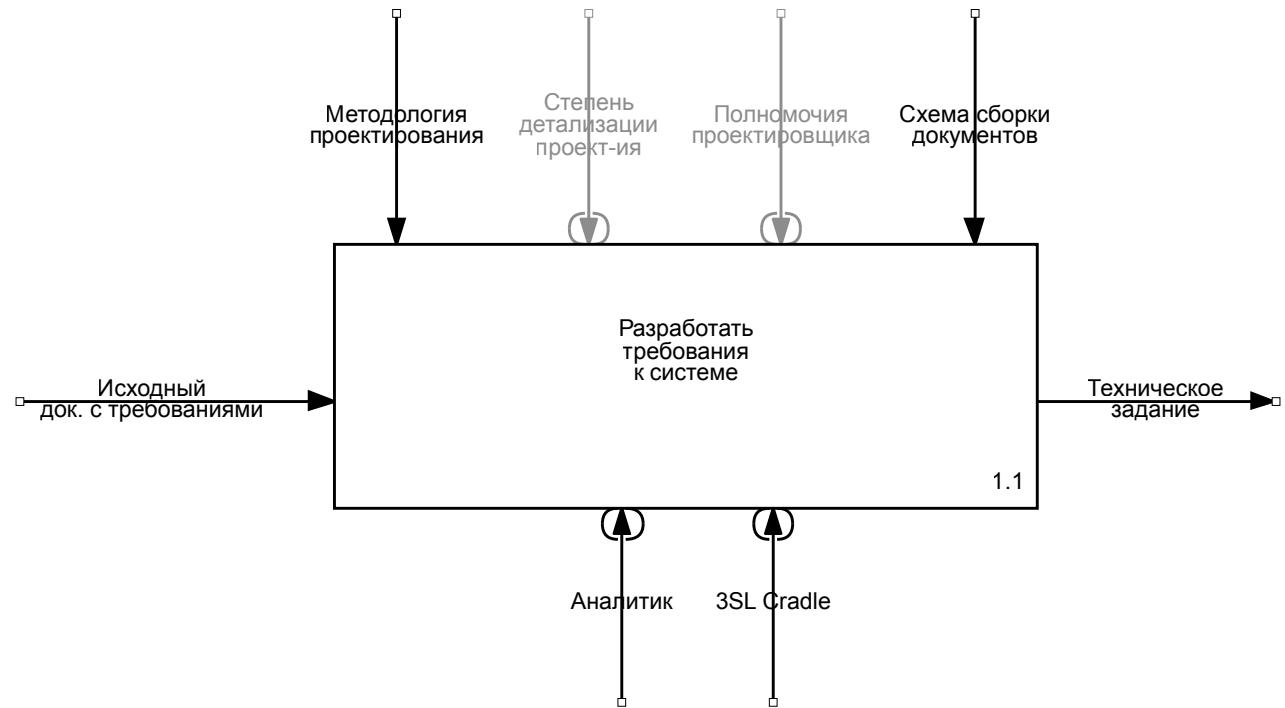


Рис. 2-9. Диаграмма A0 функциональной модели учебного проекта

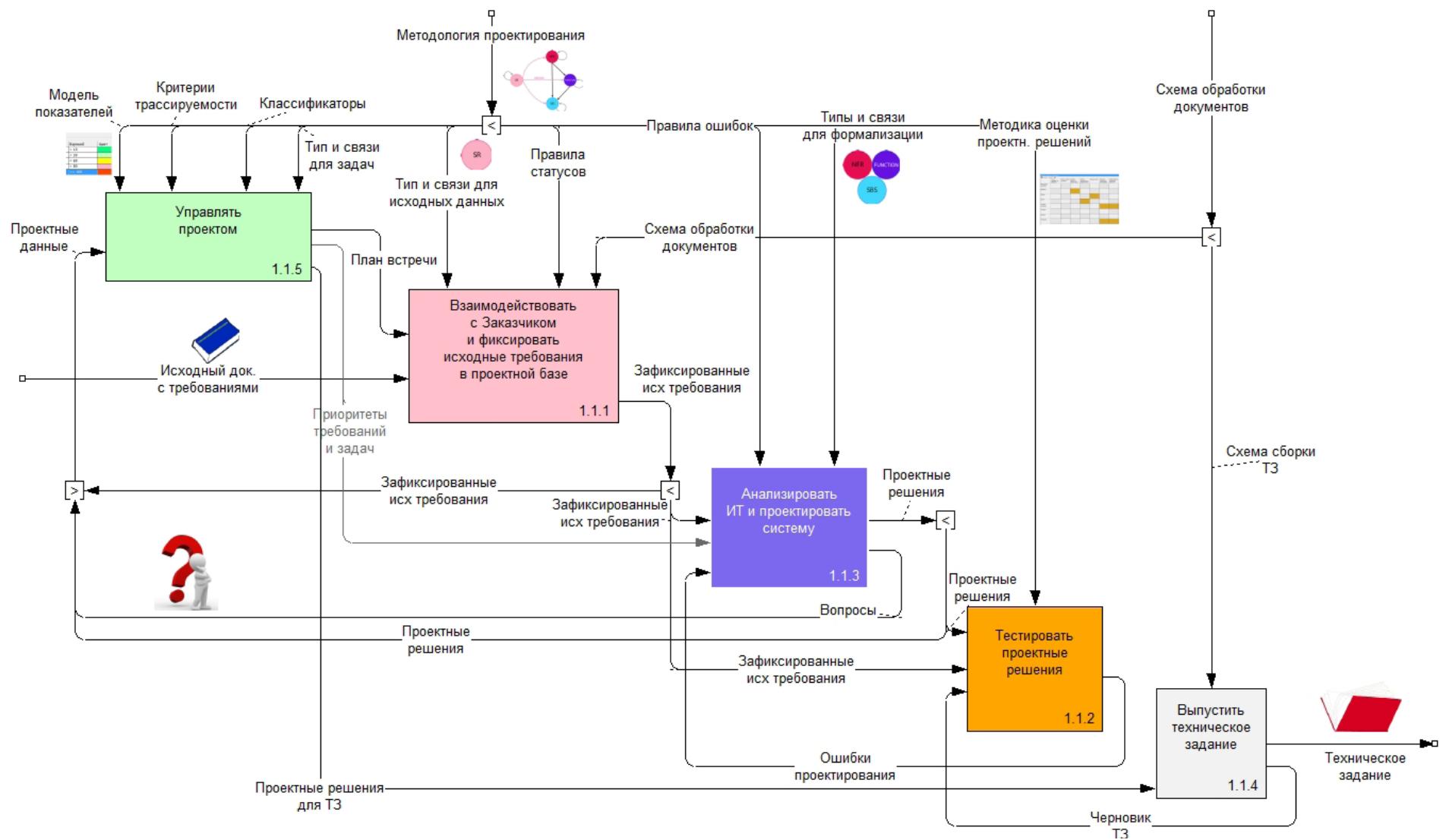


Рис. 2-10. Диаграмма A1 функциональной модели учебного проекта

Обратите внимание, что диаграмма А1 не отражает последовательность выполнения эти функций, не отражает процесс, а только информационные связи между задачами, которые необходимо будет решать. Цепочка из этих пяти функций может быть пройдена многократно, при каждом проходе мы можем уточнять и детализировать наши проектные решения. Длительность этих итераций вы можете определить для себя самостоятельно.

Проектирование по данной схеме может осуществляться как сверху вниз - от общего к частному (нисходящее проектирование), так и снизу вверх - от частного к общему (восходящее проектирование).

Реальность же такова, что проектирование, как правило, идет сразу в двух направлениях, а за счет постоянного учета связей между всеми требованиями и проектными решениями, его можно было бы в шутку назвать «сходящимся» проектированием. Что означает, что мы всегда будем проверять насколько согласованы решения на всех уровнях декомпозиции.

В качестве примеров использования Cradle для осуществления функций по управлению проектами мы посмотрим как его настроить и использовать для оценки неопределенности (рисков проекта), оценки объема изменений, оценки хода работ. Также посмотрим интеграцию с MS Project для управления связанным с нашим проектом планом работ и его задачами. Для этого нам понадобится ввести различные классификаторы, которые отражены на «зеленой» схеме.

Элементы на «зеленой» схеме, выделенные серым цветом часто используются в реальных проектах, но мы не будем описывать их настройку в этом проекте. Вы сможете сделать это самостоятельно, по аналогии с настройкой других классификаторов в качестве «домашнего задания».

Чтобы помочь аналитику взаимодействовать с заказчиком и фиксировать исходные требования (**SR**) мы рассмотрим возможности Cradle по загрузке исходных требований из документов. Также научимся публиковать симпатичную повестку встречи в корпоративном формате для формального взаимодействия с заказчиком и загружать обратно в базу, принятые на совещании решения.

Все поступившие исходные требования (**SR**) аналитику необходимо оценивать, чтобы вовремя корректировать оценку трудозатрат и ход работ, поэтому мы посмотрим на простейшей схеме классификации ошибок как Cradle поможет выполнять эту задачу и всегда иметь под рукой текущую оценку качества требований.

На основе исходных требований, которые не содержат ошибок и могут быть формализованы будут созданы (в соответствии с моделью трассировки, описанной в разделе выше) три класса элементов - нефункциональные требования (**NFR**), функции (**FUNCTION**) и компоненты системы (**SBS**). Обязательная задача аналитика в ходе формализации отмечать все логические связи между ними. Здесь мы покажем как это проще всего делать в Cradle и как контролировать процесс, чтобы не создать лишних связей и не забыть нужные.

Как только минимальный набор требований будет сформирован можно будет начать распределять их по разделам ТЗ (**DOC SECTION**) и пробовать выпускать техническое задание. Практика показывает, что сборка в документ - как в еще одно представление для контроля, позволяет увидеть дополнительный срез системы и, как правило, откорректировать проектную базу.

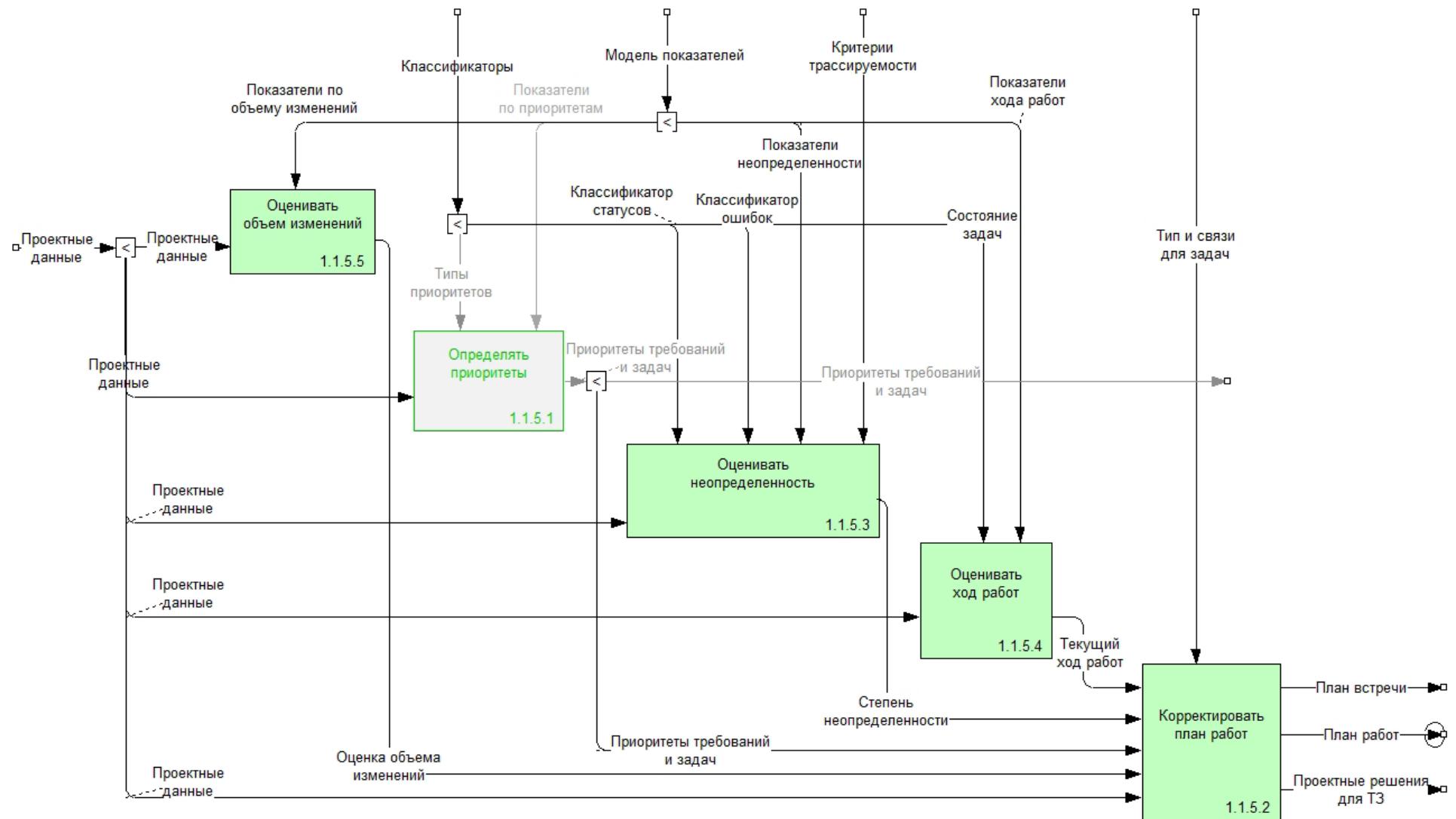


Рис. 2-11. Детализация функционального блока «Управлять проектом»

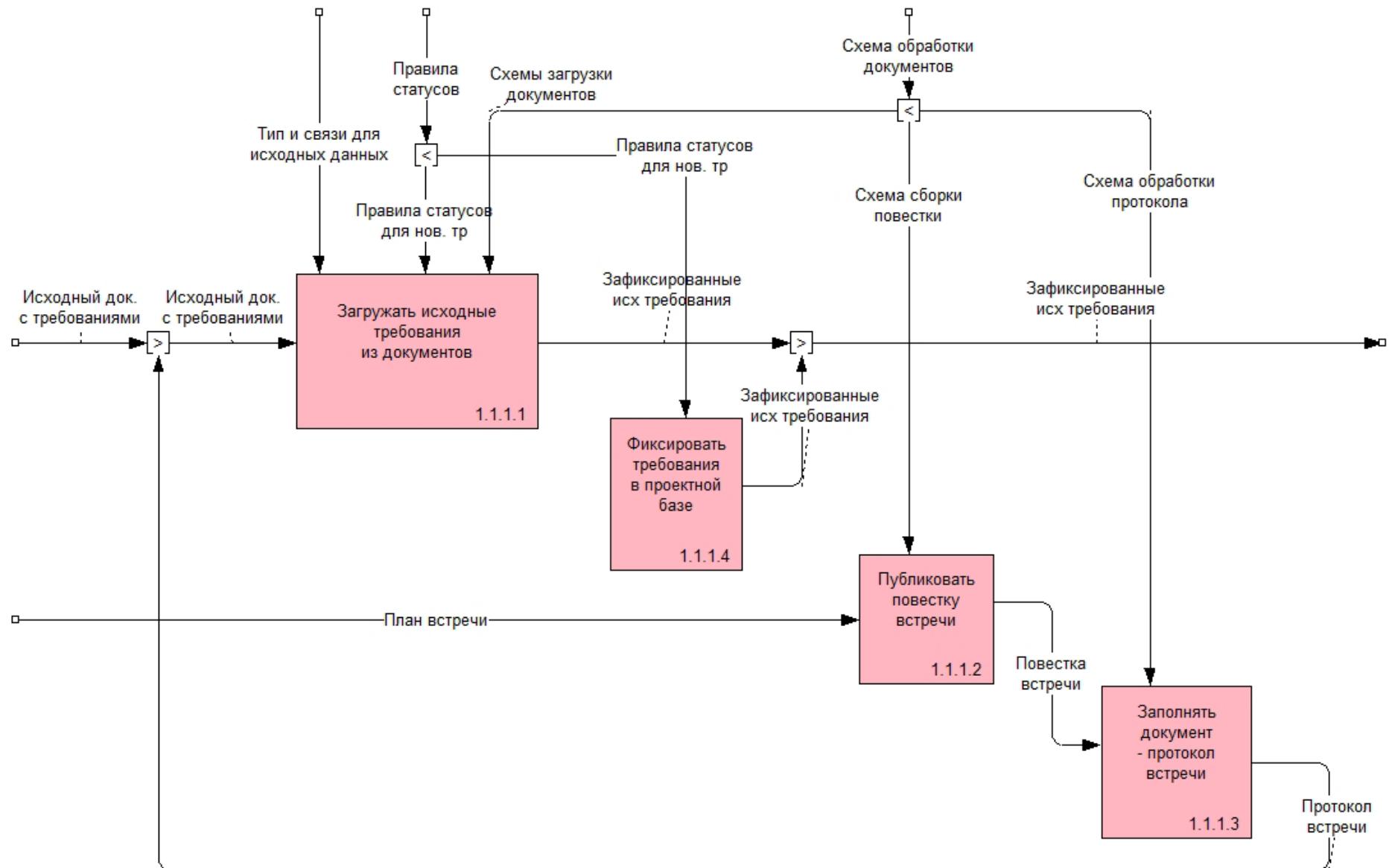


Рис. 2-12. Детализация функционального блока «Взаимодействовать с Заказчиком ...»

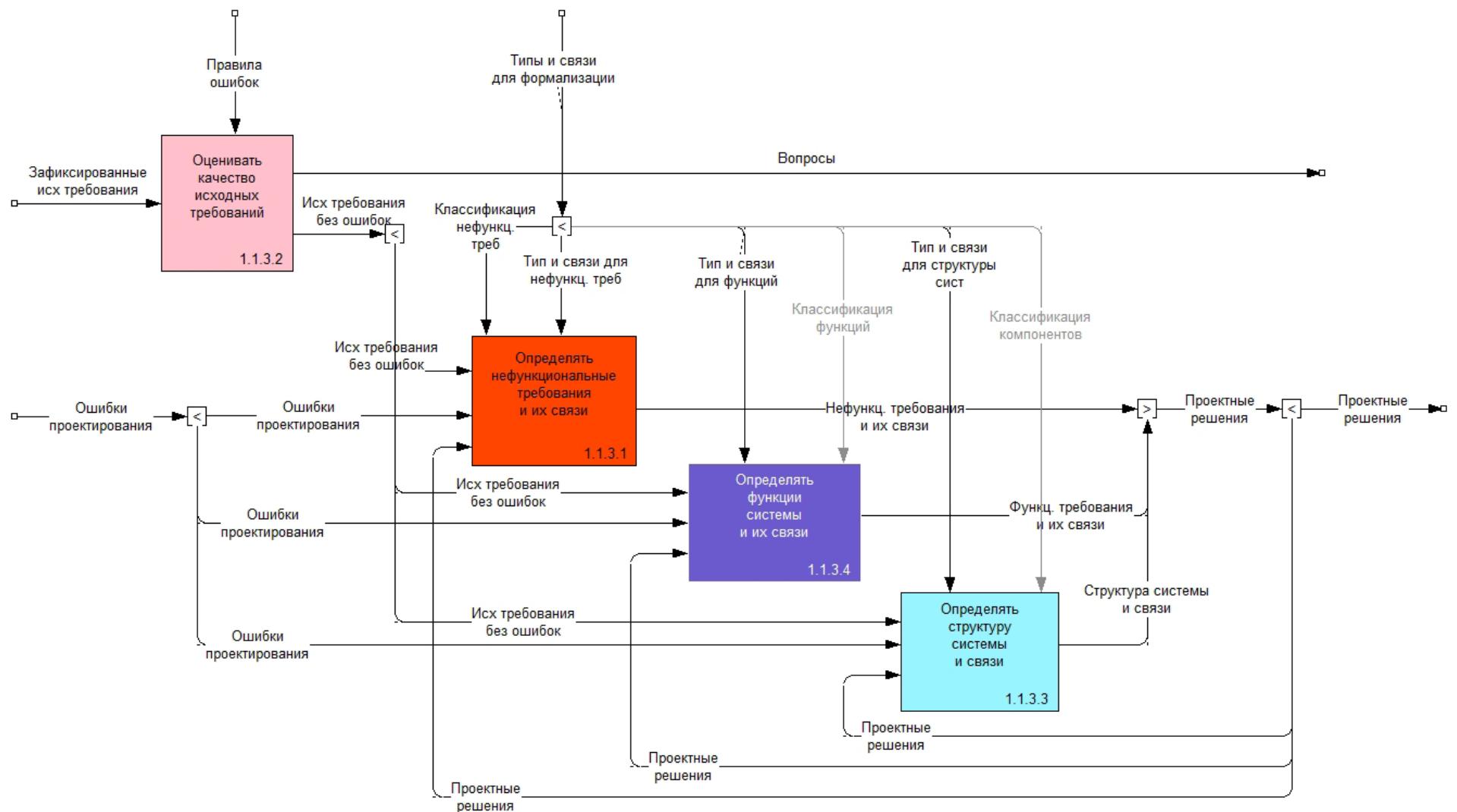


Рис. 2-13. Детализация функционального блока «Анализировать ИТ и проектировать систему»

## Шаги учебного проекта

Как уже было сказано выше, несмотря на строго зафиксированную информационную модель проекта, процесс наполнения выделенных корзинок может проходить нелинейно. Со стороны это может даже быть воспринято как «в случайном порядке».

В тоже время - изложение материала в книге возможно только последовательно. Поэтому мы рекомендуем:

- при первом чтении и там, где это возможно придерживаться хода книги, поскольку изложение каждой следующей темы базируется на предыдущих
- осознав процесс в целом, реализовывать его так, как вам будет удобнее

Последовательность изложения и первые шаги учебного проекта отражены на диаграмме ниже.

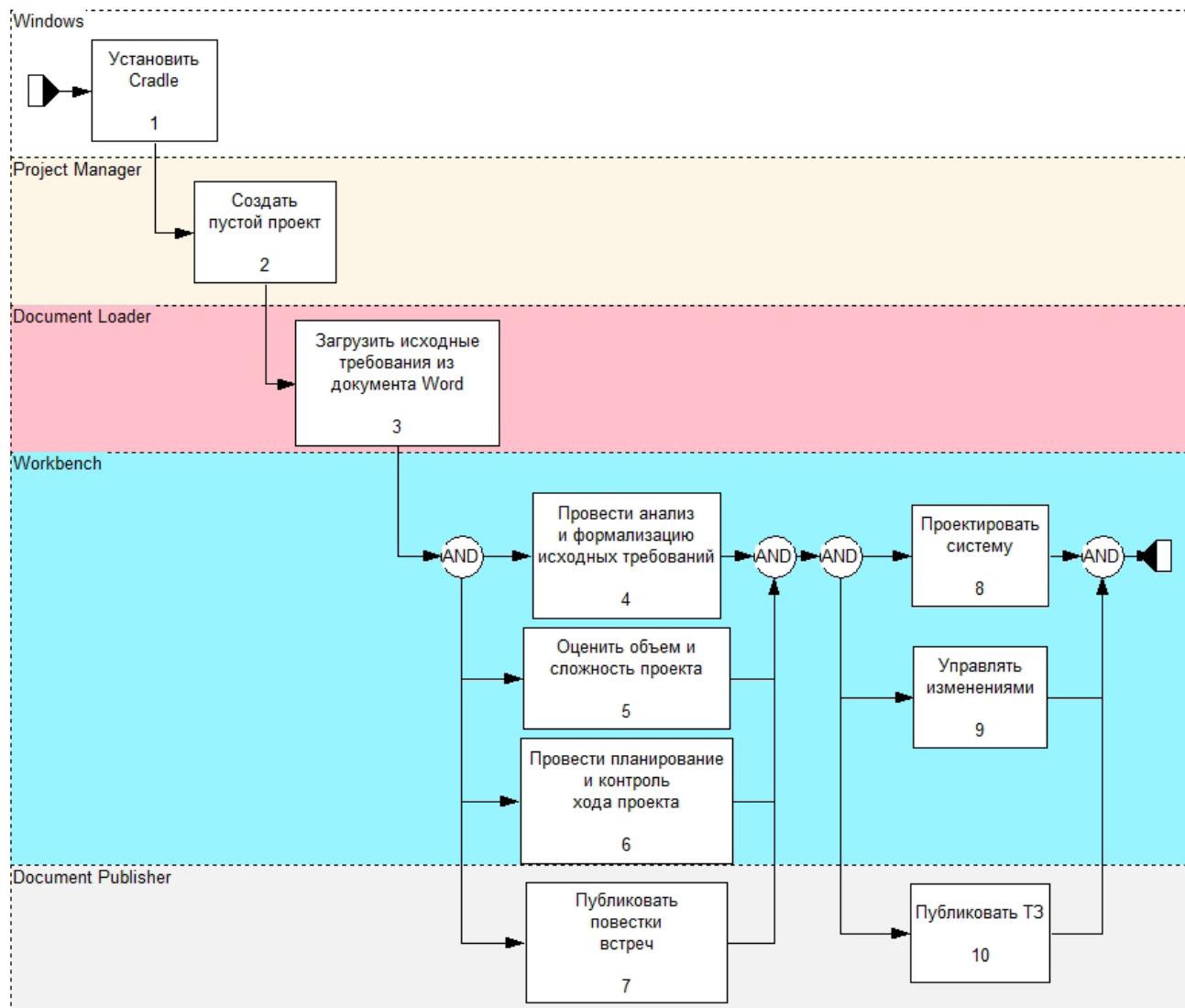


Рис. 2-14. Шаги учебного проекта

# **3SL Cradle**

## **Назначение**

Назначение 3SL Cradle - поддержка системного проектирования, включая автоматизацию типовых задач и предоставление общей интегрированной информационной среды всем участникам проектирования.

Cradle успешно применяется в аэрокосмической, телекоммуникационной, фармацевтической, автомобилестроительной, атомной, строительной и железнодорожной отраслях промышленности и многих других.

Применение Cradle целесообразно в тех областях, где крайне важны связи между элементами и эти связи не могут быть выявлены автоматически, а потому все управление связями идет «на ручном управлении». В первую очередь, это, конечно, системное проектирование. Но также это области требующие, например, проверки соответствия некоторой системы требований, например, аудит, участие в тендерах. Известны успешные примеры построения на базе Cradle системы управления рисками, системы управления взаимоотношения с клиентами.

Интересным примером является применение Cradle для подготовки юридических дел. Эта область также требует тщательной проработки множества связей между требованиями различных законодательных актов и материалов дела.

Возможности применения Cradle сильно варьируются от схемы его настройки, но чаще всего с его помощью решаются следующие классы задач:

- Разработка и управление требованиями.
- Системное проектирование, включая проектирование архитектуры.
- Управление рисками и задачами проекта.
- Управление тестами.
- Аудит проектных решений и оценка влияния изменений.
- Оценка состояния и иных показателей проекта.
- Публикация исходящих документов.

Мы попытались систематизировать все классы решаемых с помощью Cradle задач и отразили эту систематизацию на следующей иллюстрации:

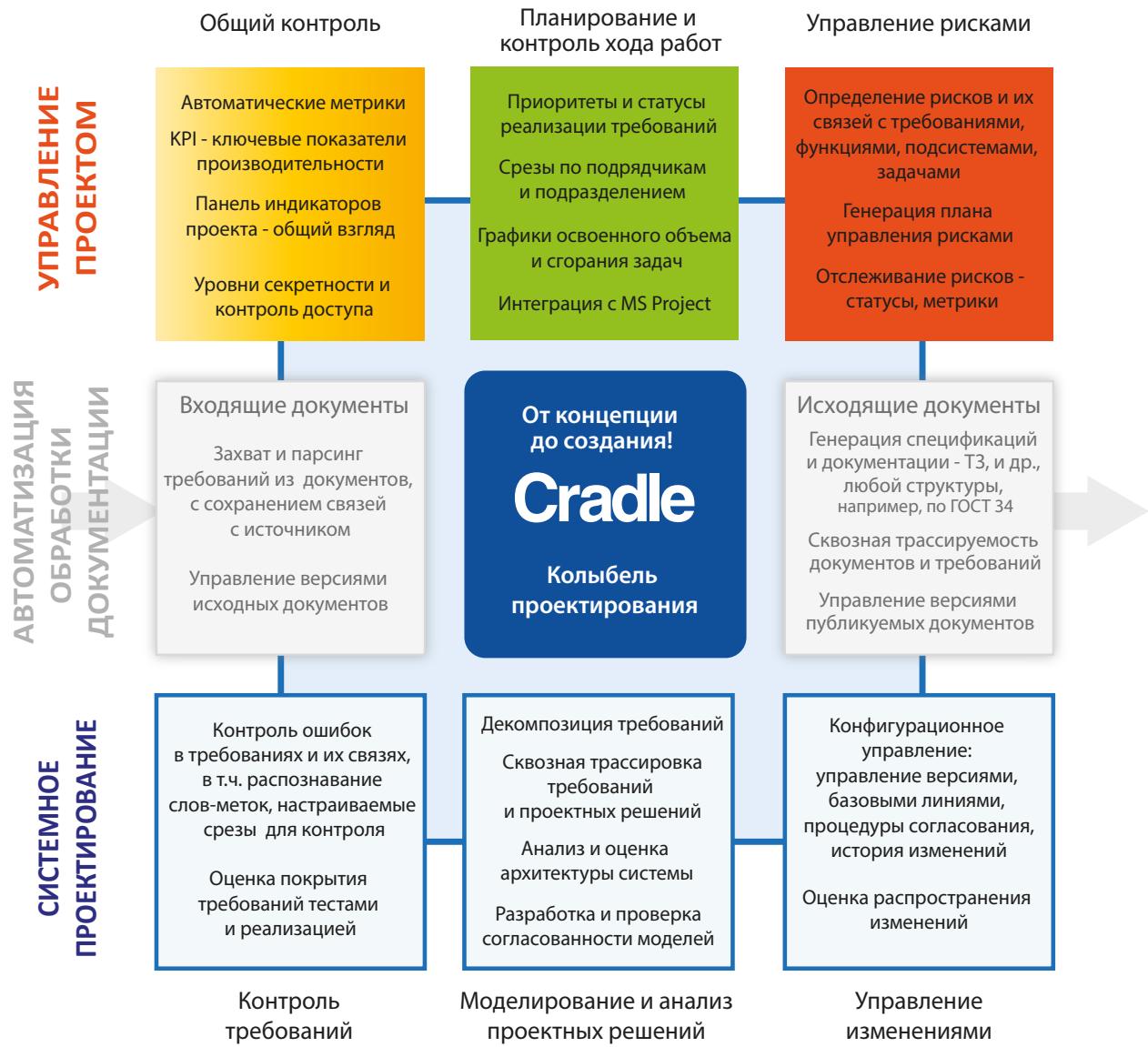


Рис. 2-15. Классы задач, решаемые с помощью 3SL Cradle

## Поддерживаемые нотации моделирования

Помимо возможности управления текстовыми проектными данными и их связями Cradle позволяет создавать модели различных типов. Cradle поддерживает большой набор нотаций:

Архитектурные:

- Architecture Interconnect Diagram (AID),
- Physical Architecture Diagram (PAD);

Функциональные и процессные:

- Data Flow Diagram (DFD),
- Entity Relationship Diagram (ERD),
- State Transition Diagram (STD),
- Data Structure Diagram (DSD),
- IDEF0 (IDF),
- Behaviour Diagram (eFFBD),
- Process Flow Diagram (PFD);

UML:

- Use Case Diagram (UCD),
- Package Diagram (PD),
- Sequence Diagram (SQD),
- Collaboration Diagram (COD),
- Class Diagram (CD),
- Statechart Diagram (SCD),
- Activity Diagram (ACD),
- Component Diagram (CPD),
- Deployment Diagram (DPD)

а также Structure Chart (STC), Software Architecture Diagram (SAD) и Ada Structure Graph (ASG).

Элементы моделей, созданные с помощью данных нотаций, могут быть связаны с другими проектными данными - требованиями, рисками, тестами, что позволяет обеспечить их сквозную трассируемость.

При этом могут быть выделены отдельные пространства имен в рамках которых возможно повторное использование элементов различных моделей при совместимости их нотаций.

## Архитектура

«Сердцем» Cradle является сервер базы данных - CDS (Cradle Database Server). Данный сервер - собственная разработка компании 3SL, а потому пользователям не требуется приобретения и сопровождения каких-либо других серверов БД.

Основной рабочий инструмент проектировщика - Workbench - толстый клиент, взаимодействующий с CDS.

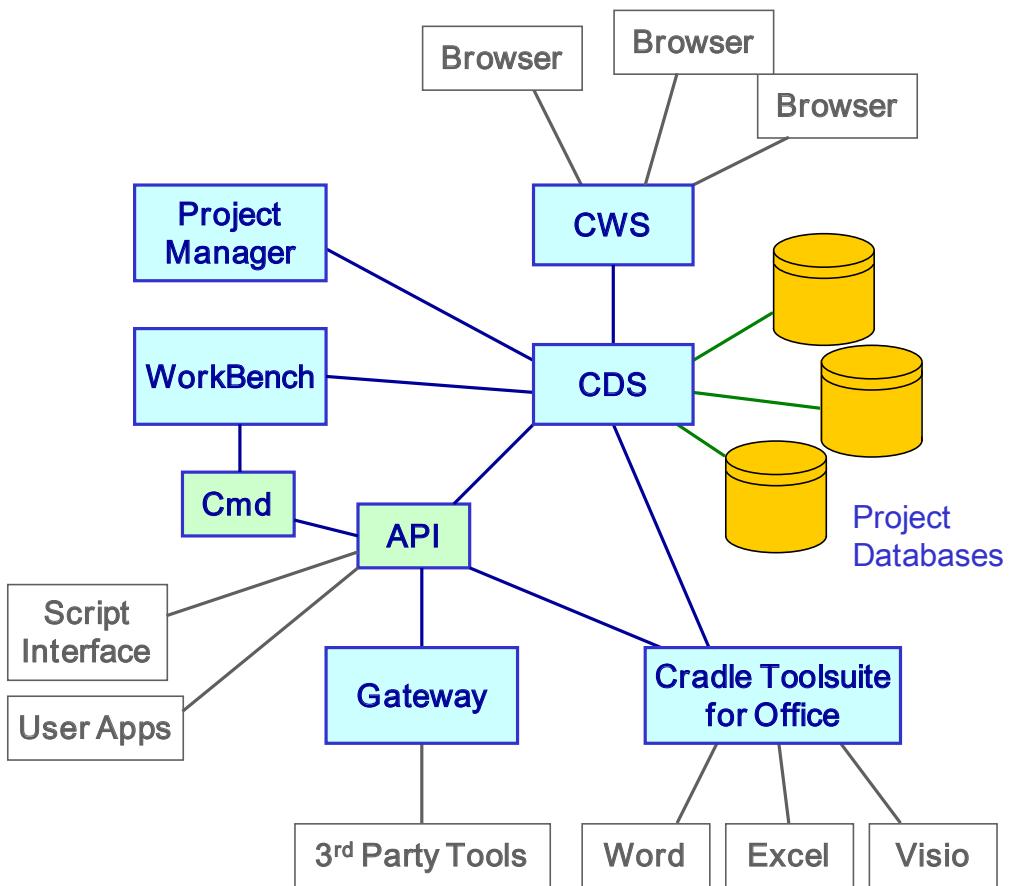


Рис. 2-16. Физическая архитектура 3SL Cradle

Важной составляющей комплектации Cradle является набор инструментов для интеграции с MS Office - это Cradle Toolsuite. Эти инструменты обеспечивают двустороннюю интеграцию и позволяют как загружать требования из документов MS Word, так и выгружать результирующие документы. В дополнение к этому реализована двусторонняя интеграция с MS Project.

Cradle также имеет собственный веб-сервер - CWS - Cradle Web Server, позволяющий работать с проектной базой с помощью веб-браузера. Вместе с Cradle поставляется три примера сборки веб-интерфейса и руководство по созданию адаптированного к задачам конкретной организации веб-интерфейса.

Cradle является хорошо масштабируемой системой:

- количество типов элементов в базе не ограничено (item types)
- количество атрибутов типов элементов не ограничено (frames)
- каждый атрибут (frame) может хранить или ссылаться на данные любого типа размером до одного терабайта.

Максимальный размер базы данных Cradle, которая используется в реальном проекте включает:

- Пользователей: 2,000
- Параллельных пользователей: 200-250
- Размер БД более 150 GBytes, включая более 15 миллионов ссылок и более 5 миллионов элементов.

Это позволяет реализовывать с его помощью крупные интеграционные проекты с длительной фазой сопровождения, ходе которой вся история изменений должна быть сохранена.

В состав Cradle корпоративной версии может быть включен прикладной программный интерфейс, который обеспечит интеграцию с корпоративными системами в случае, когда стандартных средств интеграции окажется недостаточно.

На иллюстрации ниже представлены все возможные интерфейсы, которые позволяют на информационном уровне интегрировать весь жизненный цикл проектируемой системы.

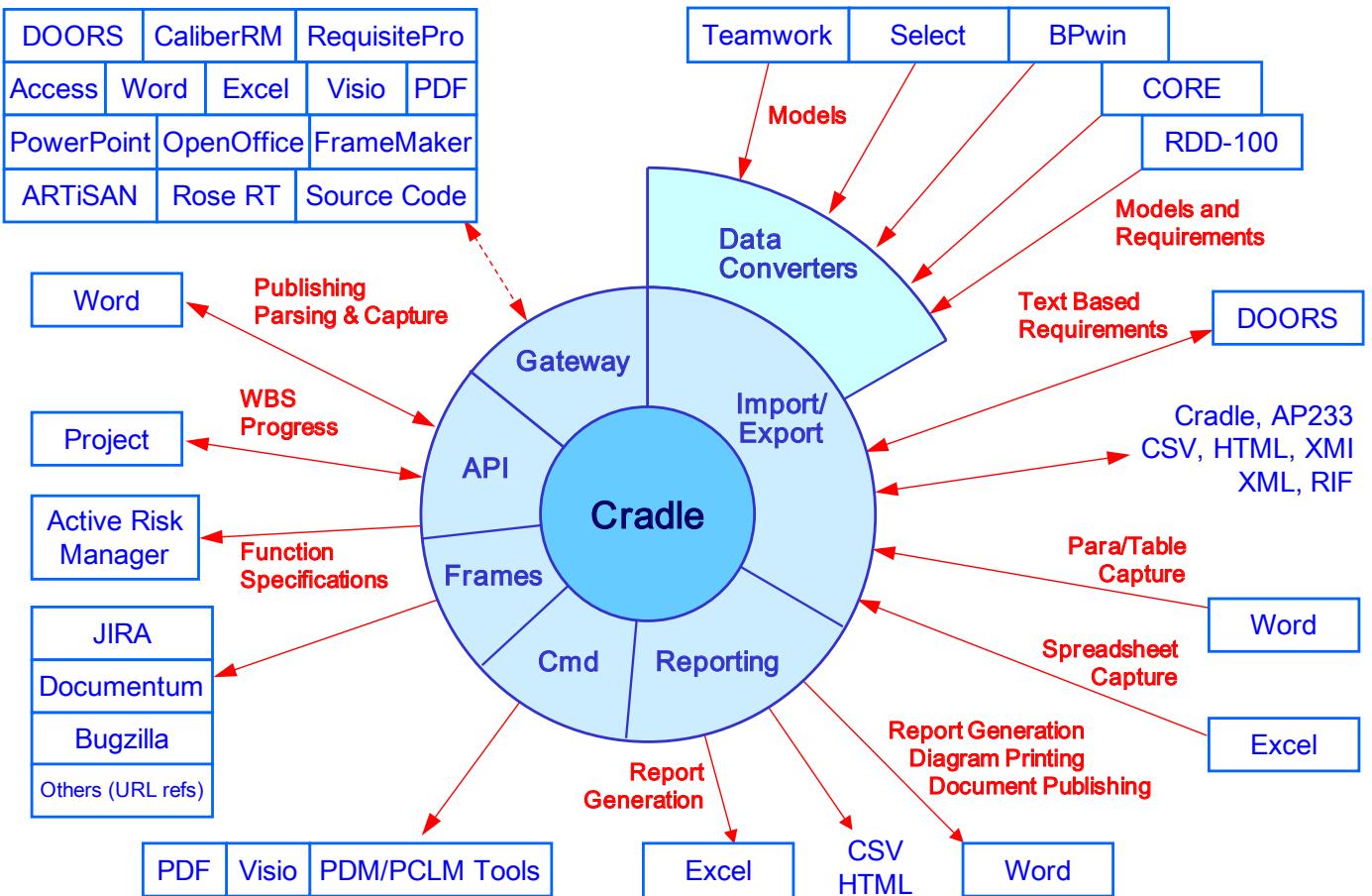


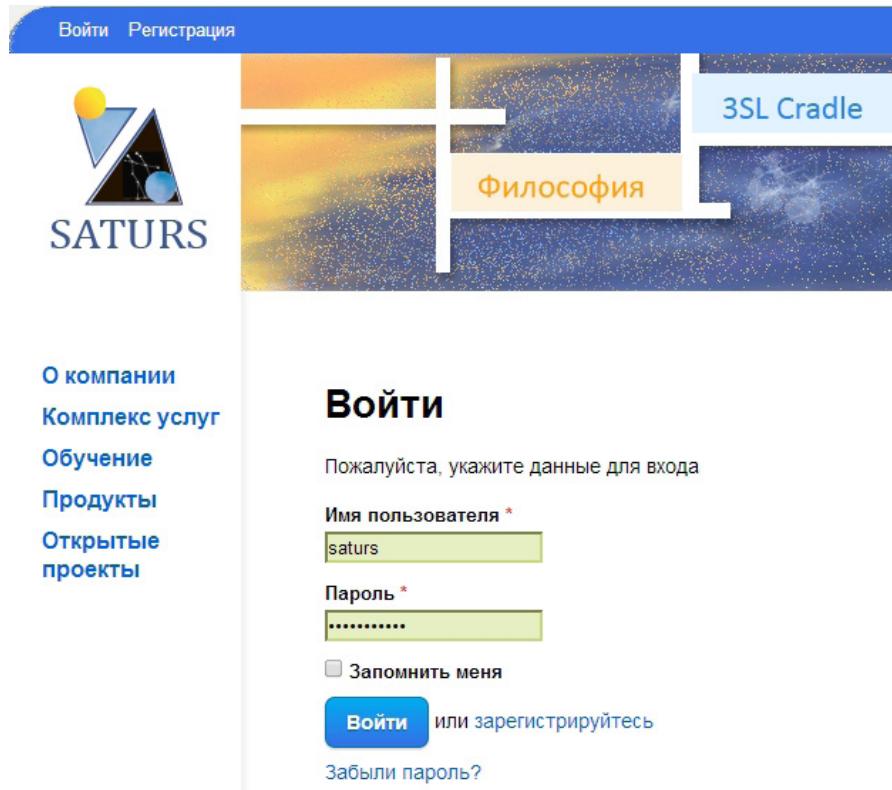
Рис. 2-17. Типы прикладных интерфейсов 3SL Cradle

## **УСТАНОВКА И ПОДГОТОВКА РАБОТЫ**

# Как получить пробную версию 3SL Cradle?

**Зарегистрироваться на сайте [www.saturs.ru](http://saturs.ru)**

1. Для установки пробной версии вам необходимо зарегистрироваться на сайте <http://saturs.ru> и войти:



Войти Регистрация

SATURS

3SL Cradle

Философия

Войти

Пожалуйста, укажите данные для входа

Имя пользователя \*

saturs

Пароль \*

.....

Запомнить меня

Войти или зарегистрируйтесь

Забыли пароль?

**Запросить trial-код для установки**

2. Перейти в раздел сайта [Получить Cradle](#)

[Получить Cradle](#)

3. Перейти в подраздел [Trial-код](#)
4. Заполнить форму запроса trial-кода, указав Host ID

## Скачать Cradle и получить 14-дневный

Для скачивания дистрибутивов или документации Cradle, а также по следующие данные:

Я хочу: \*  Дистрибутивы  Trial-код  Документацию

Версия Cradle \*

6.8.3 ▼

Платформа \*

Windows ▼

Вариант комплектации \*

Enterprise ▼

[Сравнение версий](#)

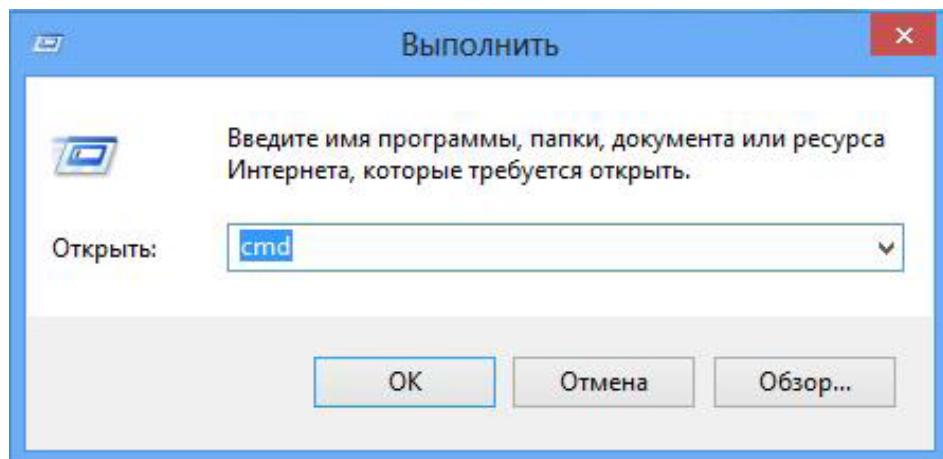
Host ID \*

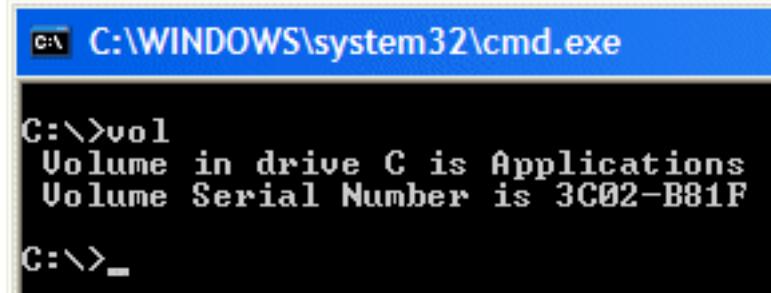
[Как узнать Host-id?](#)

Host ID компьютера используют для привязки лицензионной программы к вашему ПК. Это восемь цифр без пробелов, дефисов и двоеточий.

\* Чтобы узнать ваш Host ID для Windows

Запустите командную строку **cmd** в окне **Выполнить**, нажав сочетание клавиш **Win + R** и наберите команду **vol**.



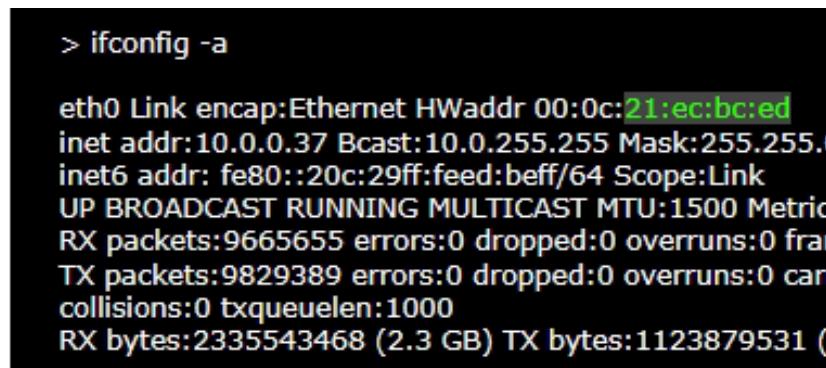


```
C:\>vol
Volume in drive C is Applications
Volume Serial Number is 3C02-B81F

C:\>_
```

\* Чтобы узнать ваш Host ID для Linux

Вызвать команду **ifconfig -a**. Последние 8 цифр - это то, что необходимо. В данном случае - 21ecbced (подсвечен зеленым).



```
> ifconfig -a

eth0 Link encap:Ethernet HWaddr 00:0c:21:ec:bc:ed
inet addr:10.0.0.37 Bcast:10.0.255.255 Mask:255.255.
inet6 addr: fe80::20c:29ff:feed:beff/64 Scope:Link
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
      RX packets:9665655 errors:0 dropped:0 overruns:0 frame:0
      TX packets:9829389 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:2335543468 (2.3 GB) TX bytes:1123879531 (
```

\* Чтобы узнать ваш Host ID для Sparc

Выполнить команду hostid

5. Отправить форму запроса, нажав кнопку:

**Запросить trial-код к 3SL Cradle**

6. После этого в течении рабочего дня на ваш e-mail будет выслан trial-код.

## Скачать дистрибутив

1. Перейти в раздел сайта [Получить Cradle](#)

[Получить Cradle](#)

2. Перейти в подраздел [Дистрибутивы](#)

### Скачать Cradle и получить 14-дневный код активации

Для скачивания дистрибутивов или документации Cradle, а также получения кода активации и следующие данные:

Я хочу:  Дистрибутивы  Trial-код  Документацию  Обзоры

Версия Cradle \*

6.8.3 ▾

Платформа \*

Windows ▾

Для установки Cradle вам потребуется trial-код

[Получить ссылки на дистрибутивы](#)

Ссылка	Описание	Язык	P
<a href="#">Дистрибутив для Windows</a>	Полный дистрибутив для Windows, включая Toolsuite. Для того, чтобы инсталлировать Cradle, Вам понадобится код активации. Вы можете запросить trial-код в этой же форме.	мульти	3

\* Для Windows - достаточно скачать первый файл

Это полный дистрибутив для Windows, включая Office Toolsuite.

Office Toolsuite - это набор инструментов для взаимодействия с пакетами MS Office, включая Document Loader (загрузка исходных требований из документов Word в базу данных Cradle), Document Publisher (публикация результирующих документов любой сложности, например, Т3)

Для работы с модулем Document Loader и Document Publisher на вашем компьютере **должен быть установлен MS Office 32bit**.

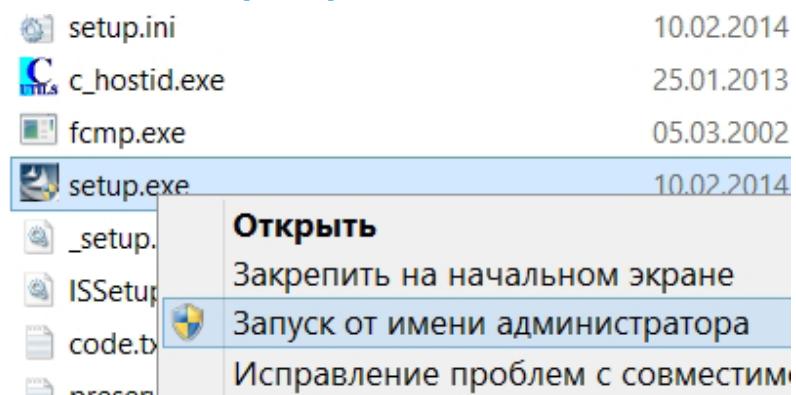
\* Для Linux и Sparc скачать все три файла.

## Установить Cradle

Получив trial-код по почте, можно начинать установку.

1. Распаковать архив с дистрибутивом в любую папку
2. **Запустить установку от имени администратора**

Для этого щелкните по файлу setup.exe правой кнопкой мыши и выберите **Запуск от имени администратора** в контекстном меню



При инсталляции Cradle, мы рекомендуем оставить параметры установки по умолчанию, для выбора индивидуальных параметров воспользуйтесь руководством по установке.

Для установки Document Loader и Document Publisher на вашем компьютере **должен быть установлен MS Office 32bit**.

Поддерживаемые версии Windows и MS Office

- Windows XP (SP2 или выше)
- Windows 2003 R2 (SP2 или выше)
- Windows Vista (SP1 или выше) and 2008
- Windows 7
- Microsoft Office:
  - 2003 (SP3 или выше)
  - 2007 (SP1 или выше)
  - 2010 (только 32 bit) <=====

Полная документация по установке 3SL Cradle для платформы Windows описана в документе:

- «Windows Installation Guide»

для UNIX платформ в документе:

- «UNIX Installation Guide»

которые доступны в разделе сайта Документация:

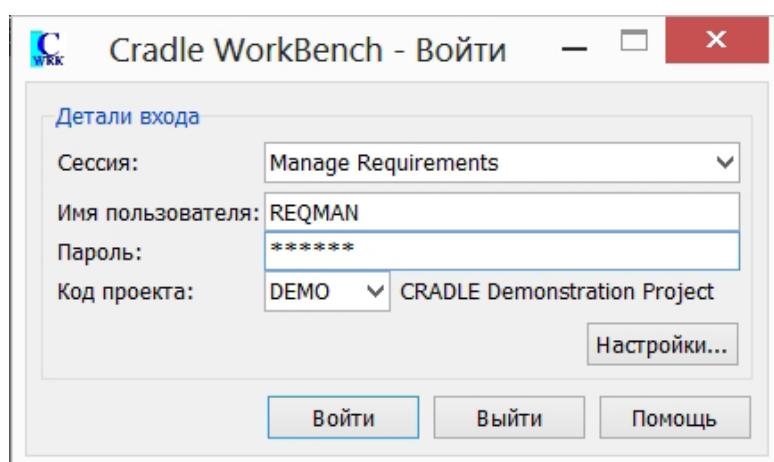
<a href="#">Windows Installation Guide</a>	<a href="#">Руководство по установке 3SL Cradle для Windows</a>
<a href="#">Unix Installation Guide</a>	<a href="#">Руководство по установке 3SL Cradle для UNIX</a>

# Проверить установку

## Запустить DEMO-проект

После установки убедитесь в успешности проведенной инсталляции. Просто запустите DEMO-проект, поставляемый вместе с Cradle. Он поможет вам узнать, что система установилась нормально и проиллюстрирует возможности системы. Для этого:

1. Запустить WorkBench через меню Пуск -> Программы -> Cradle -> WorkBench (Windows).
2. В появившемся диалоге входа выберите сессию Manage Requirements
3. Введите данные для входа:
  - Имя пользователя: REQMAN
  - Пароль: REQMAN
  - Код проекта: DEMO



Также вы можете войти и под другими пользователями, имеющие разные настройки прав доступа.

Пользователь с максимальными правами

- логин: MANAGER/ пароль: MANAGER

Пользователь, управляющий разработкой

- логин: DEVMAN/ пароль: DEVMAN

Пользователь, управляющий тестированием

- логин: TESTMAN/ пароль: TESTMAN

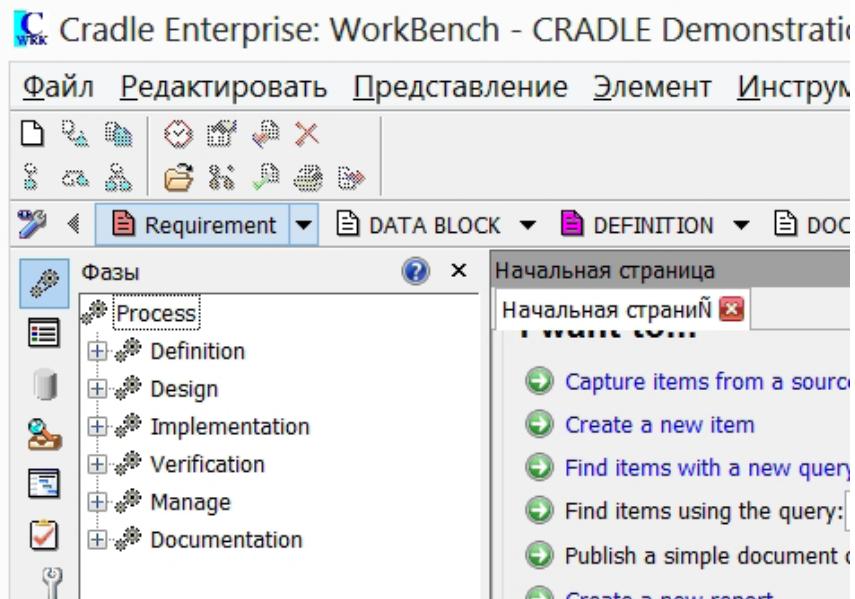
Клиент (заказчик)

- логин: CUSTOMER/ пароль: CUSTOMER

## Проверить работу DEMO-проекта, используя Панель фаз

В версии Enterprise, чтобы быстро познакомится с проектом и проверить его работу вы можете воспользоваться Панелью фаз, в рамках которой структурированы все наиболее часто используемые операции. После запуска Workbench она будет доступна с левой стороны.

Вызов панели фаз производится нажатием иконки, размещенной на левой панели пользовательского интерфейса.



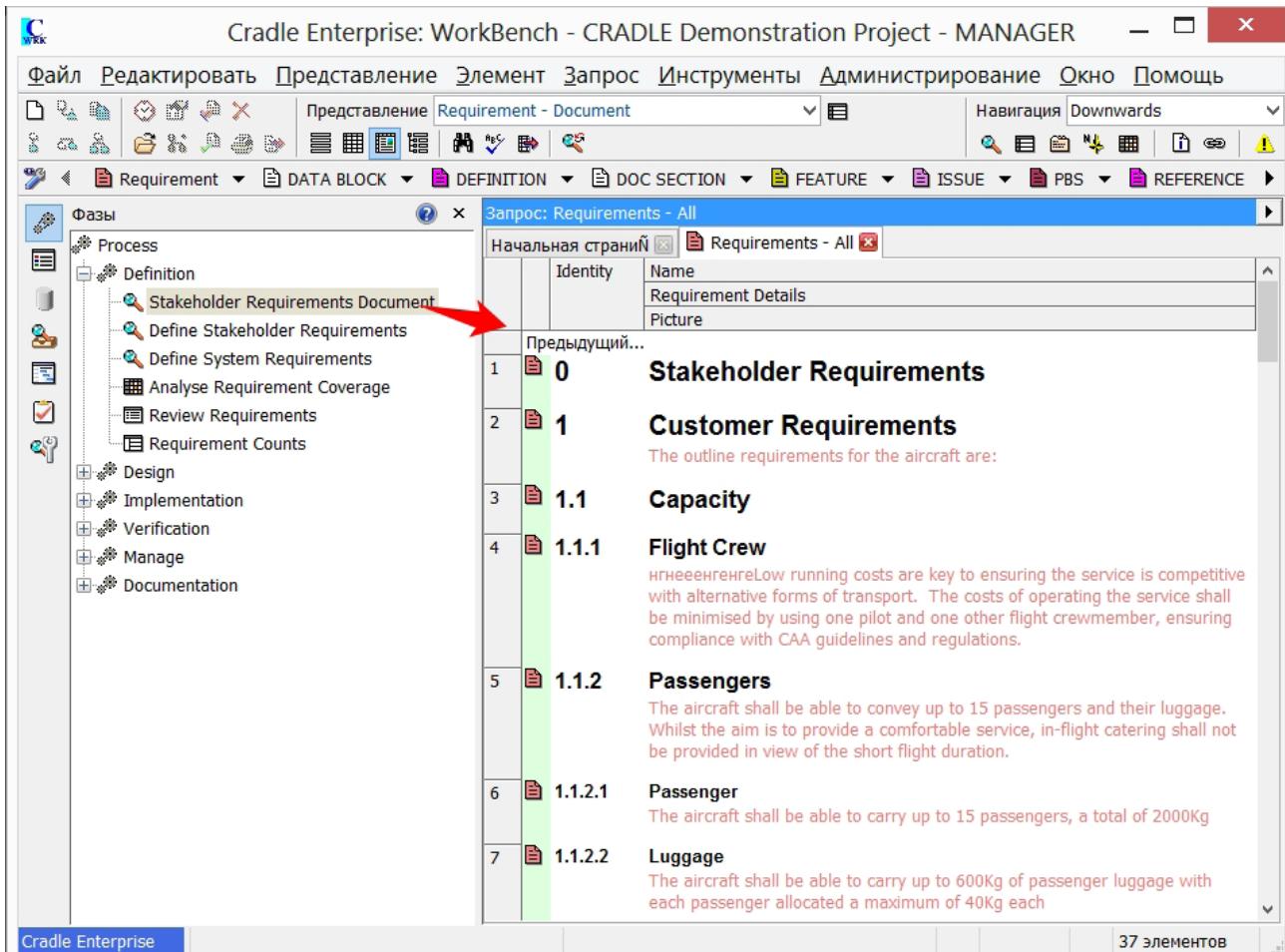
В версии Desktop панель фаз отсутствует и для этих задач можно использовать Стартовые страницы, которые могут быть настроены для каждого пользователя.

По двойному щелчку производится открытие выбранного представления.

Пройдитесь по разделам панели фаз, чтобы посмотреть возможные способы представления данных в Cradle.

Например, в разделе Definition вызовите запрос Stakeholder Requirements Document. Справа, в активную панель будут выведены все требования заказчика в виде документа.

При этом вы можете редактировать каждое отдельное требование прямо в этом представлении. Попробуйте установить курсор в любое текстовое поле.



С

Cradle Enterprise: WorkBench - CRADLE Demonstration Project - MANAGER

Файл Редактировать Представление Элемент Запрос Инструменты Администрирование Окно Помощь

Представление Requirement - Document

Навигация Downwards

Requirement DATA BLOCK DEFINITION DOC SECTION FEATURE ISSUE PBS REFERENCE

Фазы

Process

Definition

- Stakeholder Requirements Document
- Define Stakeholder Requirements
- Define System Requirements
- Analyse Requirement Coverage
- Review Requirements
- Requirement Counts

Design

- Implementation
- Verification
- Manage
- Documentation

Запрос: Requirements - All

Начальная страница Requirements - All

Identity Name Requirement Details Picture

Предыдущий...

1 0 Stakeholder Requirements

2 1 Customer Requirements

3 1.1 Capacity

4 1.1.1 Flight Crew

5 1.1.2 Passengers

6 1.1.2.1 Passenger

7 1.1.2.2 Luggage

37 элементов

В следующем разделе панели фаз - Design будет любопытно взглянуть на матрицу трассировки, которая используется для сравнения нескольких базовых продуктов, которые должны лежать в основу проекта.

Обратите внимание, что Cradle позволяет отражать в матрице трассировке не только связи, но и элементы, через которые осуществляется связь, что делает такие матрицы более информативными.

Cradle Enterprise: WorkBench - CRADLE Demonstration Project - MANAGER

Файл Редактировать Представление Элемент Матрица Инструменты Администрирование Окно Помощь

Навигация: Downwards

Requirement DATA BLOCK DEFINITION DOC SECTION FEATURE ISSUE PBS REFERENCE RESULT RISK SBS

Фазы

Process

- Definition
  - Stakeholder Requirements Document
  - Define Stakeholder Requirements
  - Define System Requirements
  - Analyse Requirement Coverage
  - Review Requirements
  - Requirement Counts
- Design
  - Design Architecture
  - Functional Allocation
  - Product Feature Comparison
  - Design Elements
- Implementation
- Verification
- Manage
- Documentation

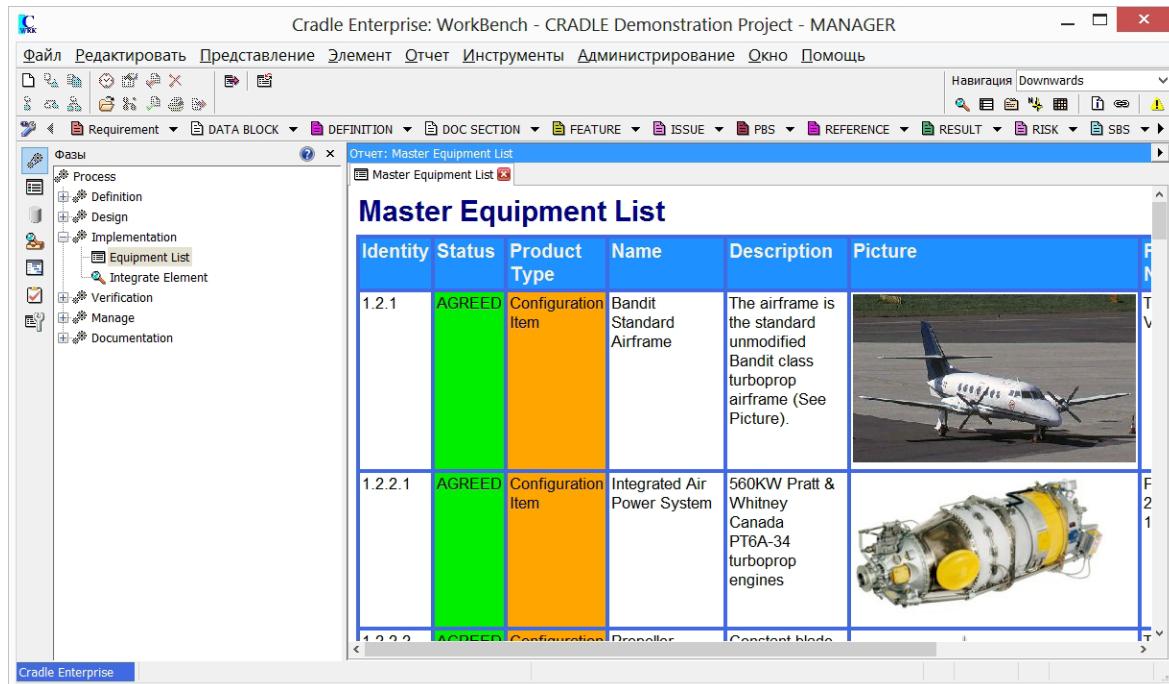
Матрица: Product Feature Comparison

Начальная	Requirement	Requirements	Requirement C	Requirement - XRD	Requirement	Requirement	Product Feature Comparison
1.1 : Multi-turboprop Features : : A : PBS	2.1 : Turbofan Features : : A : PBS	3.1 : Single Prop Features : : A : PBS					
FEA-1 : Capacity : 15 - 18 Passenger, 2	Up to 3500 lbs. Can configure for up to 18 passengers in rows of three seats abreast with a 29 inch pitch. Seating is PBS	15 Passengers and 3 aircrew (2 pilots and 1 steward)	10 - 12 passengers				
FEA-2 : Range : 2000 Miles	2000Km at long range cruising speed with reserves	7030 Km at cruising speed and altitude	2000 nm				
FEA-3 : Cruise Speed : 300 - 350 Kts	Economical cruise speed 326Km/h	480 Kts	250 Kts				
FEA-4 : Take-off distan : No more than 3000 Ft	Take-off distance with full load 2,500 Ft	Requires at least 4000 ft runway for take-off with 15 passengers and luggage	Take-off distance with full load 1,500 Ft				
FEA-5 : Through-Life Co : \$10 Million	Acquisition costs \$4 Million per aircraft	Unit purchase costs are \$6 Million per aircraft, notwithstanding modifications required to comply with CAA regulations	Acquisition costs of \$1.5 million per aircraft				
FEA-6 : Reliability : 95% Probability of no failures	Operating costs are estimated at \$5 Million over a 10 year operating cycle	Operating costs estimated at \$2 Million per annum, based on 3 flight crew and a typical short-haul operating profile.	Annual running costs are \$1 million, based on predicted operating cycle.				
FEA-7 : Certification : Baseline of	Maintenance required every 3 months, down time 2 days	Requires monthly maintenance schedule lasting 1 day	Requires monthly maintenance schedule lasting 1 day				
	The basic airframe is fully certified to CAA and FAA standards.	Aircraft holds CAA and FAA Airworthiness certificates, although no modifications are allowed.	Awaiting full certification to operate with passengers. Certified for flight tests only.				

Cradle Enterprise

Cradle позволяет хранить в требованиях и других типах элементов изображения, документы и любые другие типы данных. При этом сохраняется история изменений всех составляющих (необходимо включить режим хранения).

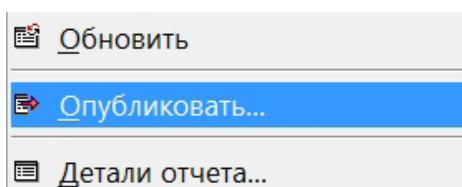
Представление Equipment List в разделе Implementation иллюстрирует эту возможность.



The screenshot shows the Cradle Enterprise WorkBench interface with the title bar 'Cradle Enterprise: WorkBench - CRADLE Demonstration Project - MANAGER'. The menu bar includes 'Файл', 'Редактировать', 'Представление', 'Элемент', 'Отчет', 'Инструменты', 'Администрирование', 'Окно', and 'Помощь'. The toolbar has various icons for file operations. The left sidebar shows a tree structure under 'Фазы' (Phases) with 'Process', 'Definition', 'Design', 'Implementation' (selected), 'Verification', 'Manage', and 'Documentation'. The main area displays a report titled 'Master Equipment List' with the following data:

Identity	Status	Product Type	Name	Description	Picture
1.2.1	AGREED	Configuration Item	Bandit Standard Airframe	The airframe is the standard unmodified Bandit class turboprop airframe (See Picture).	
1.2.2.1	AGREED	Configuration Item	Integrated Air Power System	560KW Pratt & Whitney Canada PT6A-34 turboprop engines	
1.2.2.2	AGREED	Configuration Item	Propeller	Constant blade	

Вы можете щелкнуть правой кнопкой мыши по данному представлению и опубликовать его в виде Html или RTF.



Сделайте это, не меняя настроек публикации.

Поскольку в Cradle вы можете настроить и использовать любые типы проектных данных, то можно управлять не только требованиями Заказчиков, но и самими Заказчиками. Это особенно удобно, если Заказчиков много и надо успеть правильно сориентироваться.

Cradle позволяет настроить любые подсветки, чтобы быстрее ориентироваться в множестве элементов. Вы можете оценить это выбрав раздел **Manage -> Manage Stakeholders**

С

Cradle Enterprise: WorkBench - CRADLE Demonstration Project - MANAGER

Файл Редактировать Представление Элемент Запрос Инструменты Администрирование Окно Помощь

Навигация Downwards

Requirement DATA BLOCK DEFINITION DOC SECTION FEATURE ISSUE PBS REFERENCE RESULT RISK

Фазы

- Process
- Definition
- Design
- Implementation
- Verification
- Manage
  - Manage Stakeholders
  - Manage Issues
  - Manage Risks
  - Manage References
  - Define Items
- Documentation

Запрос: Stakeholder - All

	Identity	Name	Stakeholder Type	Stakeholder Priority	Description	Narrative
1	STA-1	J Charles	SUPPLIER	Medium	Point of contact at Manchester Aircraft Engineering, providing maintenance, servicing and repair facilities at Manchester Airport.	1 March capture initial requirements
2	STA-2	B Hoskins	SUPPLIER	Low	Represents Air Traffic Controllers covering Manchester air space. Providing support to ensure compliance with ATC and air navigation	
3	STA-3	G Smith	OPERATOR	High	Pilot representative	
4	STA-4	A Clarke	CUSTOMER	High	The customer representative on the project, providing requirement input and change review and comment on costs. A member of the	
5	STA-5	B Manning	SUPPLIER	Medium	Manager of Walney Airport ground staff, provides comment on check in staff, baggage handling and aircraft position.	
6	STA-6	S Pennington	USER	High	Works for Pennington Marketing services, providing a surrogate stakeholder representing the passenger. Ensures decisions are	
7	STA-7	J R Ewing	CUSTOMER	Medium	General manager for Cumbrian Fuel services, who provide fuel at Walney Airfield. A stakeholder as fuel is provided at 24 hours'	
8	STA-8	G Khan	COMPANY	Medium	Head of the production department. A member of the review and change management boards, ensuring the design is achievable and that	

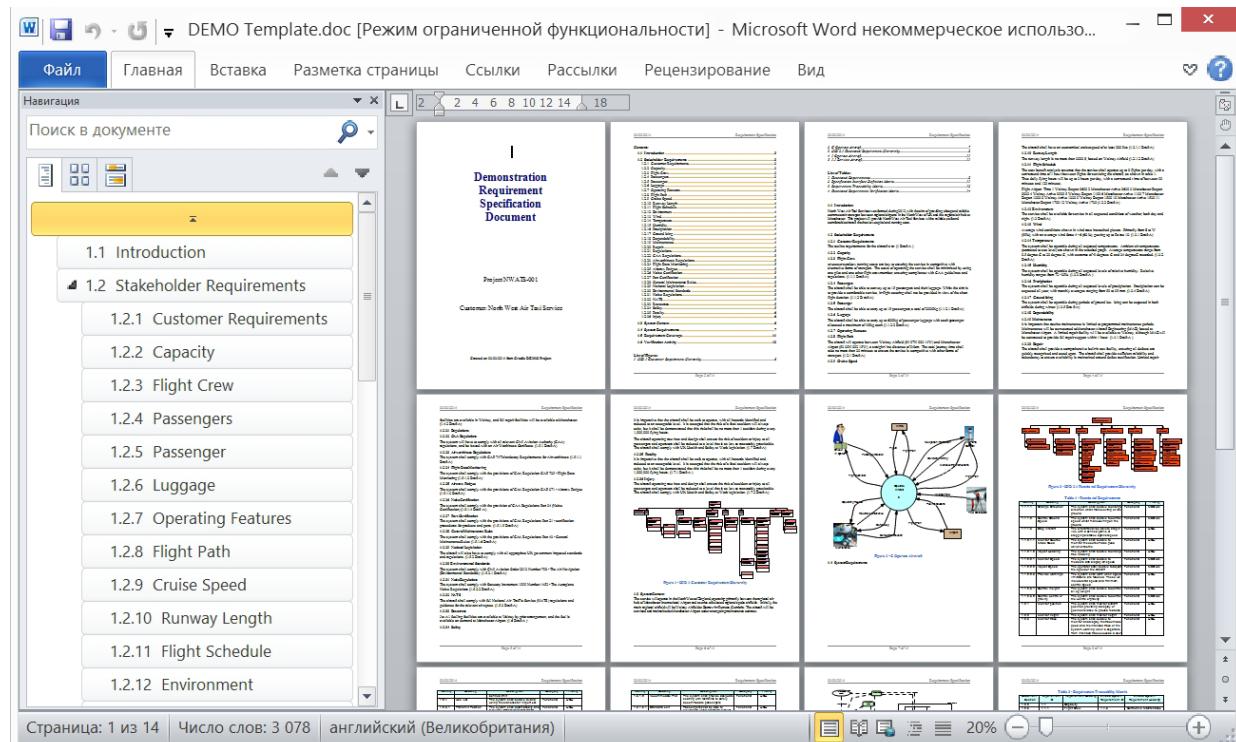
Cradle Enterprise 9 элементов

И наконец одна из ключевых возможностей - генерация документации. Cradle позволяет генерировать любые, сколько угодно сложные документы, основанные на пользовательских шаблонах. Чтобы посмотреть пример генерации, сначала закройте MS Word, если он был открыт и запустите в разделе Documentation фазу Requirements Specification Document.

После запуска, если вы устанавливали Cradle Toolsuite, вы увидите как генерируется документ. Когда генерация закончится, будет выведено сообщение

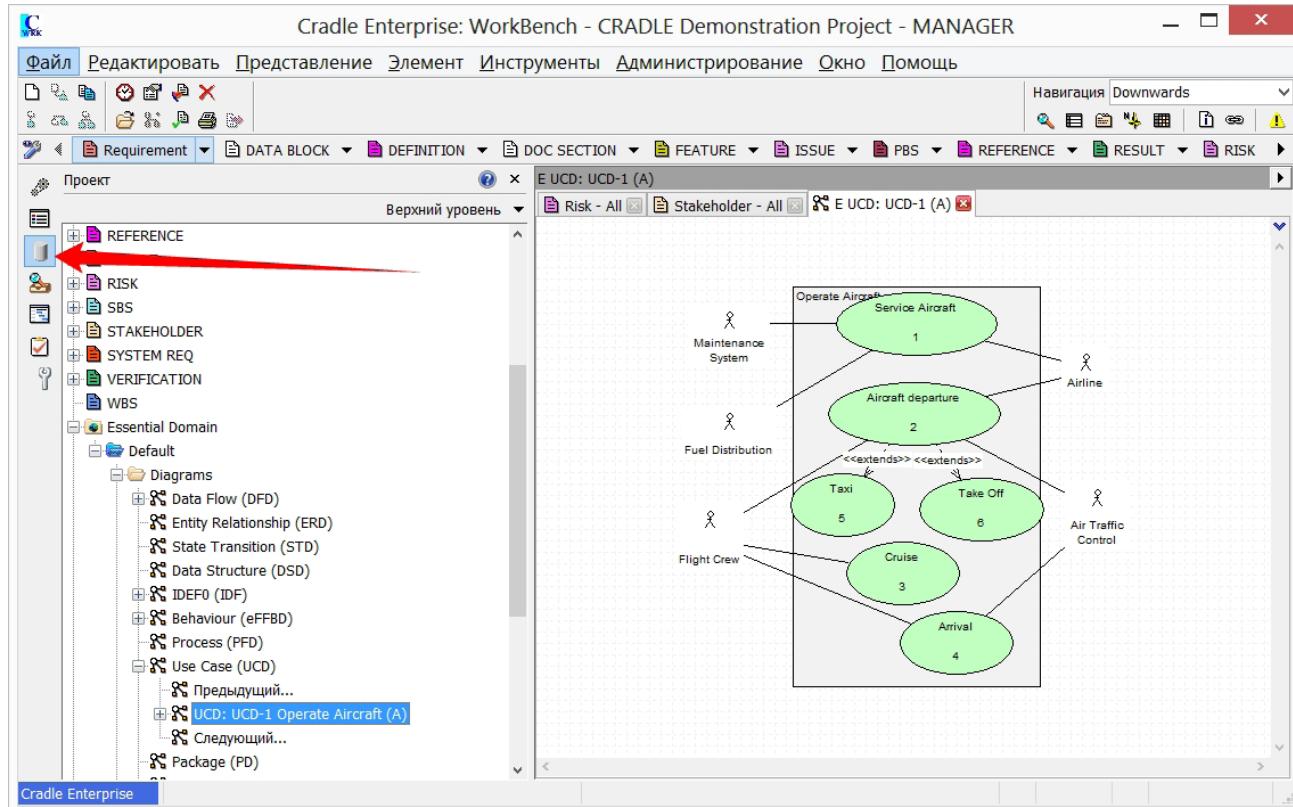


после чего MS Word закроется и через некоторое время откроет сгенерированный документ. Подождите немного.



Чтобы познакомиться с возможностями моделирования перейдите на панель Базы данных, в раздел Essential Domain. Там Вы найдете различные диаграммы. Щелкните на иконку любой диаграммы, чтобы открыть ее.

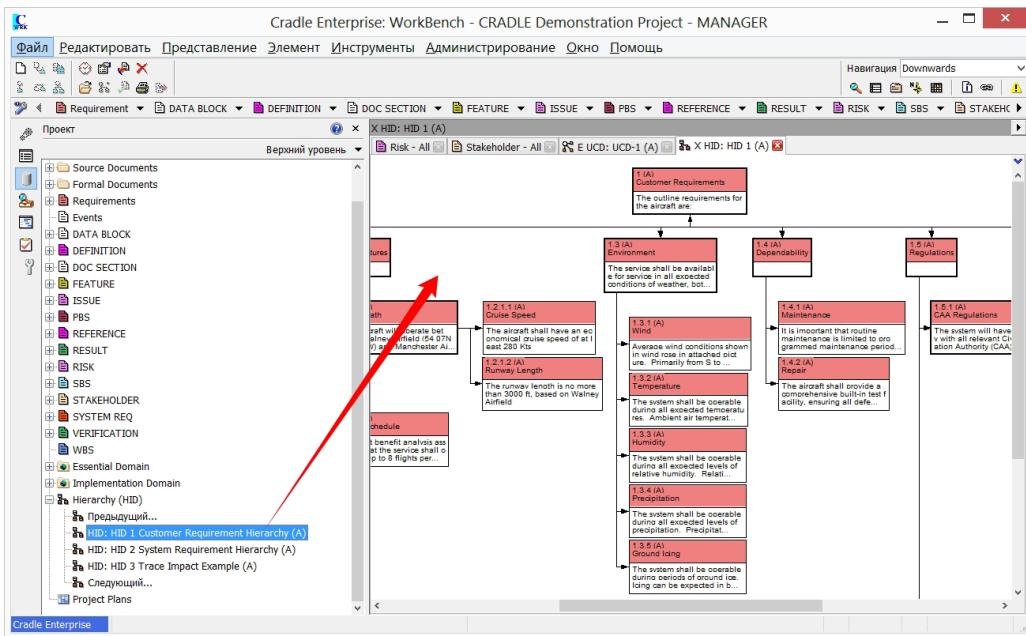
В Cradle вы можете связывать требования, риски и другие элементы напрямую с элементами диаграмм.



На той же панели, но несколько ниже, найдите специальный тип диаграмм - HID - иерхические диаграммы. Это еще один способ отображения требований и их связей, данные диаграммы строятся автоматически (в отличие от моделей). Вы можете выбрать какие типы элементов отображать, какие связи и в каком направлении, а также атрибуты элементов.

Также можно поиграть с расположением диаграммы. Все настройки можно сохранить и использовать повторно.

С помощью HID удобно, например, отображать архитектуру системы, структуру модулей, организационную структуру компании.



Если все проверки прошли успешно, то можно переходить к следующему этапу - созданию и настройке своего проекта.



## **ТИПОВЫЕ ЗАДАЧИ**

# Управление проектами *Cradle*

Управление проектами включает создание и удаление проектов, их упорядочивание с помощью папок, назначение атрибутов для описания проектов. Папки и атрибуты проектов помогают быстрее ориентироваться в созданных проектах.

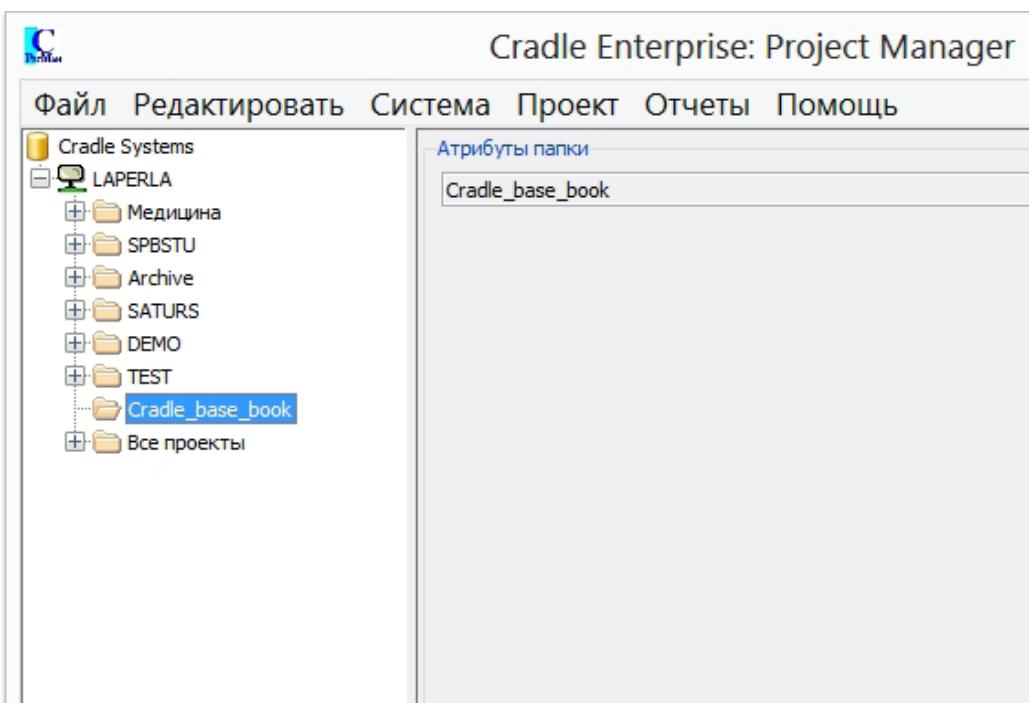
Для управления проектами чаще всего используется приложение Project Manager, но также возможно создавать проекты из командной строки. Это удобно, если необходимо создать серию однотипных проектов или для их автоматического создания при интеграции Cradle с другими системами.

## Создать проект

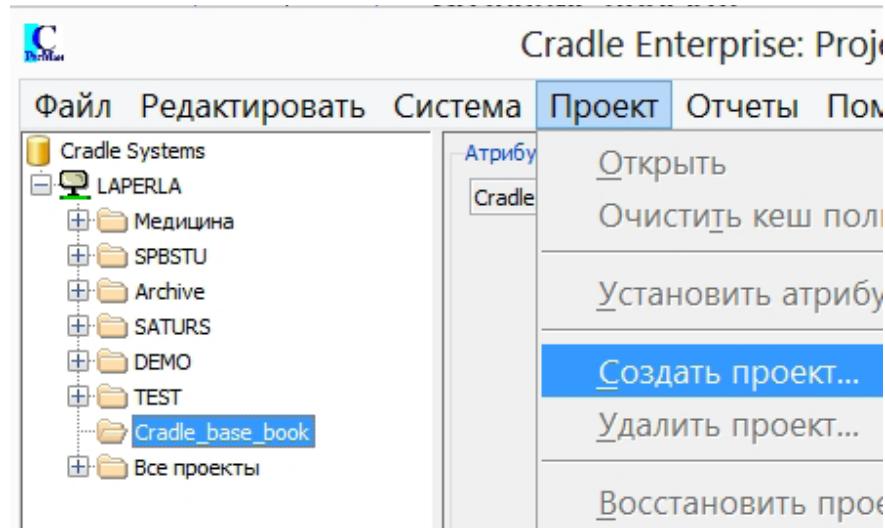
1. Запустить Project Manager.



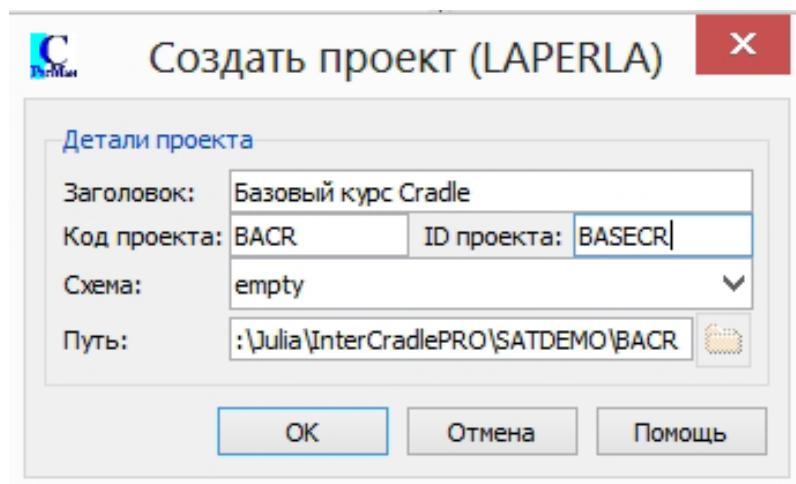
2. В Project Manager выбрать папку для создания проекта.



3. В меню **Проект** выбрать пункт **Создать проект**.



4. Ввести *Заголовок* - наименование проекта, *Код проекта* - четырехзначный код.



*Код проекта* (анг. *Project Code*), используется для однозначной идентификации проектов Cradle. Вы будете использовать его для входа в проект. Cradle использует код проекта для поиска пути к проекту в реестре проектов (файл %CRADLEHOME%\proj\_reg.\* ).

Каждый элемент в проекте (например, требования) также имеет свой уникальный идентификатор, который называется *Project Database Unique ID (PDUID)*. Для того, чтобы элементы имели уникальные идентификаторы не только в рамках одного проекта, но и в рамках всех проектов, в *PDUID* включается *Project ID*, который вы можете задать при создании проекта.

Таким образом, используя *Project ID*, при экспорте/импорте данных всегда можно понять, был ли элемент создан в данной базе данных (проекте) или был импортирован из другого проекта.

5. Выбрать *Схему* и *Путь* - директорию на диске, где будет храниться проект.
6. Нажать кнопку **OK**.

Хорошая практика - путь к проекту должен оканчиваться четырехзначным кодом проекта. Например, **C:\CradleProjects\TES5**

7. При успешном выполнении операции по созданию проекта Cradle откроет браузер и выведет html-отчет.

**Проект**

Код проекта	PID	Заголовок	Схема	Расположение базы данных
BACR	BASECR	Базовый курс Cradle	empty	C:\Julia\InterCradlePRO\SATI

**Пользователи**

The project's user register has been initialised with one User Profile. You login to your new project and change this password as soon as possible.

Имя пользователя Cradle	MANAGER
Пароль	MANAGER

Для любого вновь созданного проекта определен один пользователь - **MANAGER**. Это супер-администратор системы, имеющий максимальные права.

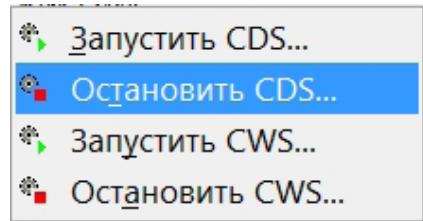
Для настольной версии Cradle, **MANAGER** - это единственный пользователь.

Для корпоративной версии рекомендуется сразу настроить дополнительных пользователей с меньшими правами и не использовать супер-администратора.

## Удалить проект

Для удаления проектов необходимо сначала остановить сервер базы данных Cradle (CDS).

1. Для этого в Project Manager в меню **Система** выбрать **Управление сервером**, и выбрать **Остановить CDS**.

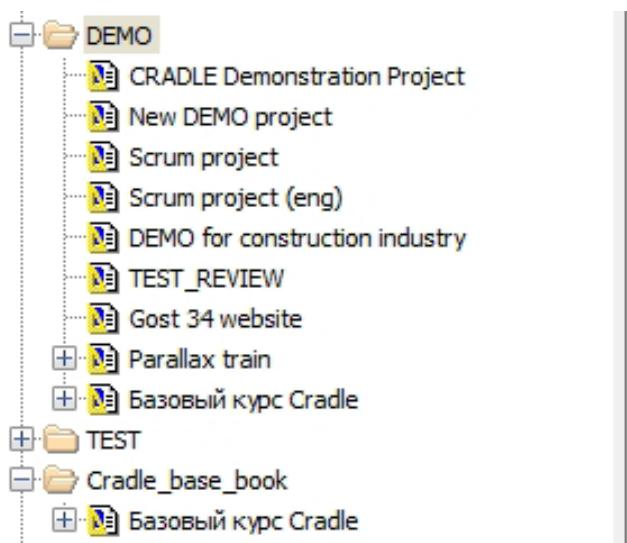


2. Выбрать проект и удалить его через меню **Проект ->Удалить проект**.

## Структурировать проекты

Папки, которые вы видите в дереве проектов, - это логические директории, помогающие структурировать ваши проекты.

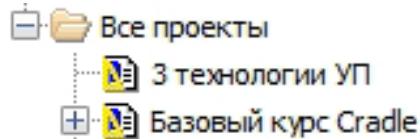
В папках хранятся не сами проекты, а только ссылки на них. Поэтому вы можете разместить ссылки на один и тот же проект в нескольких папках-классификаторах.



Чтобы поместить проект в папку, выберите проект и просто перетащите его в нужную вам папку. При этом в папке будет создана ссылка на данный проект, физически месторасположение проекта не поменяется.

Также вы можете воспользоваться контекстным меню - **Копировать** и **Вставить**.

Если вы случайно удалили ссылки на проект из всех папок (используя меню **Вырезать** или кнопку **Delete**) - не волнуйтесь, он всегда останется в папке *Все проекты*.

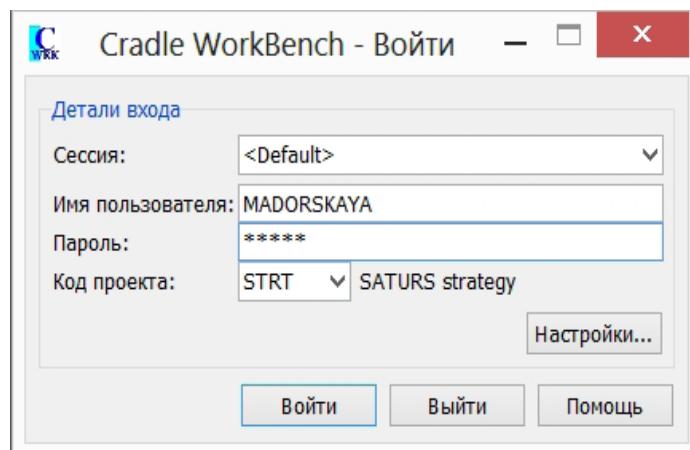


### Войти в проект

1. Обычно, чтобы войти в проект, запускается приложение Workbench:



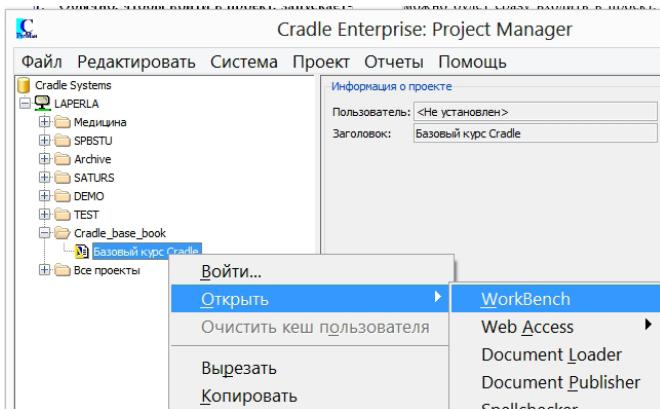
2. После запуска откроется окно входа, в которое необходимо ввести *Имя пользователя*, *Пароль* и выбрать *Код проекта*.



Если на сервере много проектов, то удобнее сразу ввести четырехзначный код проекта, в котором вы хотите поработать.

Если вы забыли код проекта и хотите воспользоваться помощью классификаторов, то:

1. Откройте Project Manager.
2. Найдите необходимый проект в дереве папок.
3. В контекстном меню выберите **Открыть -> Workbench** или другое приложение Cradle.



Если вы часто используете один и тот же проект, то создайте bat-файл, нажав который можно будет сразу входить в проект, не вводя свое имя пользователя и пароль. Для этого:

1. Откройте Блокнот.
2. Введите следующий текст, установив свои параметры доступа:

```
start <> <<с:\Program Files (x86)\Cradle\bin\exe\win32pc\work>> -cds ваш.сервер.Cradle -nobanner -login ИмяПользователя, Пароль, КодПроекта
```

Команда start является командной Windows, которая запускает файл work.exe, что соответствует запуску Workbench. Далее следуют параметры доступа к вашему проекту.

С помощью опций вы указываете:

- cds - сервер Cradle, на котором хранится проект.
  - nobanner - говорит о том, что при старте Workbench не должен показывать свой логотип, это позволяет вам быстрее получить доступ к проекту.
  - login - определяет параметры пользователя, его логин, пароль и четырехзначный код проекта, к которому необходимо получить доступ.
3. Сохраните файл на рабочий стол с расширением bat.

Удобно называть файл по четырехзначному коду проекта, например, DEMO.bat. Вы можете создать такие файлы под все часто используемые проекты. Это позволит избежать ситуации, когда требования записываются на бумаге и теряются только потому, что необходимо потратить время на ввод данных для входа в проект.

В корпоративной сети возможно применение интеграции с LDAP.

## **Загрузить схему проекта**

Загрузить схему проекта можно с помощью операций экспорта/импорта в Workbench или с использованием Project Manager.

Для того, чтобы экспортировать схему, используя Workbench:

1. Откройте Workbench и зайдите в проект, схему которого вы хотите использовать в другом проекте
2. Выберите меню Экспорт и уточните настройки экспорта, выбрав Project Schema для экспорта только схемы.

Для того, чтобы импортировать схему:

3. Откройте Workbench и зайдите в проект для которого заготовили схему на предыдущем этапе.
4. Импортируйте схему из сохраненного файла экспорта с помощью меню Импорт.

# Загрузка исходных требований из спецификаций

## Понять принципы захвата документов

Для загрузки исходных требований из документов MS Word используется приложение **Document Loader**, входящее во все комплектации Cradle и не требующее отдельной лицензии. Document Loader - это часть Cradle Toolsuite для Microsoft Office.

Document Loader позволяет поэлементно загрузить требования в базу Cradle чтобы далее управлять ими как отдельными элементами - назначать статусы, приоритеты, проводить декомпозицию на производные требования, связывать с другими проектными данными для сквозного анализа.

Типичная структура документов с исходными требованиями - это группы параграфов, организованных в иерархию. Document Loader анализирует структуру документа, выделяет все примененные стили и на основе их уровней, а также ваших настроек, создает элементы и иерархические связи между ними в базе проекта Cradle.

Настройки по умолчанию для загрузки документов работают довольно хорошо для большинства случаев.

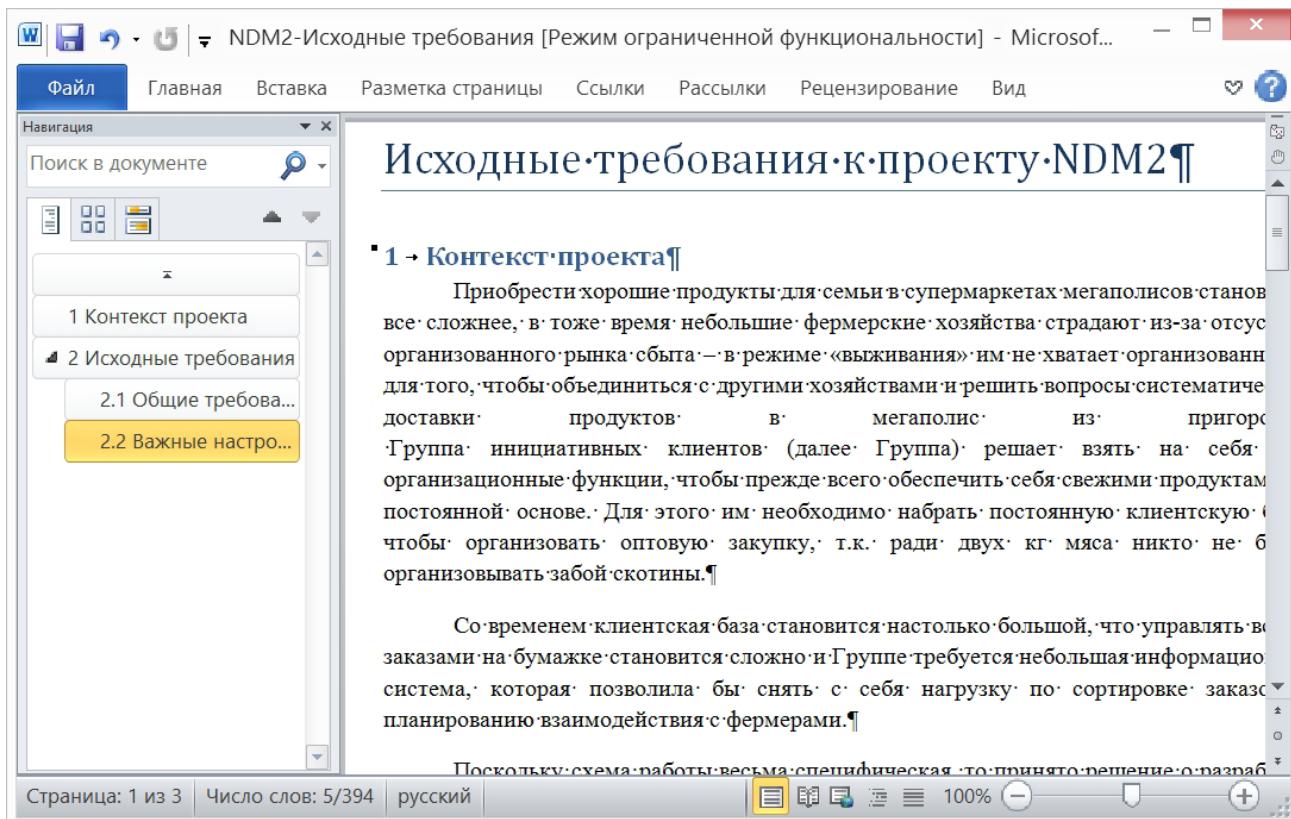
Для того, чтобы сделать свои настройки загрузки и получить иную разбивку спецификации на отдельные требования, необходимо понимать из чего состоит исходный документ Word.

Давайте изучим. Для этого откроем документ в MS Word в режиме отображения специальных символов:

Найдите и включите на панели инструментов следующую кнопку:



Посмотрим на документ:



Файл Главная Вставка Разметка страницы Ссылки Рассылки Рецензирование Вид

# Исходные требования к проекту NDM2

## 1 • Контекст проекта

Приобрести хорошие продукты для семьи в супермаркетах мегаполисов станов все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутс организованного рынка сбыта – в режиме «выживания» им не хватает организованн для того, чтобы объединиться с другими хозяйствами и решить вопросы систематиче доставки продуктов в мегаполисы из пригорода. Группа инициативных клиентов (далее Группа) решает взять на себя организационные функции, чтобы прежде всего обеспечить себя свежими продуктами постоянной основе. Для этого им необходимо набрать постоянную клиентскую базу, чтобы организовать оптовую закупку, т.к. ради двух кг мяса никто не будет организовывать забой скотины.

Со временем клиентская база становится настолько большой, что управлять всеми заказами на бумажке становится сложно и Группе требуется небольшая информационная система, которая позволила бы снять с себя нагрузку по сортировке заказов, планированию взаимодействия с фермерами.

Поскольку схема работы весьма специфическая, то принято решение о разрыве

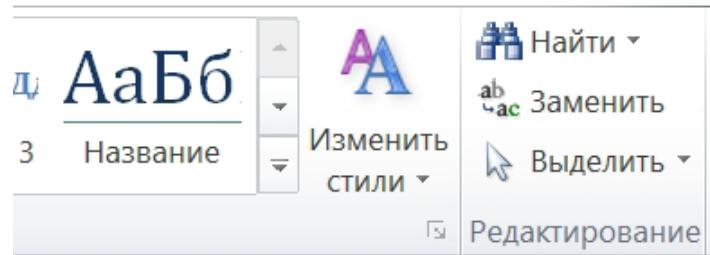
Страница: 1 из 3 | Число слов: 5/394 | русский | 100%

## Применение служебных символов

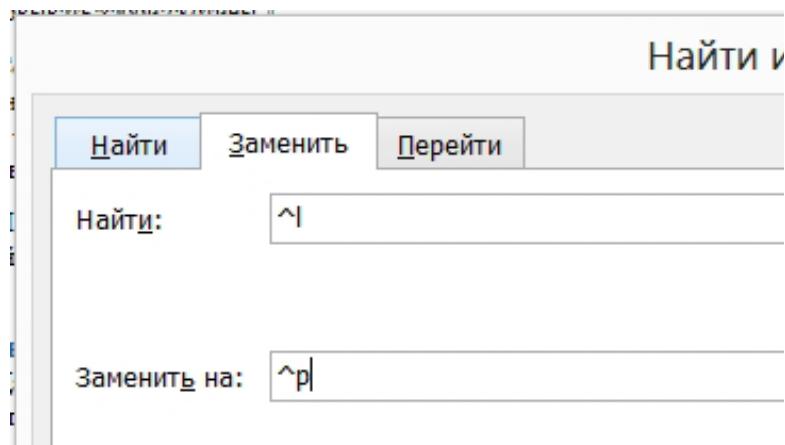
Самый важный специальный символ для нас – это символ завершения абзаца ¶. Именно по нему Cradle разделяет текст документа на отдельные требования, в случае, если к ним применен один и тот же стиль форматирования MS Word.

Обратите внимание, что символ Принудительный разрыв строки (↵), который может использоваться в документах и часто появляется в документах, скопированных из html, не считается как отделение абзаца, поэтому два требования, разделенные этим символом в исходном документе MS Word будут помещены в один элемент базы данных Cradle.

Есть быстрый способ провести замену в документе всех символов (↵) на символы абзаца ¶. Для этого при включенном отображении специальных символов (для контроля) запустите поиск/замену на панели Главная в MS Word:



И в открывшемся диалоге введите:



- $^l$  (эль) - соответствует символу принудительного разрыва строки.
- $^p$  (пэ) - соответствует символу абзаца.

Нажмите **Заменить** для поэлементной замены или **Заменить все** для обработки всего документа за один проход:

Полный список других служебных символов, которые вам может понадобиться заменять можно найти на сайте Microsoft по ссылке:

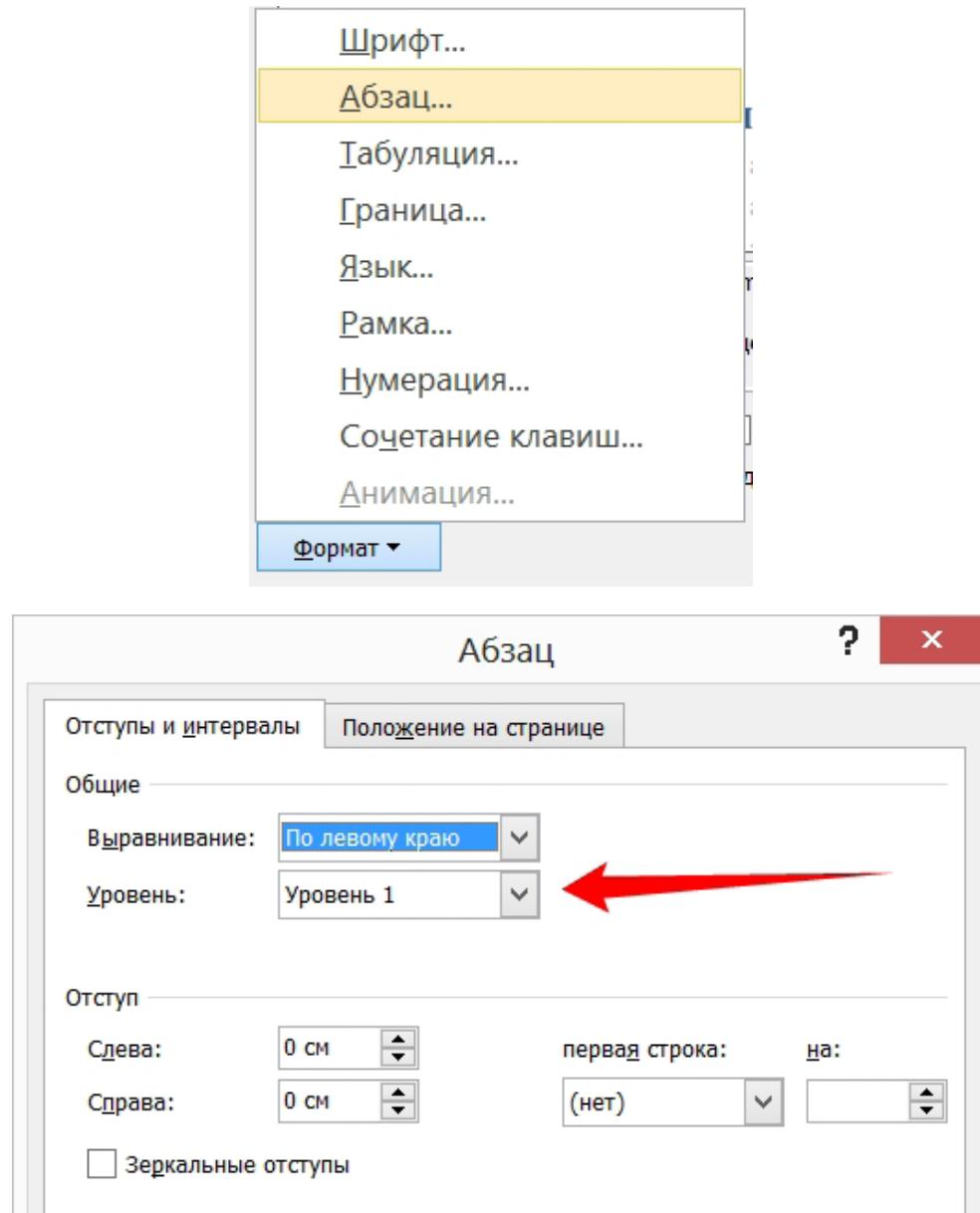
<http://office.microsoft.com/ru-ru/word-help/HA001230392.aspx>

### Применение стилей форматирования текста

Второй, наиболее важный аспект, - это примененные стили, которые обычно называются стилями заголовков. Их отличие от других стилей Word в том, что для каждого стиля заголовка определен уровень заголовка венъ заголовка. Например, на иллюстрации с документом вы видите, что в документе использован всего один стиль заголовка - Заголовок 1, который имеет первый, самый высокий, уровень. Word использует данные стили для отображения панели навига-

ции и автоматического составления содержания документа.

Уровень каждого стиля можно посмотреть и настроить в установках абзаца для каждого стиля.



Document Loader анализирует стили документа и их уровни и по этим данным воссоздает структуру загруженных требований. Структура создается за счет создания связей между требованиями.

## Захват рисунков

Документы с требованиями часто содержат иллюстрации, которые являются или самостоятельными требованиями, либо пояснениями к требованиям. В примере ниже таблица, отражающая требования к формату вывода накладной, представляет собой рисунок.

Такой рисунок можно загрузить в поле (в терминах Cradle -фрейм) типа JPEG или RTF (Rich Text Format).

Водителю требуется накладная на каждый заказ с тем, чтобы он мог правильно  
пакеты из общего груза.

Форма накладной:

«Шанс»  западная часть	Общество с ограниченной ответственностью «Белая ромашка»					
	Накладная на внутреннее перемещение № 17 от 27 сентября 2011 года					
	От подразделения Склад отдела снабжения					
	Подразделению Торговый отдел					
	№ п/п	Наименование	Единица измерения	Количество	Цена	Сумма
1	Стол компьютерный	Штука	1	9 000-00	9 000-00	
2	Стул компьютерный	Штука	2	3 500-00	7 000-00	

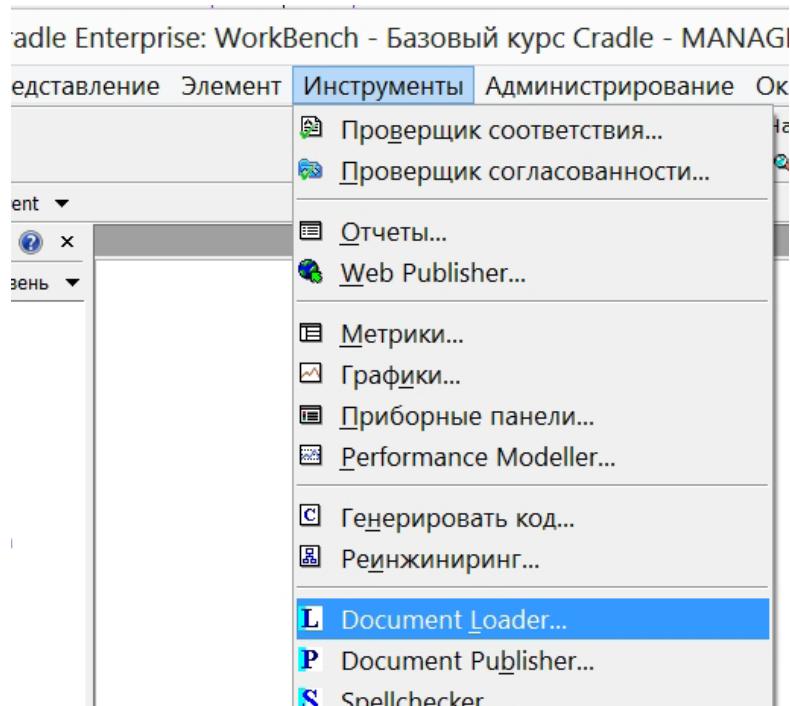
## Захват таблиц

Следующая важная часть документов с требованиями - таблицы. Document Loader может отразить таблицу в одно требование (таблицу целиком) или в несколько отдельных требований (построчно или поэлементно) .

## Загрузить с базовыми настройками Document Loader

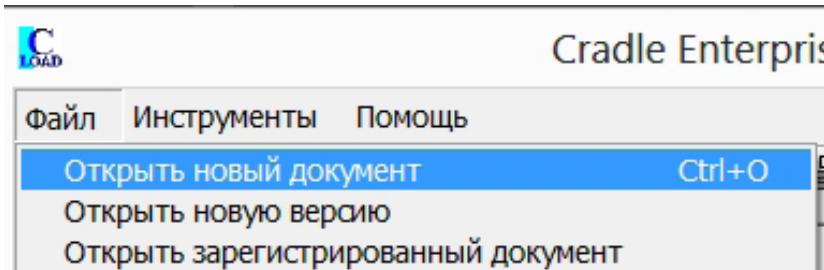
Перед тем как глубоко изучать пользовательские настройки загрузки, выполним ее с настройками по умолчанию и разберем то, что получилось.

### 1. Открыть Document Loader



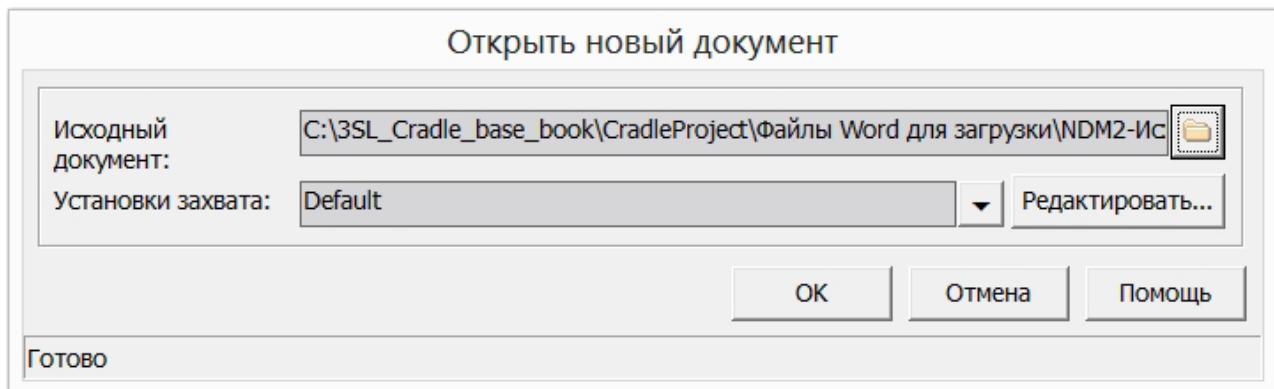
Вы можете также запустить Document Loader и с рабочего стола, но если у вас уже открыт Workbench и вы запустите Document Loader из него, то вам не придется вводить данные для входа. Для запуска Document Loader также можно создать bat-файл, однако это приложение не используется постоянно, поэтому тратить на это время не стоит.

### 2. В меню **Файл** выбрать **Открыть новый документ** или нажать **CTRL+O**



### 3. В появившемся диалоге указать *Исходный документ*.

При этом автоматически выставляются установки захвата по умолчанию (Default).



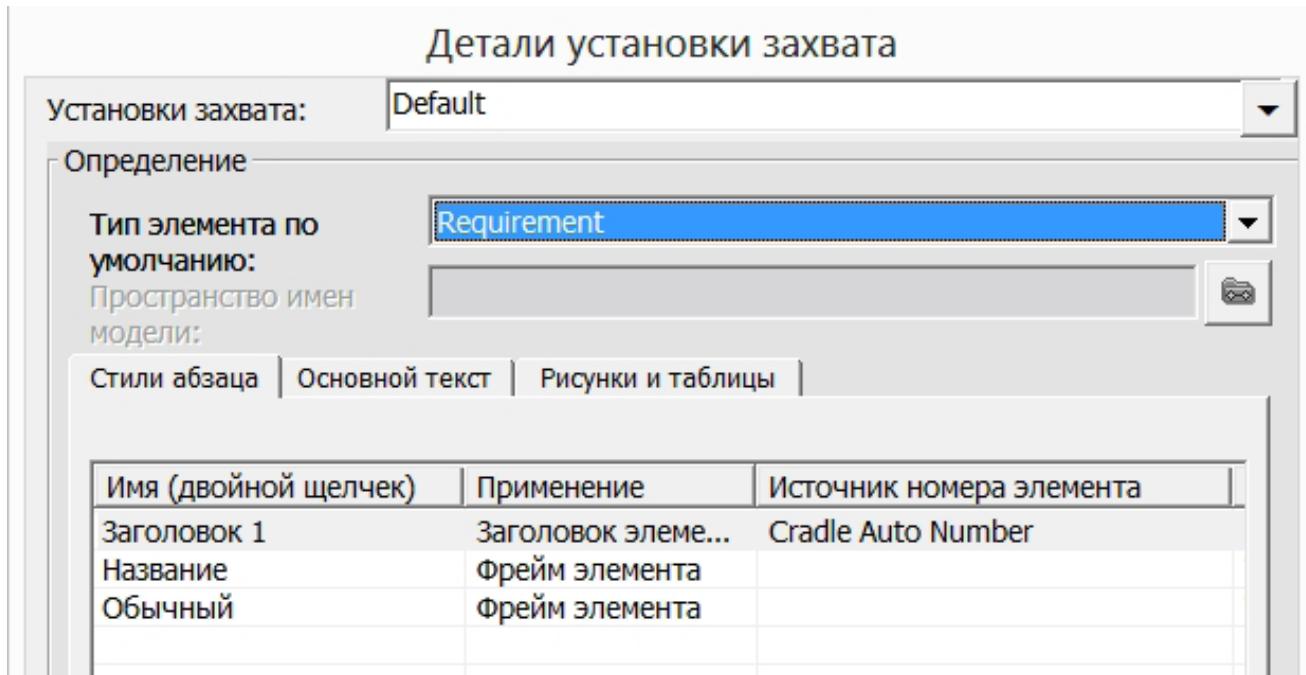
Посмотрим, что стоит за этими установками. Для этого нажмите кнопку **Редактировать**.

Первая, самая важная установка - это тип элемента в базе данных Cradle, в который будет проводиться загрузка выделенных в документе абзацев. По умолчанию - это тип **Requirement**.

На первой вкладке «**Стили абзаца**» для каждого стиля форматирования, найденного в документе, указывается как отображать абзацы с данным стилем:

- Создать отдельный элемент или
- Записать абзац в одно из полей элемента, созданного последним.

По умолчанию Cradle для стилей типа Заголовок 1, Заголовок 2 и т.д., т.е. стилей, имеющих установленный уровень, создает отдельный элемент для каждого абзаца, а для элементов, отформатированными стилями, не имеющими уровня, содержание абзаца дописывается в заданный фрейм последнего созданного элемента.

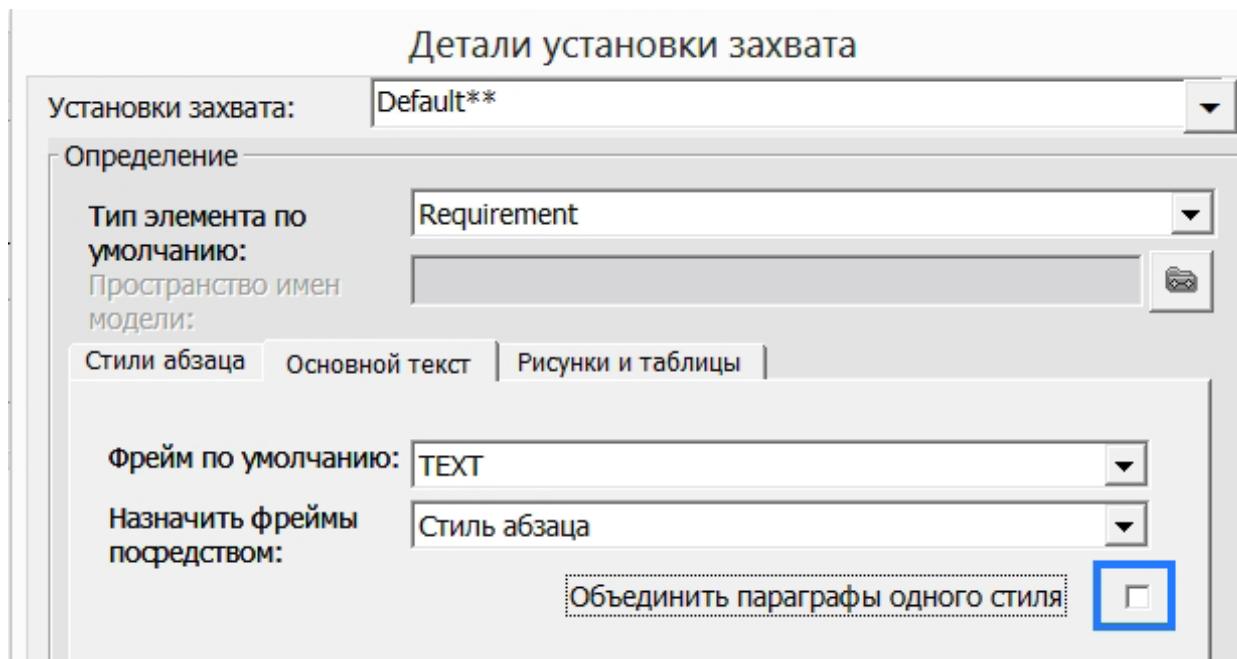


Переопределить эти настройки можно щелкнув на имя стиля (Колонка Имя).

На следующей вкладке основного настроичного диалога загрузки задается фрейм по умолчанию, в который записывается абзац. Очень важным параметром на этой вкладке является чекбокс *Объединить параграфы одного стиля*.

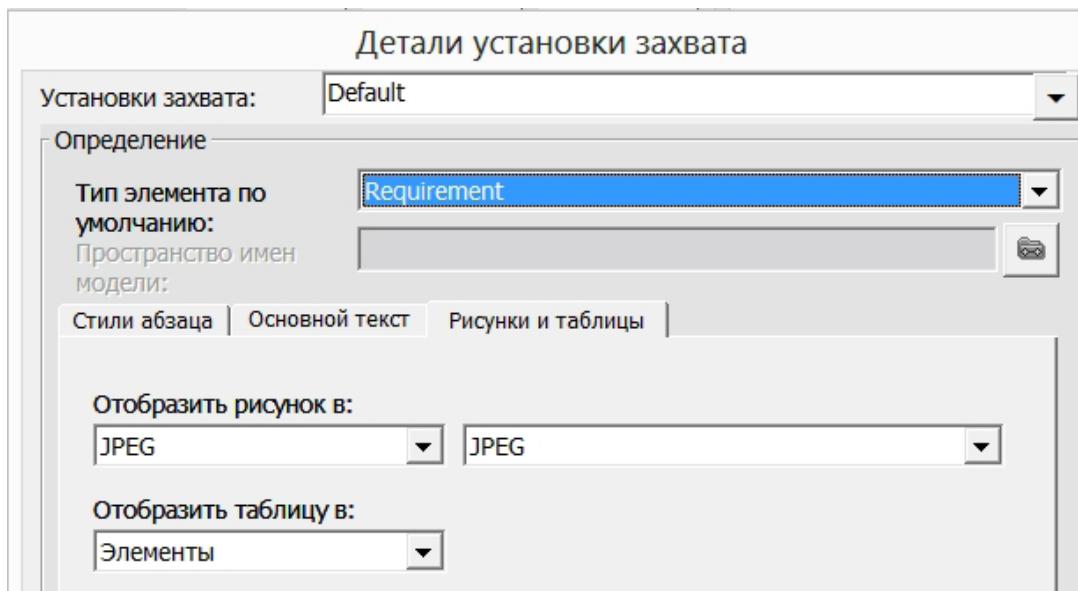
Эта установка включена по умолчанию - абзацы одного стиля объединяются и записываются в один элемент. Однако чаще всего удобнее создать для каждого абзаца отдельный элемент. Это позволяет разбить спецификацию на более мелкие единицы и точнее проводить управление статусами, приоритетами, осуществлять трассировку.

Для этой загрузки снимем галку.



Последняя вкладка определяет способ обработки изображений и таблиц. Как правило для сохранения изображений используют отдельное поле типа JPEG, но можно также загружать рисунки в поле типа RTF или бинарный документ Word.

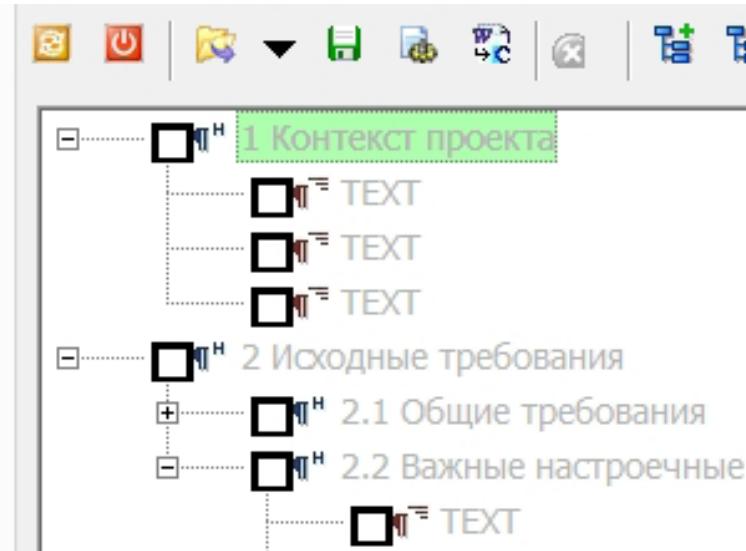
Для таблиц, по умолчанию, способ загрузки - поэлементный, но их можно загружать и в виде картинок, RTF-фреймов и бинарных документов Word. Для простоты последующей обработки мы рекомендуем загружать таблицы поэлементно.



Итак, мы с вами немного познакомились с тем, что стоит за настройками по умолчанию, которые собираемся использовать для первой загрузки.

#### 4. Нажать OK

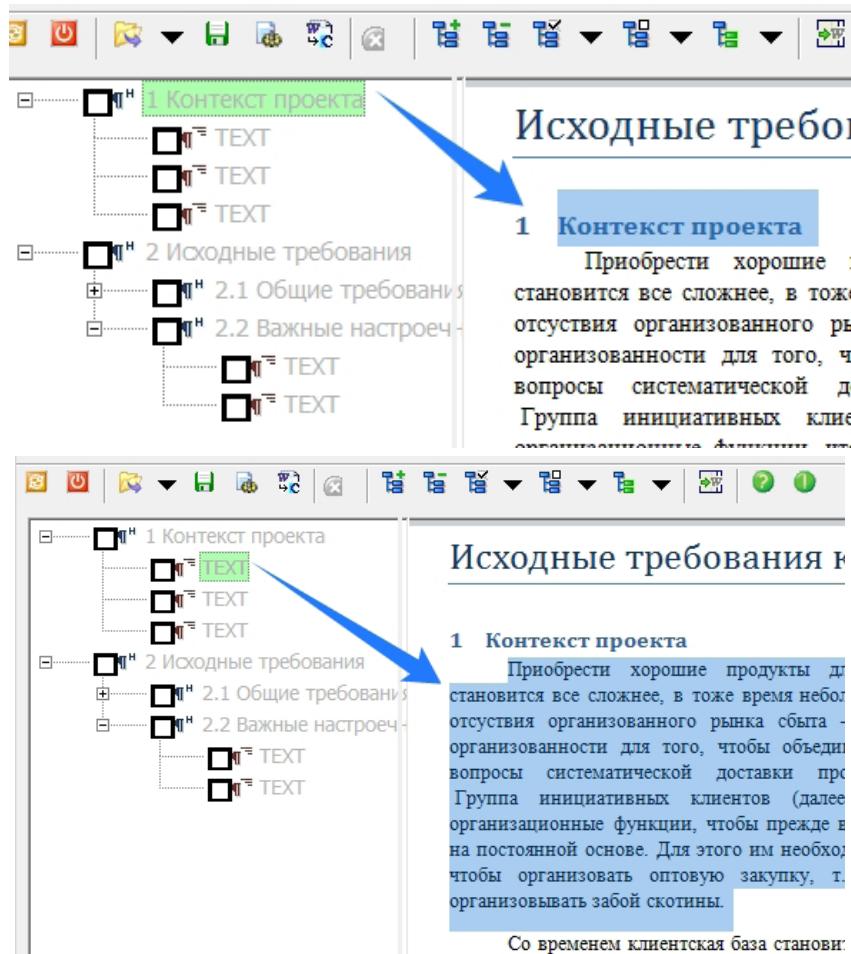
После этого Document Loader проведет анализ документа и покажет полученные на его основе элементы. Далее вы сможете выбрать те из них, которые являются требованиями и которые необходимо загрузить в базу проекта.



Для того, что контроллировать выбор элементов, откройте панель исходного документа с помощью кнопки

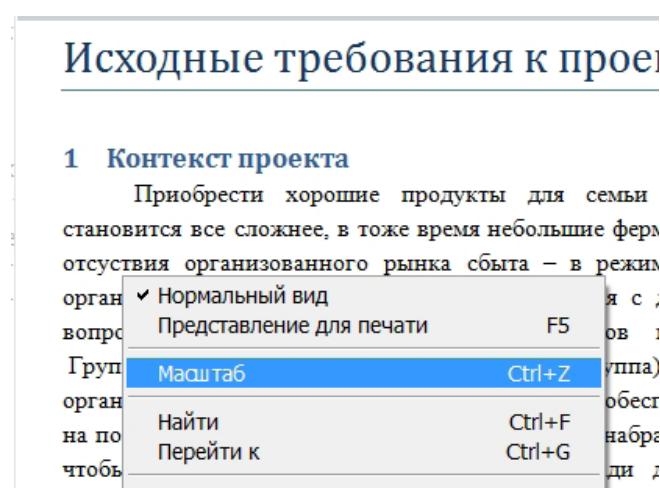


Теперь, когда вы будете вы будете переходить по элементам в дереве, в исходном документе будут подсвечиваться соответствующие абзацы

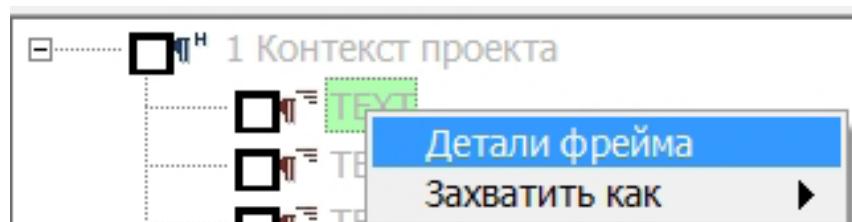


Попробуйте это, щелкая на элементы в дереве.

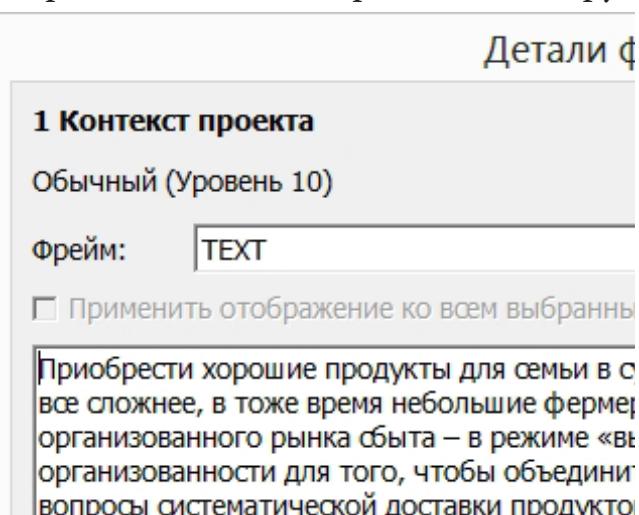
При необходимости вы можете воспользоваться контекстным меню открытой панели документа и увеличить/уменьшить масштаб отображения исходного документа:



В контекстном меню **Детали элемента** для каждого элемента можно посмотреть и при необходимости установить индивидуальные параметры загрузки:



В деталях указано, в какой тип элемента будет загружаться данный абзац, его номер, а также конкретное поле, в которое пойдет загрузка:



Обратите внимание, что при необходимости можно изменить данные настройки для отдельного элемента или группы выделенных элементов. Таким образом, можно точечно скорректировать правила загрузки элементов.

После того, как все установки сделаны, можно выбрать элементы для загрузки, в данном случае - все.

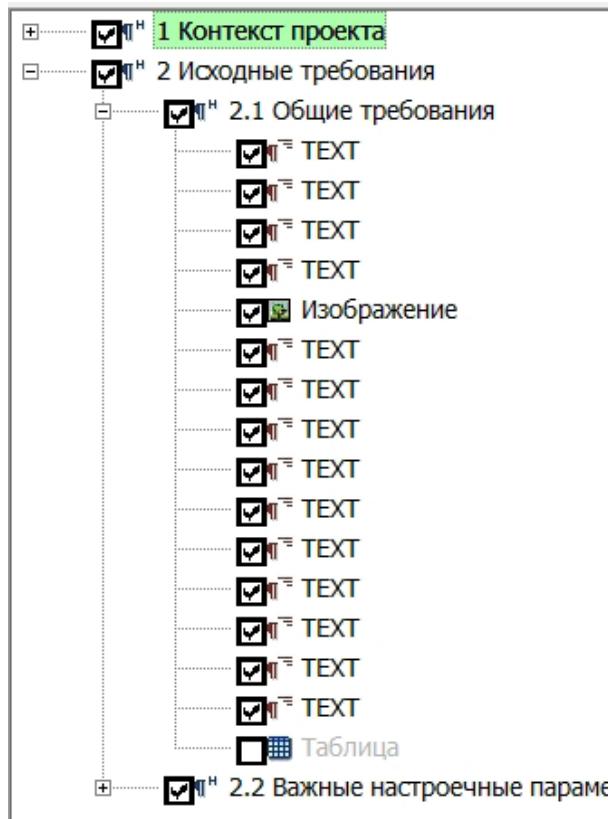


Таблица тоже подлежит загрузке, однако при ее выборе нам понадобится сделать дополнительные настройки. Если в документе много однотипных таблиц, то такие настройки выполняются один раз.

Есть несколько вариантов отображения таблиц в элементы базы данных:

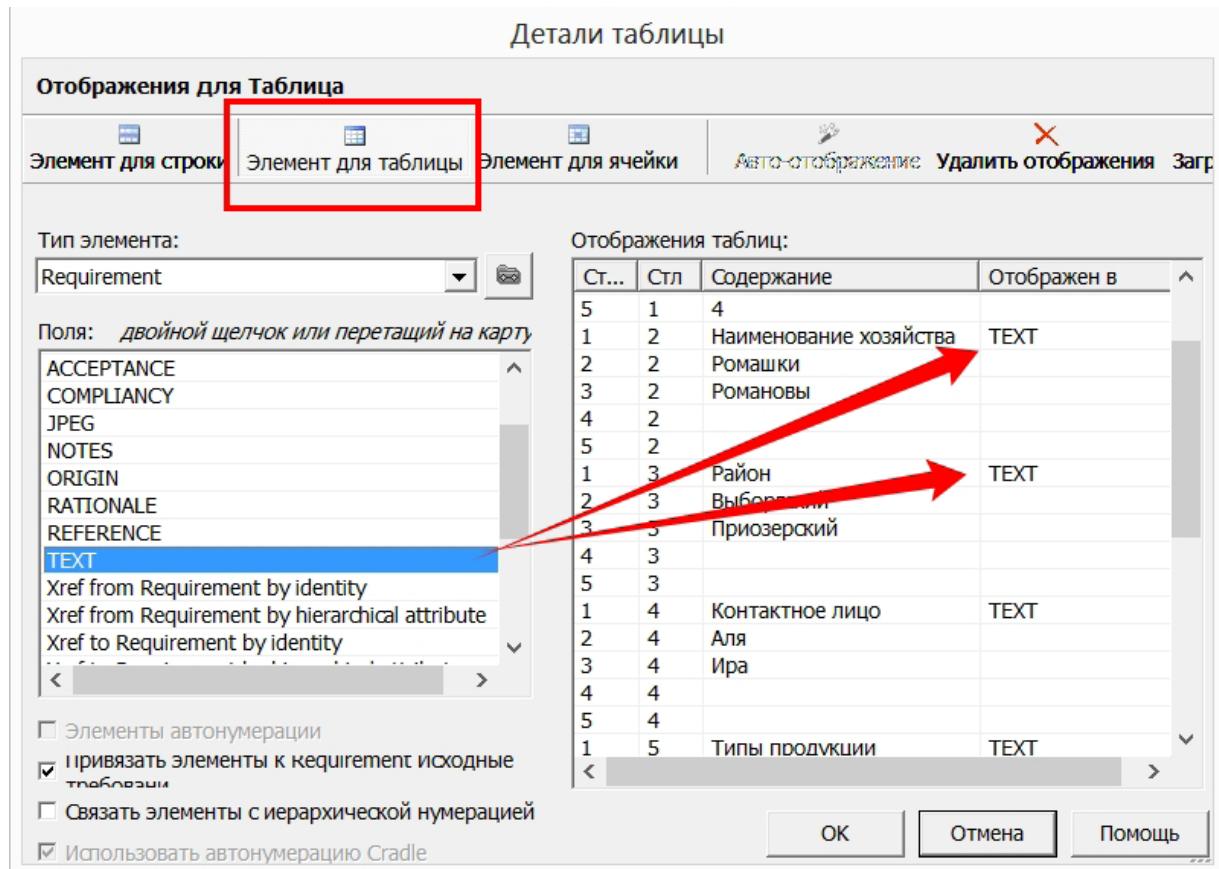
- один элемент для каждой строки таблицы;
- один элемент для каждой ячейки таблицы;
- один элемент для всей таблицы.

Отобразим данную таблицу целиком в один элемент. При этом можно указать, какие ячейки необходимо отображать и в какие поля. В данном случае нас интересуют только заголовки столбцов, поскольку они лягут в основу информационной модели предметной области, поэтому все эти заголовки отобразим в поле TEXT. Для этого:

1. Выбрать режим **Элемент для таблицы**.
2. Выбрать необходимую строку таблицы.
3. Выбрать и два раза щелкнуть по полю.

4. Повторить для каждой требуемой строки.

5. Нажать кнопку **OK**.



Всё, теперь можно загружать документ. Для этого:

1. Нажать кнопку **Захватить документ**



2. Установить атрибуты загружаемого документа:

- *Имя* (рекомендуем вводить короткое наименование на английском. Это упростит перенос проектов между Linux/Windows системами)
- *Выпуск*
- *Ссылка*
- *Дата*
- *Имя верхнеуровневого элемента*, который будет соответствовать документу в целом.

После того как все атрибуты установлены можно нажимать кнопку **Захватить**.

Результаты захвата выводятся в виде отчета. Обязательно просматривайте его весь.

Захватить документ

Детали исходного документа

Имя:	SourceReq
Версия:	01
Заголовок:	NDM2-Исходные требования
Тип документа:	▼
Классификация:	UNCLASSIFIED
Выпуск:	1
Дата выпуска:	фев 2014
Ссылка:	1
Комментарии:	
Расположение:	C:\3SL_Cradle_base_book\CradleProject\Файлы Word для загрузки\NDM2-Ио <input type="button" value="&lt;"/> <input type="button" value="&gt;"/>

Примечание: Типлэт - issue, дата issue и связи должен быть уникальный.

Элемент верхнего уровня

Тип элемента:	Requirement	▼	<input type="button" value=""/>
Индекс:	<auto generated>		
Имя:	Исходные требования Заказчика		

Настройки захвата

Воспроизвести внутренние ссылки документа как связи

Разрешить отмену

С

Результаты захвата

Операция	Детали	Результат	Время
✓ Создать элемент	Requirement 16 Исходные требования Заказчика (A)	Выполнено	14:28:27
✓ Зарегистрировать исходные требования	SOURCEREQ:01	Выполнено	14:28:27
✓ Создать элемент	Requirement 17 Контекст проекта (A)	Выполнено	14:28:27
✓ Создать связь	От Requirement 16 (A) к Requirement 17 (A)	Выполнено	14:28:27
✓ Создать связь	От Source Statement BM_01_2 к Requirement 17 (A)	Выполнено	14:28:27
✓ Добавить фрейм	TEXT фрейм Requirement 17 Контекст проекта (A)	Выполнено	14:28:27
✓ Добавить фрейм	TEXT фрейм Requirement 17 Контекст проекта (A)	Выполнено	14:28:27
✓ Добавить фрейм	TEXT фрейм Requirement 17 Контекст проекта (A)	Выполнено	14:28:27
✓ Создать элемент	Requirement 18 Исходные требования (A)	Выполнено	14:28:28
✓ Создать связь	От Requirement 16 (A) к Requirement 18 (A)	Выполнено	14:28:28
✓ Создать связь	От Source Statement BM_01_6 к Requirement 18 (A)	Выполнено	14:28:28
✓ Создать элемент	Requirement 19 Общие требования (A)	Выполнено	14:28:28
✓ Создать связь	От Requirement 18 (A) к Requirement 19 (A)	Выполнено	14:28:28
✓ Создать связь	От Source Statement BM_02_7 к Requirement 19 (A)	Выполнено	14:28:28
✓ Добавить фрейм	TEXT фрейм Requirement 19 Общие требования (A)	Выполнено	14:28:28
✓ Добавить фрейм	TEXT фрейм Requirement 19 Общие требования (A)	Выполнено	14:28:28
✓ Добавить фрейм	TEXT фрейм Requirement 19 Общие требования (A)	Выполнено	14:28:28
✓ Добавить фрейм	TEXT фрейм Requirement 19 Общие требования (A)	Выполнено	14:28:28
✓ Добавить фрейм	JPEG фрейм Requirement 19 Общие требования (A)	Выполнено	14:28:29
✓ Добавить фрейм	TFXT фрейм Requirement 19 Общие требования (A)	Выполнено	14:28:29
✓ Добавить фрейм	TFXT фрейм Requirement 19 Общие требования (A)	Выполнено	14:28:29

Создано 6 элементов  
Изменено 0 элементов

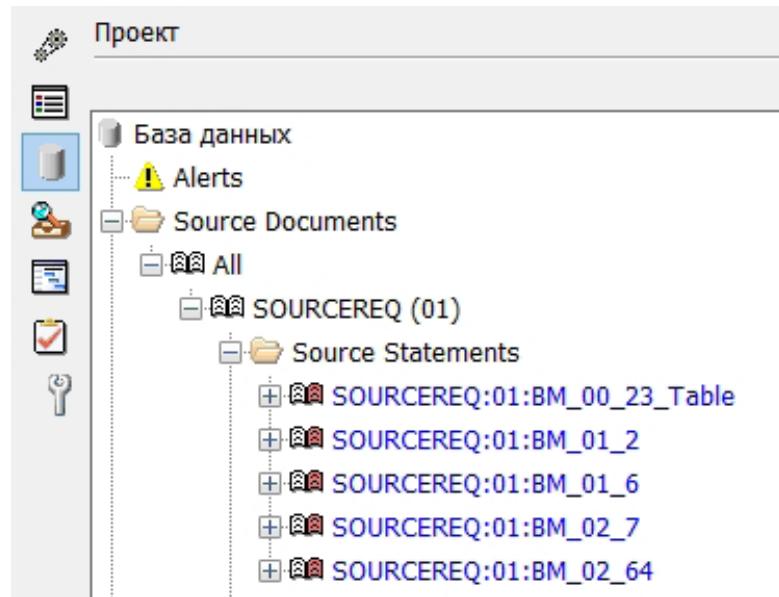
Закрыть Отчет Помощь

Одна из наиболее частых причин появления ошибок в данном отчете - невозможность создать связи между элементами, т.к. в схеме проекта установлено правило связей «Запретить все остальные», а необходимые типы связей не разрешены. В данном проекте такое правило не установлено и загрузка прошла успешно.

Всё, элементы загружены, теперь можем работать с ними в проекте. Проверим загрузку непосредственно в Workbench.

## Проверить загрузку исходных требований

1. Открыть Workbench
2. Перейти в раздел База Данных -> Source Documents

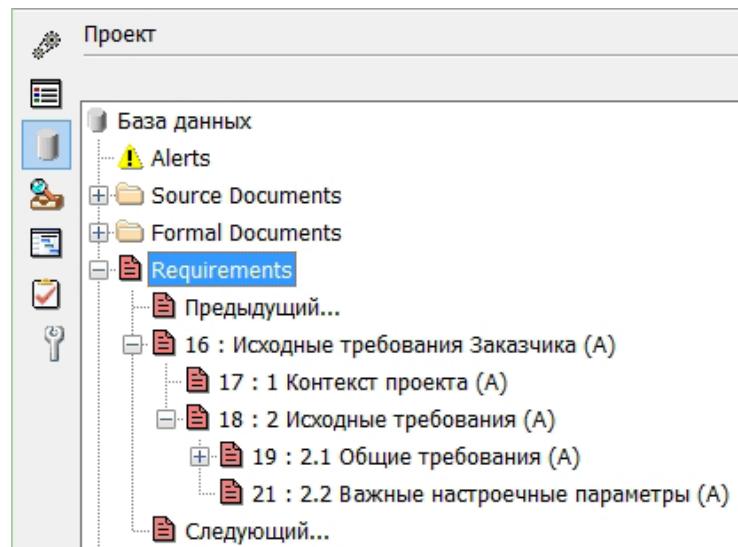


При загрузке документа Cradle создает в исходном документе закладки, чтобы всегда можно было быстро вернуться к источнику. Source Statements - это и есть ссылки на закладки. Щелкнув на любую из них, вы увидите, что исходный документ откроется ровно в том месте, которому соответствует данная закладка.

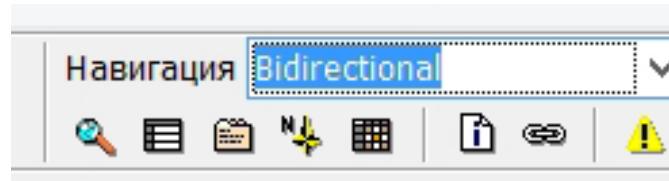
Данные элементы (Source Statements) не содержат исходного текста требований, а являются всего-лишь ссылками на соответствующие элементы документа. Чтобы посмотреть сами загруженные элементы, необходимо:

### 3. Перейти в раздел базы данных Requirements

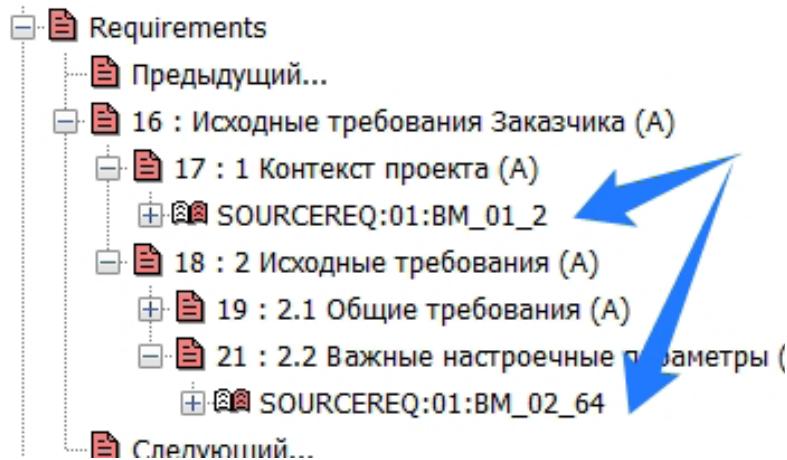
Это тот тип элемента, который мы видели в настройках загрузки документа.



Здесь мы видим уже загруженные элементы. Если мы переключим навигацию (в правом верхнем углу) в двухсторонний режим:

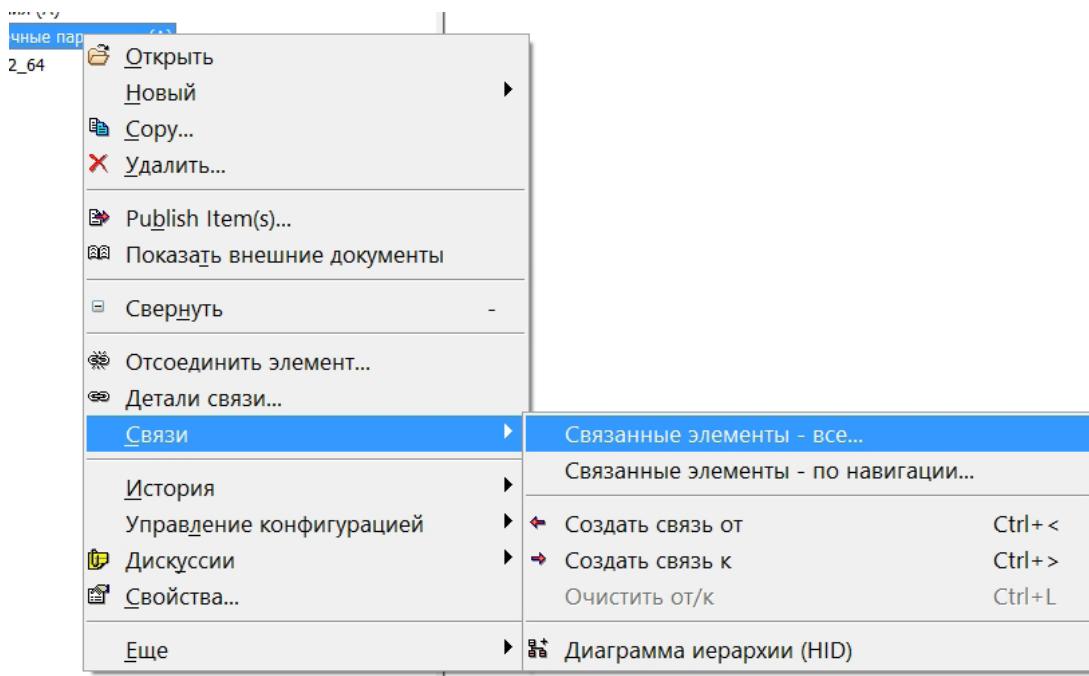


и щелкнем несколько раз на любое из загруженных требований:

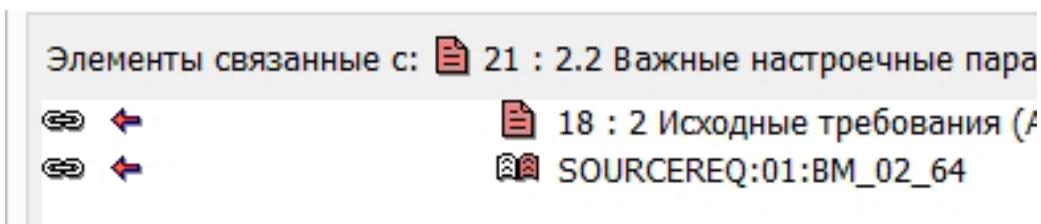


то увидим, что каждое загруженное требование связано с соответствующей закладкой в исходном документе. Таким образом, в каком бы далее представлении мы не открыли исходные требования, мы легко сможем перейти к источнику. Щелкните на любой из связанных с требованиями элементов SOURCEREQ и Word откроет документ, выделив соответствующее исходное требование. Попробуйте это!

Даже если мы работаем с другой навигацией, при которой связанные элементы типа Source Statement не отображаются, мы можем легко их найти с помощью контекстного меню каждого элемента **Связи ->Связанные элементы -все**:

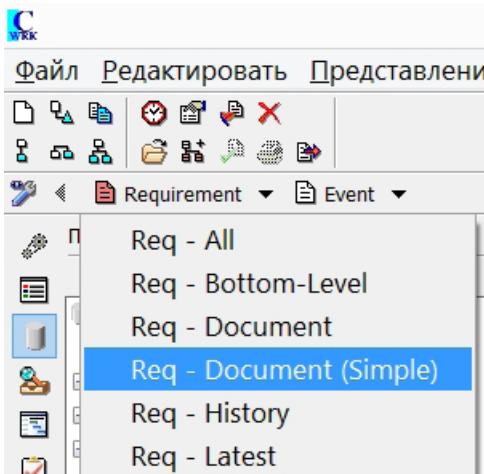


После выбора этого меню откроется диалог, который отразит все элементы, связанные с выбранным. В этом списке легко найти ссылки на Source Statements по их иконкам в виде открытой книжки . В открывшемся диалоге можно выбрать соответствующий элемент и он откроется.



Далее откроем представление, которое поможет на увидеть все загруженные элементы в виде документа.

#### 4. Запустите запрос Req-Document (Simple)



На рабочей панеле откроется представление Req - Document (Simple), которое линейно отобразит все загруженные элементы.

Обратите внимание, что номера заголовков были автоматически загружены в поле Key и благодаря этому Cradle применяет форматирование и в этом представлении.

Cradle Enterprise: WorkBench - Базовый курс Cradle - MANA

Файл Редактировать Представление Элемент Запрос Инструменты Администрирование Окно Помощь

Представление Req - Document (Simple)

Requirement Event

Проект Верхний уровень

База данных

- Alerts
- Source Documents
- Formal Documents
- Requirements
  - Предыдущий...
  - 16 : Исходные требования Заказчика (A)
    - 17 : 1 Контекст проекта (A)
      - SOURCEREQ:01:BM\_01\_2
    - 18 : 2 Исходные требования (A)
      - 19 : 2.1 Общие требования (A)
        - 21 : 2.2 Важные настроочные параметры (A)
          - SOURCEREQ:01:BM\_02\_64
    - Следующий...
  - Events
  - Essential Domain
  - Implementation Domain
  - Hierarchy (HID)
  - Project Plans

Запрос: Req - Document (Simple)

1 Исходные требования Заказчика

2 Контекст проекта

Приобрести хорошие продукты для семьи в супермаркетах мегаполисов становится все сложнее, не хватает организованности для того, чтобы объединиться с другими хозяйствами и решить проблему инициативных клиентов (далее Группа) решает взять на себя эти организационные функции клиентской базы, чтобы организовать оптовую закупку, т.к. ради двух кг мяса никто не будет

Со временем клиентская база становится настолько большой, что управлять всеми заказами, сортировкой заказов и планированием взаимодействия с фермерами.

Поскольку схема работы весьма специфическая, то принято решение о разработке заказной и

2 Исходные требования

2.1 Общие требования

Данная ИС должна обеспечивать интерфейс для заказывающих, с возможностью указать по

Время и день доставки определяет Группа на основе возможностей фермеров, в основном д

Водителю требуется накладная на каждый заказ с тем, чтобы он мог правильно выбрать пак

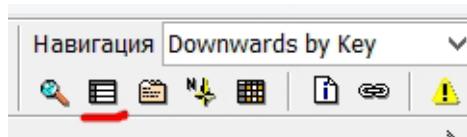
Форма накладной:

При составлении заказа клиент должен иметь возможность указать примечания по разделке

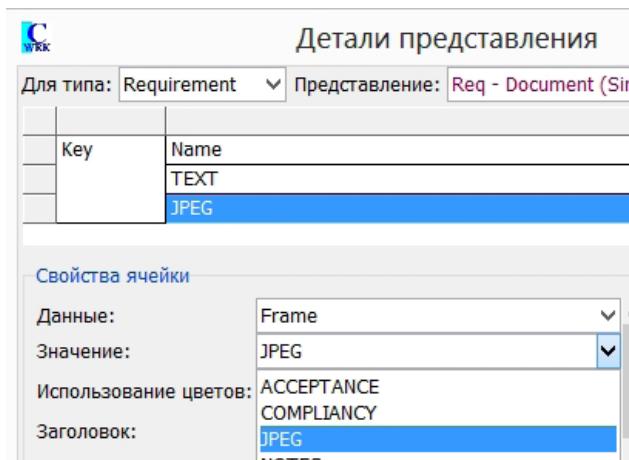
Если нам необходимо, мы можем редактировать элементы прямо в таких представлениях, однако редактирование исходных требований - это плохая практика.

Просмотрев все представление полностью мы увидим, что рисунки, загруженные в фрейм JPEG не отобразились. Внесем корректировки в настройку представления, для этого:

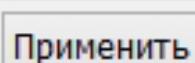
##### 5. Вызвать диалог **Детали представления**



##### 6. Заменить поле NOTES на поле JPEG.



##### 7. Применить представление, нажав кнопку:

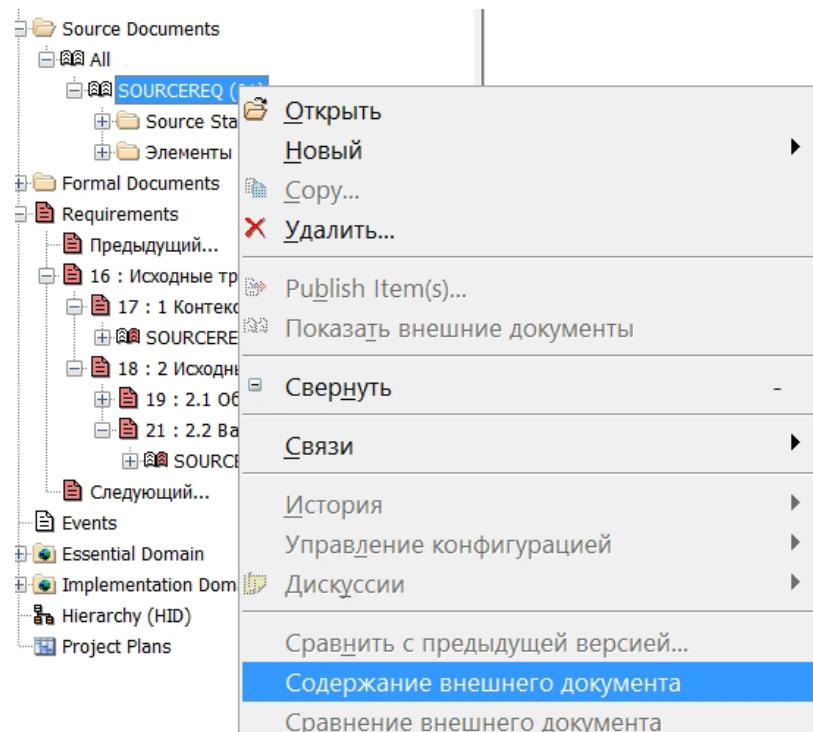


Представление обновиться и мы увидим, что теперь таблица, которая была в виде рисунка, также вывена в общий поток требований.

Мы также можем щелкнуть на любой из элементов и посмотреть, как выглядит каждое отдельное загруженное требование. Попробуйте это сделать.

Итак, мы проверили, что наши исходные требования успешно загружены.

## Запустить отчет о содержании

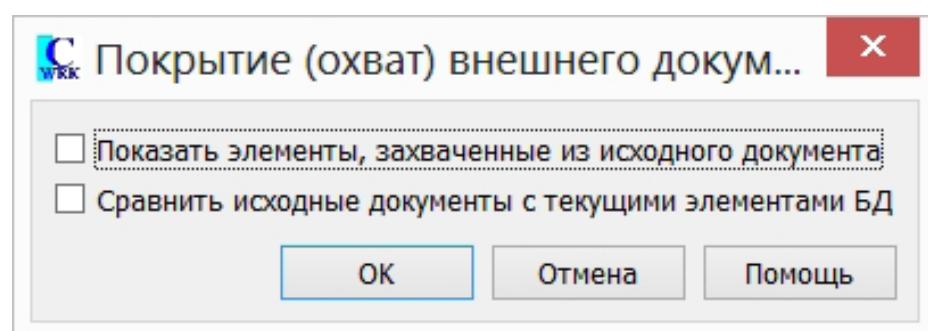


## Содержание внешнего документа

Items associated with Source Document - SOURCEREQ (01):

Info Type	Domain	Namespace	Subtype	Identity	Version	Draft
Requirement				17		A
Requirement				18		A
Requirement				19		A
Requirement				20		A
Requirement				21		A

## Запустить отчет о покрытии



SOURCEREQ-coverage [Режим ограниченной функциональности] - Microsoft Word некомм...

Файл Главная Вставка Разметка страницы Ссылки Рассылки Рецензирование Вид

## Исходные требования к проекту NDM2

### 1 Контекст проекта

Приобрести хорошие продукты для семьи в супермаркетах мегаполисов становится все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выживания» им не хватает организованности для того, чтобы объединиться с другими хозяйствами и решить вопросы систематической доставки продуктов в мегаполис из пригородов.

Группа инициативных клиентов (далее Группа) решает взять на себя эти организационные функции, чтобы прежде всего обеспечить себя свежими продуктами на постоянной основе. Для этого им необходимо набрать постоянную клиентскую базу, чтобы организовать оптовую закупку, т.к. ради двух кг мяса никто не будет организовывать забой скотины.

Со временем клиентская база становится настолько большой, что управлять всеми заказами на бумажке становится сложно и Группе требуется небольшая информационная система, которая позволила бы снять с себя нагрузку по сортировке заказов и планированию взаимодействия с фермерами.

Поскольку схема работы весьма специфическая, то принято решение о разработке заказной ИС.

### 2 Исходные требования

#### 2.1 Общие требования

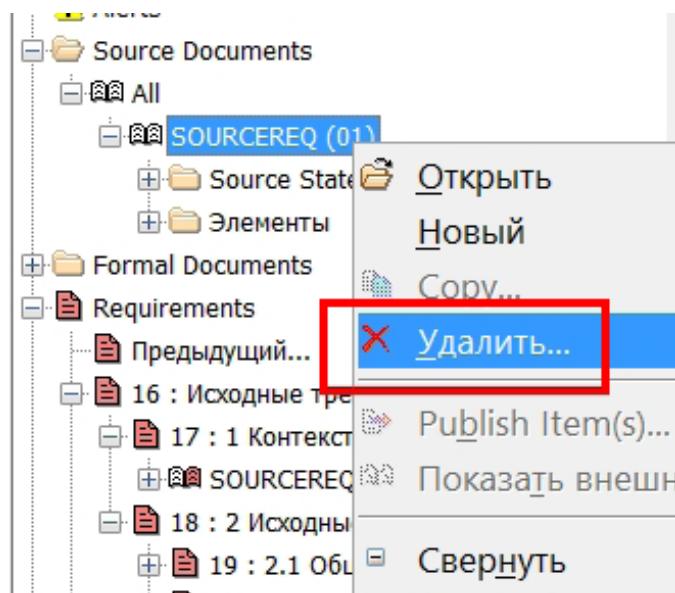
Ланная ИС должна обеспечивать интерфейс для заказывающих, с возможностью указать

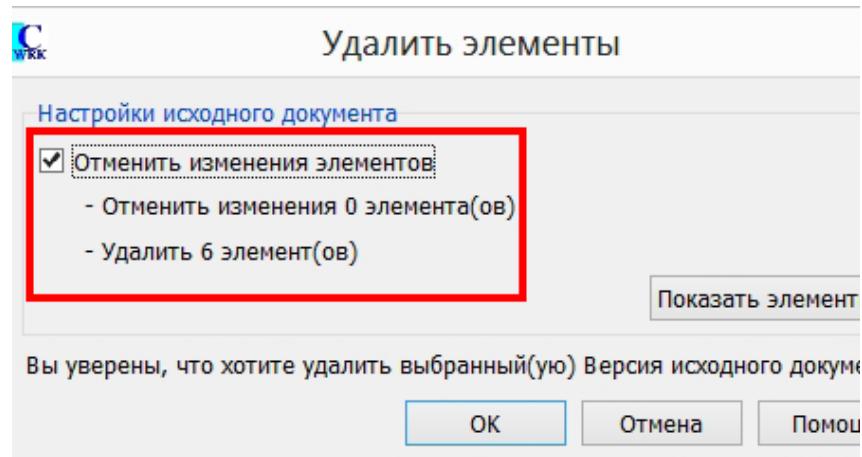
Число слов: 394 | русский |

Примечание [CRADI Requirement 17 Контекст проекта (A)]

Примечание [CRADI Requirement 18 Исходные требования (A)]

## Отменить загрузку



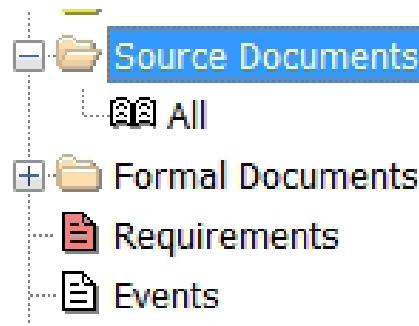


## Удалить версию исходного документа

*SOURCEREQ:01 - NDM2-Исходные требования*

### Отменить изменения элементов

Тип элемента	Элемент	Операция	Комментарии
Requirement	16 : Исходные требования Заказчика	Удалить	
Requirement	17 : 1 Контекст проекта	Удалить	
Requirement	18 : 2 Исходные требования	Удалить	
Requirement	19 : 2.1 Общие требования	Удалить	
Requirement	20 :	Удалить	
Requirement	21 : 2.2 Важные настроочные параметры	Удалить	

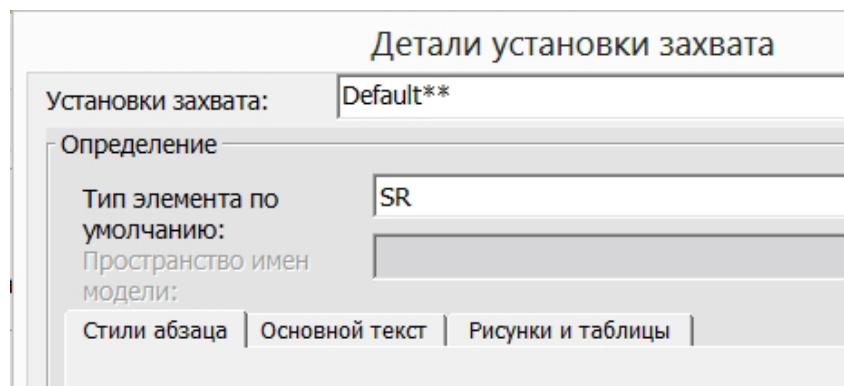


## Уточнить настройки загрузки и повторно загрузить документ

Скорректируем базовые настройки так, чтобы загружать исходные требования в элементы типа SR, в поле TEXT, независимо от того заголовок это или нет. Это упростит нам дальнейшее построение представлений для анализа и публикацию отчетов. Для этого:

1. В установках захвата указать

*Тип элемента по умолчанию* = SR



2. Для стиля *Абзац списка* в настройках отображения указать

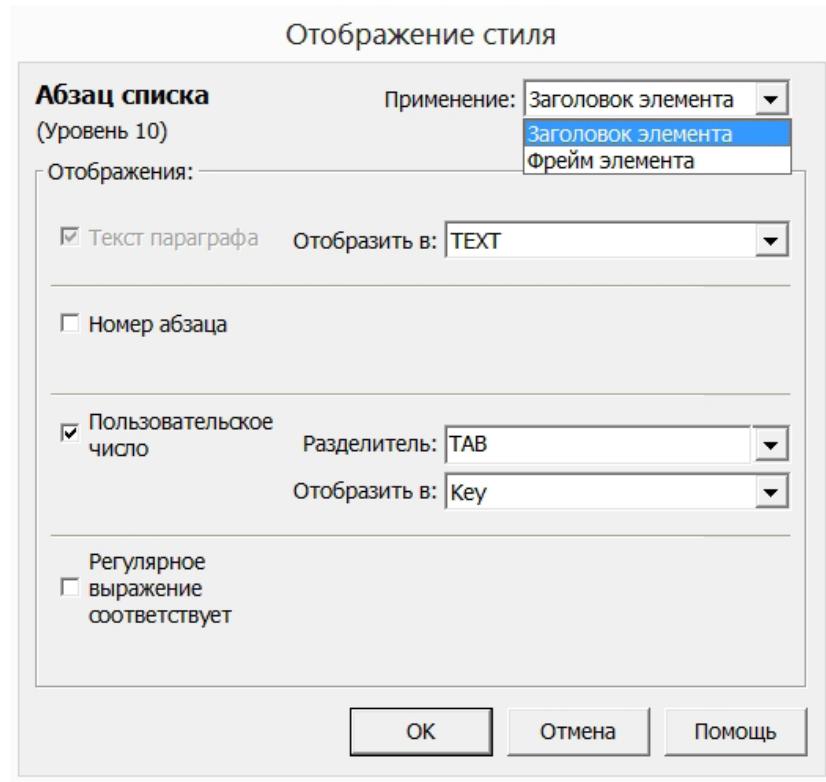
*Применение* = Заголовок элемента

Эту настройку необходимо выполнить для всех найденных в документе стилей, щелкнув по каждому стилю на закладке **Стили абзаца**

Для стиля *Абзац списка* также отобразим номер в списке в поле Key, для этого:

3. Поставить галку напротив *Пользовательское число, Разделитель установить в значение TAB, а для параметра Отобразить в установить значение Key*.

- 4.



В итоге у вас должны получиться следующие настройки для всех элементов:

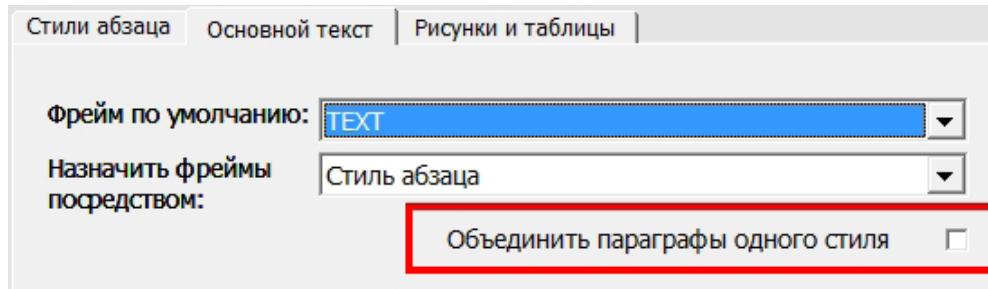
- *Применение* = Заголовок элемента
- *Источник номера* = Cradle Auto Number
- *Paragraph Text Mapping* = TEXT

Вы можете проконтролировать эти настройки на вкладке **Стили абзаца**

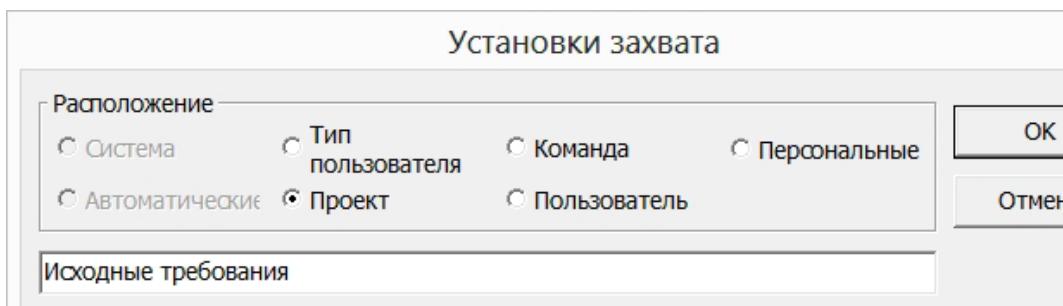
Стили абзаца						
Имя (двойной щелч...)	Применение	Источник номера элемента	Para. Text Mapping	Para. Num Ma...	User-Def...	
Абзац списка	Заголовок элемент...	Cradle Auto Number	TEXT		Key	
Заголовок 1	Заголовок элемент...	Cradle Auto Number	TEXT	Key		
<b>Заголовок 2</b>	<b>Заголовок элемент...</b>	<b>Cradle Auto Number</b>	<b>TEXT</b>			
Название	Заголовок элемент...	Cradle Auto Number	TEXT			
Обычный	Заголовок элемент...	Cradle Auto Number	TEXT			

Выполним последнюю важную настройку и сохраним установки захвата.

5. Снять галку *Объединить параграфы одного стиля*



6. Сохранить выполненные установки захвата в разделе Проект под новым именем «Исходные требования»



Все настройки в Cradle могут быть сохранены с разными уровнями доступа. При сохранении в Проект, настройки становятся доступны всем пользователям проекта (версия Enterprise).

После сохранения установок Cradle проведет анализ документа и откроется дерево элементов для захвата.

Также как и в первый раз Вы можете открыть панель MS Word и пробежаться по элементов, проконтроллировав их соответствие.

Среда разработки Cradle Enterprise Document Loader, открытый документ "NDM2-Исходные требования.docx".

Левая панель: Структура документа в виде деревьев требований. Узлы 1.1 и 1.2 выделены в зеленый цвет.

Правая панель: Текстовый редактор с открытым документом "NDM2-Исходные требования.docx".

Содержание документа:

- 1. Исходные требования к проекту NDM2
- 1.1 Контекст проекта
  - Приобрести хорошие продукты для семьи в супермаркете
  - Со временем клиентская база становится настолько большой, что...
  - Поскольку схема работы весьма специфическая, то принцип...
- 1.2 Исходные требования
  - 2.1 Общие требования
    - Данная ИС должна обеспечивать интерфейс для заказов...
    - Время и день доставки определяет Группа на основе...
    - Водителю требуется накладная на каждый заказ с...
    - Форма накладной:
      - Изображение
    - При составлении заказа клиент должен иметь возможность...
    - Доставка окончательно согласовывается по телефону...
    - С заказами работает несколько человек – в зависимости от доступного времени.**
    - Поскольку необходимо брать по много кг мяса, то...
    - Окончательную сумму заказа можно узнать только...
    - За полчаса до подъезда доставки также отсылается смс.
    - Поступившие заказы необходимо сортировать по фермерам и датам...
    - Возможность делиться и обсуждать рецепты.
    - Интерфейс минималистичный
  - 2.2 Важные настроочные параметры
    - Время оповещения перед доставкой
    - Главный менеджер

Выбрав любой из элементов можно посмотреть и при необходимости изменить его персональные установки.

## Детали элемента

Далее необходимо:

## 7. Выбрать все элементы, кроме первого

8. Настроить установки захвата для таблицы, так же как мы делали в первый раз.

The screenshot shows the Cradle Enterprise software interface. At the top, there is a menu bar with 'Файл', 'Инструменты', and 'Помощь'. Below the menu is a toolbar with various icons. The main window displays a requirements tree on the left and a configuration dialog on the right.

**Requirements Tree (Left):**

- Исходные требования к проекту NDM2
- 1 Контекст проекта
  - Приобрести хорошие продукты для семьи
  - Со временем клиентская база становится
  - Поскольку схема работы весьма специфич
- 2 Исходные требования
  - Общие требования
    - Данная ИС должна обеспечивать инт
    - Время и день доставки определяет Г
    - Водителю требуется накладная на к
    - Форма накладной:
      - Изображение
  - При составлении заказа клиент долж
  - Доставка окончательно согласовыва
  - С заказами работает несколько челов
  - Поскольку необходимо брать по мно
  - Окончательную сумму заказа можно
  - За полчаса до подъезда доставки так
  - Поступившие заказы необходимо сор
  - Возможность делиться и обсуждать
  - Интерфейс минималистичный
  - Таблица фермерских хозяйств

- Таблица

**Table View Configuration (Right):**

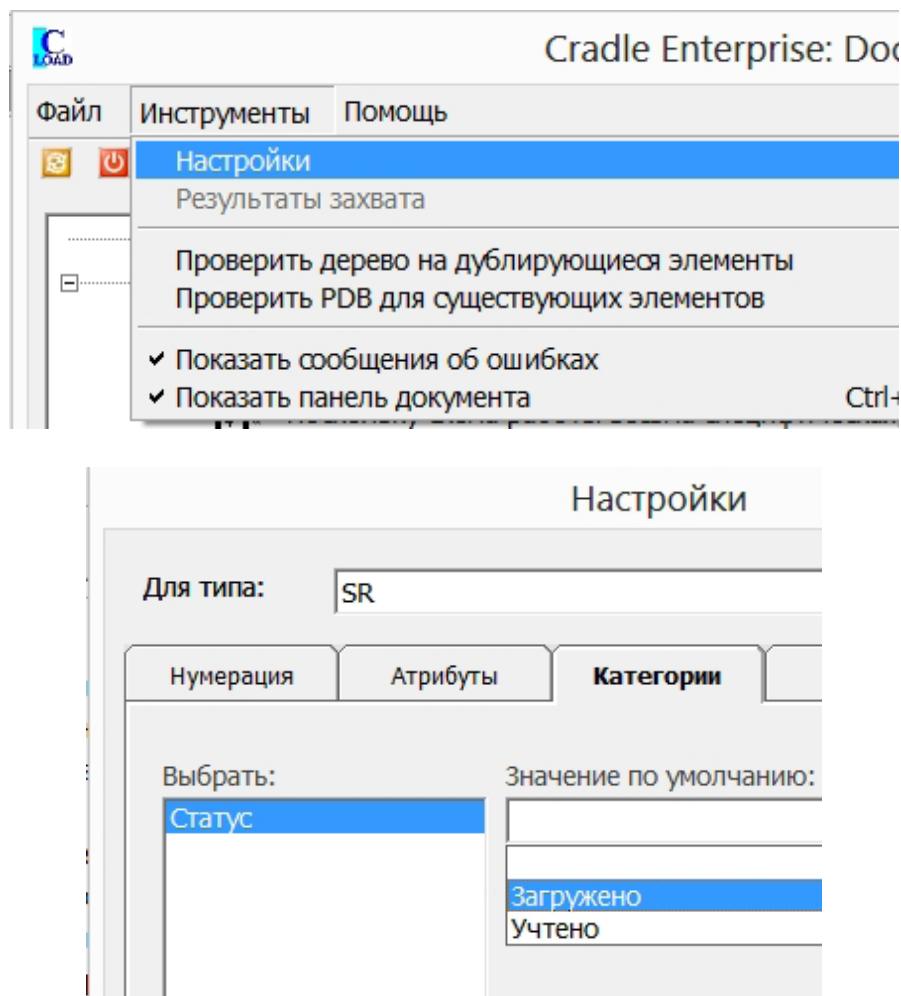
Caption: Отображения таблиц:

Ст...	Стл	Содержание	Отображен
4	1	3	
5	1	4	
1	2	Наименование хозяйства	TEXT
2	2	Ромашки	
3	2	Романовы	
4	2		
5	2		
1	3	Район	TEXT
2	3	Выборгский	
3	3	Приозерский	
4	3		
5	3		
1	4	Контактное лицо	TEXT
2	4	Аля	
3	4	Ира	
4	4		

Кроме этого сделаем еще ряд дополнительных настроек.

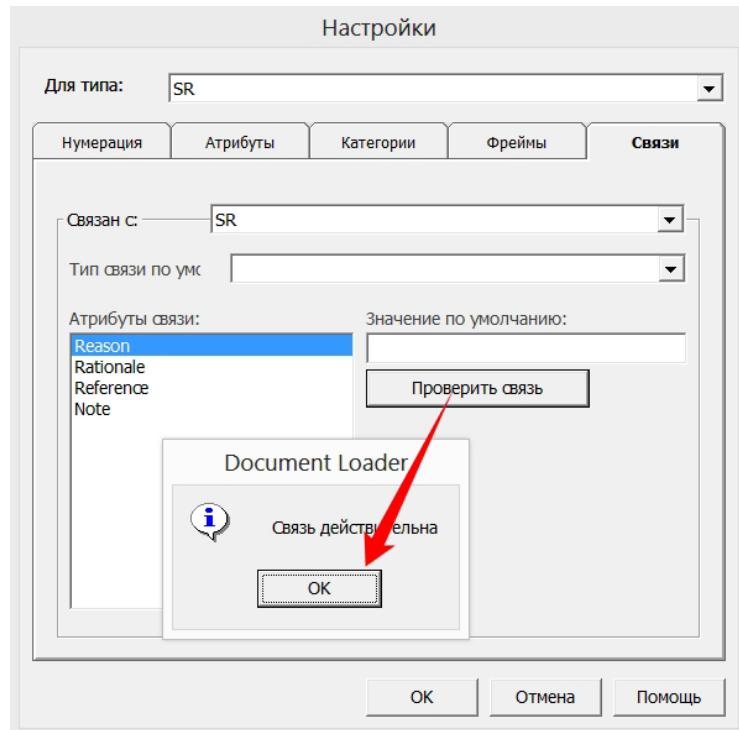
9. Установить значение *Статуса* для загружаемых элементов.

Для этого в меню **Инструменты->Настройки** перейти на вкладку **Категории** и установить для категории *Статус* значение *Загружено*. Этот статус мы будем назначать всем загруженным элементам, анализ которых еще не проводился.



10. Проверить, действительна ли связь, которая будет использоваться для организации иерархии загружаемых элементов.

Для этого на вкладке **Связи** щелкнуть кнопку **Проверить связь**.



11. Запустить загрузку

12. Установить атрибуты документа

Захватить документ

Детали исходного документа	
Имя:	SourceReq
Версия:	01
Заголовок:	Исходные требования
Тип документа:	
Классификация:	UNCLASSIFIED
Выпуск:	1
Дата выпуска:	фев 2014
Ссылка:	1
Комментарии:	
Расположение:	C:\3SL_Cradle_base_book\CradleProject\Файлы Word для загрузки\NDM2-Ис
Примечание: Типлэт - issue, дата issue и связи должен быть уникальный.	
Настройки захвата	
Тип элемента:	SR
Индекс:	<auto generated>
Имя:	Исходные требования
<input type="checkbox"/> Воспроизвести внутренние ссылки документа как связи <input checked="" type="checkbox"/> Разрешить отмену	

### 13. Проверить отчет о загрузке

Операция	Детали
✓ Создать элемент	SR 1 Исход
✓ Зарегистрировать исход...	ИСХТР:01
✓ Добавить фрейм	TEXT фрей
✓ Создать элемент	SR 2 (A)
✓ Создать связь	От SR 1 (A)
✓ Создать связь	От Source :
✓ Добавить фрейм	TEXT фрей
✓ Создать элемент	SR 3 (A)
✓ Создать связь	От SR 2 (A)
✓ Создать связь	От Source :
✓ Добавить фрейм	TEXT фрей
✓ Создать элемент	SR 4 (A)
✓ Создать связь	От SR 2 (A)
✓ Создать связь	От Source :
✓ Добавить фрейм	TEXT фрей
✓ Создать элемент	SR 5 (A)
✓ Создать связь	От SR 2 (A)
✓ Создать связь	От Source :
✓ Добавить фрейм	TEXT фрей
✓ Создать элемент	SR 6 (A)

Создано 25 элементов  
Изменено 0 элементов

### 14. Перейти в **WorkBench** и открыть запрос **SR-Document Simple**

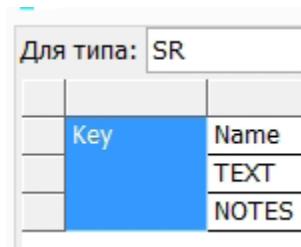
Запрос: SR - Document (Simple)	
	SR - Document (Simple) 
1	Предыдущий...
2	Исходные требования
3	1 Контекст проекта
4	Приобрести хорошие продукты для семьи в супермаркете. небольшие фермерские хозяйства страдают из-за отсутствия организованности для того, чтобы обеспечить систематической доставки продуктов в мегаполис из-за отсутствия транспорта. Решение заключается в том, что система должна решать взять на себя эти организационные функции, постоянно обновляя базу данных о ценах и наличии товаров, чтобы обеспечить быструю доставку. Для этого им необходимо набрать команду из 5 человек, которые будут заниматься организацией закупок, т.к. ради двух кг мяса никто не будет организовывать доставку. Для этого им необходимо набрать команду из 5 человек, которые будут заниматься организацией закупок, т.к. ради двух кг мяса никто не будет организовывать доставку.
5	Со временем клиентская база становится настолько большой, что требуется небольшая информационная система для управления заказами и планированием взаимодействия с клиентами. Для этого им необходимо набрать команду из 5 человек, которые будут заниматься организацией закупок, т.к. ради двух кг мяса никто не будет организовывать доставку.
	Поскольку схема работы весьма специфическая, то предполагается, что система должна быть адаптирована под конкретные потребности клиента.

Мы видим, что все наши элементы успешно загружены. Если мы откроем какой-либо из элементов, то увидим, что вся информация занесена в поле TEXT, а заголовок у исходного требования пустой. В версии 7.0 возможно будет использовать пользовательские представления и для древовидных отображений, поэтому отсутствие заголовков нам не помешает.

Теперь, что бы проверить статусы, добавим к текущему представлению вывод категории Статус, для этого:

15. Открыть диалог **Детали представления**

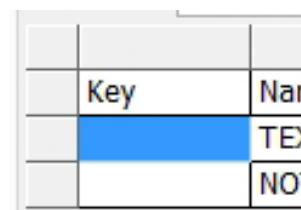
16. Выделить ячейку с полем *Key*



Для типа: SR	
Key	Name
	TEXT
	NOTES

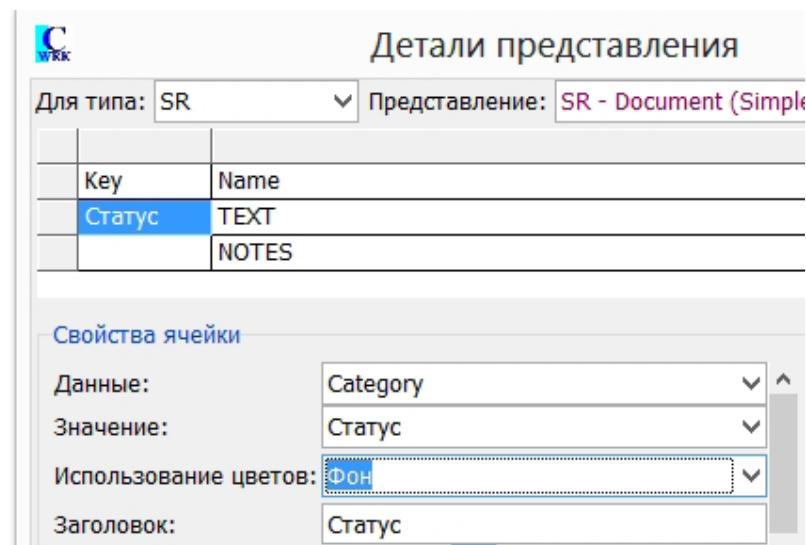
17. Нажать кнопку **Разделить ячейку** 

18. Выделить первую ячейку под полем *Key*



Key	Name
	TEXT
	NOTES

19. Назначить категорию *Статус* для ячейки



Свойства ячейки

Данные:	Category
Значение:	Статус
Использование цветов:	Фон
Заголовок:	Статус

20. Установить *Использование цветов* = **Фон**

21. Нажать кнопку **Применить**

## Применить

После этого мы представление обновиться и мы увидим, что все элементы, которые мы загрузили имеют *Статус Загружено*, который подсвечивается желтым цветом в соответствии с цветовыми настройками категории.

SR - Document (Simple)		
Предыдущий...		
1	Загружено	Исходные требования
2	1	Загружено Контекст проекта
3	Загружено	Приобрести хорошие продукты для семьи в супермаркете небольшие фермерские хозяйства страдают из-за отсутствия «выживания» им не хватает организованности для того, чтобы объединяться с другими хозяйствами и решить вопросы систематической доставки продуктов в мегаполисы (далее Группа) решает взять на себя эти организационные функции, чтобы организовать оптовую закупку, т.к. ради двух кг мяса никто не будет организовывать забой скотины).
4	Загружено	Со временем клиентская база становится настолько большой, что управлять всеми заказами на бумаге становится сложно и группе требуется небольшая информационная система, которая позволила бы снять с себя нагрузку по сортировке заказов и планированию взаимодействия с фермерами.
5	Загружено	Поскольку схема работы весьма специфическая, то принято решение о разработке заказной ИС.

Если мы теперь откроем отчет о покрытии внешнего документа из контекстного меню загруженного документа

## Покрытие (охват) внешнего документа...

то увидим, что элементов создано намного больше и каждый элемент соответствует отдельному абзацу.

Исходные требования к проекту NDM2	
1	Контекст проекта
	Приобрести хорошие продукты для семьи в супермаркетах мегаполисов становится все сложнее. В тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выживания» им не хватает организованности для того, чтобы объединяться с другими хозяйствами и решить вопросы систематической доставки продуктов в мегаполисы из пригородов. Группа инициативных клиентов (далее Группа) решает взять на себя эти организационные функции, чтобы прежде всего обеспечить себе свежими продуктами на постоянной основе. Для этого им необходимо набрать постоянную клиентскую базу, чтобы организовать оптовую закупку, т.к. ради двух кг мяса никто не будет организовывать забой скотины).
	Со временем клиентская база становится настолько большой, что управлять всеми заказами на бумаге становится сложно и группе требуется небольшая информационная система, которая позволила бы снять с себя нагрузку по сортировке заказов и планированию взаимодействия с фермерами.
	Поскольку схема работы весьма специфическая, то принято решение о разработке заказной ИС.
2	Исходные требования

Мы можем также запустить второй отчет о покрытии, который покажет, были ли сделаны какие-либо изменения в исходных требованиях после их загрузки.

Всё, на этом загрузка исходных требований успешно завершена, в рамках этого раздела мы рассмотрели:

- загрузку исходных требований в проектную базу с базовыми настроичными параметрами захвата
- загрузку исходных требований в проектную базу с небольшими корректировками настроек параметров.

# *Анализ и формализация исходных требований*

## **Познакомиться с моделью формализации исходных требований**

Итак, мы загрузили Исходные требования Заказчика. Чаще всего такие требования:

- плохо структурированы
- неформализованы - не классифицированы и содержат неточности

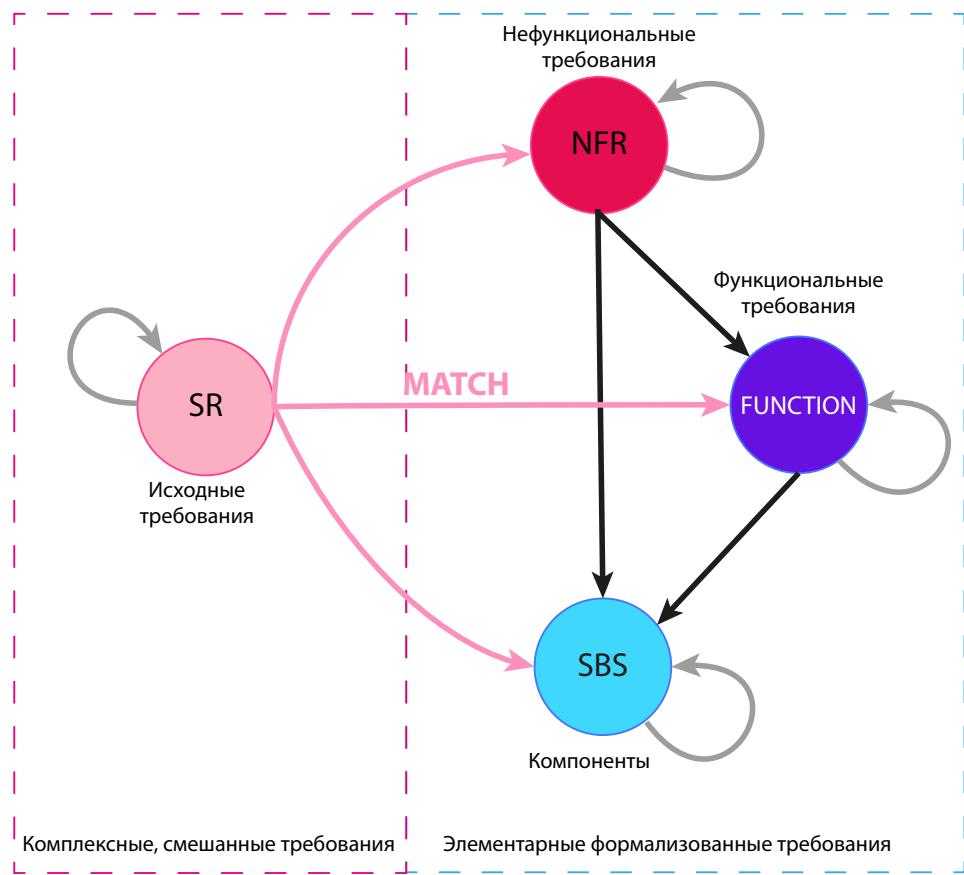
т.е. являются сырьем материалом, который подлежит глубокому анализу и обработке.

В рамках данного примера мы будем проводить формализацию исходных требований, раскладывая их на три взаимосвязанных корзинки и проставляя связи между конкретными требованиями:

- NFR (Non Functional Requirements - нефункциональные требования,
- FUNCTION - функциональные требования,
- SBS - структура системы.

Все вместе - NFR, FUNCTION и SBS будем называть производные требования.

При этом связи будут идти в соответствии с выбранной моделью трассировки:



Такая формализация позволит нам быстро строить различные аналитические представления, которые помогут грамотно спроектировать систему.

Кроме этого мы будем использовать связи:

- **MATCH** для отражения соответствия исходного требования и производного. Направление связи от **SR** к любому типу производного требования.
- **CONFLICT** для отражения конфликтов между требованиями любых типов

Также при анализе мы будем устанавливать значение категории *Ошибки*, если обнаружим ошибку в любом из требований.

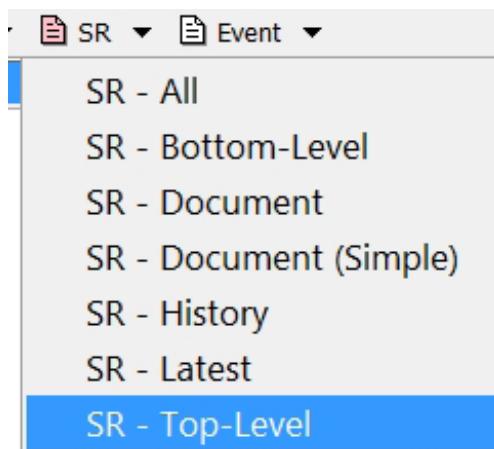
На основе результатов выставления ошибок исходным требованиям мы настроим панель индикаторов, которая будет автоматически пересчитываться и отражать качество исходных требований. Это поможет спрогнозировать время, необходимое для уточнения исходных требований, дополнительную работу с

Заказчиком, а значит точнее проводить планирование работ по проекту.

## Настроить представление для анализа исходных требований

Сначала настроим представление, которое будет отражать иерархию исходных требований, их статус, а также связи с производными требованиями (по типу связи MATCH). Для этого:

### 1. Вызвать запрос SR-Top

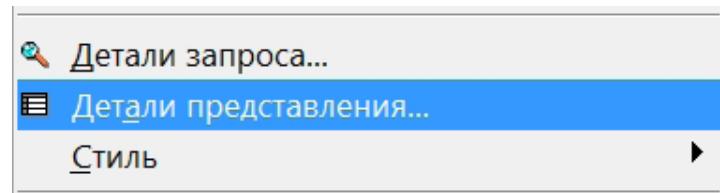


Запрос отобразит один элемент SR верхнего уровня в представлении по умолчанию - SR-Basic.

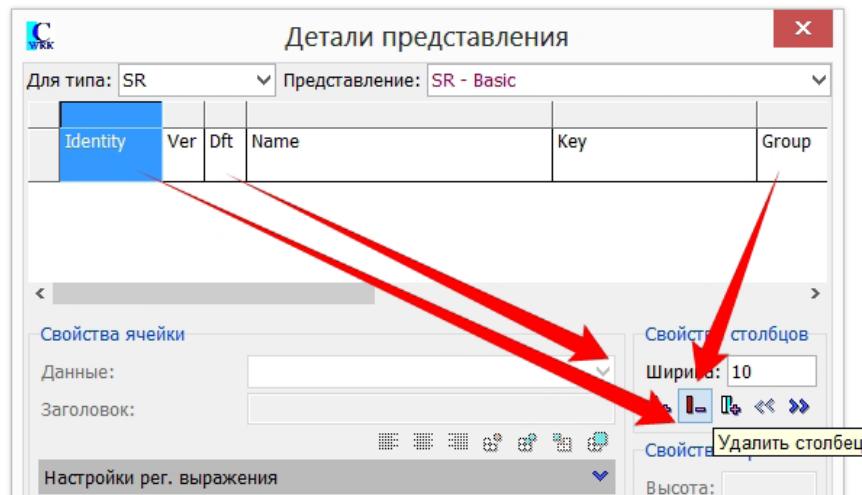
Запрос: SR - Top-Level						
SR - Top-Level		SR - Basic				
	Identity	Ver	Dft	Name	Key	Group
Предыдущий...						
1	1	A		Исходные требования		ИСХТР:01
Следующий...						

Далее откорректируем представление, чтобы рекурсивно выводились все связанные элементы типа SR и любые типы производных требований. Для этого:

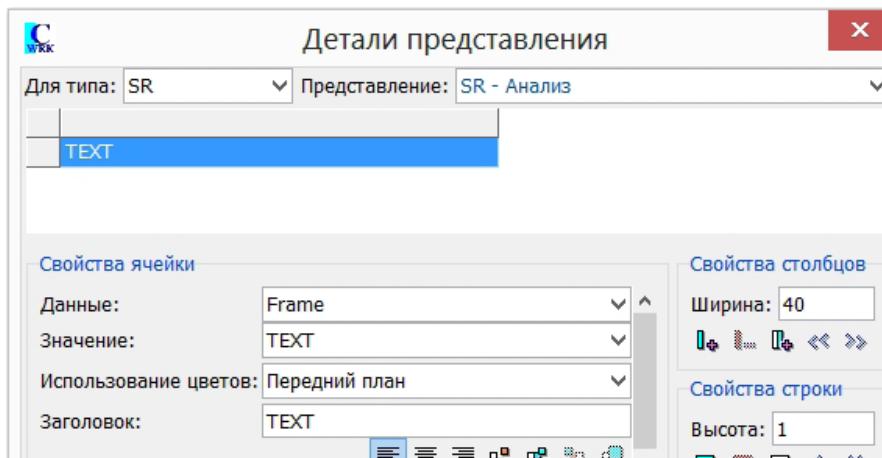
### 2. Открыть диалог **Детали представления**, щелкнув правой кнопкой мыши по открытому представлению:



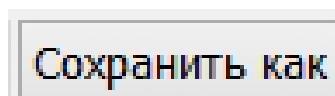
3. Удалить все столбцы, кроме Name, которое заменить на фрейм TEXT

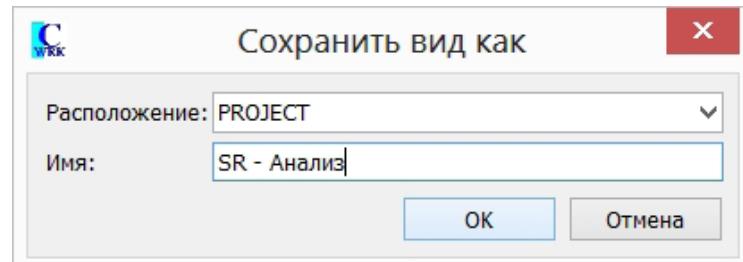


В итоге должно получиться следующее промежуточное представление:

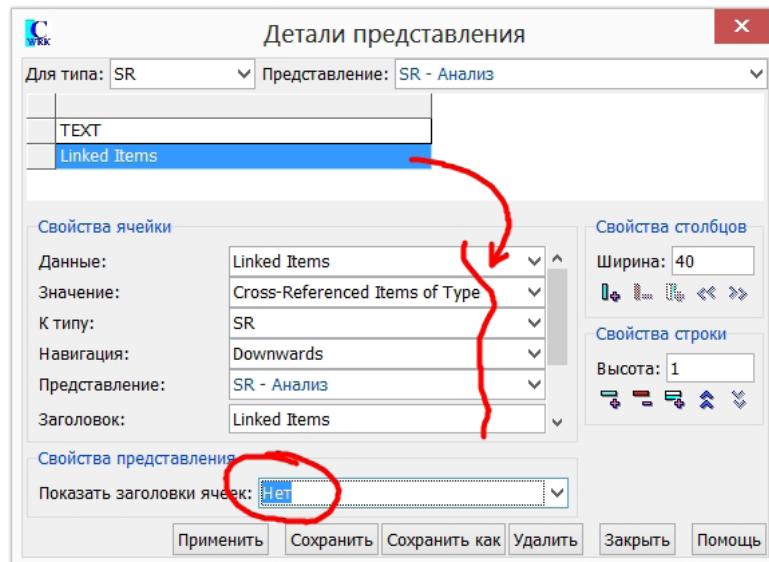


4. Сохранить его под именем SR-Анализ





5. Теперь добавить строку, в которой будут выводиться связанные элементы

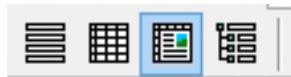


6. Сохранить и применить текущее представление.

Теперь мы можем видеть всю иерархию загруженных требований, а не только элемент верхнего уровня. Однако табличная форма здесь не совсем удобна

для использования, поэтому переключим стиль отображения в Документ.

7. Нажать кнопку с изображением документа на панели инструментов:



Или использовать сочетание горячих клавиш: CTRL+3.

A screenshot of a software interface showing a query results window titled 'Запрос: SR - Top-Level'. The window contains a list of requirements. The first requirement is expanded, showing its details. The background of the expanded requirement is highlighted with a light green color. The interface includes a toolbar at the top and a sidebar on the left.

Отлично, теперь сетка таблицы не будет нам мешать воспринимать данные. Сделаем еще несколько важных настроек.

8. Снова открыть детали текущего представления

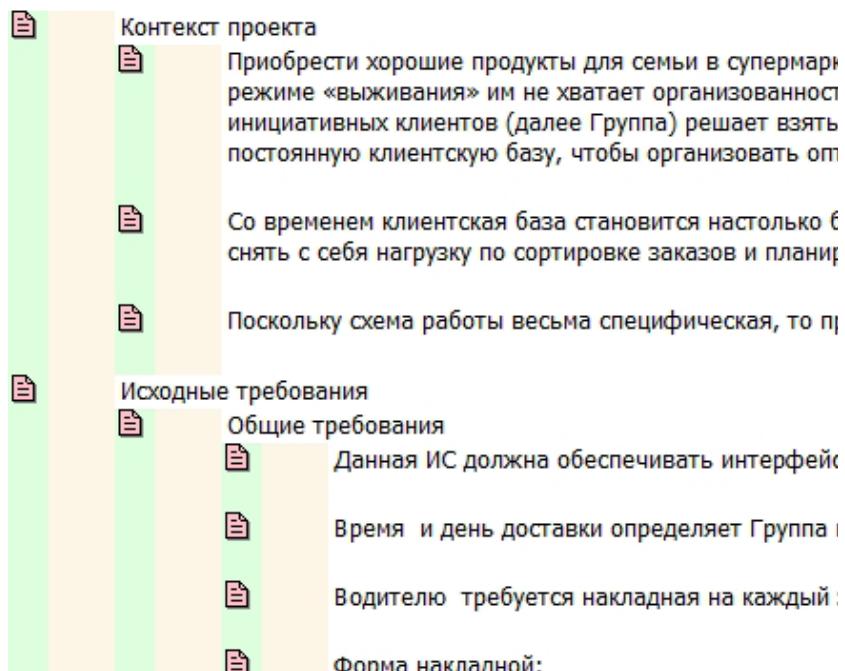
9. Создать колонку, в которую вывести поле Статус, с подсветкой фона.

10. Также включить режим замены регулярного выражения, это уберет сам текст, оставив только заливку.

11. Ширину поля Статус выставить равной 5.

Скриншот конфигуратора представления («Детали представления») в системе ЯК. Настройки для типа SR и представления SR - Анализ. Выбрана ячейка «Статус» с типом TEXT и связанными элементами (Linked Items). Панель свойств ячейки показывает: Данные - Category, Значение - Статус, Использование цветов - Фон, Заголовок - Статус. Настройки рег. выражения включают в себя использование замены на базе рег. выражения (заполнено), регулярное выражение соответствует и заменить на. Итоговая метрика не выбрана.

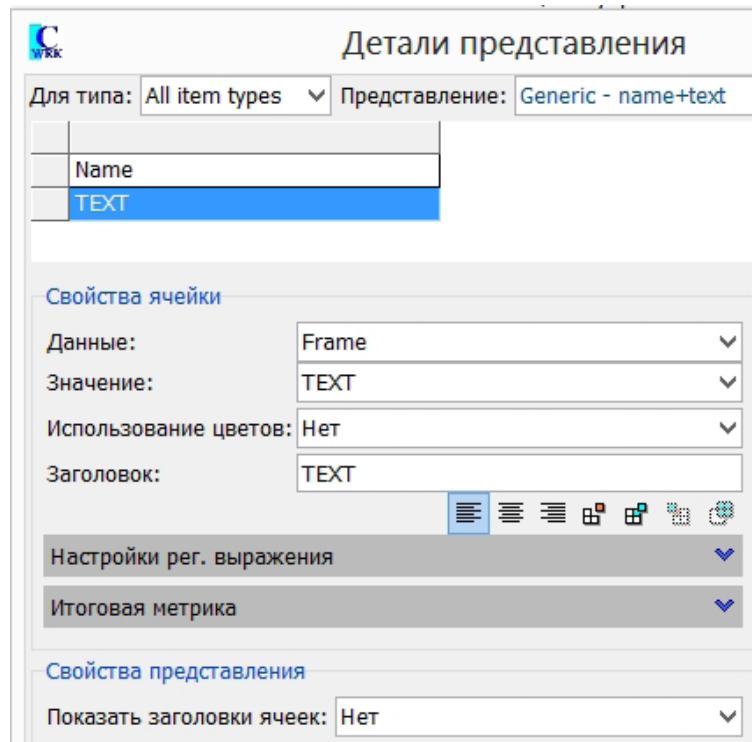
В итоге получим удобное иерархическое представление исходных требований.



Последним штрихом добавим отображение справа от каждого исходного элемента, связанных с ним производных требований. Для этого

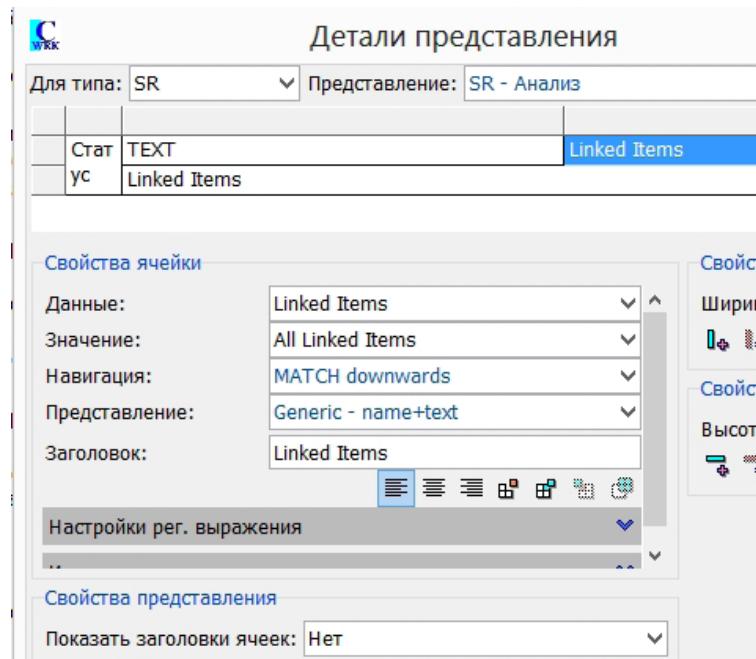
12. Снова открыть [Детали представления](#) и создать представление Generic-name+text.

Это представление будет выводить Имя и Текст для любого типа элементов и которое мы используем как вложенное представление в нашем основном представлении- SR-Анализ.



Обратите внимание, что для вложенных рекурсивных представлений обязательно должны быть отключены заголовки - настройка *Показать заголовки* = Нет.

13. Теперь вернуться к исходному представлению SR-Анализ и донастроить его, добавив справа вывод связанных элементов по навигации MATCHN downwards.



Обратите внимание, что необходимо объединить столбцы второй строки, соответствующей связанным SR.

#### 14. Все, сохранить и применить представление:

Контекст проекта

Приобрести хорошие продукты для семьи в супермаркетах мегаполисов становится все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выживания» им не хватает организованности для того, чтобы объединиться с другими хозяйствами и решить вопросы систематической доставки продуктов в мегаполис из пригородов. Группа инициативных клиентов (далее Группа) решает взять на себя эти организационные функции, чтобы прежде всего обеспечить себя свежими продуктами на постоянной основе. Для этого им необходимо набрать постоянную клиентскую базу, чтобы организовать оптовую закупку, т.к. ради двух кг мяса никто не будет

Со временем клиентская база становится настолько большой, что управлять всеми

Пока мы видим примерно такой же результат как и раньше, с той лишь разницей, что поле TEXT наших исходных требований ограничено широной ячейки. Пока мы не связывали никаких производных требований с исходными и поэтому место справа ничем не заполнено.

Такие представления отрабатываются и настраиваются однократно и могут быть затем использованы не только в одном проекте, но и импортированы в другие. Со временем у вас установится удобная вам технология, которая будет транслироваться из проекта в проект и настройки будут занимать незначительное время.

## **Провести анализ, оценку и формализацию исходных требований**

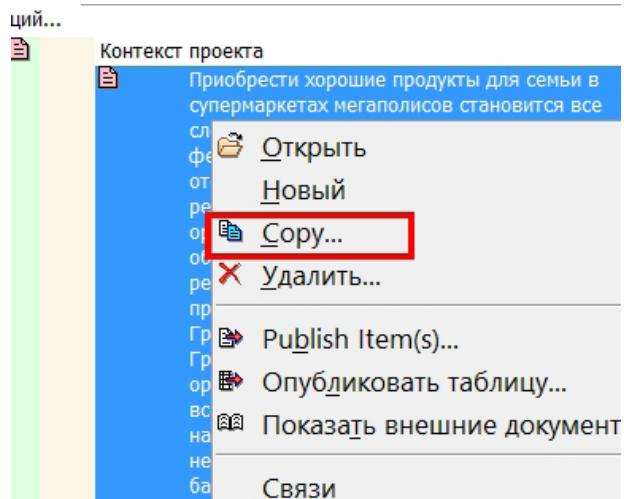
Теперь приступим непосредственно к работе аналитика. Необходимо внимательно прочитать каждое исходное требование и выделить из него три класса производных требований: функциональные, нефункциональные и структуру системы. В этом разделе мы не будем подробно разбирать отличия между этими видами требований, а рассмотрим лишь технологический аспект создания производных требований. Подробнее разобраться с классификациями требований и их практическим применением Вы можете на наших курсах.

Рассмотрим технологию, которая позволяет быстро, сохраняя все связи с исходными требованиями, разработать базовый набор производных, формализованных требований.

Выберем первое требование - контекст проекта. Прочитав внимательно, мы видим, что в нем указана главная *Цель проекта* - Обеспечить инициативную группу свежими продуктами на постоянной основе. Цели - одни из самых важных нефункциональных требований.

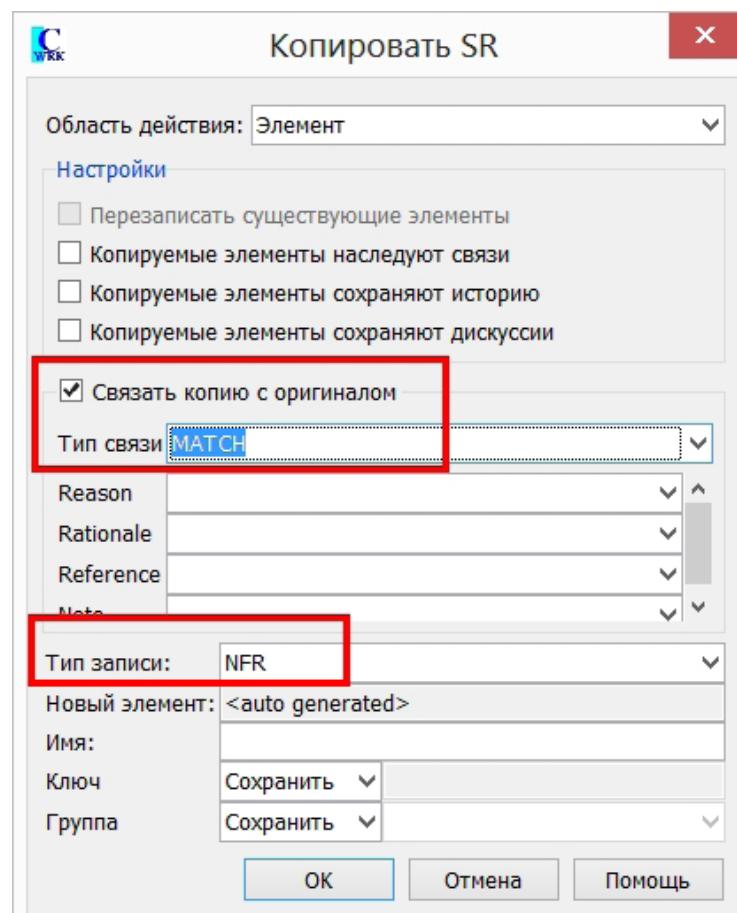
Скопируем его (это исходное требование) в NFR, удалив все, что не относится к этой цели, для этого

- 1. Выделить исходное требование.**
- 2. Из контекстного меню выбрать Copy**



### 3. Установить следующие настройки копирования:

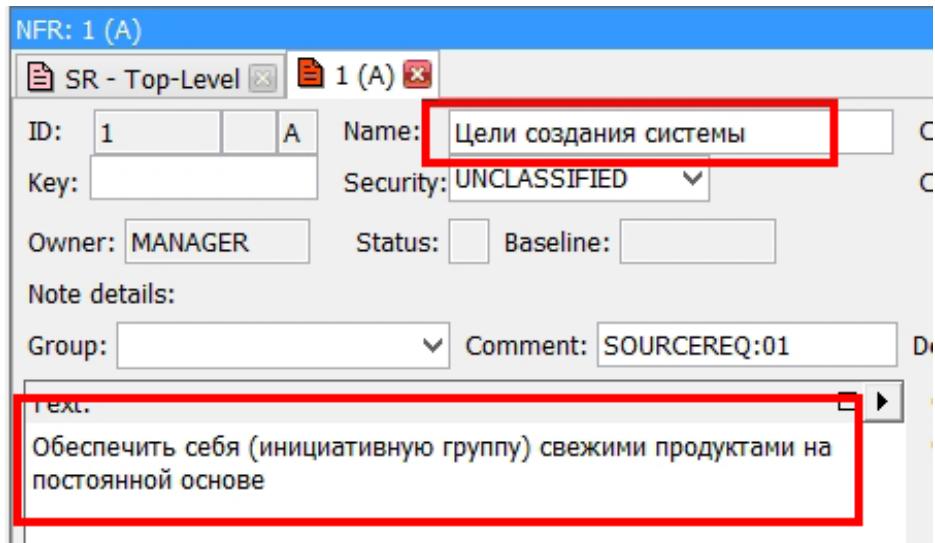
- Связать элемент с оригиналом
- Типа записи = NFR
- Тип связи = MATCH



#### 4. Нажать **OK**

После этого скопированный элемент откроется в форме и можно будет его отредактировать, удалив все лишнее и установив его персональные атрибуты.

#### 5. Отредактировать элемент следующим образом:



NFR: 1 (A)

SR - Top-Level 1 (A)

ID: 1 Name: Цели создания системы

Key: Security: UNCLASSIFIED

Owner: MANAGER Status: Baseline:

Note details:

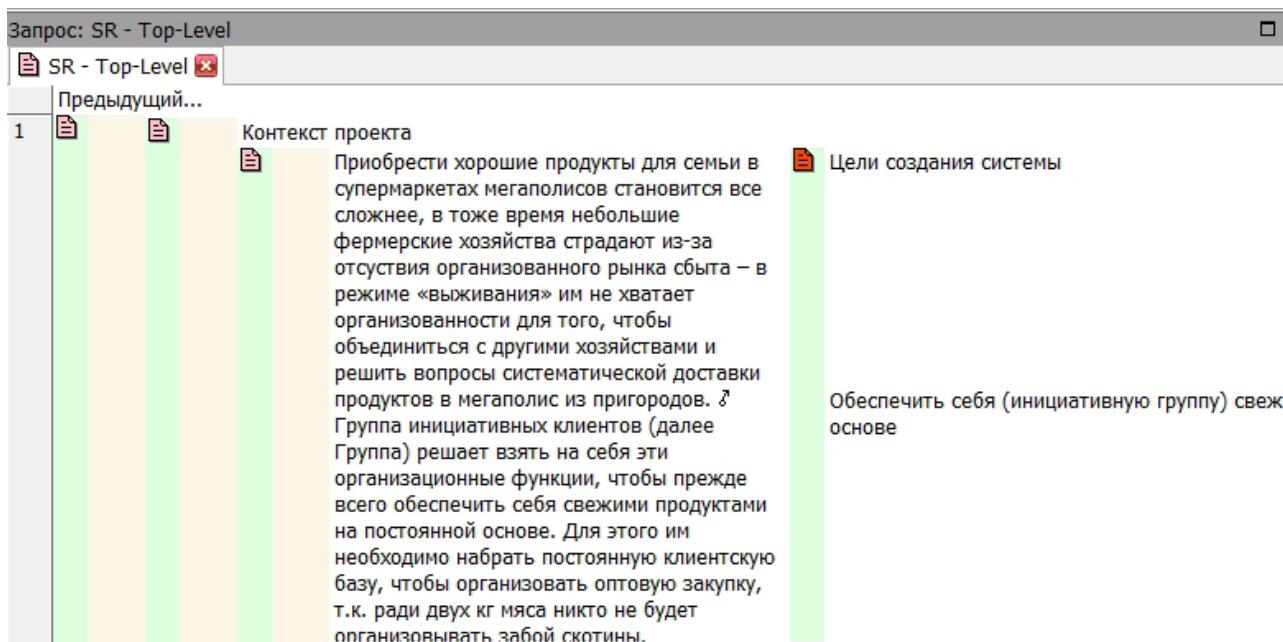
Group: Comment: SOURCEREQ:01

Text:  
Обеспечить себя (инициативную группу) свежими продуктами на постоянной основе

#### 6. Сохранить отредактированный элемент, нажав **CTRL+s**

#### 7. Закрыть созданный элемент, нажав **CTRL+w**

#### 8. Обновить запрос и представление SR-Анализ



Запрос: SR - Top-Level			
SR - Top-Level			
Предыдущий...			
1	Контекст проекта	Цели создания системы	
	Приобрести хорошие продукты для семьи в супермаркетах мегаполисов становится все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выживания» им не хватает организованности для того, чтобы объединиться с другими хозяйствами и решить вопросы систематической доставки продуктов в мегаполис из пригородов. Группа инициативных клиентов (далее Группа) решает взять на себя эти организационные функции, чтобы прежде всего обеспечить себя свежими продуктами на постоянной основе. Для этого им необходимо набрать постоянную клиентскую базу, чтобы организовать оптовую закупку, т.к. ради двух кг мяса никто не будет опланировывать забой скотины.	Обеспечить себя (инициативную группу) свежими продуктами на постоянной основе	

Теперь мы видим, что рядом с исходным требованием отображается связанное с ним производное требование.

Продолжим анализ этого абзаца и выделим все остальные производные требования.

## Изучить пример формализации

Как вы заметили на предыдущих иллюстрациях, на основании одного исходного неформализованного требования, даже не очень большого, может получиться довольно большой объем формализованных требований. Разберем один из примеров подробнее.

Исходное требование *«При составлении заказа клиент должен иметь возможность указать примечания по разделку мяса, а тот, кто разделяет, должен не забыть учесть эти пожелания»*

Из этого требования нам необходимо выделить:

- структуры данных АС
- функции АС
- условия, при которых эти функции должны выполняться
- роли

Часть *«При составлении заказа клиент»* говорит нам о том, что в системе должна быть роль *Клиент*, которой должна быть доступна функция *Оформление заказа*. Читая далее понимаем, что функция *Указать примечания по разделке мяса* должна быть подфункцией *Оформление заказа*.

Анализ этого исходного требования с точки зрения структур данных показывает, что нам понадобится такой атрибут сущности *Заказ* как *Примечания по разделке мяса*.

Аналогичным образом получаются все остальные производные формализованные требования.



При составлении заказа клиент должен иметь возможность указать примечания по разделке мяса, а тот, кто разделывает, должен не забыть учесть эти пожелания.

	Оформление заказа
	Указать примечания по разделке мяса
	Разделка мяса
	Примечания к заказу
	Пожелания клиента по разделке мяса
	Справочник продуктов
	Мясо
	Мясник
	Клиент

Благодаря такому подходу мы можем легко контролировать как список функций системы, таки ролей, которые появляются в ходе анализа предметной области. И, конечно, все нефункциональные требования, в том числе информационную модель предметной области.

Такой тщательный анализ исходных данных позволяет извлечь гораздо больше требований за один проход, а значит здорово повысить эффективности работы.

Запрос: NFR - Top-Level

Запрос: FUNCTION - Top-Level

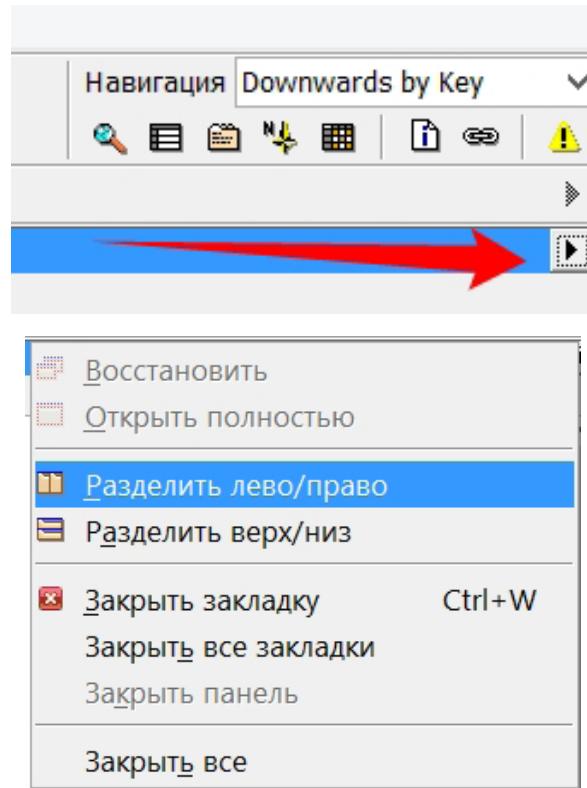
Запрос: SBS - Top-Level

	Identity	Ver	Dft	Name	Key
Предыдущий...					
SBS: 1 : Фермерское хозяйство (A)	1		A	Фермерское хозяйство	
SBS: 2 : Группа инициативных клиентов (A)	2		A	Группа инициативных клиентов	
SBS: 4 : Водитель (A)	4		A	Водитель	
<b>SBS: 5 : Мясник (A)</b>	<b>5</b>		<b>A</b>	<b>Мясник</b>	
SBS: 6 : Клиент (A)	6		A	Клиент	
Следующий...					

## Организовать рабочее пространство для анализа исходных требований

Удобно разделить рабочую область на несколько панелей, открыв каждый тип производных требований в своем окне. Для этого воспользуйтесь кнопкой настройки панели, в правом верхнем углу рабочей области и выберите **Разделить лево/право** и затем **верх/низ**. Панель базы данных можно пока закрыть, чтобы получить дополнительное рабочее пространство.

Для комфортной работы по анализу требований и проектированию в 3SL Cradle мы рекомендуем использовать монитор не менее 17 дюймов, чтобы иметь все необходимые представления на одном экране.



Добейтесь конфигурации окон аналогичной той, что представлена на следующем рисунке. Откройте в каждом дополнительном окне справа запрос Top-Level для каждого типа производных требований:

Cradle Enterprise: WorkBench - Базовый курс Cradle - MANAGER - (8 непрочитанных оповещений)

Файл Редактировать Представление Элемент Запрос Инструменты Администрирование Окно Домошь

Представление SBS - Basic

Навигация Downwards by Key

Запрос: SR - Top-Level

SR - Top-Level

Предыдущий...

Контекст проекта

Приобрести хорошие продукты для сензы в супермаркетах негаполис становятся все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выкисания» им не хватает организованности для того, чтобы объединиться с другими хозяйствами и решить вопросы систематической доставки продуктов в мегаполис из пригородов. Группа инициативных клиентов (далее Группа) решает взять на себя эти организационные функции, чтобы прежде всего обеспечить себя свежими продуктами на постоянной основе. Для этого им необходимо набрать постоянную клиентскую базу, чтобы организовать оптовую закупку, т.к. ради двух кг мяса никто не будет организовывать забой скотины.

Обеспечить себя (инициативную группу)

Со временем клиентская база становится настолько большой, что управлять всеми заказами на бумажке становится сложно и группе требуется небольшая информационная система, которая позволила бы снять с себя нагрузку по сортировке заказов и планированию взаимодействия с фермерами.

Поскольку схема работы весьма специфическая, то принято решение о разработке заказной ИС.

Исходные требования

Общие требования

Данная ИС должна обеспечивать интерфейс для заказывающих, с возможностью указать пожелания о недели доставки, например, сплэйт, доставка к Морозу, сан.

Запрос: NFR - Top-Level

NFR - Top-Level

Предыдущий...

1 1 А Цели создания системы

Следующий...

Запрос: FUNCTION - Top-Level

FUNCTION - Top-Level

Предыдущий...

Следующий...

Запрос: SBS - Top-Level

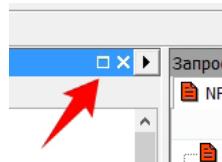
SBS - Top-Level

Предыдущий...

Следующий...

0 элементов

Однако если вам понадобится сконцентрироваться только на одном из представлений - это легко сделать, просто нажмите знак раскрытия представления, который есть у каждого окна в правом верхнем углу:

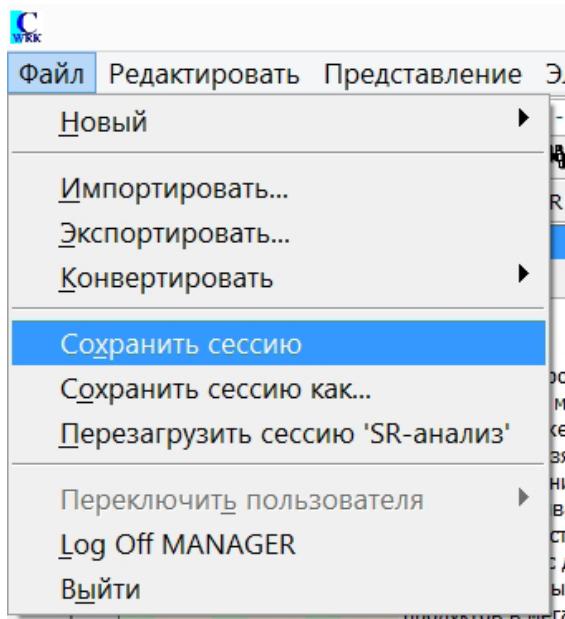


При этом выбранное окно раскроется, но остальные окна также останутся открытыми и их не придется открывать повторно. Для выхода из этого режима нажмите:

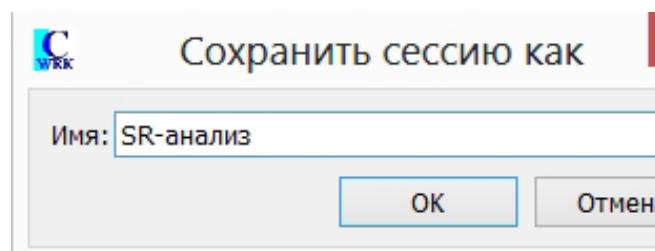


Сделанные настройки интерфейса можно сохранить, чтобы в следующий раз загружать проект с того же места. Для этого:

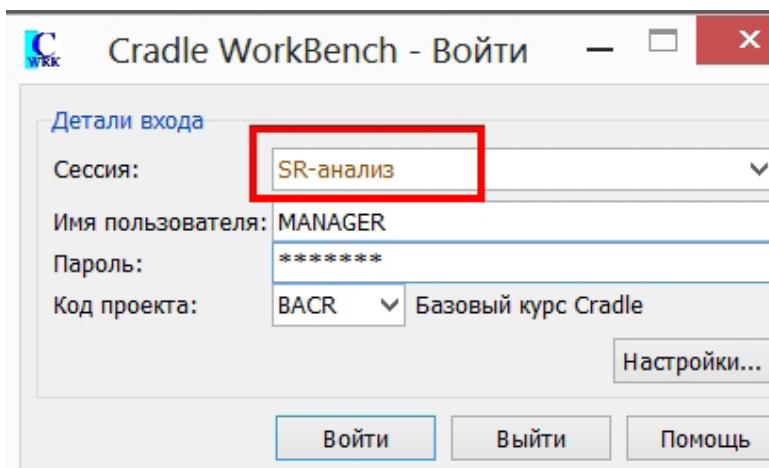
1. Выберите **Файл->Сохранить сессию**



2. Введите имя сессии



Теперь, при входе в проект Вы сможете выбрать сессию, после чего загрузится сохраненное расположение окон и открытые запросы.

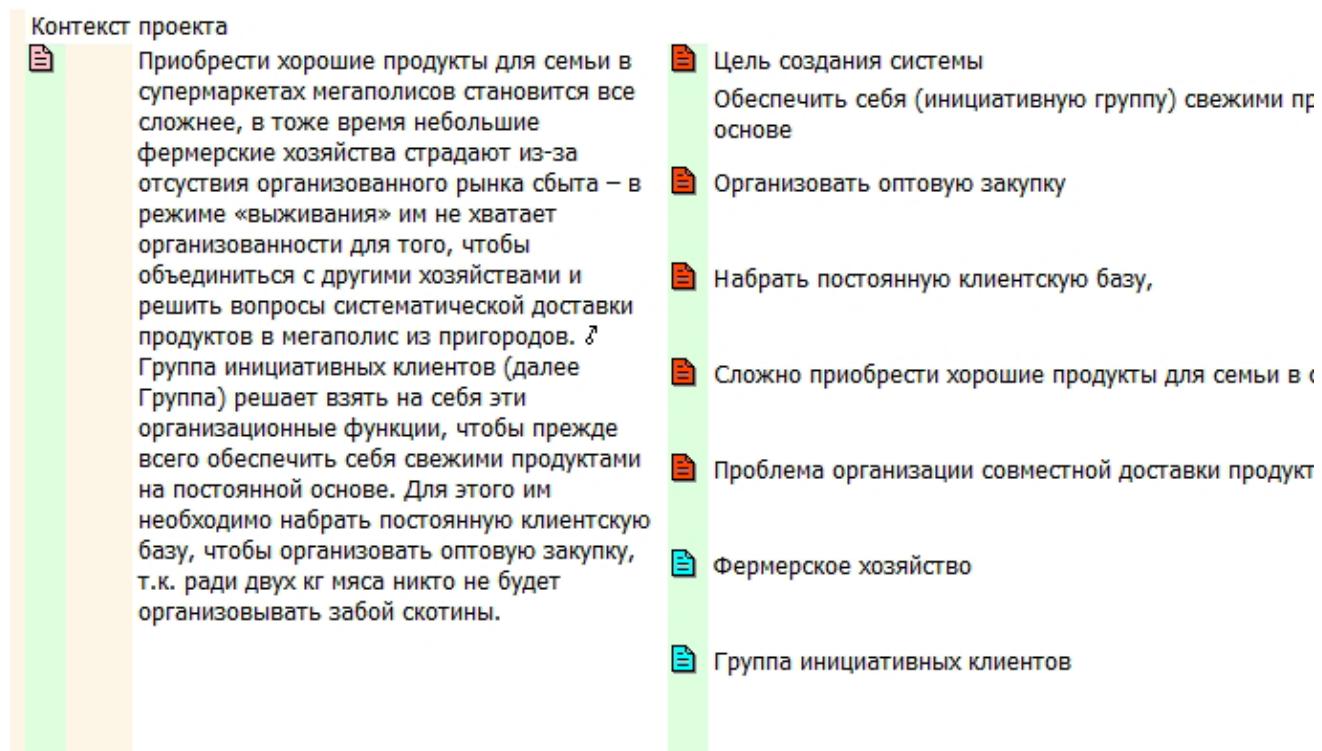


В следующих версиях Cradle будет также возможно указать в качестве пары

метра загружаемую сессию при старте Workbench из файла.

Вернемся к анализу требований.

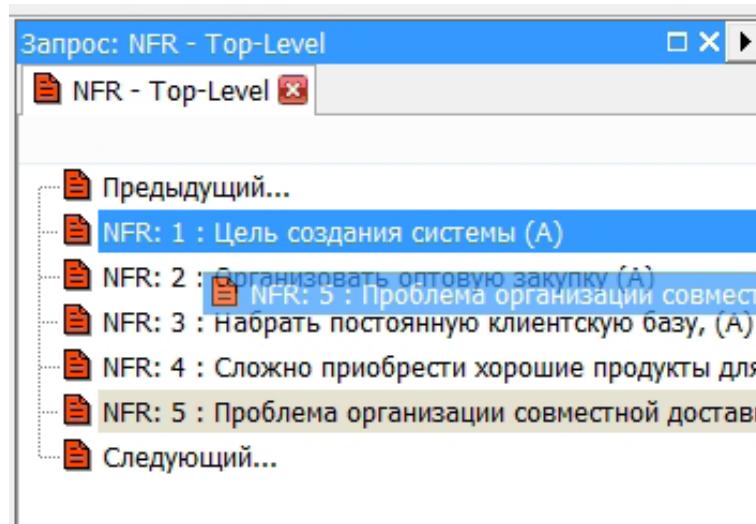
После полного анализа первого довольно большого абзаца с исходными требованиями, мы выявили пять NFR и несколько компонентов системы (в данном случае - роли). Хорошая практика - выявлять не только цели разработки системы, но и проблемы, решение которых требуется. Такие исходные требования (проблемы) позволяют в дальнейшем подобрать более точное решение, как на уровне функций системы, так и по другим ее характеристикам.



Хорошая практика - проводя формализацию исходных требований или другой анализ сразу же проставляйте все выявленные связи между отдельными элементами. Используя горячие клавиши это выполняется практически моментально и потом вы много раз поблагодарите себя за это, т.к. связи покажут свою работу в самые неожиданные моменты, не давая вам принять решение, которое заведет проект в тупик.

Пока связи между созданными NFR еще нам очевидны и мы их не забыли, проведем их небольшое реструктурирование - организуем цели и проблемы в иерархию. На верхнем уровне у нас будет самая важная цель, которая определяет все остальные подцели (подзадачи) - «Обеспечить себя....». Выявленные

проблемы напрямую связаны с этой целью, отразим это в системе, для этого в представлении NFR-Top-level перетащим этот элемент на цель:



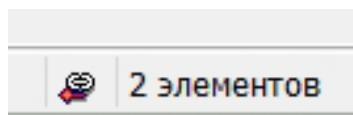
Рекомендуем сразу запомнить горячие клавиши для создания связей - они работают в любых представлениях и формах.

Создание связи ОТ - CTRL+<

Создание связи К - CTRL+>

Очистка буфера связи - CTRL+L

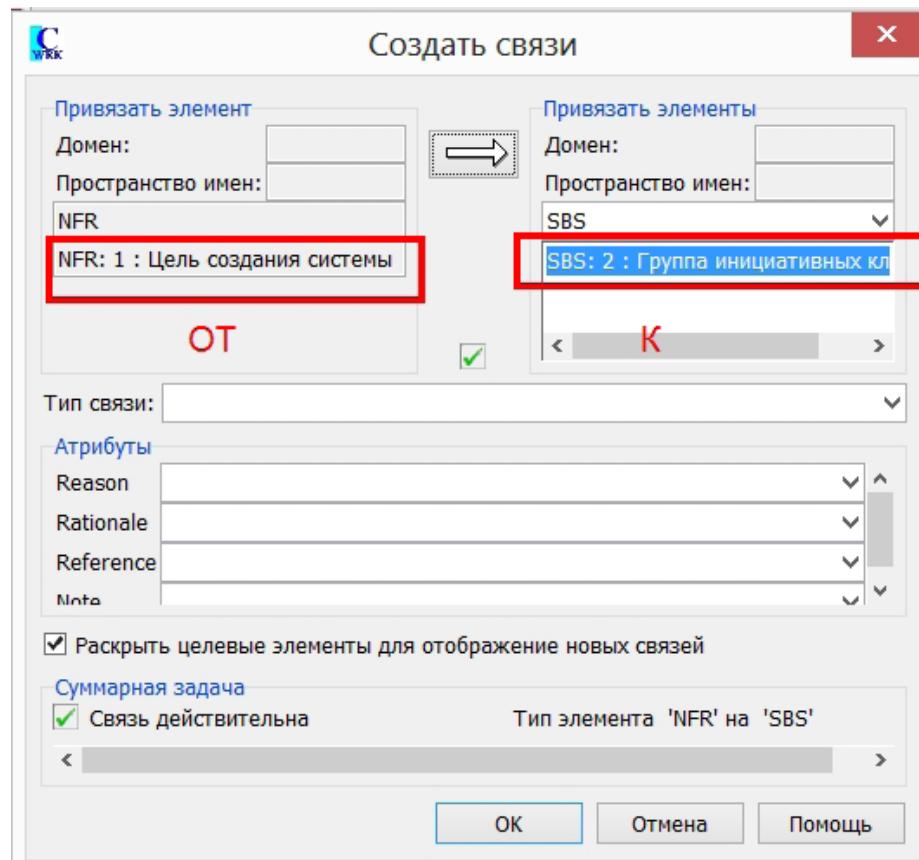
Вы можете выделить любой элемент в любом представлении или форме и нажать одну из комбинаций, например, CTRL+> (создать связь ОТ) и увидите на панели внизу справа, что данный элемент внесен в буфер:



После этого нажмите вторую комбинацию CTRL+< (создать связь К) и буфер заполнится вторым элементом:

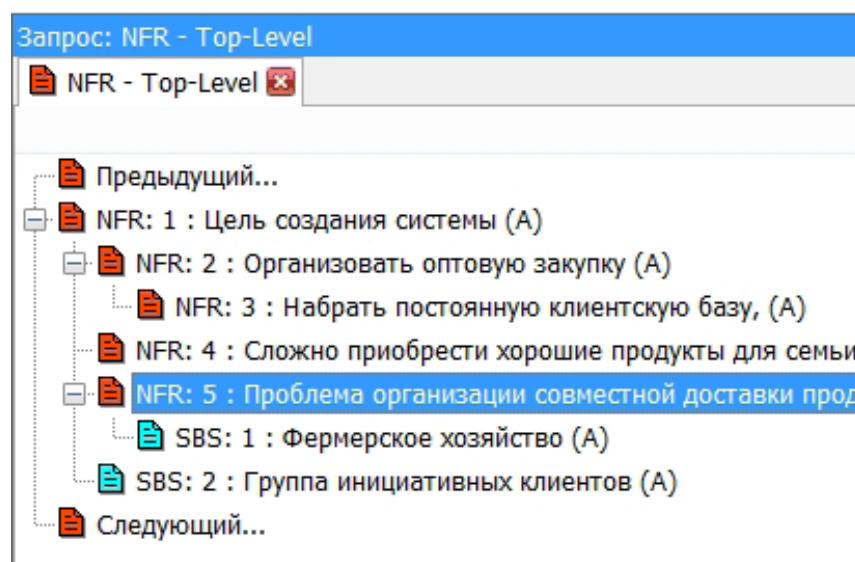


При этом откроется диалог создания связи:



Потренируйтесь и попробуйте создать остальные связи как перетаскиванием, так и с помощью горячих клавиш. Это наиболее часто используемые операции в системе и их нужно делать легко и быстро.

В итоге у вас должно получиться следующее представление:



Заметьте, что мы не только связали нефункциональные требования между собой, но и отразили их связи с выявленными компонентами системы.

## **Отразить ошибки в исходных требованиях**

А что же с ошибками?

Если мы внимательно прочитаем первое исходное требование и попытаемся точно понять его смысл, то найдем ключевое слово, которое всегда является показателем неоднозначности требования. Это слово «хорошие» продукты. Что значит хорошие?

Почему это так важно в данном случае. Слово хорошие будет в дальнейшем влиять на спектр продуктов, с которыми будет работать наша система, это повлияет на справочники продуктов, атрибуты, которые мы будем использовать, поэтому очень важно в дальнейшем уточнить это требование. Будьте уверены - узнаете много интересного.

Поэтому открываем исходное требование и устанавливаем соответствующий атрибут ошибки. Обязательно напишите в комментарии (поле NOTES) почему вы выставили такую ошибку, потом вам это здорово сэкономит время.

SR: 28 (A)

SR - Top-Level 28 (A)

ID: 28	A	Name: <input type="text"/>	Created By: MANAGER	On: 25/02/14
Key: <input type="text"/>	Security: UNCLASSIFIED		Changed By: MANAGER	On: 27/02/14
Owner: MANAGER	Status: <input type="text"/>	Baseline: <input type="text"/>		

Note details:

Group: <input type="text"/>	Comment: SOURCEREQ:01	Description: <input type="text"/>
-----------------------------	-----------------------	-----------------------------------

Text:

Приобрести хорошие продукты для семьи в супермаркетах мегаполисов становится все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выживания» им не хватает организованности для того, чтобы объединиться с другими хозяйствами и решить вопросы систематической доставки продуктов в мегаполис из пригородов. Группа инициативных клиентов (далее Группа) решает взять на себя эти организационные функции, чтобы

Categories:

Статус: Загружено

Ошибки

Неоднозначность

Discussions

Text frames: RTF

Other frames: JPEG

Notes: Хорошие продукты - неоднозначное требование

Кстати, обратите внимание, что в комментарии исходного требования указано наименование документа, из которого оно было загружено. При загрузке это делается автоматически и если вы загружали несколько документов, то всегда легко будет построить запрос, который выведет элементы только конкретного документа.

## Зафиксировать вопросы по исходным требованиям

Хорошая практика - сразу готовить вопросы на согласование с Заказчиком по мере анализа исходных требований.

Мы рекомендуем создавать отдельные элементы для вопросов и связывать их в направлении

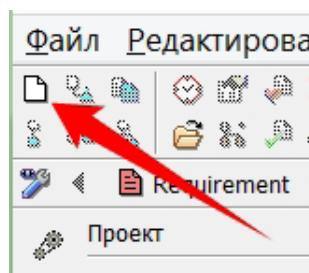
- От вопроса К исходному или производному требованию, если ответ на

вопрос изменит (уточнит) смысл исходного требования.

Такой подход к организации связей позволит в дальнейшем эффективно использовать возможности Cradle по управлению изменениями, а именно подсвечивания элементов, подозреваемых на изменение.

Чтобы удобно организовать вопросы создадим отдельный раздел SR. Для этого:

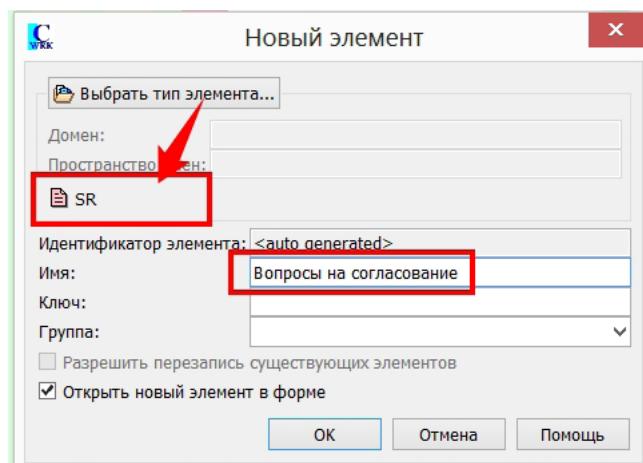
1. В режиме активной панели отображения SR нажать кнопку **Создать новый элемент**



В результате появится диалог создания нового элемента, при этом тип элемента будет установлен в SR - то, что нам надо.

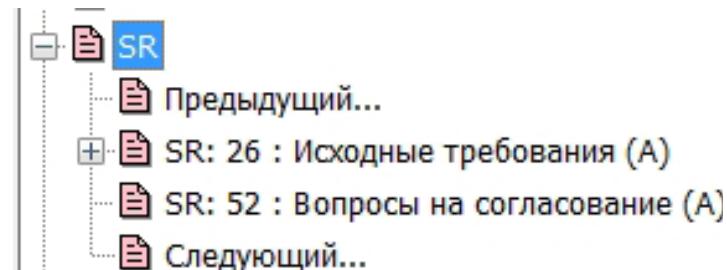
Cradle подставляет тот тип элементов, с которым вы работаете в активном окне. Если же у вас был до этого активен другой тип элементов, то в этом диалоге выберите SR. Затем:

2. Ввести имя элемента - Вопросы на согласование



3. Нажать **OK**

Вы можете увидеть, что созданный раздел появился в базе данных и с ним теперь можно связывать новые элементы.

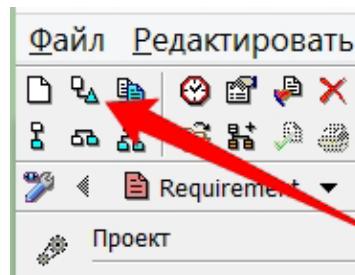


Теперь создадим вопросы относительно того, что такое же такое в понимании Заказчика «хорошие продукты» и «свежие продукты». Не поленимся и создадим два отдельных вопроса, т.к. ответы на них определят скорее всего разные свойства продуктов.

Хорошая практика - создавать отдельные элементы под каждое требование, определяющее не более одного свойства системы.

Для создания вопросов:

1. Выделить элемент SR 52 Вопросы на соглаование
2. Нажать кнопку **Создать новый связанный элемент**



3. Ввести название вопроса и, при необходимости, пояснительный текст

С Новый связанный элемент

 Выбрать тип элемента...

Домен:

Пространство имен:

 SR

Идентификатор элемента: <auto generated>

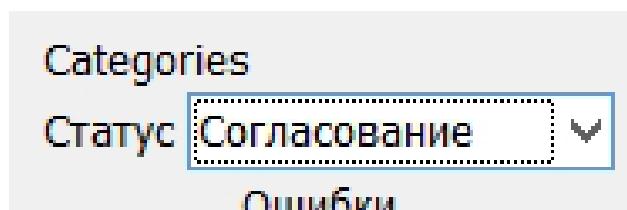
Имя: Что означает свежие| продукты?

Ключ:

Группа:

Разрешить перезапись существующих элементов

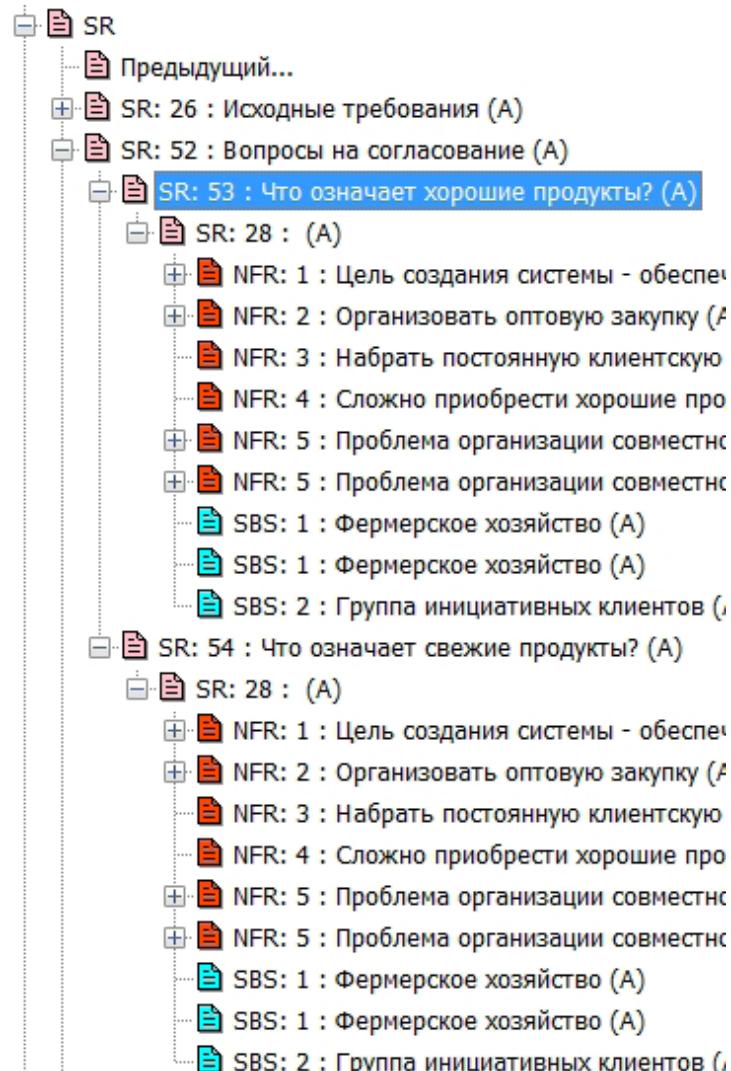
4. В открывшемся диалоге редактирования элемента установить *Статус* = согласование



5. Нажать **CTRL+S** для сохранения элемента.
6. Нажать **CTRL+W**, чтобы закрыть элемент.

Теперь, чтобы мы при работе с исходным требованием мы всегда знали, что по нему есть/были вопросы, проставим связи между созданными вопросами и исходным требованием, используя сочетания клавиш **CTRL+>**, **CTRL+<**

В результате у вас должна получиться следующая картина:



где SR 28 - это первое исходное требование, описывающее контекст проекта. Т.к. оба вопроса у нас связаны с этим требованием, то на следующем уровне мы видим одинаковые структуры.

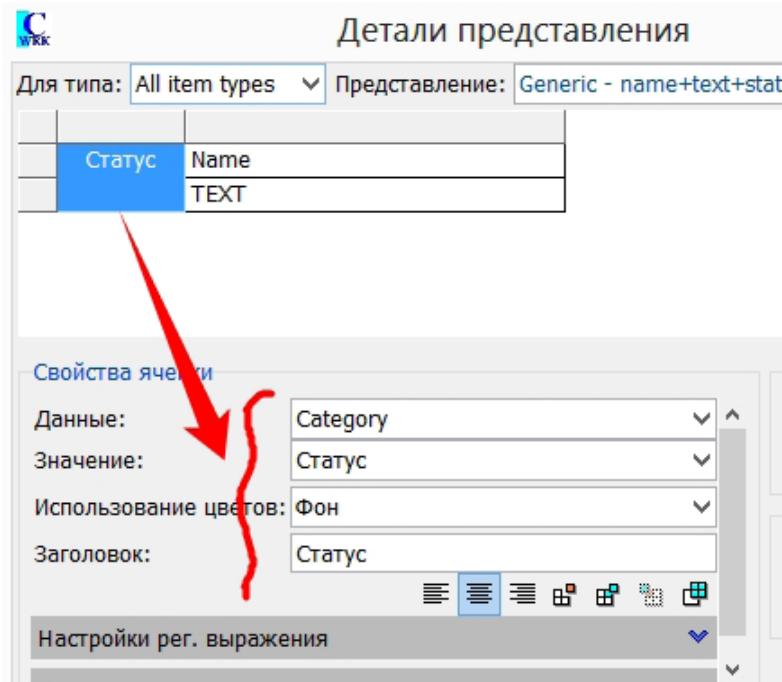
Если же мы обновим запрос и представление, которое используем для анализа исходных требований (SR-Top-Level/SR-Анализ), то не увидим связанных вопросов, т.к. направление связи для них в другую сторону.

В качестве дополнительной практики откорректируем это представление таким образом, чтобы отобразить как сами вопросы, так и их статус. Для этого:

1. Создать вспомогательное представление типа Generic (для любых типов элементов) отображающее Статус.

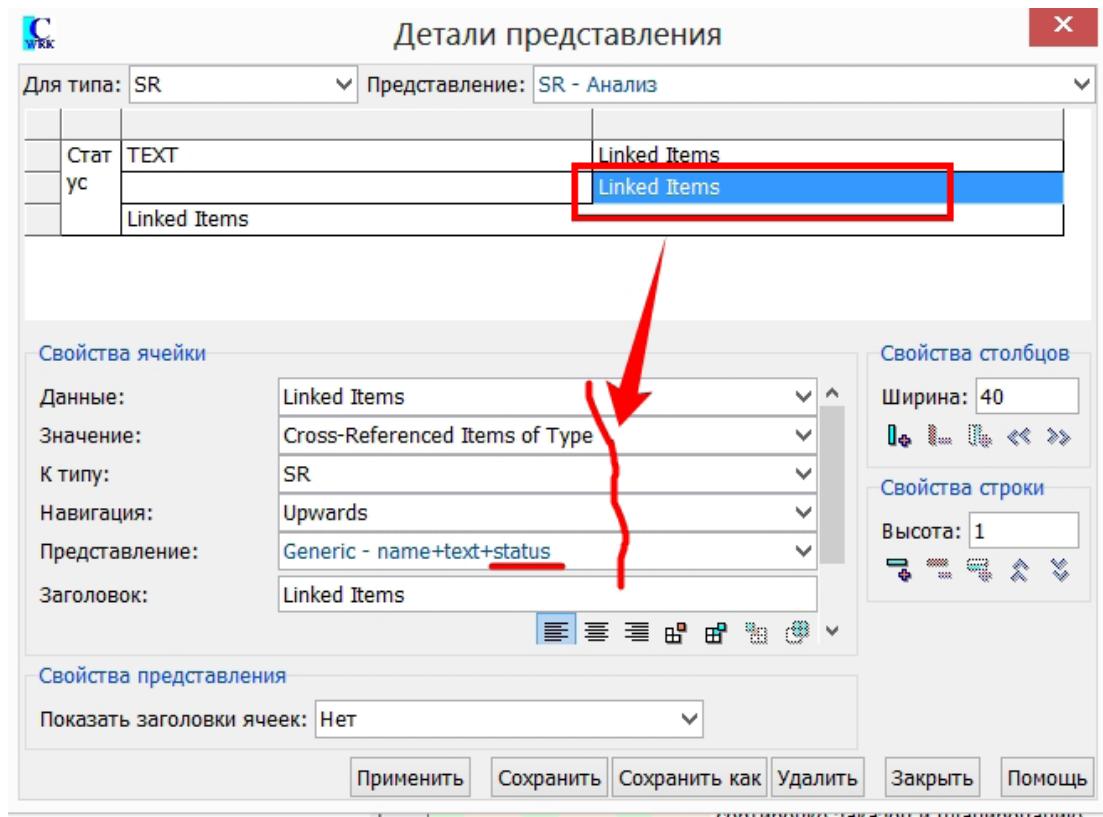
Это представление создадим на основании созданного ранее представления Generic-name+text, добавив в качестве третьего поля - Статус, настроив под-

светку фона по данной категории.



2. Отредактировать представление SR-анализ, сделав вывод связанных элементов типа SR по направлению связи «Вверх» (Upwards)

Выполнить это в соответствии с настройками, указанными на рисунке, выбрав созданное дополнительное представление.



3. Сохранить и применить представление, используя кнопки данного диалога.

В итоге должна отобразиться следующая информация:

Запрос: SR - Top-Level			
SR - Top-Level			
Предыдущий...			
1	Контекст проекта	Загружено	Исходные требования
	Приобрести хорошие продукты для семьи в супермаркетах мегаполисов становится все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выживания» им не хватает организованности для того, чтобы объединиться с другими хозяйствами и решить вопросы систематической доставки продуктов в мегаполис из пригородов. Группа инициативных клиентов (далее Группа) решает взять на себя эти организационные функции, чтобы прежде всего обеспечить себя свежими продуктами на постоянной основе. Для этого им необходимо набрать постоянную клиентскую базу, чтобы организовать оптовую закупку, т.к. ради двух кг мяса никто не будет организовывать забой скотины.	Цель создания системы - обеспечить себя (инициативную группу)	
		Организовать оптовую закупку	
		Набрать постоянную клиентскую базу,	
		Сложно приобрести хорошие продукты для семьи в супермаркетах мегаполисов	
		Проблема организации совместной доставки продуктов в мегаполис из пригородов	
		Фермерское хозяйство	
		Группа инициативных клиентов	
	Загружено	Контекст проекта	
	Согласование	Что означает хорошие продукты?	
	Согласование	Что означает свежие продукты?	
	Со временем клиентская база становится настолько большой, что управлять всеми заказами на бумажке становится сложно и		

На этом представлении мы видим, что с первым исходным требованием у нас связан ряд созданных производных требований типа NFR и SBS, а также ряд вопросов, для которых использован тип SR и эти вопросы находятся в статусе Согласование. Это представление позволяет нам оценить полную картину.

Если абзац с исходным требованием довольно большой, как в этом случае, то за несколько его просмотров может выявиться довольно много вопросов.

Обратите внимание на следующую иллюстрацию - после дополнительного анализа появились новые вопросы и внизу открыта отдельно панель с разделом - Вопросы. Каждый вопрос сначала создается как дочерний от раздела Вопросы на согласование, а затем, с помощью горячих клавиш связывается с исходным требованием.

Запрос: SR - Top-Level

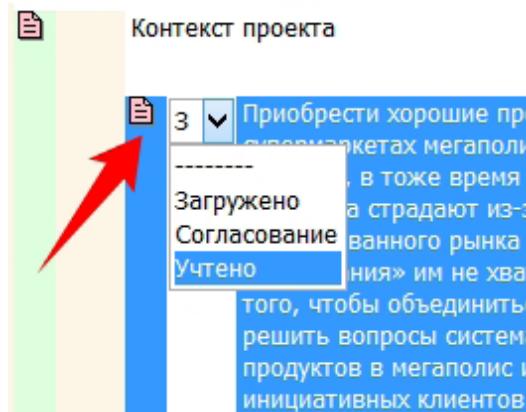
SR - Top-Level #1

Identity	Ver	Dft	Name	Key	Group	Comment
SR: 52 : Вопросы на согласование (A)	52	A	Вопросы на согласование			
SR: 53 : Что означает хорошие продукты? (A)	53	A	Что означает хорошие продукты?			
SR: 54 : Что означает свежие продукты? (A)	54	A	Что означает свежие продукты?			
SR: 55 : Что такое систематическая доставка? (A)	55	A	Что такое систематическая доставка?			
SR: 56 : Что означает постоянная клиентская база? (A)	56	A	Что означает постоянная клиентская база?			
SR: 56 : Что означает постоянная клиентская база? (A)	56	A	Что означает постоянная клиентская база?			

## Изменить статус проанализированных исходных требований

После того, как мы извлекли из исходного требования все производные и зафиксировали все вопросы, необходимо изменить его статус. Это легко сделать прямо в этом представлении для анализа. Для этого:

### 1. Щелкнуть на желтое поле, рядом с текстом



## 2. Изменить Статус на Учтено

После этого элементу присвоится статус Учтено и цвет поля изменится на зеленый.



Если вы успели проанализировать группу исходных требований, то можно выделить несколько элементов и назначить им новый статус за одну операцию.

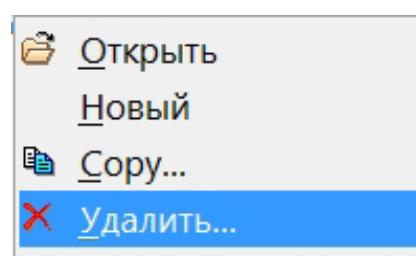
Хорошая практика - «закрывать» требование верхнего уровня (менять статус на выполнено, учтено), только когда закрыты все предыдущие.

## Удалить или восстановить элементы

Особенно в ходе обучения часто можно ошибиться и создать ненужный элемент или удалить нужный.

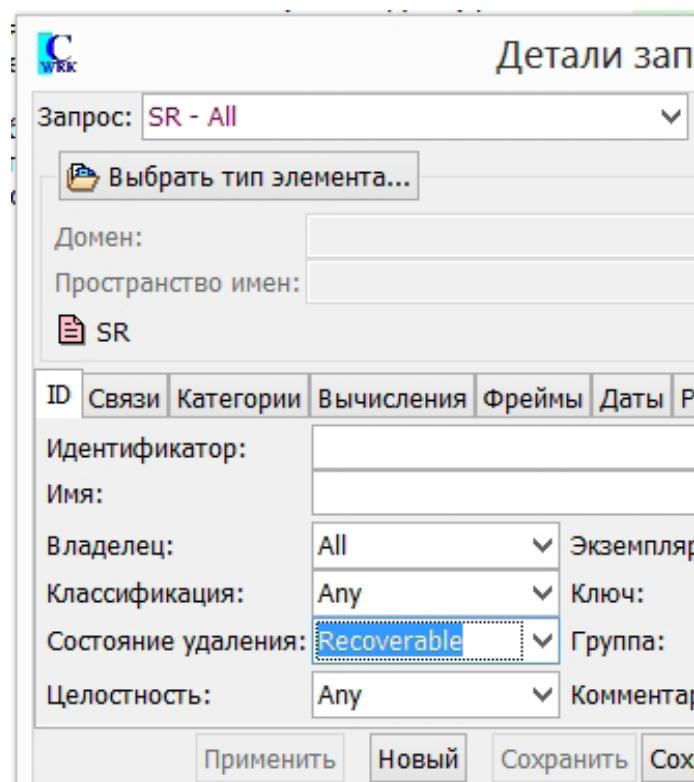
Чтобы удалить элемент(ы) необходимо:

1. Выбрать элемент(ы)
2. Нажать кнопку **Delete** на клавиатуре или из контекстного меню выбрать **Удалить**



При первом удалении элемента(ов) Cradle не удаляет элемент из базы, а меняет значение его атрибута *Состояние удаления* на Recoverable, т.е. тот, который возможно восстановить. Такие элементы не отображаются в общем потоке, но могут быть выбраны с помощью запроса по заданному типу элементов. Для этого:

1. Открыть диалог **Детали запроса**
2. Выбрать, например, запрос SR-All
3. Установить *Состояние удаления* = Recoverable



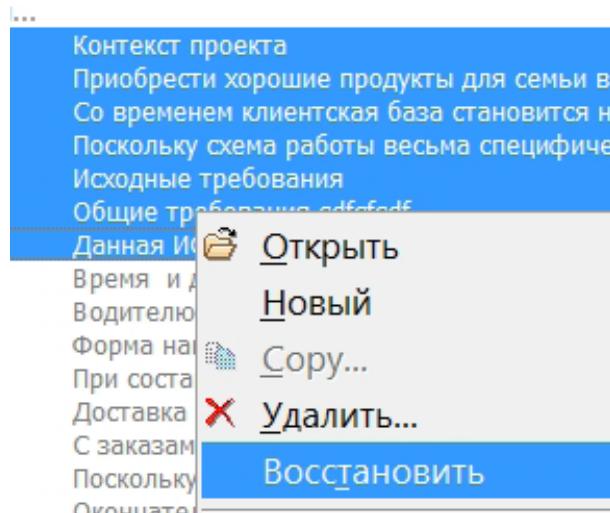
4. Нажать кнопку **Новый**

При этом на активной рабочей панели откроется новый запрос, который отразит все помеченные на удаление элементы. Чтобы вы никогда не перепутали их с действительными элементами, они выводятся серым цветом.

Запрос: SR - All				
Key	Ver	Dft	Name	TEXT
Предыдущий...				
1	A		Исходн...	
	A			Контекст проекта
	A			Приобрести хорошие продукты
	A			Со временем клиентская база
	A			Поскольку схема работы весьма
2	A		Исходные требования	
	A			Общие требования sdfsfsdf
	A			Данная ИС должна обеспечив
	A			Время и день доставки опред
	A			Водителю требуется накладн
	A			Форма накладной:

Вы можете восстановить или окончательно удалить эти элементы. Для этого:

- Выделить необходимые элементы
- В контекстном меню выбрать **Удалить** или **Восстановить**

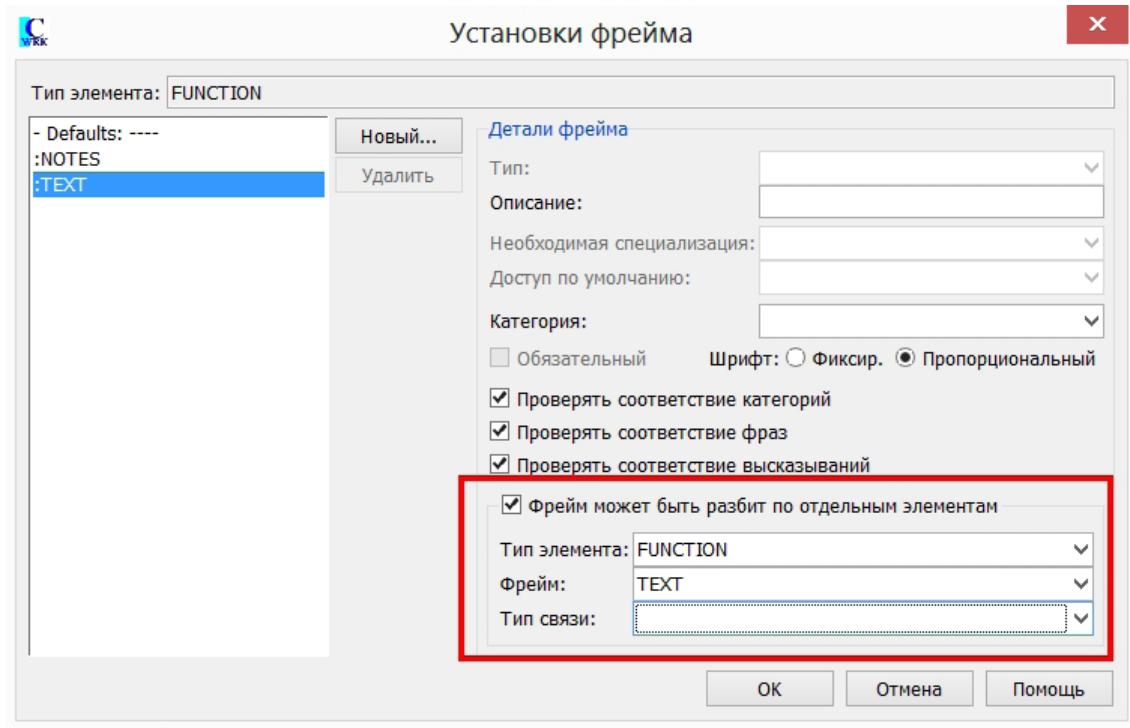


После такого удаления элементы полностью удаляются из базы.

### Разделить объемные требования на элементарные

Удобной функцией Cradle является функция разделения элементов по заданному фрейму. При ее запуске на основании текста фрейма (как правило фрейм TEXT) создается столько новых элементов, сколько строк в этом фрейме.

Для того, чтобы эта функция работала для каждого типа элементов в настройках необходимо указать по какому фрейму возможно проводить разделение. Пример настроек для типа элементов FUNCTION указан на рисунке ниже. Аналогичные настройки сделаны для всех типов элементов проекта.



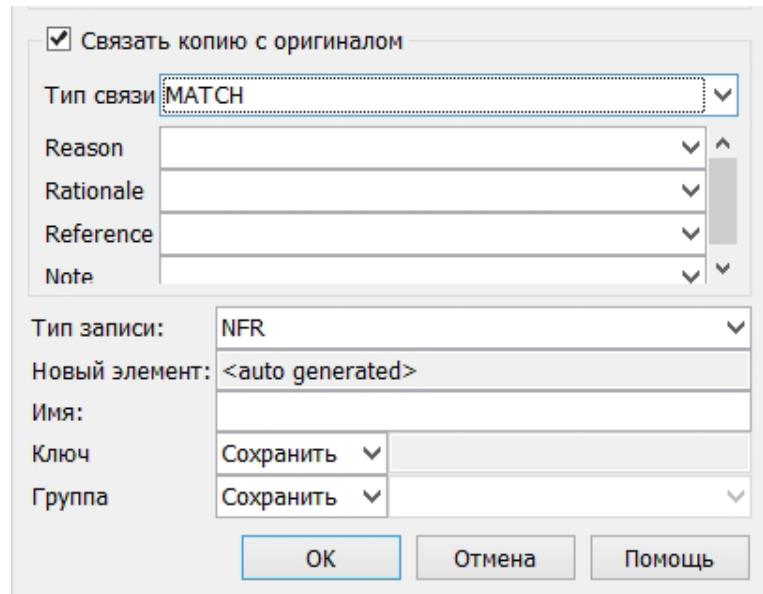
Посмотрим как выполнить разделение на примере. Требование SR34

*«Время и день доставки определяет Группа на основе возможностей фермеров, в основном доставка производится с утра. Конкретное время доставки зависит от маршрута водителя.»*

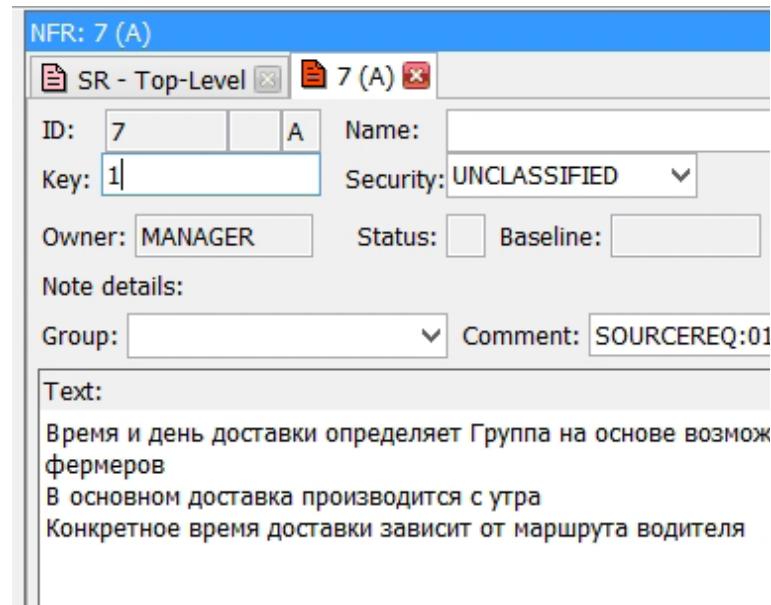
на самом деле включает три элементарных требования:

- Время и день доставки определяет Группа на основе возможностей фермеров
- В основном доставка производится с утра.
- Конкретное время доставки зависит от маршрута водителя

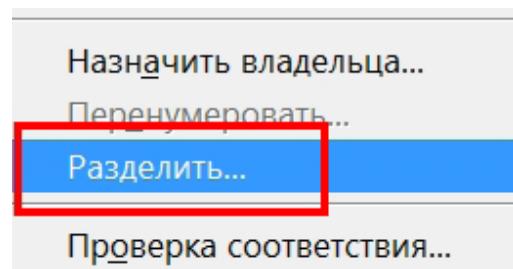
1. Скопировать SR34 в NFR со следующими настройками:



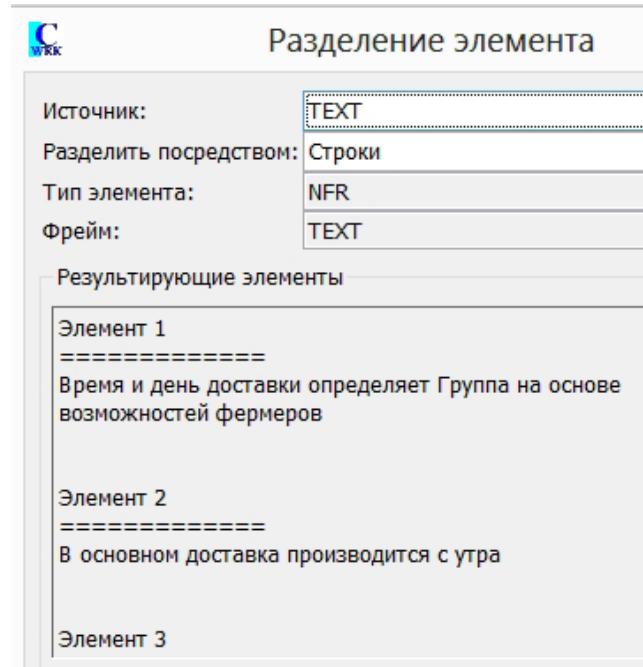
2. Отредактировать открывшееся NFR так, чтобы каждое элементарное требование оказалось на одной строчке.
3. Установить значение Key (любое)



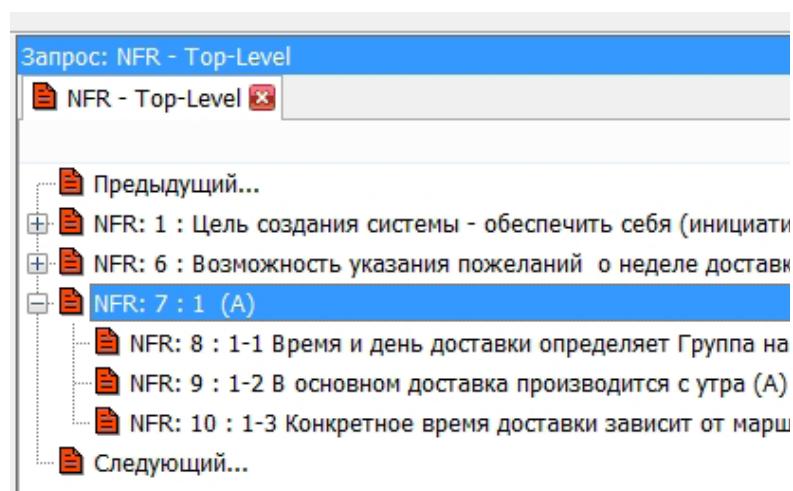
4. Сохранить NFR, нажав CTRL+S
5. Из контекстного меню элемента выбрать **Еще ->Разделить**



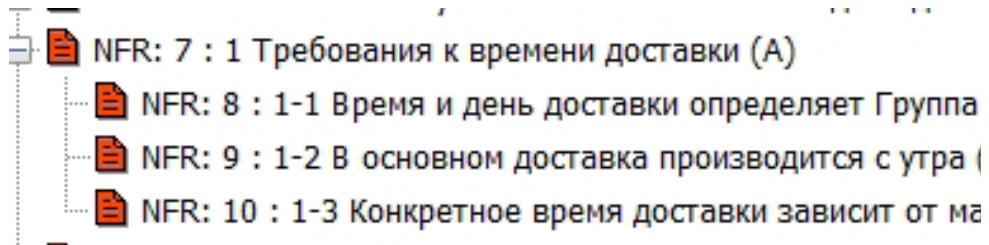
6. Разделить элемент, воспользовавшись стандартными установками:



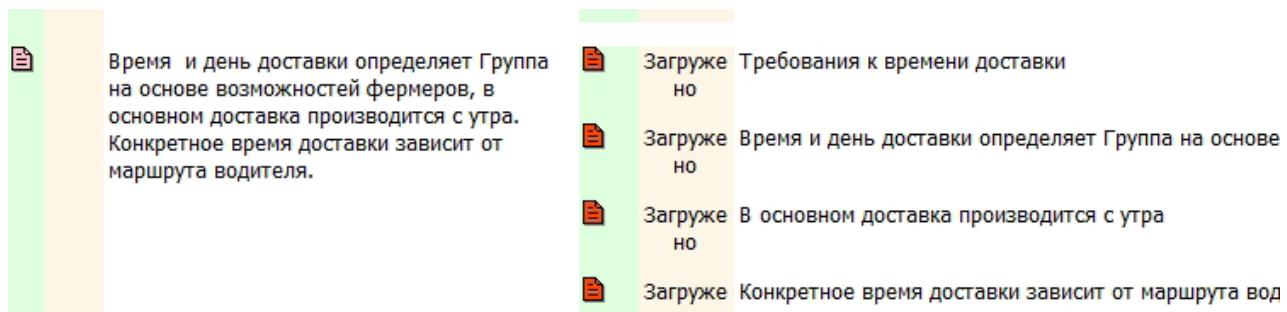
В окне с запросами NFR-Top-Level мы увидим созданные элементы, связанные с тем, который мы делили:



Рекомендуем удалить в верхнем элементе исходный текст и добавить ему поясняющее название, объединяющее три этих элемента. Например, вот так:



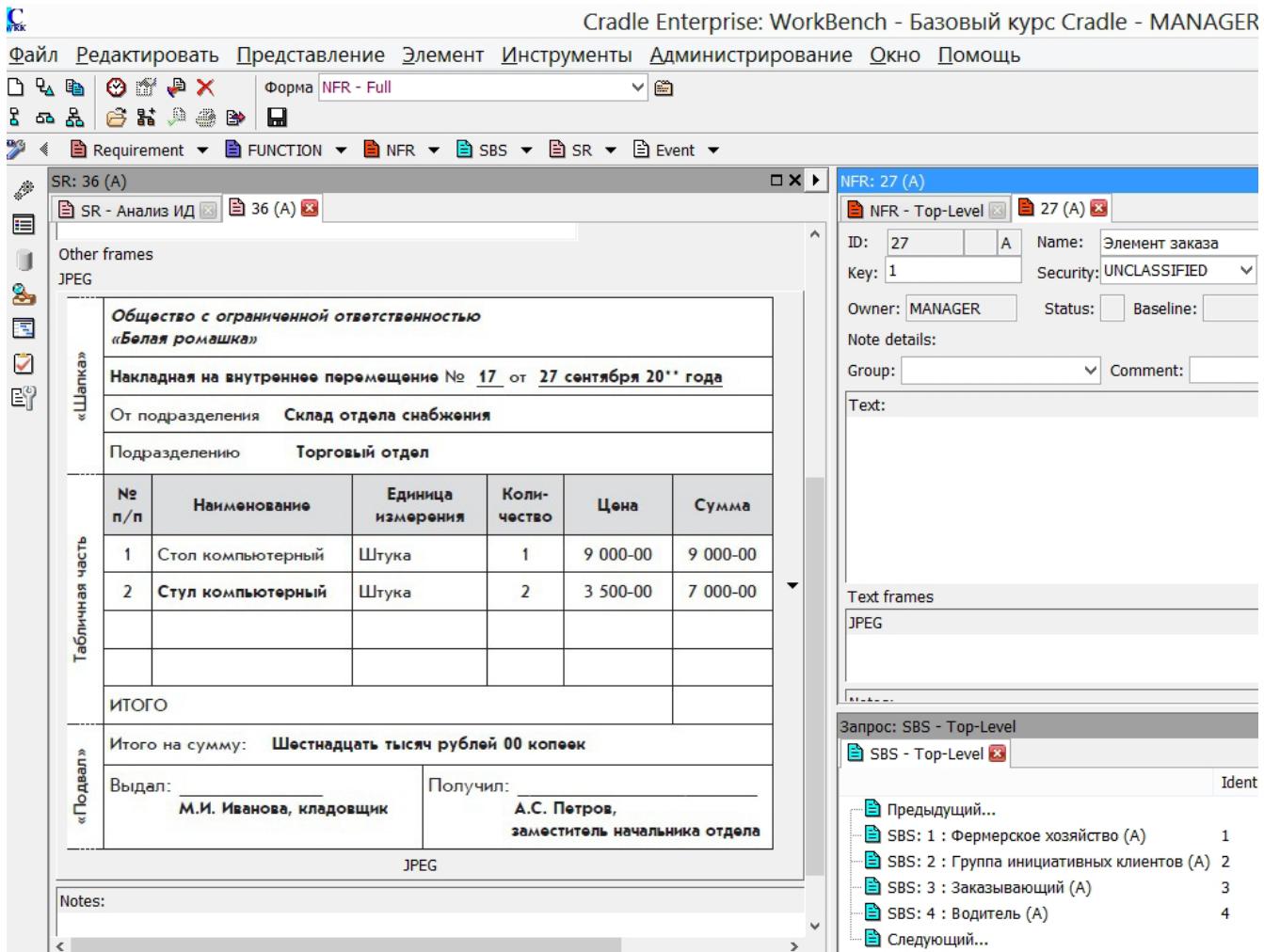
Теперь необходимо выделить все три созданных элемента и связать их с исходным SR, чтобы сохранить трассировку. У вас должна получиться следующая картина, отраженная на рисунке ниже. Конечно, этого можно было бы и не делать, т.к. элемент верхнего уровня связан с исходным SR, но предложенный подход позволяет сохранить трассировку даже, если вам захочется переструктурировать требования, а вам обязательно это захочется сделать.



Функция разделения удобна не только, когда к нам поступили комплексные исходные требования, но и тогда, когда надо быстро набрать серию элементов, подчиненных одному общему, например, описать поля структуры данных.

Тогда мы можем набирать все данные в одном элементе и после этого воспользоваться функцией **Разделить**. На этом мы сэкономим себе немного дополнительного времени. Посмотрим, на примере. В исходных данных у нас есть таблица накладной, в виде картинке, нам необходимо описать ее структуру. Для этого:

1. Открыть элемент с накладной в одном окне
2. В другом окне открыть созданный элемент, соответствующей всей структуре данных в целом.



### 3. Ввести данные о структуре на основании полей накладной.

Каждый элемент вводить на новой строке

### 4. Назначить любой ключ элементу (Key)

### 5. Сохранить элемент CTRL+S

SR: 36 (A)

Other frames  
JPEG

«Шапка»

Общество с ограниченной ответственностью  
«Белая ромашка»

Накладная на внутреннее перемещение № 17 от 27 сентября 20\*\* года

От подразделения Склад отдела снабжения

Подразделению Торговый отдел

Табличная часть

№ п/п	Наименование	Единица измерения	Коли-чество	Цена	Сумма
1	Стол компьютерный	Штука	1	9 000-00	9 000-00
2	<b>Стул компьютерный</b>	Штука	2	3 500-00	7 000-00
ИТОГО					
Итого на сумму: Шестнадцать тысяч рублей 00 копеек					
Выдал: М.И. Иванова, кладовщик		Получил: А.С. Петров, заместитель начальника отдела			

«Подвал»

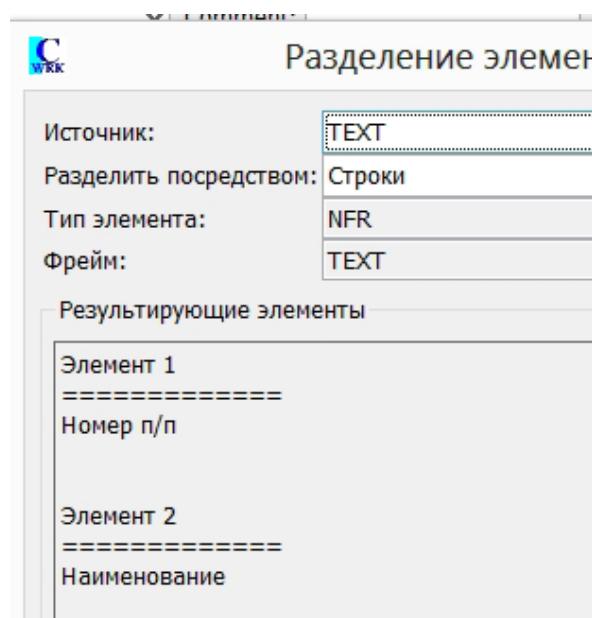
NFR: 27 (A)

ID: 27 A Name: Элем  
Key: 1 Security: UNCL  
Owner: MANAGER Status: B  
Note details:  
Group: Com  
Text:  
Номер п/п  
Наименование  
Единица измерения  
Количество  
Цена  
Сумма  
Text frames  
JPEG

Запрос: SBS - Top-Level  
SBS - Top-Level

Предыдущий...  
SBS: 1 : Фермерское хозяйство (A)

6. В контекстном меню выбрать **Разделить**
7. В настройка разделения оставить «Посредством строки»



8. Нажать кнопку **Разделить**

Разделить

9. Очистить поле текст исходного элемента.

10. Закрыть исходный элемент (CTRL+w)

11. Проверить разделение, открыв элементы в представлении типа Дерево:



Обратите внимание, что при разделении Cradle добавляет в каждый элемент нижнего уровня имя исходного элемента. В данном случае это очень удобно, т.к. при трассировке мы часто можем не увидеть элемент верхнего уровня и такие названия как «Количество» и др. могут оказаться непонятными, выпав из контекста.

Хорошая практика - всегда давайте самостоятельно понятные наименования элементам.

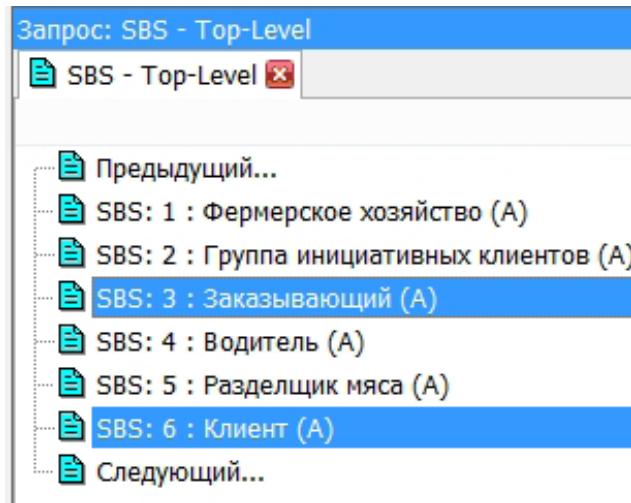
12. Соедините все созданные дочерние элементы с исходным SR.

13. Очистите поле TEXT для созданных дочерних NFR, т.к. в данном поле про-дублировалось поле Name

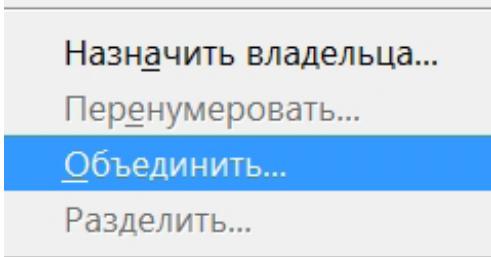
## Объединить дублирующиеся требования

Нередко в ходе формализации исходных требований могут быть созданы элементы, имеющие разные наименования, но одинаковые по смыслу. Такие элементы стоит объединить, сохранив запись об их объединении. Для этого:

1. Выбрать элементы, подлежащие объединению:



2. В контекстном меню выбрать **Еще -> Объединить:**



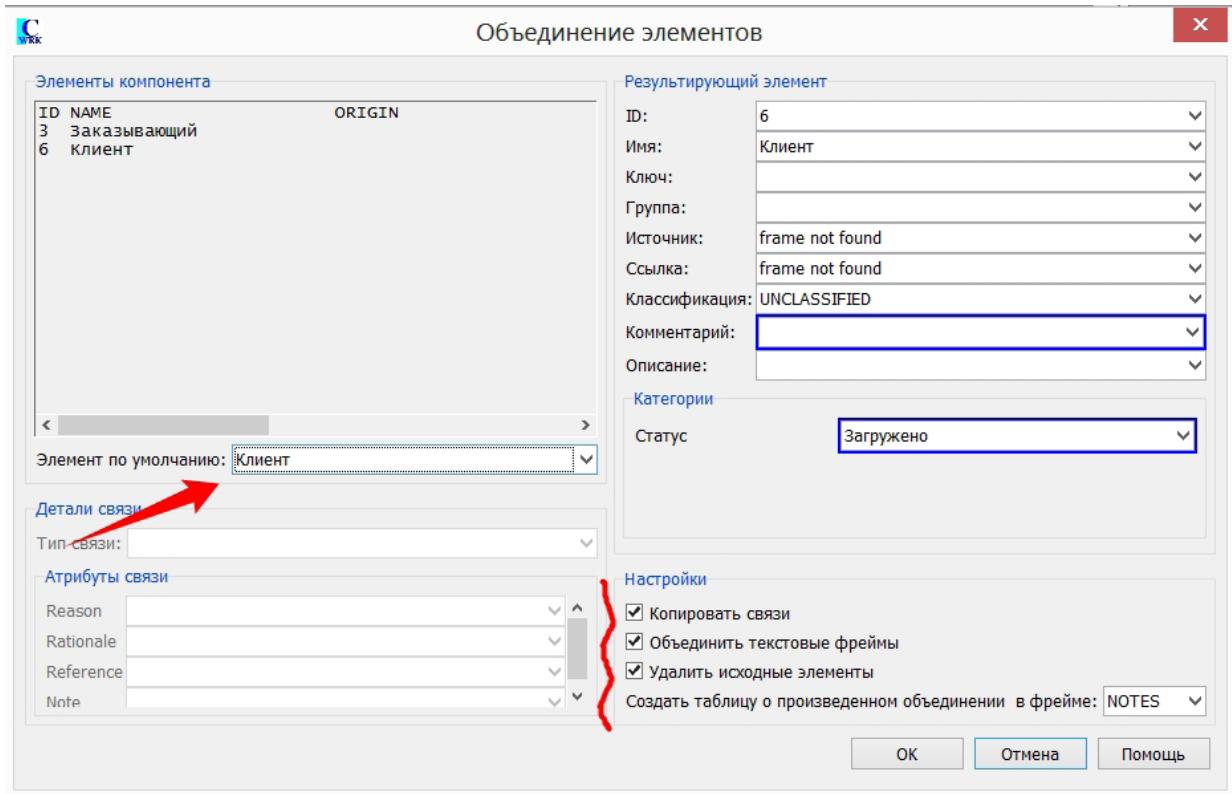
3. В диалог **Объединение элементов** выбрать **Элемент по умолчанию** - это один из объединяемых элементов, чьи настройки станут основными

4. Указать необходимость

- Копировать связи
- Объединить текстовые фреймы
- Удалить исходные документы

5. Информацию об объединении записывать в фрейм NOTES.

Для того, чтобы фрейм NOTES возможно было выбрать в этом диалоге, в установках проекта для выбранного типа элементов должна быть указана возможность разделения элемента по этому типу фрейма.

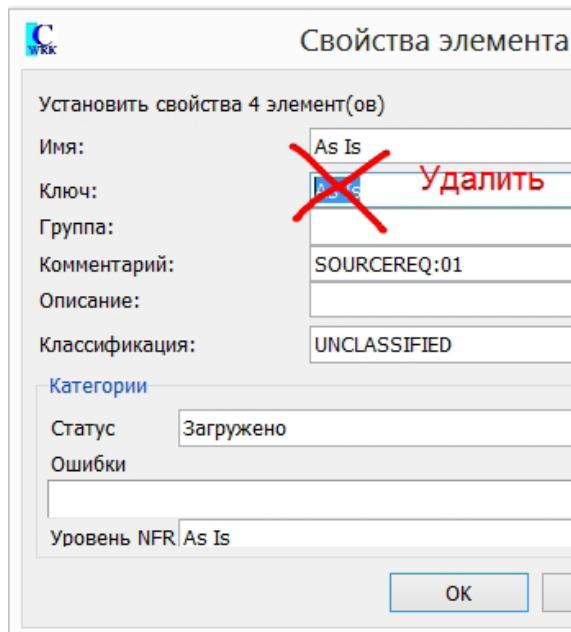


## 6. Нажать кнопку **OK**

После этого в выбранный элемент по умолчанию добавится информация из текстовых полей второго элемента, в фрейм NOTES добавится запись об объединении, а также на элемент по умолчанию перекинутся все связи от второго элемента. После того, как эти операции будут выполнены системой, автоматически удалится второй элемент.

## Установить общие атрибуты для производных требований

Обратите внимание, что дочерние элементы были созданы с ключами типа 1-1, 1-2 и т.д. Если вам не нужны эти ключи для дальнейшей работы, то вы можете просто выделить все четыре элемента и очистить поле Key в их общих свойствах:



Запомните этот метод, его можно использовать для групповой корректировки свойств элементов. Особенно часто это используется для корректировки значений категории Статус.

### Проставлять связи от уже созданных элементов

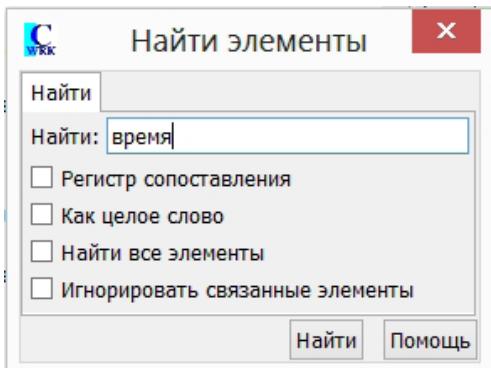
По мере того, как вы будете продвигаться по исходным требованиям, все больше элементов будет повторяться. Нет необходимости заново создавать повторяющиеся элементы, достаточно проставлять связи от созданных к исходным требованиям.

Если элементов создано уже довольно много, но вы помните атрибуты необходимого элемента (например, слово из наименования), то можно воспользоваться быстрым поиском. Для этого:

1. Откройте запрос и представление, которое потенциально может содержать ваш элемент и атрибут, по которому будет идти поиск.

Как правило это запрос All (например, NFR-All, SBS-All), а представление - Document (Simple) (например, NFR-Document (Simple)).

2. Сделав активной панель с данным запросом нажмите **CTRL+F**
3. Введите строку для поиска



#### 4. Нажмите кнопку **Найти**

Если элементы, содержащие указанную строку существуют, то Cradle переведет фокус на первый из них. Чтобы перейти к следующему элементу, не закрывая диалог поиска, снова нажмите **Найти**

Identity	Ver	Dft	Name
9	9	A	В основном доставка производится с утра
10	10	A	Конкретное время доставки зависит от маршрута водителя
11	11	A	Номер наклейки на пакет должен соответствовать номеру
12	12	A	Заказ
13	13	A	Номер накладной
14	14	A	Накладная должна

### Зафиксировать конфликтующие требования

Если в процессе анализа исходных требований выявлены конфликтующие пары, то между ними можно создать связь типа CONFLICT. Для этого просто перетащите один элемент на другой (или воспользуйтесь горячими клавишами создания связей **CTRL+>**, **CTRL+<**, **CTRL+L**).

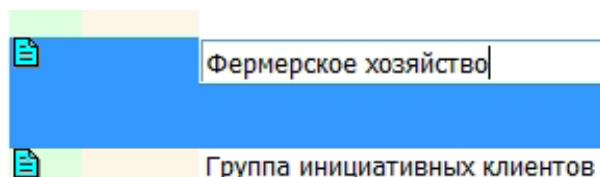
В появившемся диалоговом окне создания связи выберите тип связи CONFLICT.

Это позволит не терять из виду противоречащие друг другу требования и своевременно разрешать конфликты.

## Реструктурировать формализованные требования

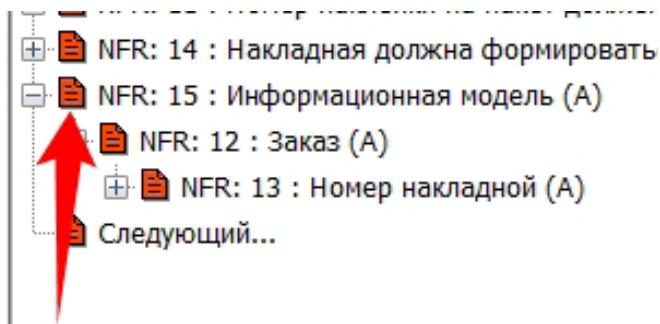
Анализ и проектирование - итеративные процессы. Невозможно сразу увидеть структуру среди исходных или производных формализованных требований, поэтому одна из наиболее частых групп операций - это переименование и переструктурирование элементов. В Cradle это выполняется очень легко - переименовать любой элемент можно прямо в открытом представлении (в табличной форме или в форме документа), для этого:

1. Выделить необходимый элемент
2. Щелкнуть на поле, подлежащее редактированию
3. Ввести требуемое значение:



Если элемент выведен в режиме Дерево, то необходимо:

1. Щелкнуть на иконку элемента, что приведет к его открытию



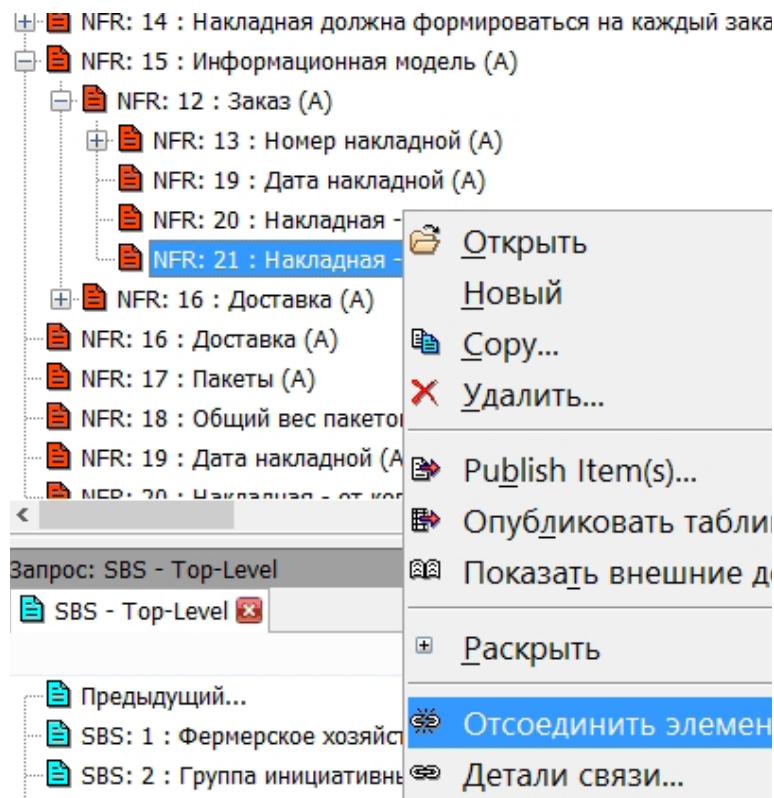
2. Изменить необходимые поля.
3. Сохранить и закрыть элемент.

Реструктуризацию элементов - изменение их связей возможно проводить либо с помощью drag&drop (перетаскиванием), либо с помощью горячих клавиш для создания связей.

Чтобы удалить лишние связи:

1. Выделите элемент, который необходимо отсоединить

## 2. В контекстном меню выберите **Отсоединить**



Вы также можете выделить и отсоединить несколько элементов.

## Завершить первичный анализ и формализацию исходных требований

Итак, все первичные формализованные требования выявлены и структурированы, список вопросов по загруженным требованиям сформирован. Из исходных неформализованных требований извлечены:

- Цели и задачи создания системы
- Требования к информационной модели
- Правила выполнения основных функций - временные и другие характеристики
- Требования к функциям системы
- Требования к ролям системы
- Вопросы на согласование

Далее необходимо оценить состояние проекта, провести его планирование (если это не было сделано раньше) и согласовать вопросы с Заказчиком.

Cradle Enterprise: WorkBench - Базовый курс Cradle - MANAGER - (Анализ ИД) - (15 непрочитанных оповещений)

Файл Редактировать Представление Элемент Запрос Инструменты Администрирование Окно Помощь

Requirement FUNCTION NFR SBS SR Event

Запрос: SR - Анализ ИД

Клиент

37

Доставка окончательно согласовывается по телефону за два дня, после этого уже отменить заказ нельзя. Обзвонить необходимо обязательно всех, но часто сразу дозвониться не получается, поэтому надо, чтобы была возможность указать до кого уже дозвонились и результаты дозвона.

Согласование доставки

Обзвон клиентов для согласования доставки

Позвонить клиенту для согласования доставки

Зафиксировать результаты звонка по со

Выбрать клиентов, с которыми доставка

Нельзя отменить заказ после согласован

Доставка согласовывается за два дня

Обязательно обзвонить всех клиентов, с

Доставка согласована или нет

Результаты согласования доставки

38

Запрос: SR - Вопросы

SR - Вопросы

Identity Ver Df

Предыдущий...

SR: 52 : Вопросы на согласование (A) 52 A

SR: 53 : Что означает хорошие продукты? (A) 53 A

SR: 54 : Что означает свежие продукты? (A) 54 A

SR: 55 : Что такое систематическая доставка? (A) 55 A

SR: 56 : Что означает постоянная клиентская база? (A) 56 A

SR: 57 : Какие типы интерфейсов для заказа необходимы? (A) 57 A

SR: 59 : Что если человек не сможет принять заказ (A) 59 A

SR: 60 : Кто осуществляет обзвон? (A) 60 A

SR: 61 : Что значит "работает" (A) 61 A

SR: 62 : Что значит "ближайший" (A) 62 A

SR: 63 : Как должна происходить доставка заказа между договорившимися? (A) 63 A

Запрос: NFR - Top-Level

NFR - Top-Level

Предыдущий...

NFR: 1 : Цель создания системы - обеспечить себя (инициативную г 1 A

NFR: 15 : Требования к информационной модели (A) 2 A

NFR: 12 : Заказ (A) 3 A

NFR: 22 : Сумма заказа (A) 4 A

NFR: 26 : Накладная (A) 5 A

NFR: 38 : Примечания к заказу (A) 6 A

NFR: 47 : Доставка согласована или нет (A) 7 A

NFR: 48 : Результаты согласования доставки (A) 8 A

NFR: 52 : Район доставки заказа (A) 9 A

NFR: 53 : Желаемая дата доставки (A) 10 A

NFR: 54 : Запланированная дата доставки (A) 11 A

NFR: 16 : Доставка (A) 12 A

NFR: 40 : Справочник продуктов (A) 13 A

NFR: 55 : Фермерское хозяйство (A) 14 A

NFR: 63 : Настройки системы (A) 15 A

NFR: 66 : Правила управления заказами (A) 16 A

NFR: 11 : Номер наклейки пакет должен соответствовать номеру заказа (A) 17 A

NFR: 14 : Накладная должна формироваться на каждый заказ отдельно (A) 18 A

NFR: 43 : Правила отмены заказа (A) 19 A

NFR: 44 : Правила согласования заказа (A) 20 A

NFR: 49 : Возможность одновременной работы с заказами неско 21 A

NFR: 51 : Окончательную сумму заказа можно посчитать только вручную (A) 22 A

Запрос: FUNCTION - Top-Level

FUNCTION - Top-Level

Предыдущий...

FUNCTION: 3 : Оформление заказа (A) 1 A

FUNCTION: 9 : Подготовка заказа (A) 2 A

FUNCTION: 22 : Печать документов (A) 3 A

FUNCTION: 23 : Доставка заказов (A) 4 A

FUNCTION: 24 : Коммуникационные функции (A) 5 A

FUNCTION: 10 : Согласование доставки (A) 6 A

FUNCTION: 11 : Обзвон клиентов для согласования (A) 7 A

FUNCTION: 14 : Выбрать клиентов, с которыми доставлять (A) 8 A

FUNCTION: 15 : Договориться с ближайшим (A) 9 A

FUNCTION: 17 : Информирование клиента (A) 10 A

FUNCTION: 16 : Проинформировать о сумме заказа (A) 11 A

FUNCTION: 18 : Отправить смс за попадающими под условия (A) 12 A

FUNCTION: 20 : Обсуждение рецептов (A) 13 A

FUNCTION: 21 : Обмен рецептами (A) 14 A

FUNCTION: 25 : Управление заказами (A) 15 A

FUNCTION: 1 : Сортировка заказов (A) 16 A

FUNCTION: 2 : Планирование взаимодействия с фермерами (A) 17 A

Следующий...

Запрос: SBS - Top-Level

SBS - Top-Level

Identity Ver Dft Name Key

Предыдущий...

SBS: 1 : Фермерское хозяйство (A) 1 A Фермерское хозяйство

SBS: 2 : Группа инициативных клиентов (A) 2 A Группа инициативных клиентов

SBS: 4 : Водитель (A) 4 A Водитель

SBS: 5 : Мясник (A) 5 A Мясник

SBS: 8 : Главный менеджер (A) 8 A Главный менеджер

SBS: 6 : Клиент (A) 6 A Клиент

Следующий...



# Оценка объема и сложности проекта

Сложность проекта коррелирует с объемом неопределенностей. В случае рассматриваемой схемы проекта, объем неопределенностей определяется

- количеством вопросов, требующих согласования
- объемом исходных требований с ошибками как типа Неоднозначности, так и Конфликт, поскольку конфликтующие требования - тоже создают неопределенность.

Возможно разработать и другие метрики, в рамках этого учебника разберем технические аспекты настройки расчета метрик в Cradle. Вы можете затем разработать свои методы оценки и настроить их по аналогии.

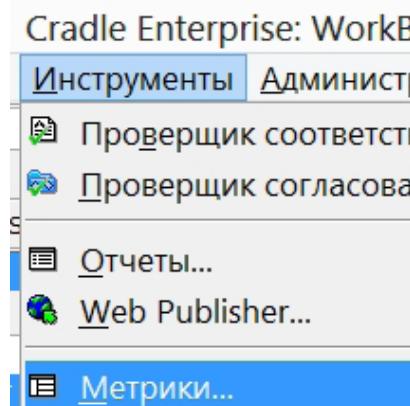
## Настроить метрики и опубликовать результаты расчетов

Метрики основываются на запросах. Для расчета любой метрики в начале мы должны построить запрос, который будет выбирать элементы, на основе которых будет производится расчет метрики.

### Пример1. Соотношение исходных требований по статусам и ошибкам

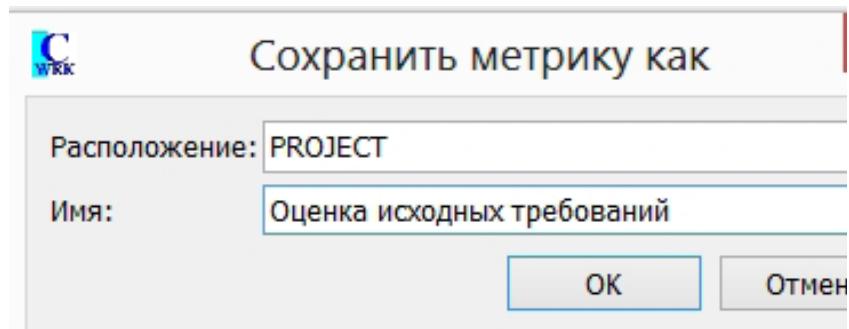
Настроим метрику, которая будет отражать процентное соотношение исходных требований, имеющих разные атрибуты - статус и ошибки. Для этого:

1. Выбрать в меню **Инструменты** -> **Метрики**



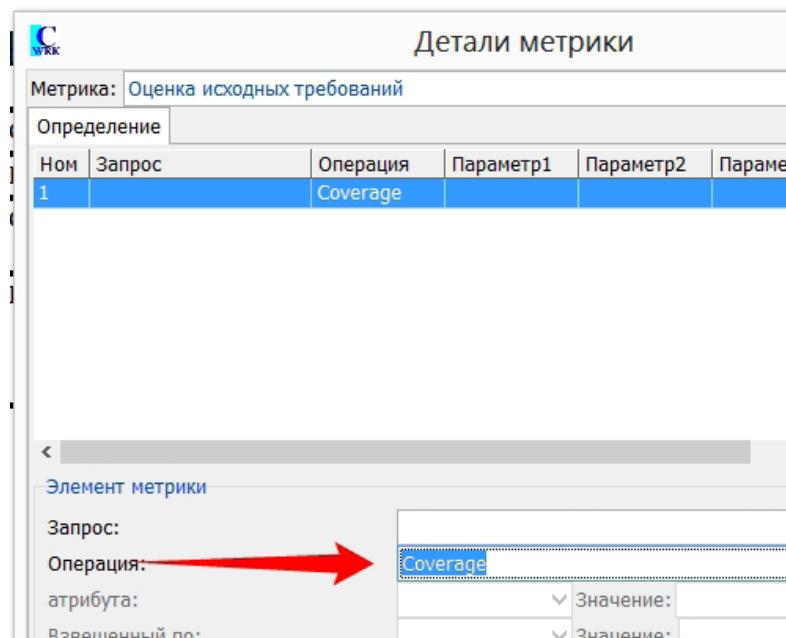
2. В диалоге **Детали метрики** нажать кнопку **Сохранить как** и ввести наименование метрики.

менование создаваемой метрики

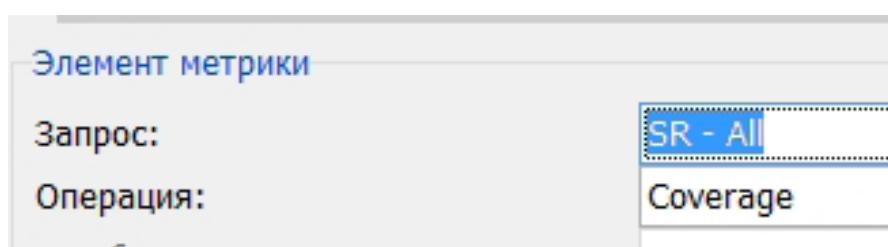


3. Нажать кнопку **Добавить**

4. Выделив появившуюся строчку, выбрать в выпадающем списке Операция = Coverage



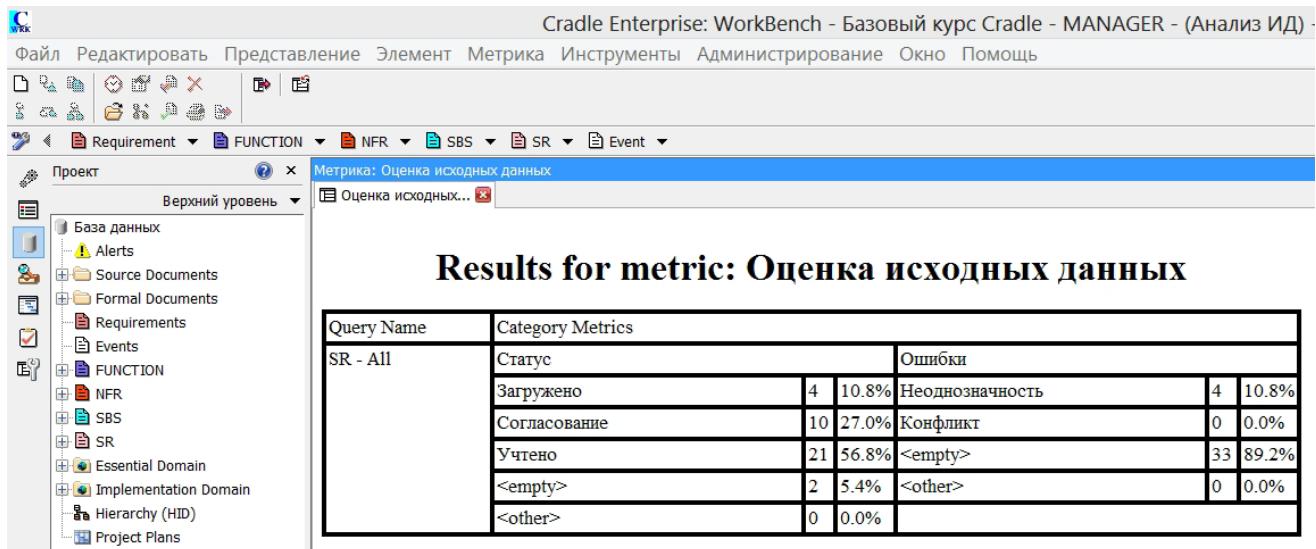
5. В выпадающем списке Запрос выбрать SR-All



6. Сохранить выполненные настройки, нажав кнопку **Сохранить**

7. Запустить расчет и вывод результатов, нажав кнопку **Новый**

В активном окне будут выведены результаты расчета метрики:

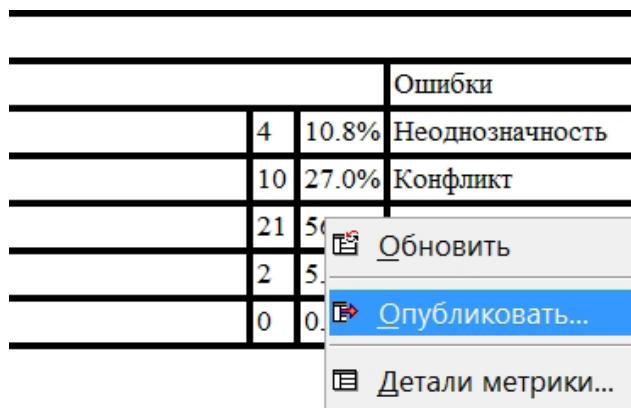


The screenshot shows the Cradle Enterprise WorkBench interface. The menu bar includes 'Файл', 'Редактировать', 'Представление', 'Элемент', 'Метрика', 'Инструменты', 'Администрирование', 'Окно', and 'Помощь'. The toolbar has icons for file operations like Open, Save, Print, and Export. The navigation bar shows 'Requirement', 'FUNCTION', 'NFR', 'SBS', 'SR', and 'Event'. The left sidebar is a tree view of the project structure under 'Проект' (Top Level), including 'База данных', 'Alerts', 'Source Documents', 'Formal Documents' (Requirements, Events), 'FUNCTION', 'NFR', 'SBS', 'SR', 'Essential Domain', 'Implementation Domain' (Hierarchy (HID)), and 'Project Plans'. The main content area is titled 'Метрика: Оценка исходных данных' and shows a table titled 'Results for metric: Оценка исходных данных'. The table has two columns: 'Query Name' (SR - All) and 'Category Metrics'. The 'Category Metrics' column is further divided into 'Статус' and 'Ошибки'. The data is as follows:

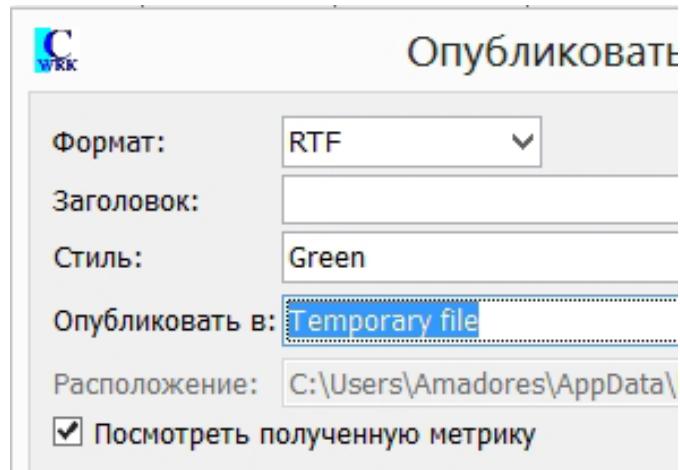
Query Name	Category Metrics					
SR - All	Статус			Ошибки		
	Загружено	4	10.8%	Неоднозначность	4	10.8%
	Согласование	10	27.0%	Конфликт	0	0.0%
	Учтено	21	56.8%	<empty>	33	89.2%
	<empty>	2	5.4%	<other>	0	0.0%
	<other>	0	0.0%			

Созданную метрику можно не только вывести в интерфейсе Workbench, но и опубликовать в форматах RTF, Html, CSV. Для этого

8. В диалоге Детали метрики нажать кнопку **Опубликовать** или выбрать **Опубликовать** в контекстном меню открытой метрики



9. В настройках публикации указать:



## 10. Нажать **OK**

После этого Cradle рассчитает значение метрики и выведет результат в RTF-файл:

### Results for metric: Оценка исходных данных

Query Name	Category Metrics					
SR - All	Статус	Ошибки				
	Загружено	4	10.8%	Неоднозначность	4	10.8%
	Согласование	10	27.0%	Конфликт	0	0.0%
	Учтено	21	56.8%	<empty>	33	89.2%
	<empty>	2	5.4%	<other>	0	0.0%
	<other>	0	0.0%			

Благодаря этим данным, которые мы получили довольно быстро, мы можем оценить картину в целом. Итак, мы видим, что на согласование у нас отложено почти одна треть элементов - 27%. Это довольно много. При этом 4 элемента в состоянии Загружено, т.е. это те элементы, которые были загружены, но еще не учтены.

По ошибкам у нас вполне неплохой показатель - на данном этапе конфликтов не обнаружено, а неоднозначность выявлена в 10,8% процентах элементов.

### Пример2. Количество формальных требований нижнего уровня

Для оценки объема проекта создадим вторую метрику, которая будет выводить количество функций, компонентов и нефункциональных требований нижнего уровня. На нижнем уровне находятся элементы, не являющиеся клас-

сификаторами, это позволит нам точнее оценивать объем проекта. Особенно полезна эта информация будет тогда, когда накопится статистика по нескольким проектам и можно будет точнее переводить количество элементов в трудозатраты.

Для создания такой метрики выполните настройки, указанные на следующем рисунке.

1. Создать новую метрику

2. Добавить строки для запросов

- NFR - Bottom-Level
- FUNCTION-Bottom-Level
- SBS-Bottom-Level

3. Установить для всех запросов операцию = Count items (подсчитать количество элементов)

Скриншот окна «Детали метрики» в CBR. Виджет «Определение» показывает таблицу с тремя строками:

Ном	Запрос	Операция	Параметр1	Параметр2	Параметр3
1	FUNCTION - Bottom-Level	Count			
2	SBS - Bottom-Level	Count			
3	NFR - Bottom-Level	Count			

Виджет «Элемент метрики» содержит настройки для первого запроса:

- Запрос: FUNCTION - Bottom-Level
- Операция: Count items
- атрибута: (пусто)
- Взвешенный по: (пусто)
- Сгруппировано по: (пусто)
- Подгруппа по: (пусто)
- Показать, используя представление: (пусто)

На панели инструментов в правом верхнем углу виджета расположены кнопки: Добавить, Копировать, Удалить, Наверх, Вниз. Внизу виджета расположены кнопки: Применить, Новый, Опубликовать..., Сохранить, Сохранить как, Удалить, Закрыть, Помощь.

4. Запустить метрику, нажав кнопку **Новый** или **Применить**

Вот что у нас получилось на данный момент:

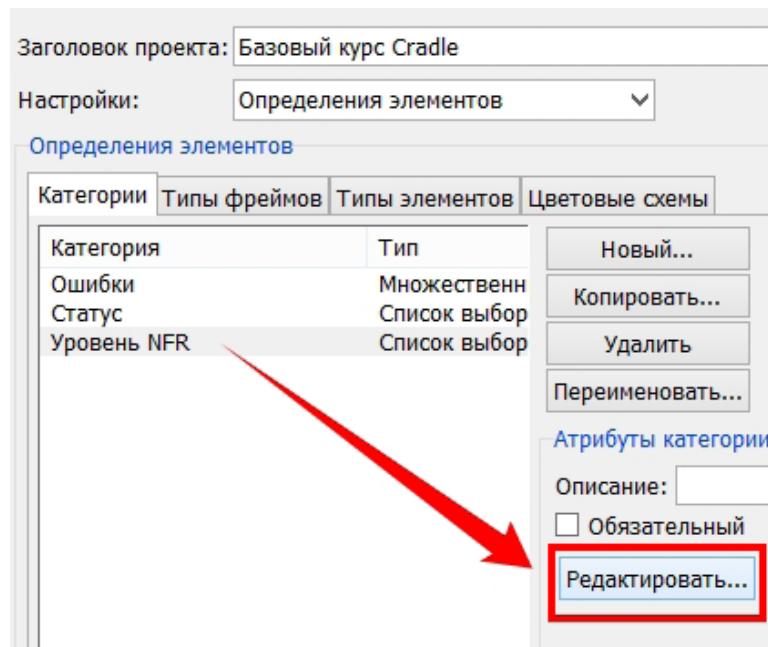
## Results for metric: 1

Query Name	Count
FUNCTION - Bottom-Level	15
Query Name	Count
SBS - Bottom-Level	5
Query Name	Count
NFR - Bottom-Level	43

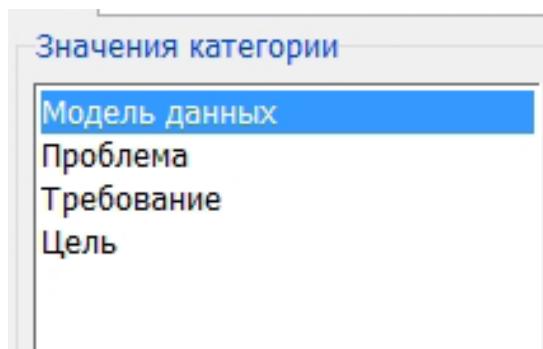
### Пример3. Размер модели данных

Требования к модели данных у нас размещены как подраздел NFR, поэтому для того, чтобы легко выделить эти элементы из других нам необходимо добавить атрибут, по которому возможно было бы осуществлять такой выбор. Для этого

1. Открыть меню **Администрирование** -> **Установки проекта**
2. Открыть раздел -> **Определения элементов** -> **Категории**
3. Выбрать категорию Уровень NFR и нажать кнопку **Редактировать**

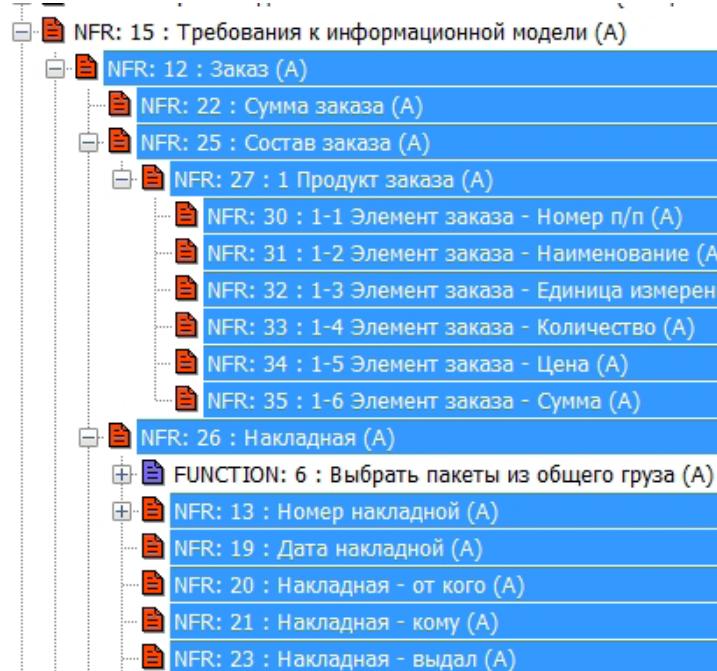


4. Добавить новую строку «Модель данных»



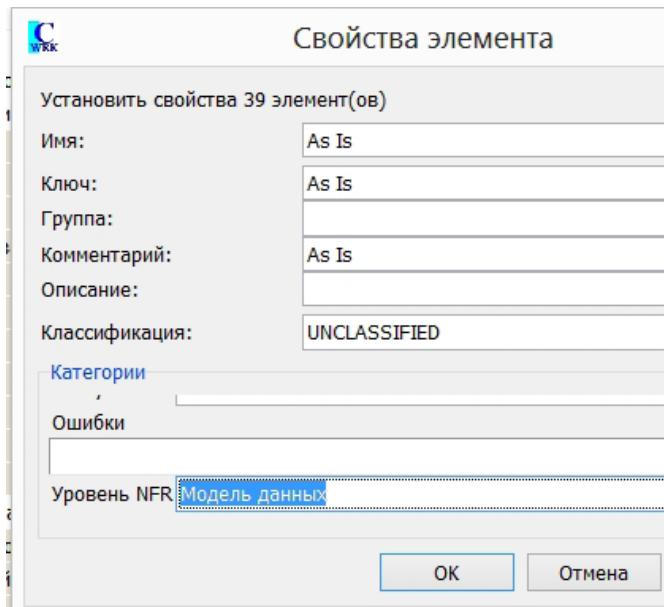
5. Сохранить все изменения схемы проекта

6. Выделить все элементы NFR, относящиеся к модели данных



7. В контекстном меню выбрать **Свойства**

8. Установить значение категории Уровень NFR в Модель данных:



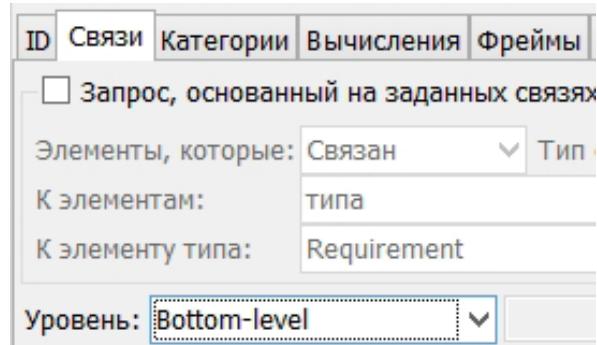
На практике удобно было выделить дополнительную детализацию, введя категории Таблица, Поле, Параметр и др. или даже создать отдельный тип элемента для отражения информационной модели проекта.

Теперь мы можем составить запрос, который выберет все элементы модели данных нижнего уровня. Для этого:

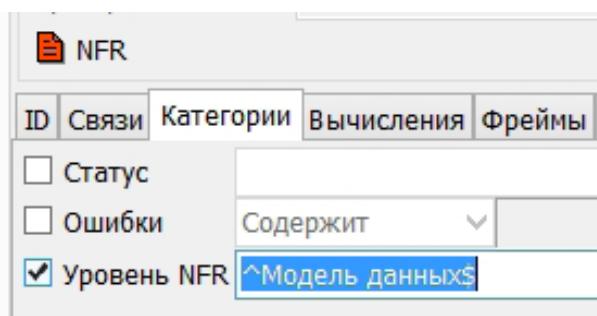
9. Открыть диалог **Детали запроса**

10. Выбрать *Тип элемента* = NFR

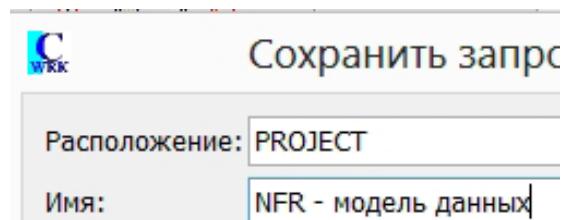
11. На вкладке **Связи** выбрать *Уровень* = Bottom-Level



12. На вкладке **Категории** выбрать *Уровень NFR* = Модель данных



13. Сохранить как «NFR - модель данных»



14. Открыть детали созданной ранее метрики «Оценка формализованных требований»

15. Добавить строку, соответствующую данному запросу

Метрика: Оценка формализ. требований

Определение

Ном	Запрос	Операция	Параметр1	Параметр2	Параметр3	П
1	FUNCTION -	Count				
2	SBS - Bottom-Level	Count				
3	NFR - Bottom-Level	Count				
4	NFR - модель данных	Count				

Элемент метрики

Запрос: NFR - модель данных

Операция: Count items

## 16. Применить изменения.

В результате у нас получилась такая таблица:

Query Name	Count
FUNCTION - Bottom-Level	15
Query Name	Count
SBS - Bottom-Level	5
Query Name	Count
NFR - Bottom-Level	43
Query Name	Count
NFR - модель данных	29

По данной метрике мы видим, что основная составляющая наших нефункциональных требований - это требования к модели данных.

Попробуйте и сами настройте метрику, которая отражала бы отдельно нефункциональные требования, не являющиеся моделью данных. Для этого в категориях используйте следующую строку:

Восклицательный знак используется для математического отрицания (НЕ).

#### Пример 4. Сводная таблица исходных требований

Рассмотрим еще один интересный пример типа метрик - сводные таблицы, создадим сводную таблицу для исходных требований. Для этого:

1. Открыть диалог **Детали метрики**
2. Выбрать метрику «Оценка исходных данных»
3. **Сохранить как** «Исходные данные сводка»
4. Установить следующие настройки для метрики:
  - Запрос = SR-All
  - Операция = Pivot table
  - Значени = Статус
  - Значение = Ошибки
  - Показать, используя представление =SR-Анализ
5. Сохранить и применить метрику

Detali metriki

Метрика: Исходные данные - сводка

Определение

Ном	Запрос	Операция	Параметр1	Параметр2	Параметр3	П
1	SR - All	Pivot	Статус	Ошибки		

Добавить  
Копировать  
Удалить  
Наверх  
Вниз

Элемент метрики

Запрос: SR - All  
Операция: Pivot table  
атрибута:  
Взвешенный по:  
Сгруппировано по:  
Подгруппа по:  
Показать, используя представление: SR - Анализ

Применить Новый Опубликовать... Сохранить Сохранить как Удалить Закрыть Помощь

После применения метрики будет выведена таблица, которая отражает количество элементов по каждой категории. Обратите внимание, что данная таблица является интерактивной. Щелкнув на любую из цифр вы получите в активной панели все элементы, которые соответствуют условию ячейки.

Query Name: SR - All

Ошибки	Статус				
	<Unset>	Загружено	Согласование	Учтено	TOTALS
<Unset>	2	3	10	18	33
Неоднозначность	0	1	0	3	4
Конфликт	0	0	0	0	0
TOTALS	2	4	10	21	37

Например, щелкнув на элемент «3», находящийся на пересечении столбца Учтено и строки Неоднозначность, вы получите быстрый доступ к трем исходным требованиям, которые были уже учтены, но которым присвоена ошибка Неоднозначность. При этом требования будут открыты в заданном в настройках метрики представлении. В данном случае - это SR-Анализ и поэтому мы сра-

зую же увидим все связанные с ними производные требования.

**Results for metric: Исходные данные - сводка**

**Query Name: SR - All**

Ошибки	Статус				
	<Unset>	Загружено	Согласование	Учтено	TOTALS
<Unset>	2	3	10	18	33
Неоднозначность	0	1	0	3	4
Конфликт	0	0	0	0	0
<b>TOTALS</b>	<b>2</b>	<b>4</b>	<b>10</b>	<b>21</b>	<b>37</b>

**Запрос: SR - All**

Номер	Описание	Статус
1	Приобрести хорошие продукты для семьи в супермаркетах мегаполисов становится все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выживания» им не хватает организованности для того, чтобы объединиться с другими хозяйствами и решить вопросы систематической доставки продуктов в мегаполис из пригородов. Группа инициативных клиентов (далее Группа) решает взять на себя эти организационные функции, чтобы прежде всего обеспечить себе свежими продуктами на постоянной основе. Для этого им необходимо набрать постоянную клиентскую базу, чтобы организовать оптовую закупку, т.к. ради двух кг мяса никто не будет организовывать забой скотины.	Цель создания
2	С заказами работает несколько человек – в зависимости от доступного времени.	Организовать
3	Поскольку необходимо брать по много кг мяса, то необходима возможность клиентам договориться с ближайшим.	Набрать постс

## Запустить сохраненную метрику

Чтобы вызвать созданную ранее метрику необходимо:

1. Перейти на панель **Запрос**, в раздел **Метрики**.
2. Нажать на иконку сохраненной метрики и она откроется в активной рабочей панели.

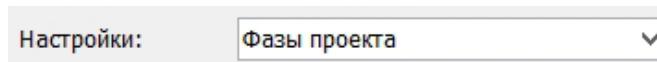
## Настроить панель фаз для быстрого вызова метрик

В проекте может быть настроено довольно много запросов, представлений, метрик, при этом лишь часть из них будет использоваться на постоянной основе. Такие часто используемые вызовы, стандартные для всей команды проекта удобно сохранять на панели Фаз.

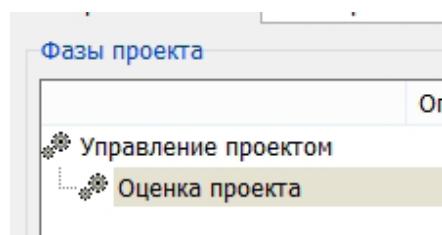
Панель Фаз представляется собой дерево папок, которые могут включать вызовы запросов, матриц, метрик, отчетов, графиков, генерации документов, отображения плана и даже вызов команд.

Настроим быстрый вызов одной из созданных метрик. Для этого:

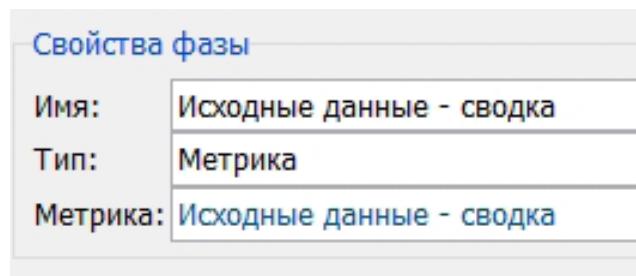
1. Открыть меню **Администрирование** -> **Установки проекта**, раздел **Фазы проекта**



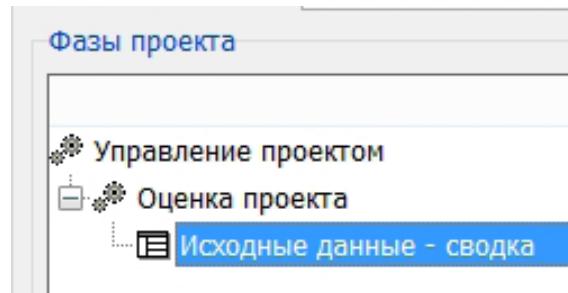
2. Нажать **Добавить** несколько раз
3. Поименовать созданные элементы-разделы панели фаз следующим образом:



4. В разделе Оценка проекта создать еще один узел и настроить на вызов метрики «Исходные данные - сводка»

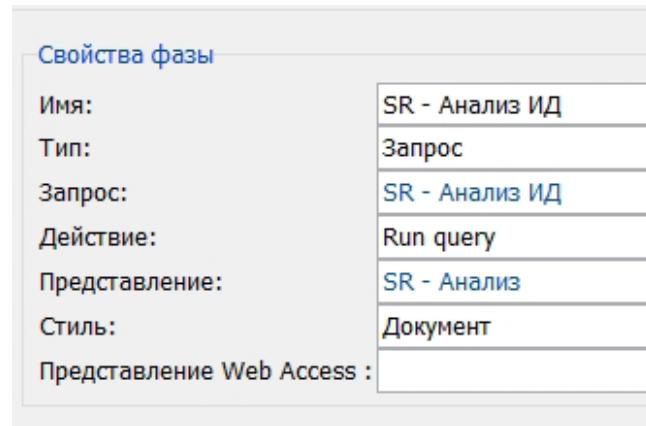


После чего панель Фаз будет выглядеть следующим образом:

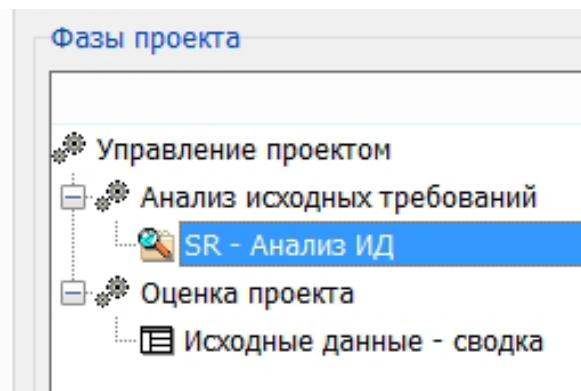


Добавим на панель Фаз также вызов запроса и представления для анализа исходных данных. Для этого:

5. Добавить раздел Анализ исходных требований, в который добавить вызов запроса «SR-анализ ИД»



В итоге должна получиться следующая картина Фаз:

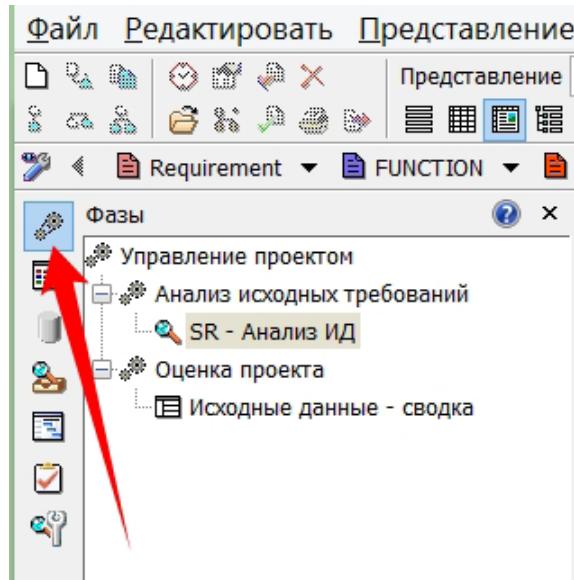


На этом пока все. Сохраним настройки и перейдем в панель Фаз, чтобы оценить ее работу.

Для этого:

6. Нажать **OK**

7. Открыть панель Фаз, нажав на шестеренки:



Теперь, чтобы запустить любой из элементов панели Фаз нажмите два раза на соответствующий заголовок или один раз на иконку элемента. Проверьте, что все созданные элементы работают и запрос/метрика открываются.

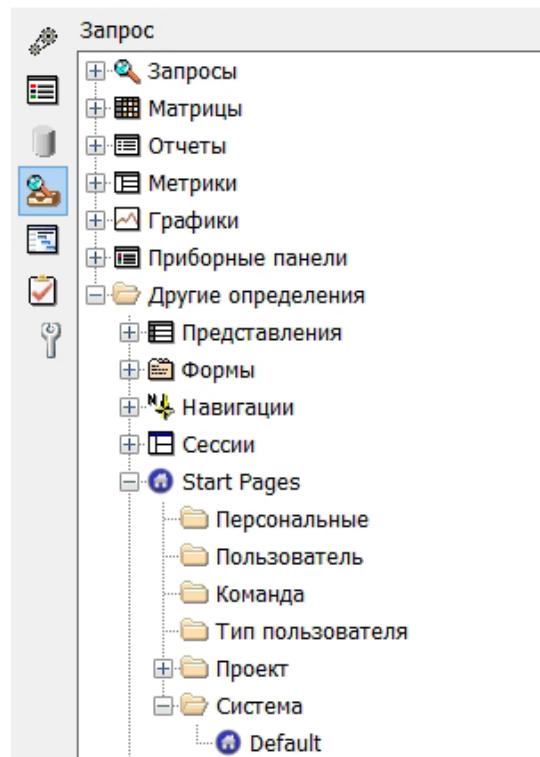
### **Настроить стартовую страницу для быстрого вызова метрик**

Стартовые страницы, как и панель Фаз, позволяют организовать быстрый доступ к данным и функциям. Главное отличие стартовых страниц от панели Фаз в том, что Стартовые страницы могут быть индивидуальными для каждого пользователя или для групп пользователей. И эта функциональность доступна как в Enterprise, так и в персональных версиях Cradle.

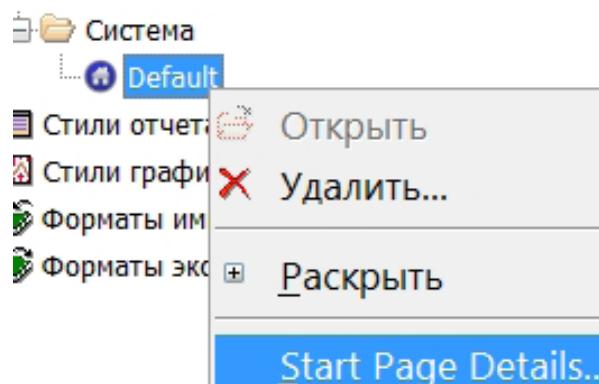
Таким образом, панель Фаз удобна для организации стандартного процесса проектирования для корпоративных пользователей, а Стартовые страницы - для индивидуальной подстройки системы по свои конкретные задачи соответствующие роли в проекте.

Для того, чтобы создать и настроить свою стартовую страницу:

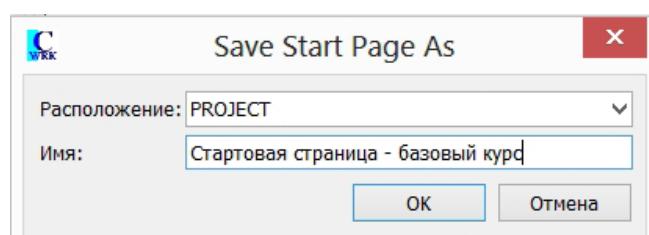
1. Перейти в **Панель запросов** и открыть раздел **Start Pages**



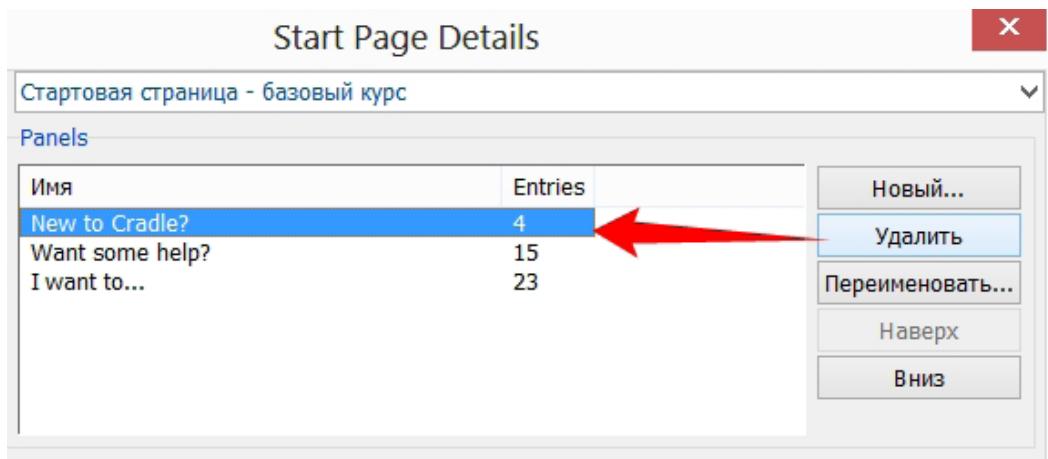
2. Выбрать стартовую страницу Default в разделе **Система**
3. В контекстном меню выбрать **Start Page Details**



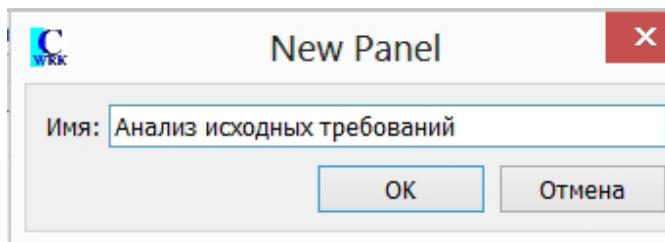
4. Пересохранить открывшиеся настройки под новым именем, нажав **Сохранить как**



5. Выделить и удалить все разделы, унаследованные от страницы по умолчанию



6. Создать первый раздел (панель), нажав кнопку **Новый**



Добавить в раздел элемент вызова запроса, который мы ранее использовали для анализа исходных требований. Для этого:

7. Нажать **Новый** в разделе **Entries**

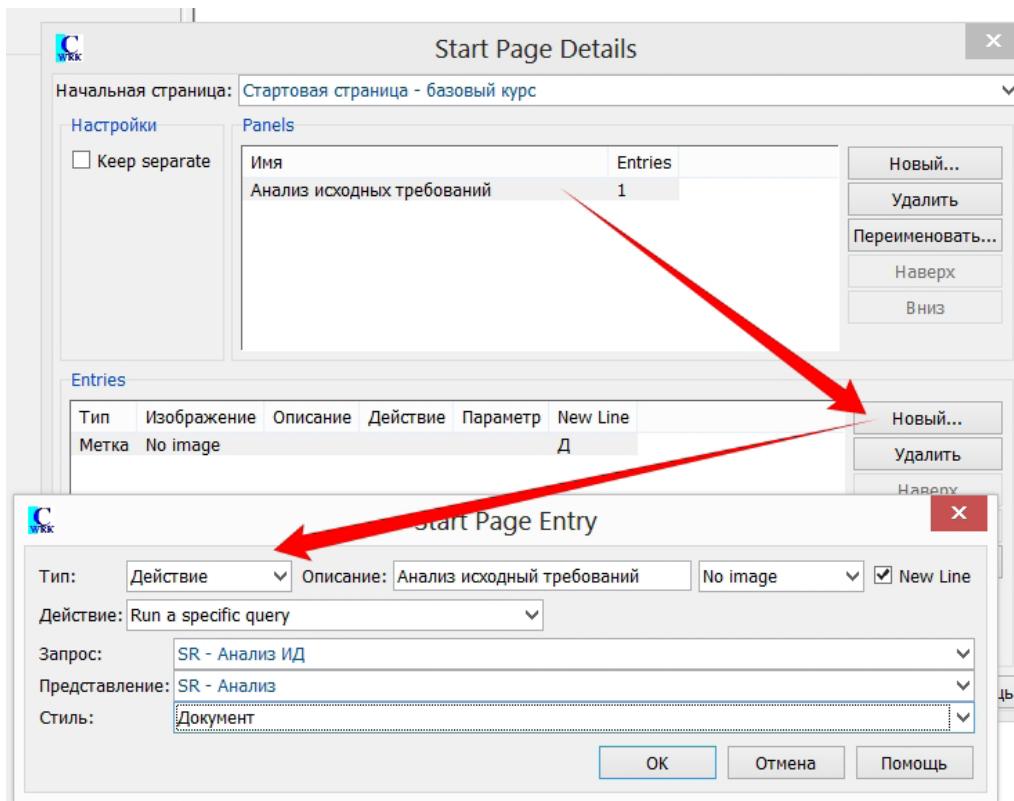
8. Выбрать *Tip* = Действие

9. Выбрать *Действие* = Run a specific query

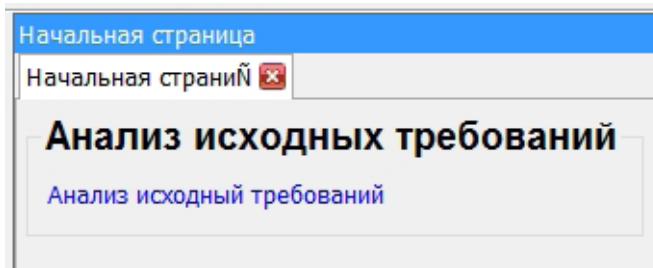
10. Выбрать Запрос = SR - Анализ ИД и соответствующее представление в стиле Документ.

11. Нажать **OK** и Сохранить

12. Нажать **Preview**



После этого будет отображено текущее состояние стартовой страницы



По аналогии добавьте второй раздел, который будет содержать уже вызов созданных в предыдущем разделе метрик. Для этого в качестве *Действия* выберите «Generate a specific metric».

Start Page Details

Начальная страница: Базовый курс 3SL Cradle

Настройки

Keep separate

Panels

Имя	Entries	
Анализ исходных требований	1	<input type="button" value="Новый..."/>
Оценка объема и сложности проекта	3	<input type="button" value="Удалить"/>
		<input type="button" value="Переименовать..."/>
		<input type="button" value="Наверх"/>
		<input type="button" value="Вниз"/>

Entries

Тип	Изображение	Описание	Действие	
Действие	No image	Оценка исходных данных	Generate a specific metric	<input type="button" value="Новый..."/>
Действие	No image	Оценка формализованных требований	Generate a specific metric	<input type="button" value="Удалить"/>
Действие	No image	Исходные данные - сводка	Generate a specific metric	<input type="button" value="Наверх"/>

Start Page Entry

Тип: Действие Описание: Исходные данные - сводка Действие: Generate a specific metric Метрика: Исходные данные - сводка

New Line

В итоге у вас должна получится следующая стартовая страница, которая далее может быть дополнена:

Начальная страница

Начальная страница

**Анализ исходных требований**

Анализ исходных требований

**Оценка объема и сложности проекта**

Оценка исходных данных  
Оценка формализованных требований  
Исходные данные - сводка

## Настроить панель индикаторов (KPI)

Ключевые показатели производительности (KPI - Key Performance Indicators) - это вычисляемые показатели производственных процессов, отражающие ситуацию в целом.

Мы считаем, что KPI - крайне полезный вспомогательный инструмент для контроля проекта, при условии, что решения принимаются не на основании только лишь одних значений индикаторов, а на основе комплексного анализа производственной ситуации. Индикаторы - это всего лишь повод своевременно задать необходимые вопросы о ходе процессов, а не средства измерения людей. Пренебрежение этим правилом приводит к крайне негативным последствиям. Подробнее мы обсуждаем данные вопросы на наших занятиях.

Суть работы панели индикаторов заключается в том, что в начале рассчитываются метрики проекта, а затем, в зависимости от их значений и настроенных привязок диапазонов значение метрики раскрашивается соответствующим цветом. Это позволяет быстро оценить ситуацию, не вдаваясь в конкретные цифры.

Диапазонов для каждой метрики, а соответственно цветов не должно быть много, т.к. их сложно будет запомнить и принимать решения.

Настроим пример панели индикаторов KPI, на которой будут собраны все показатели, отражающие неопределенность (а значит риски!) проекта:

- Соотношение исходных требований со статусами (Согласование, Загружено) к общему числу исходных требований.
- Соотношение исходных требований для которых указаны какие-либо ошибки к общему числу исходных требований.
- Процент нетрассируемых к функциям нефункциональных требований (NFR) нижнего уровня по отношению к общему числу NFR нижнего уровня.
- Процент нетрассируемых к функциям элементов SBS нижнего уровня по отношению к общему числу SBS нижнего уровня.

Чтобы посчитать все выбранные показатели необходимо построить запросы, которые выберут соответствующие элементы. Нам не достает следующих

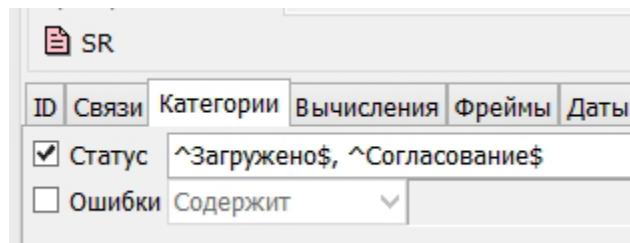
запросов:

- Выбрать все SR, имеющие статус Согласование или Загружено.
- Выбрать все SR, имеющие одну из ошибок.
- Выбрать все NFR нижнего уровня, не имеющие связей с функциями.
- Выбрать все SBS нижнего уровня, не имеющие связей с функциями.

Рассмотрим построение первого и третьего запроса, т.к. два других строятся по аналогии.

### **Выбрать все SR со статусом Согласование или Загружено**

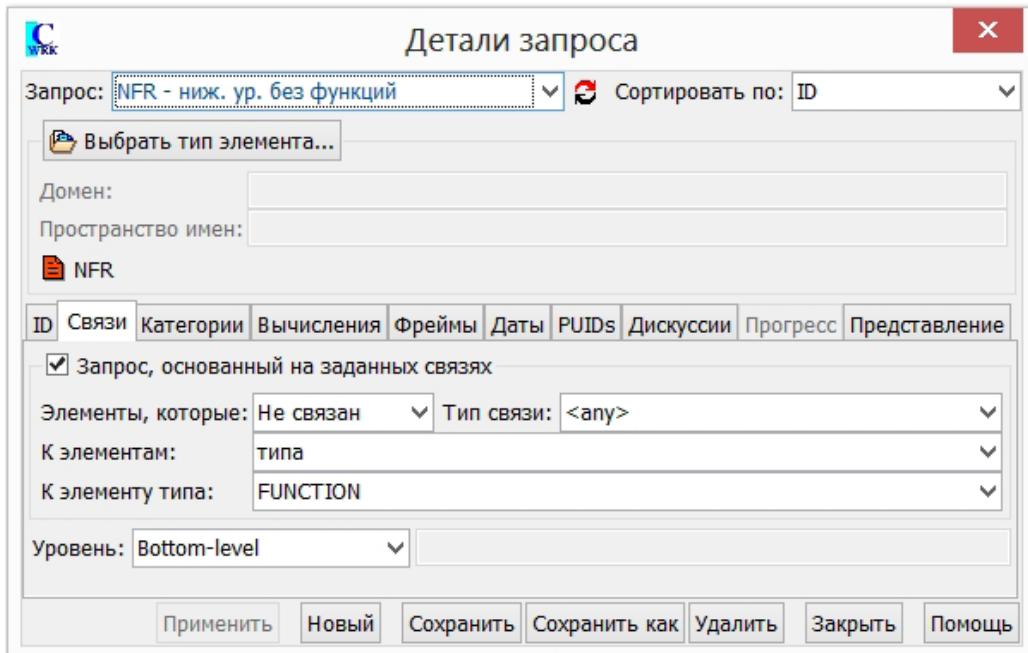
1. Открыть Детали запроса
2. Выбрать в качестве типа элемента - SR
3. Перейти на вкладку Категории
4. Выбрать в выпадающем списке *Статус* Загружено, а затем Согласование



5. Сохранить запрос под именем «SR - согласование и загружено».

### **Выбрать все NFR нижнего уровня, не имеющие связей с функциями**

1. Открыть Детали запроса
2. Выбрать в качестве типа элемента - NFR
3. Перейти на вкладку Связи
4. Установить Уровень Bottom-Level и остальные настройки, как указано на иллюстрации:



## 5. Сохранить запрос под именем «NFR - нижн. ур. без функций».

По аналогии создайте два других запроса, а запросы, которые позволят нам посчитать общее число SR, NFR нижнего уровня и SBS нижнего уровня, созданы в системе автоматически при создании этих типов элементов, что здорово экономит нам времени.

Теперь необходимо собрать все запросы в одной метрике, которая будет считать количество элементов в каждом запросе. Эти расчеты могут быть распределены и по разным метрикам, но сейчас будет удобнее собрать все в одном месте.

### Создать метрику, рассчитывающую количество элементов в каждом запросе

Чтобы создать такую метрику необходимо:

1. Открыть диалог Детали метрики
2. Открыть пустую метрику и сохранить под новым именем «Неопределенность проекта»
3. Добавить в метрику все 7 запросов, установив в качестве операции = Count.

Результат этих настроек отражен на иллюстрации ниже.

Детали метрики				
Метрика: Неопределенность проекта				
Определение				
Ном	Запрос	Операция	Параметр1	Параметр2
1	SBS - ниж. ур. без функций	Count		
2	SBS - Bottom-Level	Count		
3	NFR - ниж. ур. без функций	Count		
4	NFR - Bottom-Level	Count		
5	SR-ошибки	Count		
6	SR- согласование и загружено	Count		
7	SR - All	Count		

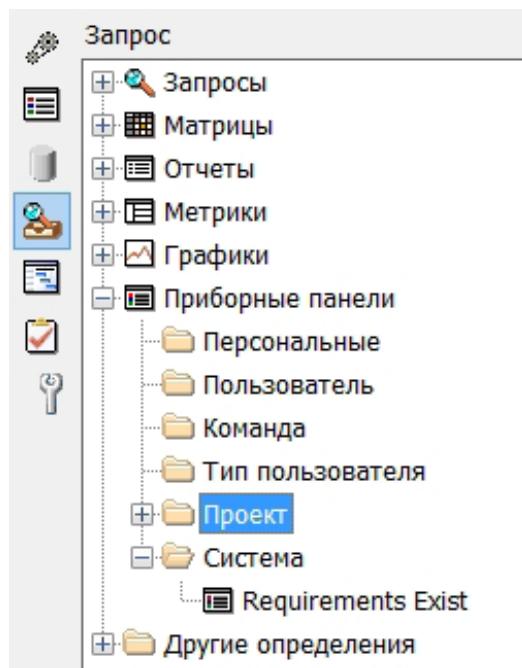
После выполнения всех настроек не забудьте их сохранить. Запустите метрику, нажав «Новый» в диалоге ее настройки и проверьте, что все отражено верно и все элементы подсчитаны.

Query Name	Count
SBS - ниж. ур. без функций	2
Query Name	Count
SBS - Bottom-Level	5
Query Name	Count
NFR - ниж. ур. без функций	34
Query Name	Count
NFR - Bottom-Level	43
Query Name	Count
SR-ошибки	4
Query Name	Count
SR- согласование и загружено	14
Query Name	Count
SR - All	37

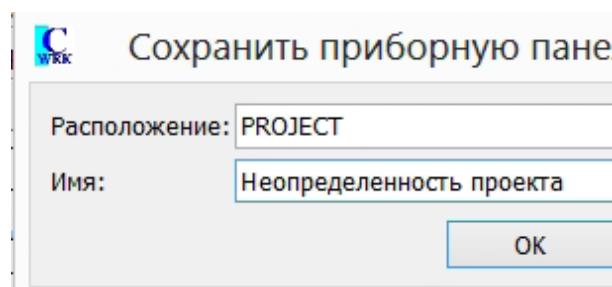
Теперь можно переходить к настройке самой панели индикаторов.

## Создать панель индикаторов с цветовыми привязками

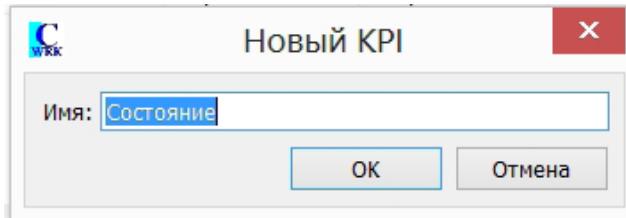
1. Открыть панель **Запрос**
2. Перейти в раздел **Приборные панели**



3. В контекстном меню панели Requirements Exist выбрать **Детали панели**
4. Удалить все строки настроек, сохранив Панель под новым именем «Неопределенность проекта»



5. Нажать «Новый», чтобы добавить первый индикатор.
6. Дать наименование «Состояние»



7. Установить следующие настройки:

Источник = Процент

Для первой метрики:

Метрика = Неопределенность

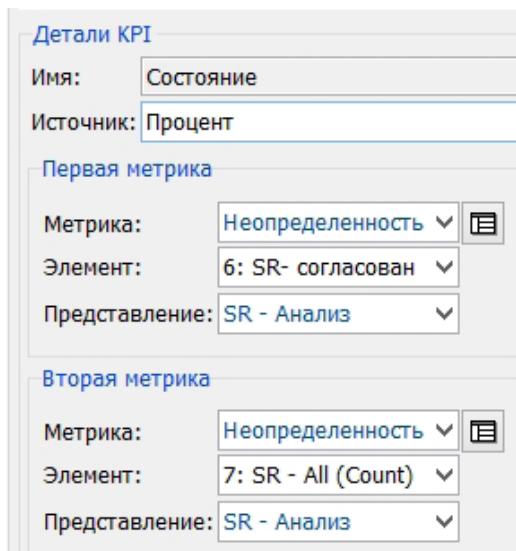
Элемент = SR -согласование и загружено

Представление - SR -Анализ

Для второй метрики, аналогично, кроме

Элемент = SR -All

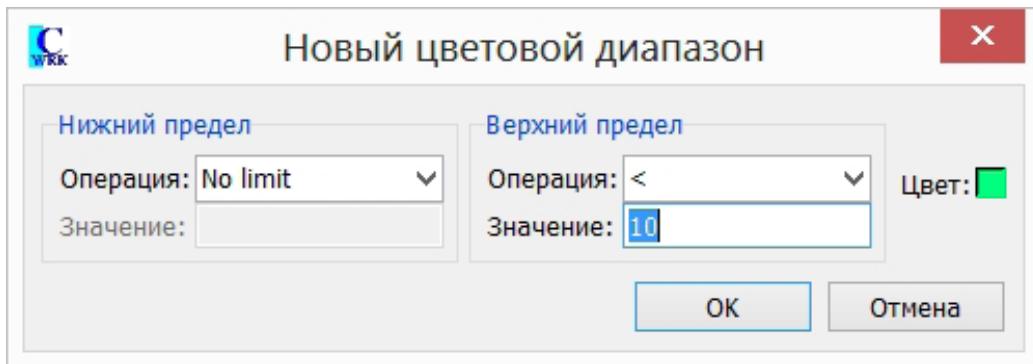
Эти настройки проиллюстрированы ниже:



Теперь необходимо настроить цветовые привязки

8. Нажать кнопку **Новый** в разделе **Настройки привязки**

9. Настроить привязку следующим образом:



Это означает, что при значении соотношения количества исходных требований на согласовании или только что загруженных к общему количеству исходных требований менее 10% ситуация довольно хороша и ее можно отражать зеленым цветом.

10. Сохранить настройку, нажав **OK**, после чего привязка отобразится в таблице:

Настройки привязки			
	Нижний	Верхний	Цвет
1	No limit	< 10	Green

11. Создать другие привязки цвета для диапазонов 10-30, 30-60, 60-100.

В итоге таблица привязок будет выглядеть следующим образом:

Настройки привязки			
	Нижний	Верхний	Цвет
1	No limit	< 10	Green
2	$\geq 10$	$< 30$	Yellow
3	$\geq 30$	$< 60$	Pink
4	$\geq 60$	No limit	Orange

12. По аналогии настройте остальные три индикатора.

Настройки для них указаны на следующих рисунках:

С

Detали приборной

Приборная панель: Неопределенность проекта

KPI	Источник	Первый элемент	Второй э
Состояние	Процент	SR- согласование и загружено (Count)	SR - All (
Ошибки	Процент	SR-ошибки (Count)	SR - All (
Нетрассируемость NFR	Процент	NFR - ниж. ур. без функций (Count)	NFR - Во
Нетрассируемость SBS	Процент	SBS - ниж. ур. без функций (Count)	SBS - Во

Детали KPI

Имя: Ошибки

Источник: Процент

Первая метрика

Метрика: Неопределенность проекта

Элемент: 5: SR-ошибки (Count)

Представление: Generic - name+text+status

Вторая метрика

Метрика: Неопределенность проекта

Элемент: 7: SR - All (Count)

Представление: Generic - name+text+status

Приборная панель: Неопределенность проекта

KPI	Источник	Первый элемент	Второй эл
Состояние	Процент	SR- согласование и загружено (Count)	SR - All (C)
Ошибки	Процент	SR-ошибки (Count)	SR - All (C)
<b>Нетрассируемость NFR</b>	<b>Процент</b>	<b>NFR - ниж. ур. без функций (Count)</b>	<b>NFR - Bottom-Level (Count)</b>
Нетрассируемость SBS	Процент	SBS - ниж. ур. без функций (Count)	SBS - Bottom-Level (Count)

## Детали KPI

Имя: Нетрассируемость NFR

Источник: Процент

## Первая метрика

Метрика:	Неопределенность проекта	▼	≡
Элемент:	3: NFR - ниж. ур. без функций (Count)	▼	
Представление:	Generic - name+text+status	▼	

## Вторая метрика

Метрика:	Неопределенность проекта	▼	≡
Элемент:	4: NFR - Bottom-Level (Count)	▼	
Представление:	Generic - name+text+status	▼	

С

Детали приборной

Приборная панель: Неопределенность проекта

KPI	Источник	Первый элемент	Второй э.
Состояние	Процент	SR- согласование и загружено (Count)	SR - All (
Ошибки	Процент	SR-ошибки (Count)	SR - All (
Нетрассируемость NFR	Процент	NFR - ниж. ур. без функций (Count)	NFR - Bot
Нетрассируемость SBS	Процент	SBS - ниж. ур. без функций (Count)	SBS - Bot

Детали KPI

Имя: Нетрассируемость SBS

Источник: Процент

Первая метрика

Метрика: Неопределенность проекта

Элемент: 1: SBS - ниж. ур. без функций (Count)

Представление: Generic - name+text+status

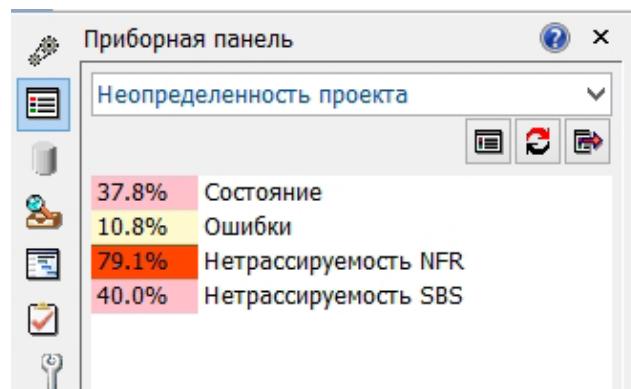
Вторая метрика

Метрика: Неопределенность проекта

Элемент: 2: SBS - Bottom-Level (Count)

Представление: Generic - name+text+status

Чтобы посмотреть результаты своих настроек откройте Приборную панель:



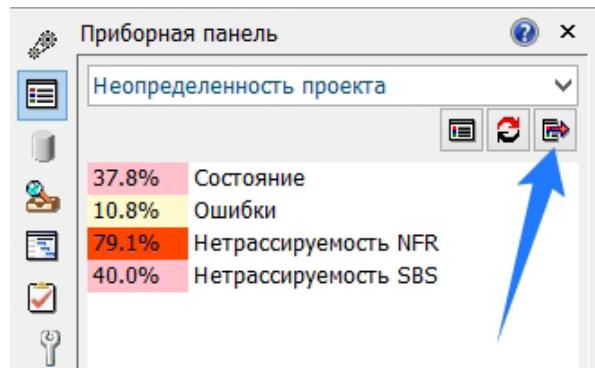
Теперь за один взгляд мы можем комплексно оценить текущее состояние рисков проекта в четырех срезах.

Вы можете нажать на любой из индикаторов и в активной рабочей панели

откроются соответствующие ему запросы. Это позволяет быстро переключиться от оценок к тем данным, на основании которых они построены и, например, посмотреть, какие именно требования содержат ошибки или находятся на согласовании.

В одном проекте может быть настроено много панелей индикаторов, которые будут объединять значимые для вас метрики.

Так же как и для метрик работает функция публикации данных панели для этого нажмите правую иконку и выберите настройки публикации.



Панель индикаторов может быть вывена в любой веб-интерфейс, например, как часть Sharepoint.

# ***Планирование и контроль проекта***

## **Три подхода к учету трудозатрат и управлению задачами**

Cradle позволяет реализовать несколько вариантов учета трудозатрат и управления проектными задачами.

Вариант1. Планирование и учет трудозатрат прямо в требованиях.

Этот вариант подходит более для фазы Сопровождения, когда новое требование одновременно является и новой задачей.

Вариант2. Создание отдельных элементов типа Задача (Task) и связывание требований с ними. При этом отчетность производится по задаче в целом за весь период.

Задача может иметь все стандартные атрибуты, такие как Дата начала, Дата окончания, Фактические трудозатраты и любые другие. Задаче может быть назначен исполнитель - пользователь Cradle, имеющий права доступа к данному проекту.

Cradle позволяет построить различные аналитические представления - отражающие статус задач и другие поля, выводить отчеты.

Вариант 3.

При необходимости использовать расширенные функции автоматического ресурсного планирования и анализа: строить сетевые графики, проводить анализ критического пути и автоматическое перестроение плана работ на основе связей, отображаемых на диаграмме Ганта, более подойдет вариант интеграции с MS Project.

Этот вариант дополняет Вариант 2, при этом для выбранной иерархии задач создается план работ MS Project, который автоматически синхронизируется с задачами в Cradle по временным характеристикам.

В этом примере мы используем Вариант 2 для формирования стека задач, еще не распланированных на конкретное время и Вариант 3 для планирования задач и отчетности по трудозатратам.

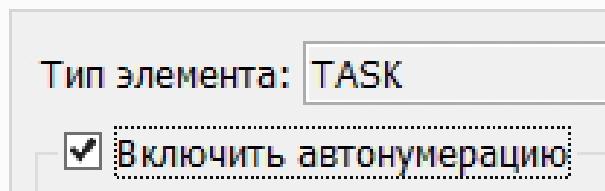
Таким образом, мы реализуем часто встречающийся на практике подход -

когда сначала задачи попадают в некоторую очередь, в которой они анализируются, расставляются их приоритеты и уже после их анализа происходит включение план работ и пересчет остальных работ.

Обратите внимание, что большая часть настроек, которые будут иллюстрироваться дальше, выполняется однократно и может быть повторно использована в любых других проектах.

## Настроить схему проекта

1. Создать тип элемента TASK в разделе меню **Администрирование ->Установки проекта ->Определения элементов** вкладка **Типы элементов**
2. Установить для него автонумерацию

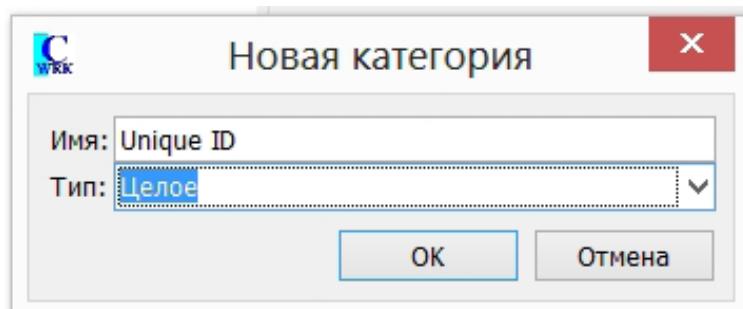


3. Создать фрейм PERFORMERS

Это поле будет использоваться для назначения исполнителей - пользователей Cradle на задачи.

Далее необходимо создать ряд категорий, которые будут использоваться для отражения временных характеристик задач, а также последующей интеграции с MS Project.

4. В этом же диалоге на вкладке **Категории** создать категории, указанные в таблице



Поле	Описание	Тип
<b>Unique ID</b>	Уникальный идентификатор задачи в MS Project	Положительное целое
<b>Summary</b>	Признак суммарной задачи	Свободный формат или значения Yes/No
<b>Milestone</b>	Признак вехи	Свободный формат или значения Yes/No
<b>Start</b>	Дата начала задачи	Дата
<b>Finish</b>	Дата окончания задачи	Дата
<b>Duration</b>	Длительности задачи	Положительное целое
<b>Actual Start</b>	Дата фактического начала задачи	Дата
<b>Actual Finish</b>	Дата фактического окончания задачи	Дата
<b>Actual Duration</b>	Фактическая длительность задачи	Положительное целое
<b>Percent Complete</b>	Процент выполнения задачи	Положительное целое
<b>Приоритет</b>	Приоритет выполнения задачи	Определяемый пользователем: Высокий, Средний, Низкий

В итоге у вас должен получится такой список категорий:

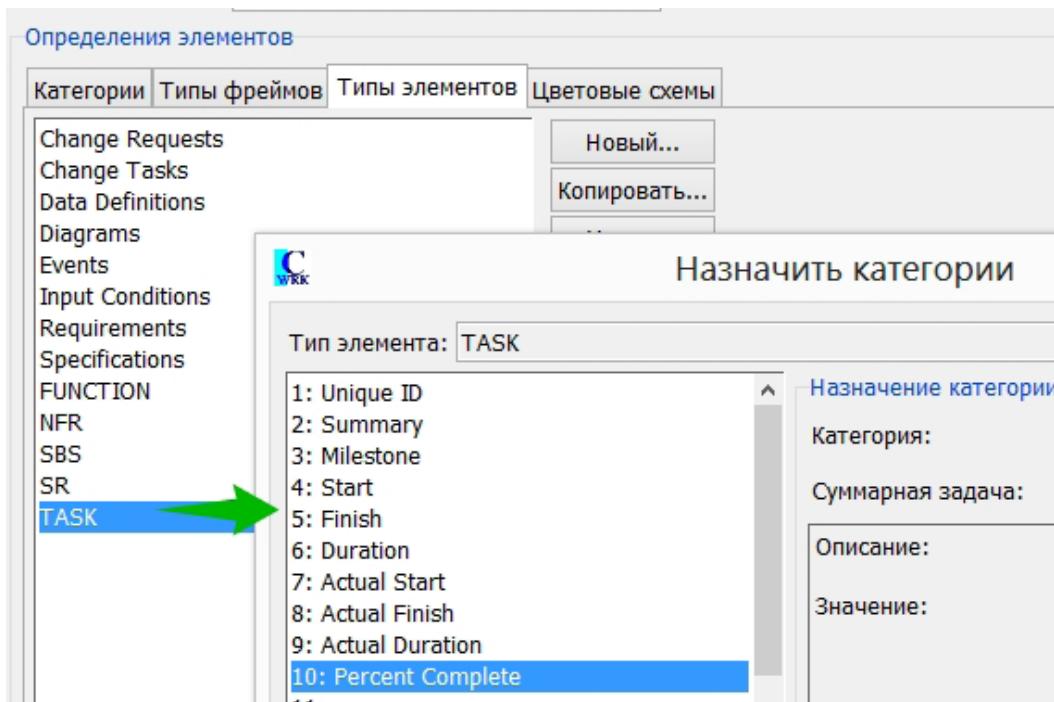
Категория	Тип
Actual Duration	Положительное
Actual Finish	Дата
Actual Start	Дата
Duration	Положительное
Finish	Дата
Milestone	Определяемый
Percent Complete	Положительное
Start	Дата
Summary	Определяемый
Unique ID	Целое
Ошибки	Множественное
Статус	Список выбор
Уровень NFR	Список выбор

5. Вернуться на вкладку **Типы элементов**

## 6. Назначить созданные категории стипу элементов TASK

Для этого выбрать TASK в списке элементов и нажать кнопку **Категории**. Выбирая свободную строку назначать категории из выпадающего списка.

После этого мы сможем использовать тип TASK для создания новых задач с указанием их временных характеристик.



Хорошая практика - если вы назначаете одну и ту же категорию нескольким типам элементов, то используйте позицию под одинаковым номером.

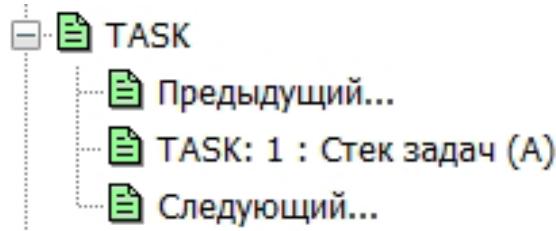
## Подготовить очередь задач

### 7. Перейти на панель База данных

### 8. Выделить тип TASK



9. В контекстном меню выбрать **Новый->Элемент** и создать элемент верхнего уровня для стека задач.



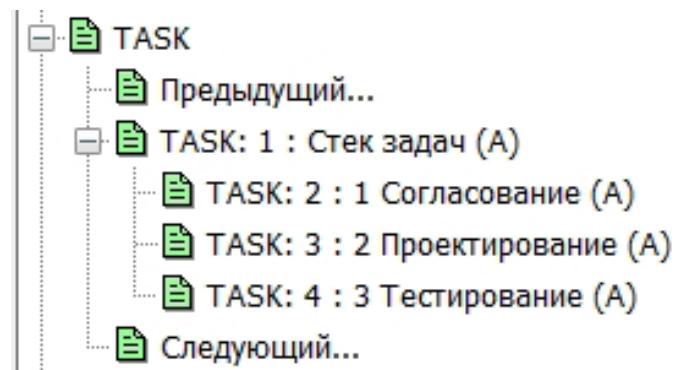
В рамках стека вы можете ввести разделы, которые вам будут удобны для сортировки.

Для примера введем разделы - Согласование, Проектирование, Тестирование. Для этого создадим связанные элементы как дочерние. Это поможет нам потом легко выполнять их сортировку и перенос в план работ. Для этого

10. Выделить элемент «Стек задач» и нажать кнопку **Создать дочерний**



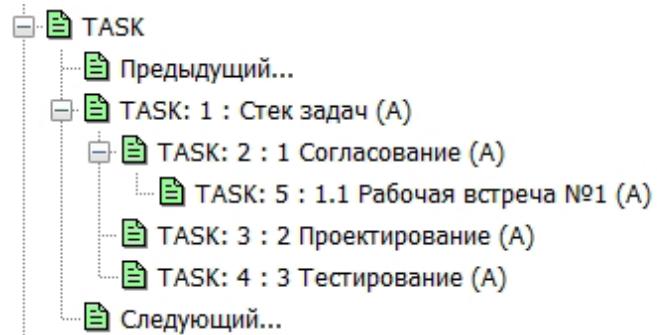
В результате создания трех дочерних элементов должно получиться:



Теперь можно создавать конкретные задачи и связывать с ними другие проектные данные.

По ходу проекта ближайшая задача, которую нам необходимо выполнить - это провести рабочую встречу и получить ответы на подготовленные вопросы. Создадим эту задачу как дочернюю в подразделе Согласование и свяжем с ней сформированные ранее вопросы.

11. Создать дочернюю задачу «Рабочая встреча №1»



12. Выбрать все вопросы, со статусом согласование, составив запрос:

ID	Связи	Категории	Вычисления	Фреймы	Даты	PUIDs	Д
<input checked="" type="checkbox"/> Статус	<input type="text" value="^Согласование\$"/>						
<input type="checkbox"/> Ошибки	<input type="text" value="Содержит"/>						

Для создания связей между всеми вопросами и задачей

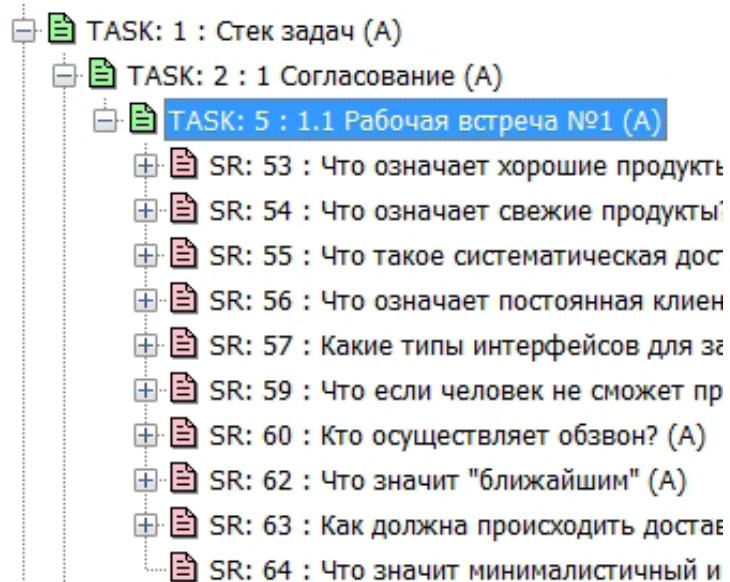
13. Выделить все выбранные вопросы

14. Нажать CTRL+>

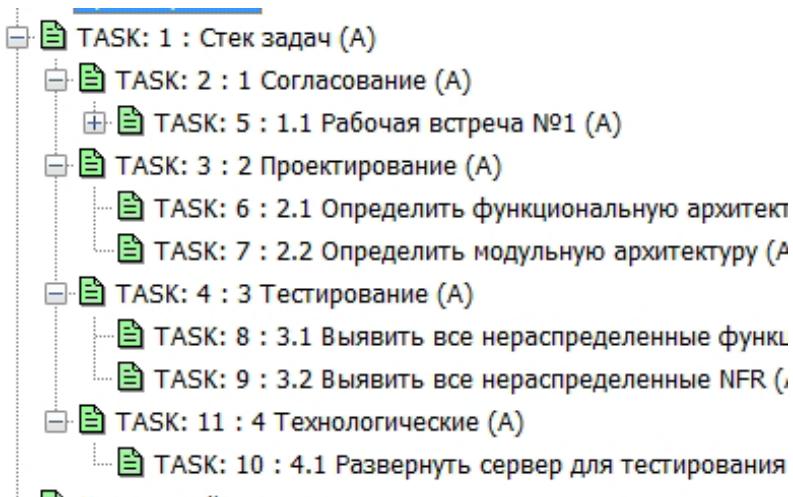
15. Выделить задачу рабочей встречи

16. Нажать CTRL+<

В итоге получим



Продумаем остальные задачи и разместим их в соответствующих разделах. На данном этапе их еще не может быть много, т.к. пока мы только знакомимся с постановкой задачи.

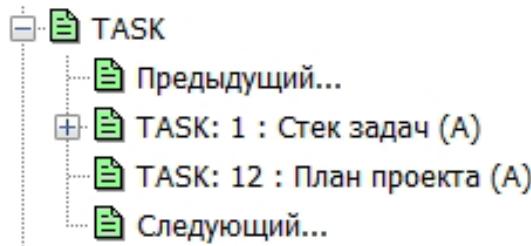


Если нам не нужен контроль сроков по задачам, а необходим просто список дел, которые необходимо выполнить, то такого решения может вполне хватить.

Если же этот проект имеет важное значение для компании и требуется контролировать исполнение сроков, то необходимо подготовить план работ, в котором будет определена очередность задач, требования к срокам выполнения, а также назначены конкретные исполнители.

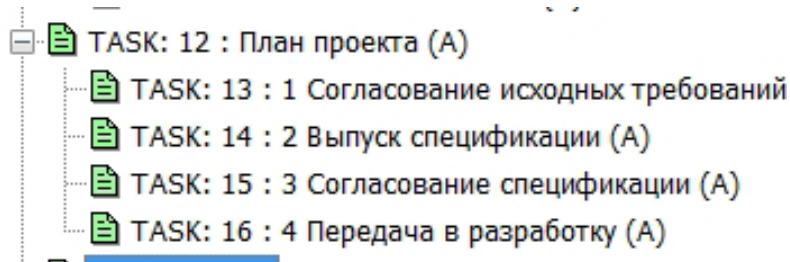
## Подготовить план работ

17. Создать отдельный элемент верхнего уровня, который будет ассоциирован с планом работ.



Далее план можно структурировать любым, удобным для вас образом, возможно по релизам или как-то иначе. Разобьем его на те этапы, которые нам очевидны на данный момент. Для этого:

18. Создать подзадачи у элемента, соответствующего плану работ

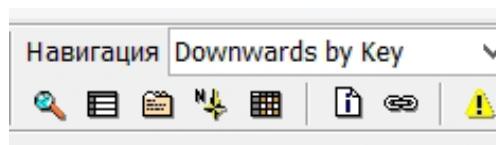


Заметьте, что если в Стеке задач, задачи сгруппированы по виду деятельности, то в рамках плана работ задачи в данном случае группируются по этапам, которые отражают последовательность выполнения работ.

После этого возьмем задачи из стека и разместим в соответствующем этапе плана работ. Для этого воспользуемся интересной функцией Cradle для переупорядочивания иерархических элементов.

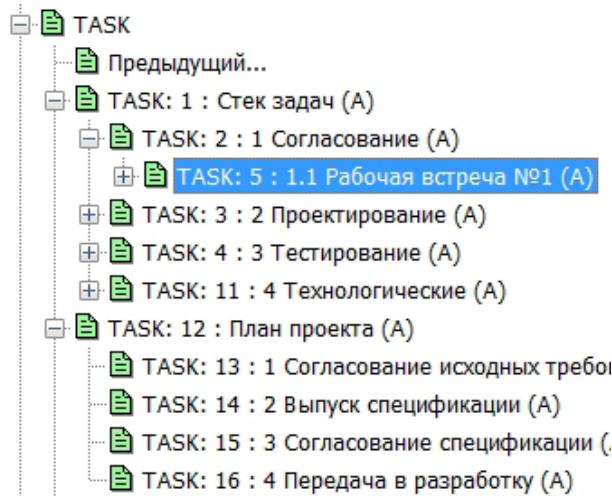
19. Сделать активной панель База данных

20. Включить режим навигации Downwards by Key



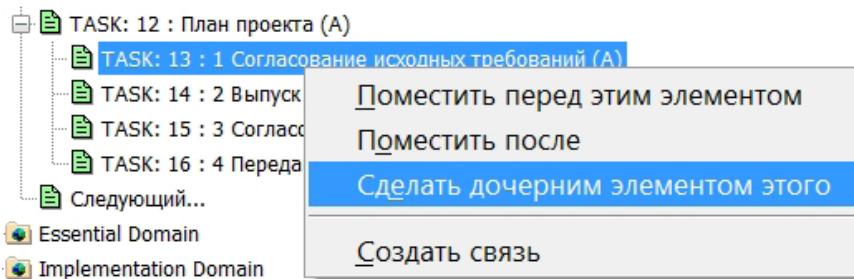
Обратите внимание, что для каждой рабочей панели может быть установлен индивидуальный режим навигации.

21. Раскрыть дерево задач так, чтобы была доступна задача из стека «Рабочая встреча №1» и этап плана работ «Согласование исходных требований»



22. Выделить элемент «Рабочая встреча №1» и перетащить его на элемент «Согласование исходных требований»

В режиме навигации *Downwards by key* это перетаскивание будет работать несколько иначе, чем в обычном режиме. Появится меню, которое позволит выбрать - переупорядочить элементы или, как обычно, создать дополнительную связь.



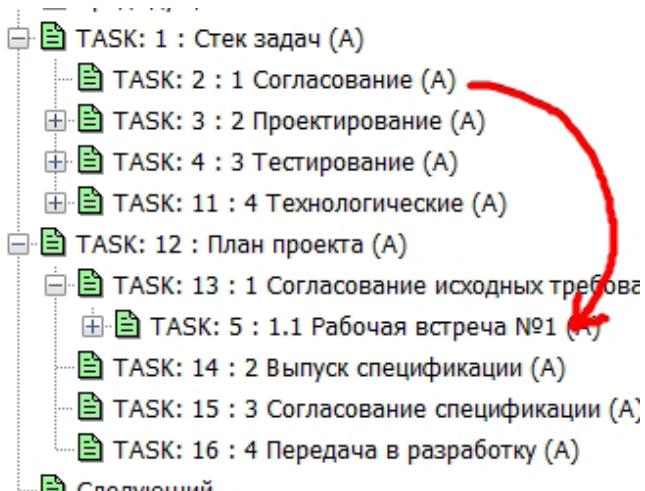
Последняя строка этого меню - **Создать связь** аналогична обычному режиму создания связи. Три остальных пункта - позволяют не только создать связь между заданными элементами, но и

- автоматически перестроить иерархические ключи элементов (переупорядочить элементы)
- удалить старую исходную связь элемента с разделом, разместив его в новом разделе.

Попробуйте и посмотрите, что получится, выбрать пункт меню **Сделать**

## дочерним элементом этого

В итоге у вас должна получиться следующая картина:

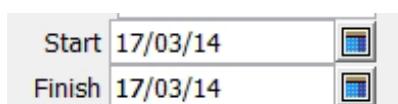


Задача «Рабочая встреча №1» больше не будет находиться в стеке задач, а разместиться в плане работ.

Если задача помещена план работ, т.е. распланирована, это означает, что мы уже можем наметить сроки ее выполнения - дату начала работ и дату окончания. Для этого

23. Открыть задачу «Рабочая встреча №1»

24. Назначить плановые дату начала и окончания задачи

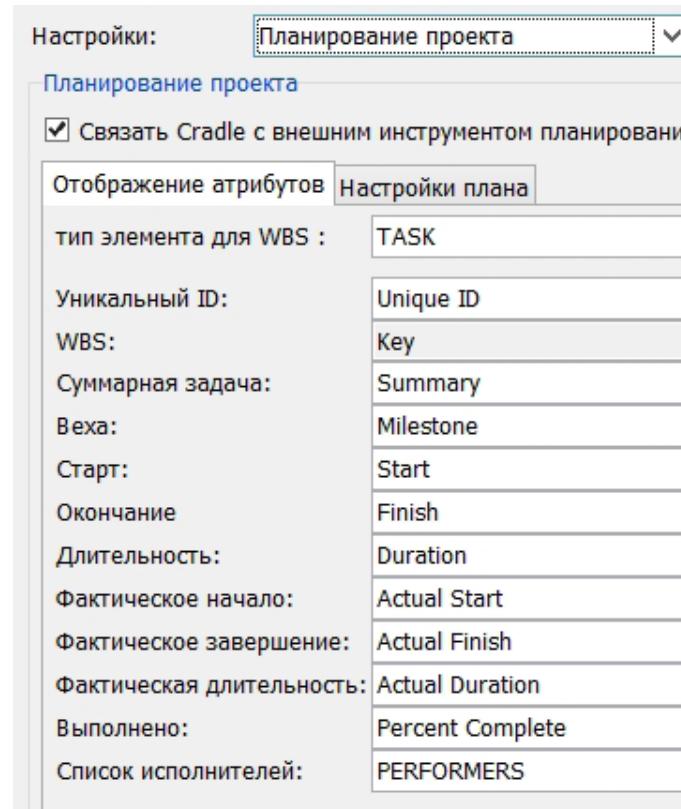


25. Сохранить и закрыть измененную задачу

Обратите внимание, что все наши манипуляции с задачей по перемещению ее из стека в план не повлияли на ее связи с требованиями, подлежащими согласованию и мы можем получить к ним доступ из иерархии плана работ.

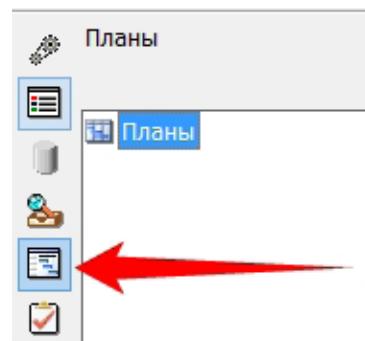
Для того, чтобы воспользоваться функциональностью по назначению ресурсов и панели персональных задач нам необходимо сделать несколько дополнительных настроек, указав Cradle, что считать планом работ. Для этого:

26. В меню **Установки проекта** в разделе **Планирование проекта** установить следующие настройки:

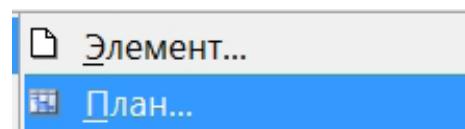


Как Вы видите, вы можете использовать любой тип элемента для работы с функциями планирования, достаточно прописать соответствие полей, необходимых для работы с планом.

## 27. Перейти на панель **Планы** и выделить элемент **Планы**

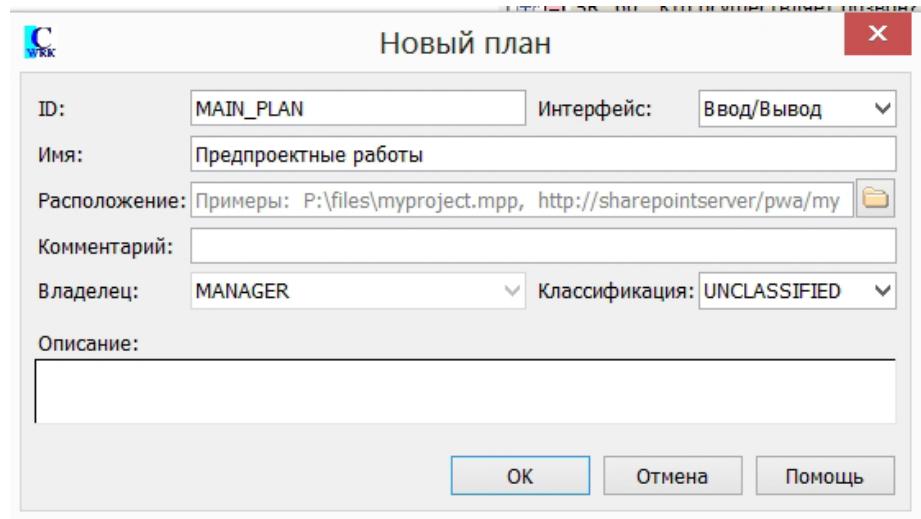


## 28. В контекстном меню выбрать **Новый ->План**



## 29. Ввести идентификатор и имя плана

Важно: идентификатор плана записывается в поле *Комментарий* для всех задач плана и это то поле, по которому определяется принадлежность задач к плану работ.

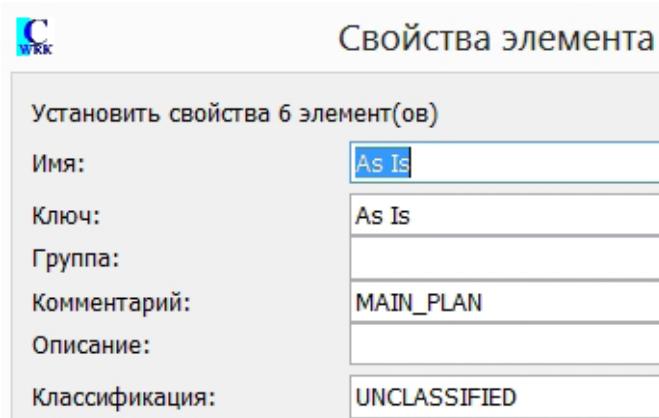


В связи с тем, что элементы плана работ мы создали до создания самого плана, чтобы подсоединить их в план необходимо:

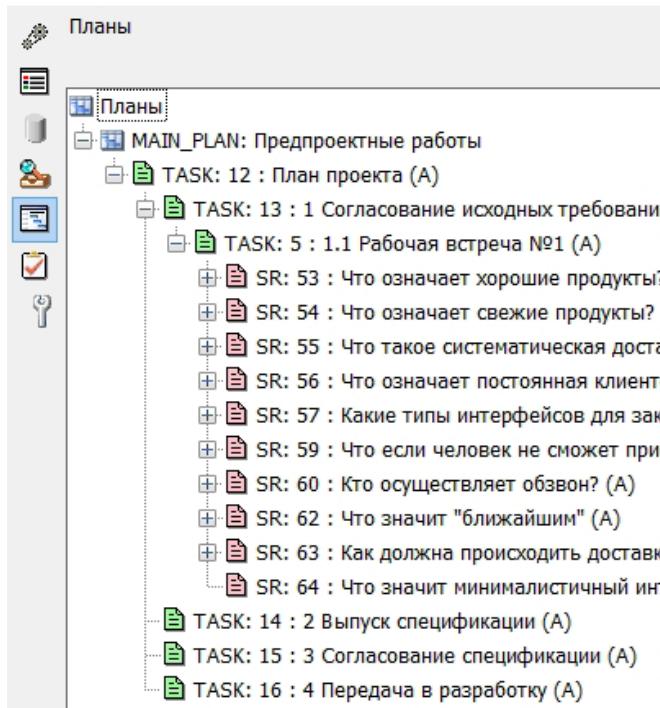
### 30. Выделить все задачи плана работ

### 31. В контекстном меню выбрать **Свойства**

### 32. В свойствах выделенных элементов установить поле *Комментарий* в значение идентификатора плана, в данном случае **MAIN\_PLAN**



Теперь вы можете вернуться на панель Планы и раскрыть план «Предпроектные работы». Вы увидите, что все задачи, подсоединились к плану работ.



Теперь, если вы будете создавать дочерние элементы от элементов плана, то комментарий будет устанавливаться автоматически. Однако, при перетаскивании задач из стека, это поле необходимо устанавливать вручную, подтверждая тем самым, что вы включили задачу в план работ.

После выполненных настроек мы можем воспользоваться расширенной функциональностью по управлению задачами:

- назначить пользователей Cradle на задачи в качестве исполнителя.
- использовать панель задач для работы с персональными задачами
- использовать интеграцию с MS Project для комплексного управления планом работ и ресурсами проекта, расчета длительности проекта.

Жизненный цикл проектной базы требований может оказаться значительно дольше, чем жизненный цикл одного проекта, поэтому один проект Cradle может иметь множество планов работ. Это могут быть подпроекты, ограниченные во времени и выполняемые как параллельно, так и последовательно.

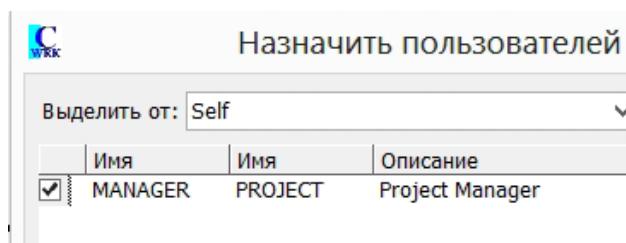
## Назначить исполнителей

Чтобы назначить исполнителей на задачу необходимо

33. Выбрать задачу

34. В контекстном меню выбрать **Назначить пользователей**

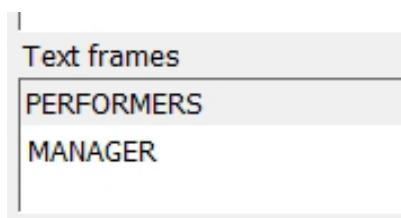
35. В диалоге **Назначить пользователей** выбрать тех пользователей, которые должны выполнять задачу



В данном проекте у нас всего один пользователь, назначим его на выбранную задачу.

36. Нажать **OK**

Теперь в поле PERFORMERS выбранной задачи появилась запись о пользователе Cradle. Мы можем также сразу же использовать это поле для назначения исполнителей, соблюдая правило - каждому исполнителю - отдельная строка.

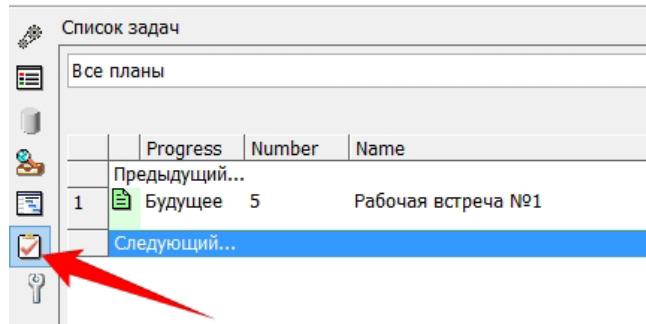


Хорошая практика - назначать исполнителей только на задачи нижнего уровня, а не на суммарные или вехи.

## Использовать панель задач для отслеживания

После того как исполнители (пользователи данного проекта Cradle) назначены на задачи, они могут воспользоваться панелью Список задач для того, чтобы отслеживать состояние назначенных на них задач. Для этого:

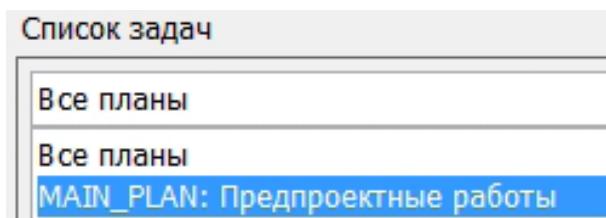
### 37. Перейти на панель Список задач



На данной панели будут отображены все задачи, которые назначены на текущего пользователя. Прямо с этой панели можно открыть свои задачи, чтобы посмотреть детали или опубликовать из в RTF/Html/CSV формате. Для этого используйте следующие три кнопки, расположенные в правом углу панели Список задач:



Верхний выпадающий список позволяет выбрать план работ, из которого необходимо выводить задачи:



### Отчитываться по задачам в Workbench

Если вы хотите вводить фактические данные по задачам, такие как:

- трудозатраты
- даты начала и окончания
- процент выполненных работ

то для этого необходимо открыть конкретную задачу и ввести эти параметры

Categories

Unique ID	
Summary	
Milestone	
Start	17/03/14
Finish	17/03/14
Duration	
Actual Start	
Actual Finish	
Actual Duration	
Percent Complete	
Приоритет	Высокий
 Discussions	

или воспользоваться готовым представлением - TASK-Schedule, в котором выводятся все необходимые поля:

Файл Редактировать Представление Элемент Запрос Инструменты Администрирование Окно Домошь

Представление TASK - Schedule

Requirement FUNCTION NFR SBS SR TASK Event

Запрос: без заголовка

	Identity	Ver	Dft	Progress	Comment	Name	Unique ID	Summary	Milestone	Start	Finish	Duration	Actual Start	Actual Finish	Actual Duration	Percent
1	5	A	В процессе MAIN_PLAN	Рабочая встреча №1						17/03/14	17/03/14		14/03/14		5	

Как и любые другие представления, вы можете отредактировать данное, чтобы оно отражало только те поля, которые вам необходимы, например так:

Запрос: без заголовка

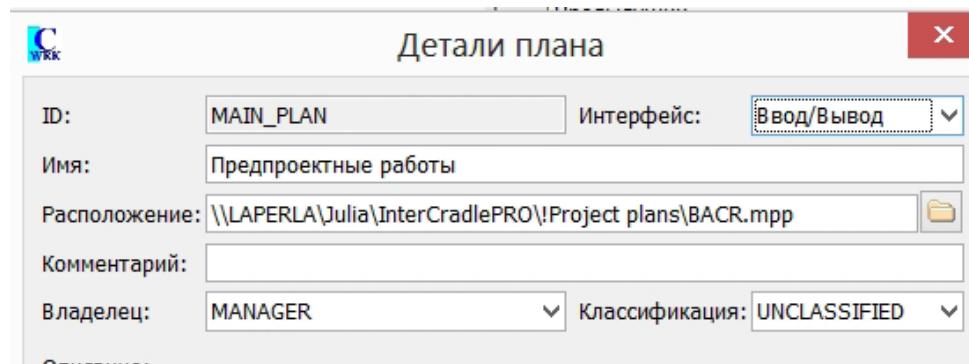
	Identity	Progress	Name	Actual	Actual	Percent Complete
1	5	В процессе	Рабочая встреча №1	14/03/14		5

## Выгрузить задачи в MS Project

Если в вашем проекте планируется довольно сложное расписание - множество задач с зависимостями друг от друга, то очень удобно будет воспользоваться функциями MS Project. Для этого

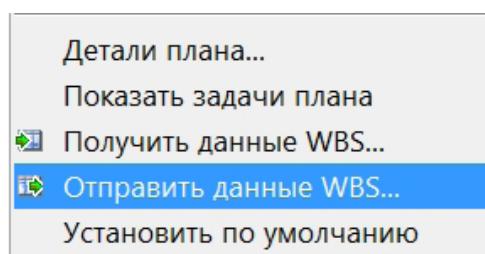
38. Создать план работ MS Project и сохранить в доступном для всех пользователей разделе сетевого диска или на сервере MS Project

39. В деталях настроенного ранее плана работ указать путь к файлу или проекту на сервере.

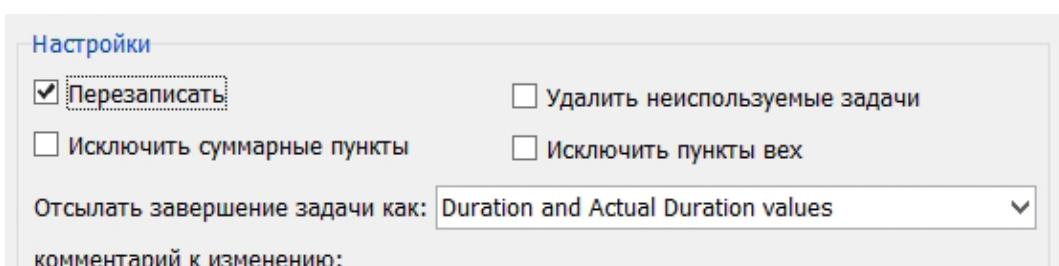


40. Перейти на панель Планы

41. Выделить План MAIN\_PLAN и в контекстном меню выбрать **Отправить данные WBS**



42. Установить следующие настройки:



#### 43. Нажать кнопку **Отправить**

**Отправить**

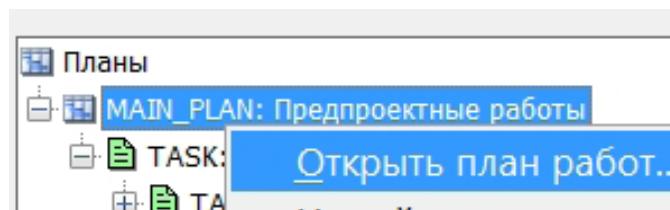
Чтобы функция отправки работала без ошибок, необходимо, чтобы суммарная задача плана работ имела ключ верхнего уровня, а остальные элементы - иерархические ключи следующих уровней.

Также важно знать, что поле Duration хранится в MS Project храниться в минутах, соответственно, если вы хотите пользоваться интеграцией с MS Project, вводя данные о длительности в Cradle, а не в MS Project, то длительность необходимо вводить в минутах.

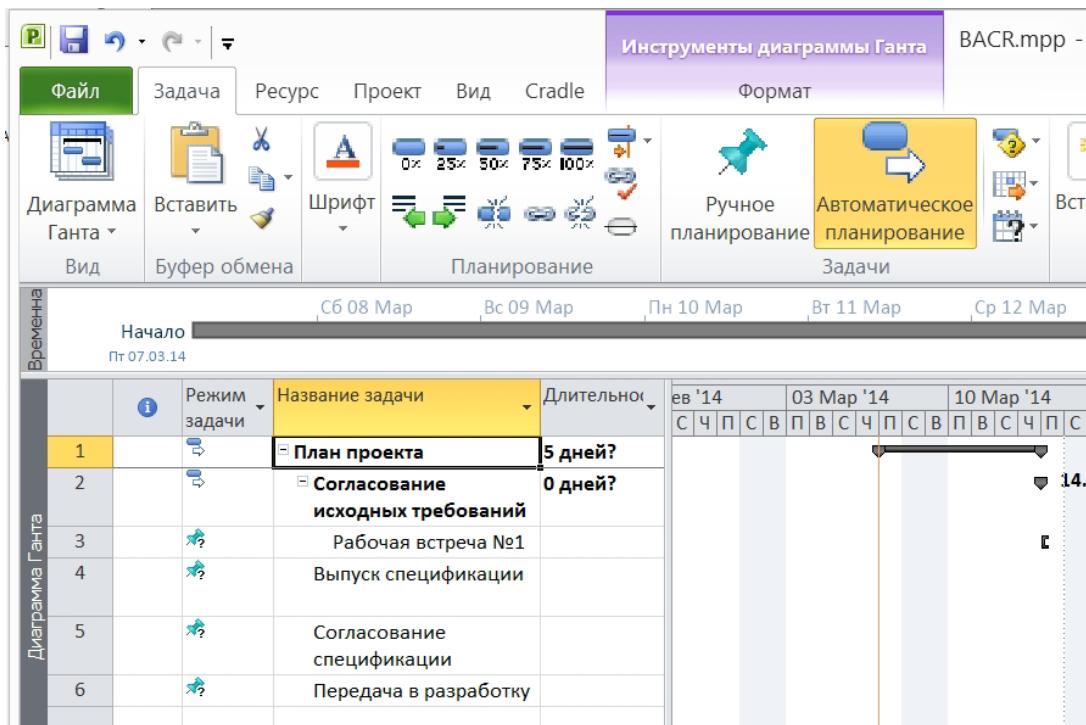
Чтобы проще ориентироваться, вы можете создать в проекте Cradle поле-калькуляцию, которая будет рассчитывать значение длительности в днях, основываясь на значении поля длительность в минутах.

#### **Провести планирование в MS Project**

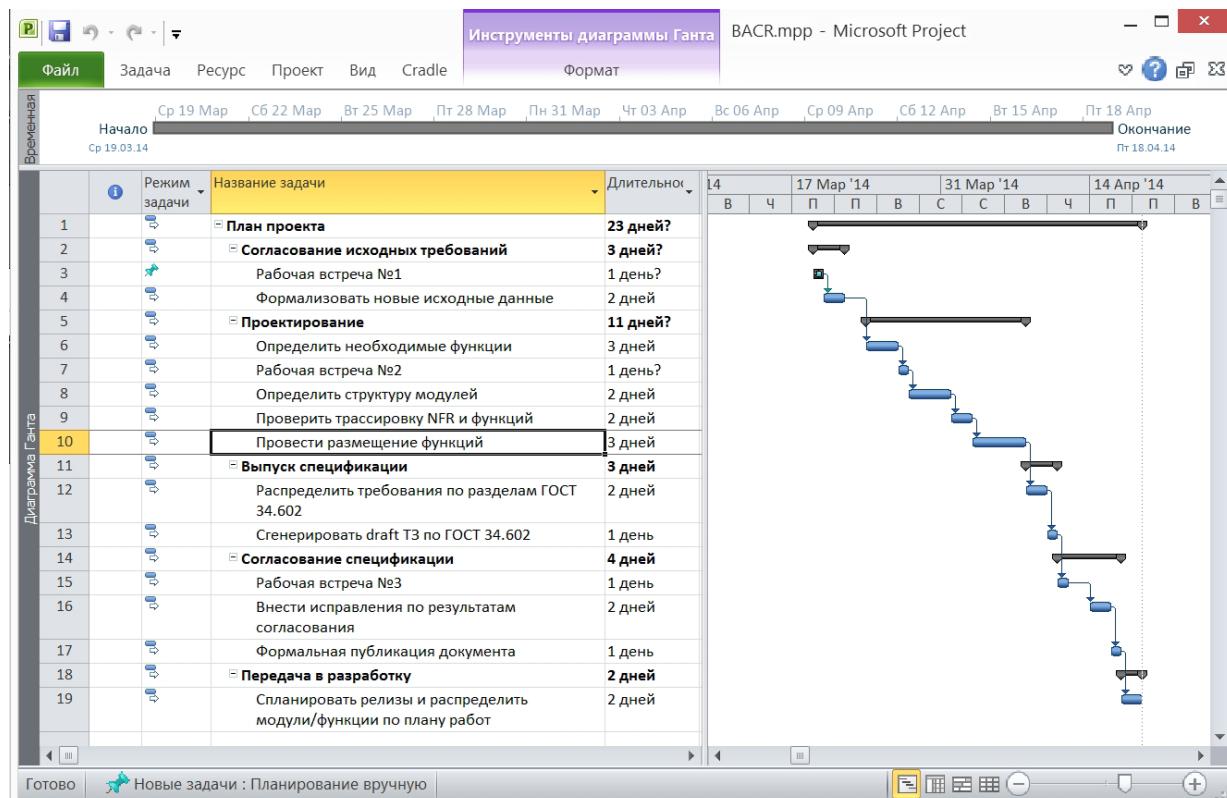
После того, как синхронизация прошла, вы можете открыть план работ, выбрав в контекстном меню плана пункт **Открыть план работ**



Откроется MS Project и соответствующий файл проекта и вы сможете уточнить план работ, пропавив временные зависимости между задачами.

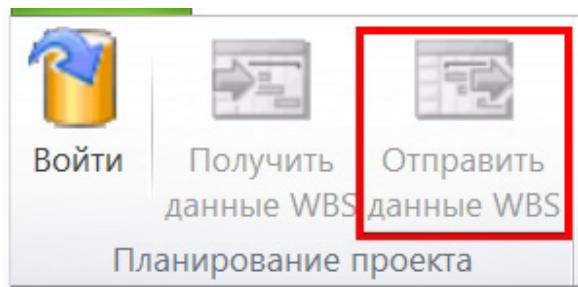


После уточнения структуры задач и связей мы получили следующий план работ, в котором с помощью стандартных функций MS Project проведен расчет длительности проекта.

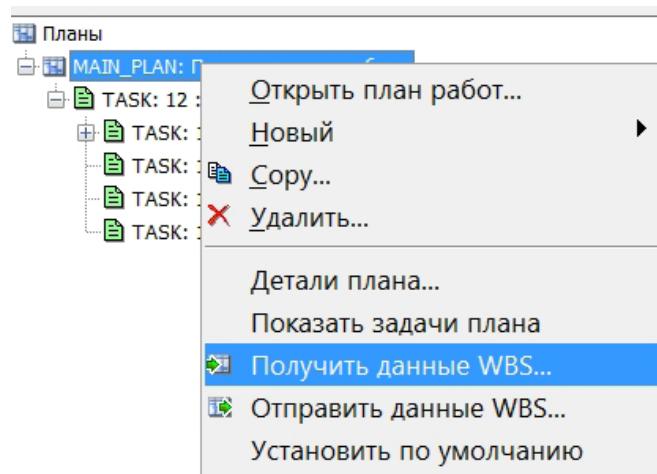


## Обновить план в Workbench

Если план работ был изменен не в Cradle, а в MS Project, то мы можем провести синхронизацию в обратную сторону и обновить данные по задачам в проекте Cradle. Эту операцию можно запустить как в MS Project из панели Cradle:

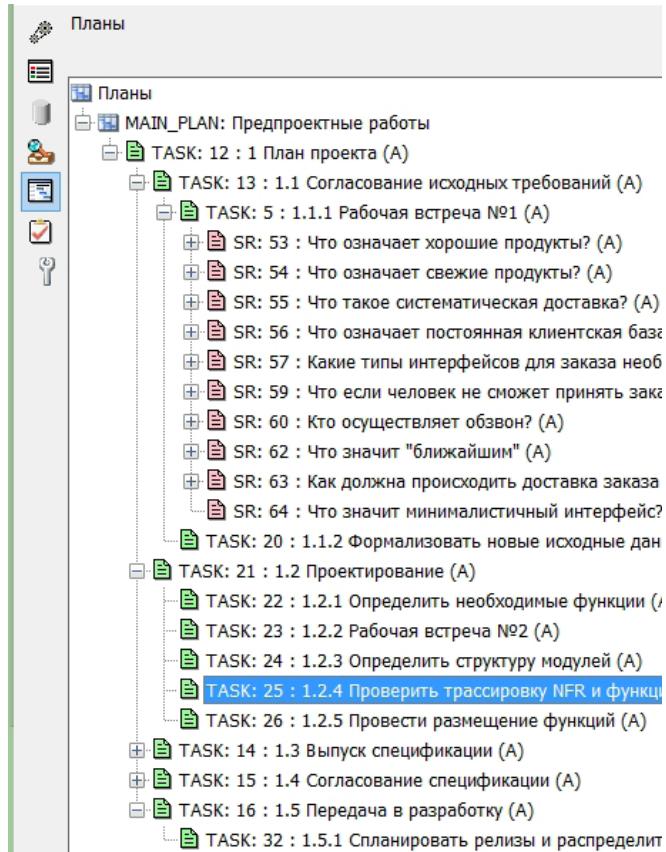


Так и из Cradle, выбрав пункт меню **Получить данные WBS**:



Первый вариант более удобен для руководителей проектов, которые не занимаются непосредственно проектированием и больше работают в MS Project.

После обновления данных задач в Cradle мы получим следующую картину:



Как вы видите, все новые задачи, которые мы создали в MS Project теперь появились и в Cradle, при этом все связи требований с задачами сохранились. Также были обновлены и временные параметры задач.

## Отчитываться по задачам

Отчитываться по задачам возможно как через Cradle, так и через MS Project. Первый вариант подходит для ситуации, когда исполнители - аналитики, проектировщики сами проставляют процент выполнения задачи прямо в Cradle. Эта информация сохраняется в целом для задачи и затем отправляется в MS Project.

Второй вариант подходит для ситуаций, когда учет выполнения ведет непосредственно руководитель проекта, соответственно он и выставляет его прямо в MS Project, отправляя затем в Cradle.

Для крупных компаний удобен также вариант (при условии внедрения сервера MS Project), когда исполнители отчитываются с использованием веб-интерфейса сервера MS Project (Web access). При этом у исполнителей появляется возможность вести отчетность не в целом для задачи, а отчитываться по зада-

чам за каждый день.

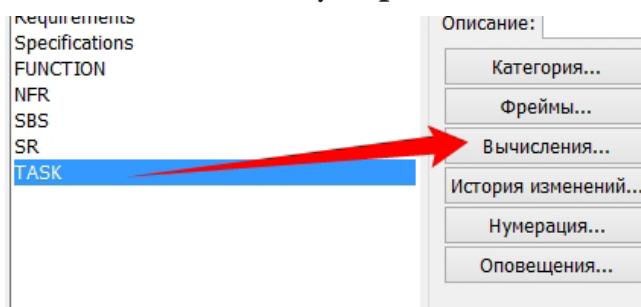
Выбор способа отчетности должен происходить с учетом многих корпоративных факторов и если вам необходима помощь в этой области, то вы можете обратиться в САТУРС за профессиональной консультацией.

## Настроить вычисляемое поле для определения длительности

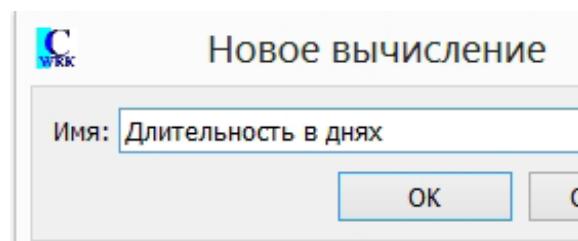
В качестве примера настройки вычисляемых полей рассмотрим настройку расчета длительности задачи в рабочих днях на основании длительности в минутах.

Как уже было сказано выше, если вы хотите использовать интеграцию с MS Project, то в связи с тем, что в этой системе учет длительности ведется в минутах, в Cradle соответствующее поле также должно заполняться в минутах, что несколько необычно. Чтобы помочь себе выбирать правильную длительность в минутах настроим вычисляемое поле, которое будет переводить минуты в рабочие дни. Для этого:

1. Открыть меню **Установки проекта**, раздел **Определения элементов**, вкладка **Типы элементов**
2. Выбрать тип **TASK** и нажать кнопку справа **Вычисления**

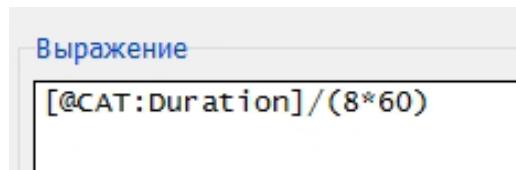


3. В диалоге **Настройки вычисления** нажать кнопку **Новый** и дать имя новому вычисляемому полю - **Длительность в днях**



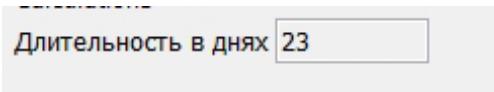
4. Формируя Выражение выбрать категорию Duration
5. Разделить полученное выражение на  $8*60$  - количество минут в рабочем дне

В итоге должно получится следующее выражение:



## 6. Сохранить настройки

Теперь открыв любую задачу проекта в форме вы увидите, сколько ее длительность составляет в днях. Вы также можете вывести это поле в любое представление



Если у вас довольно масштабный проект, то вы можете добавить аналогичные вычисляемые поля, которые будут отражать длительность задач в неделях или месяцах.

## Добавить метрики и показатели задач в KPI

Чтобы контролировать ход проекта прямо в Cradle мы можем создать метрики и Панель индикаторов для отражения показателей, связанных с планом работ.

Нам могут быть интересны, например, следующие показатели:

- Процент невыполненных задач по количеству
- Процент невыполненных задач по длительности.

Выполним это на примере первого показателя. Для этого, дополнительно к тому, что уже настроено в системе нам понадобятся запрос, который выберет все невыполненные задачи нижнего уровня и метрика, которая посчитает их количество, а также общее количество задач нижнего уровня.

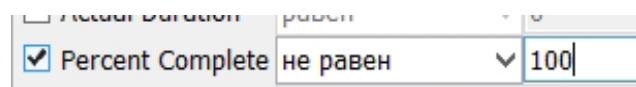
Для настройки вышеперечисленных элементов:

## 1. Создать запрос к типу TASK

2. Указать уровень Bottom-level



3. На вкладке запроса Категории указать, что приоритет не равен 100, что означает - задача еще не выполнена.



4. Сохранить запрос под именем «TASK - не завершенные»

5. Создать новую метрику «Ход проекта» и добавить в нее два запроса:

- только что созданный «TASK - не завершенные» и
- автоматический «TASK-Bottom-Level».

6. Для обоих запросов выбрать операцию Count

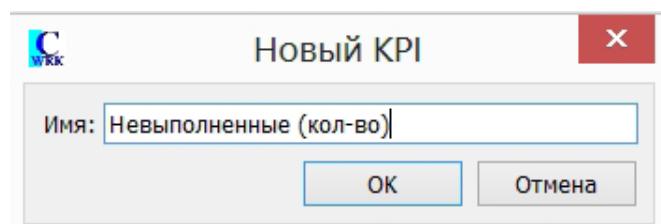
Определение		
Ном	Запрос	Операция
1	TASK - не завершенные	Count
2	TASK - Bottom-Level	Count

Запустить и проверить метрику. Если вы выполняли проект одновременно с ходом учебного пособия, то на данный момент оба значения у вас получатся одинаковыми, т.к. еще нет выполненных задач.

Далее настроим панель индикаторов.

7. Создать новую панель индикаторов «Ход проекта»

8. Создать на панели новый индикатор



9. Настроить индикатор следующим образом:

Детали KPI

Имя: Невыполненные (кол-во)

Источник: Процент

Первая метрика

Метрика: Ход проекта  
Элемент: 1: TASK - не завершено  
Представление: TASK - Schedule

Вторая метрика

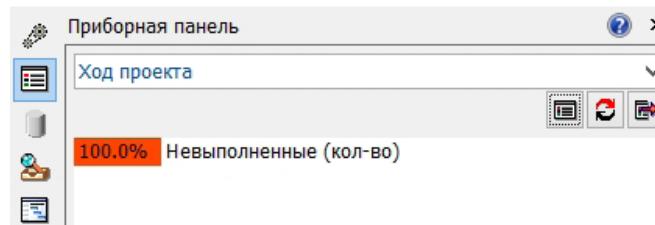
Метрика: Ход проекта  
Элемент: 2: TASK - Bottom-L  
Представление: TASK - Schedule

Настройки привязки

	Нижний	Верхний	Цвет
1	No limit	< 10	Green
2	$\geq 10$	$< 30$	Light Green
3	$\geq 30$	$< 60$	Yellow
4	$\geq 60$	$< 80$	Pink
5	$\geq 80$	$\leq 100$	Orange

## 10. Сохранить и применить настройки.

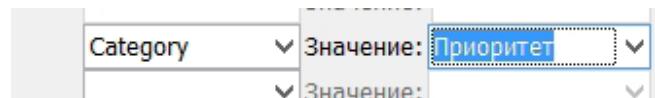
У вас должен получиться следующий результат:



По мере выполнения задач индикатор будет приобретать более радостные цвета.

Также, нам была бы интересна метрика, которая отражала бы процентное распределение задач по приоритету. Настроим ее, это довольно просто.

1. Вернуться в метрику «Ход проекта»
2. Скопировать строку «TASK - не завершенные», опустив ее вниз, соответствующей кнопкой.
3. В настройка Сгруппировано по выбрать Category и выбрать категорию приоритет:



#### 4. Сохранить и применить метрику.

TASK - не завершенные		18
	Высокий	1

Обратите внимание, что данная и предыдущие метрики вычислялись по всем задачам, присутствующем в проекте, как входящим в проект, так и нет.

Если необходимо построить метрику отдельную для запланированных задач и задач из стека, то необходимо уточнить запрос, выделив, например, необходимые задачи по значению поля Комментарий, который как вы помните, заполняется для задач из плана идентификатором плана работ.

# **Взаимодействие с Заказчиком**

Все требования проекта должны храниться в одном месте, это позволяет здорово сэкономить проектное время и силы всей команды, т.к. работа с хаосом всегда требует дополнительного времени и порождает ненужные конфликты.

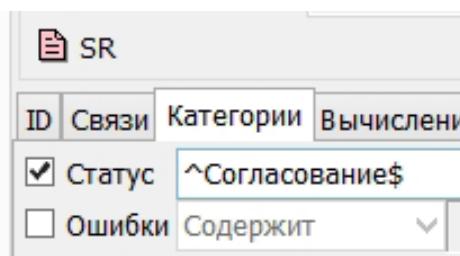
Ранее мы выделили вопросы, которые необходимо согласовать с Заказчиком. Ответы на эти вопросы, полученные на рабочей встрече необходимо также ввести в Cradle, т.к. они представляют новые исходные требования. К счастью Cradle позволяет автоматизировать процесс как подготовки совещания, так и обработки его результатов.

Рассмотрим два варианта генерации повестки - с помощью функций отчетов (не требуется лицензия на модуль DOC) и с помощью подготовки корпоративного шаблона (требуется лицензия на модуль DOC)

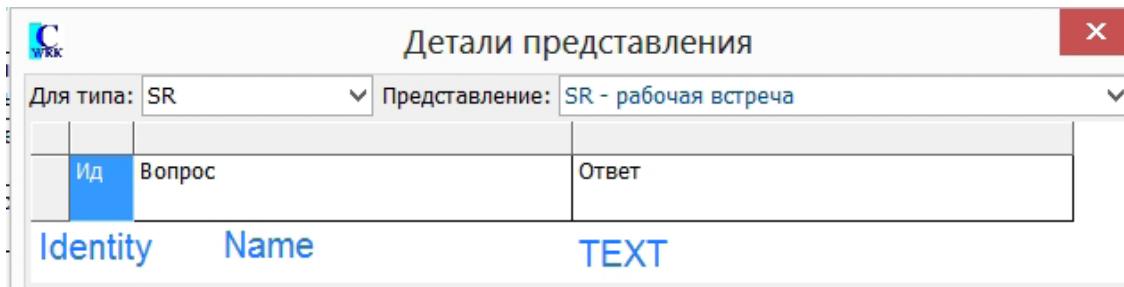
## **Настроить простой отчет для генерации повестки**

Первый, самый простой вариант, подготовить повестку встречи - построить запрос, который выведет все вопросы, подлежащие согласованию и представление, которое отразит необходимые поля. Для этого:

1. Создать запрос к типу SR
2. В разделе категории указать Статус = Согласование:



3. Сохранить запрос под именем «SR - согласование».
4. Создать представление для типа SR со следующими настройками:



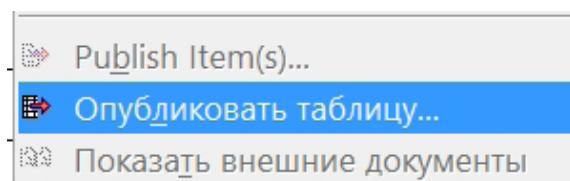
5. Сохранить представление под именем «SR-рабочая встреча»
6. Открыть созданный запрос «SR - согласование» и применить к нему представление «SR-рабочая встреча»

Поле TEXT (ответ) пока не заполнено. Это предстоит сделать в ходе рабочей встречи.

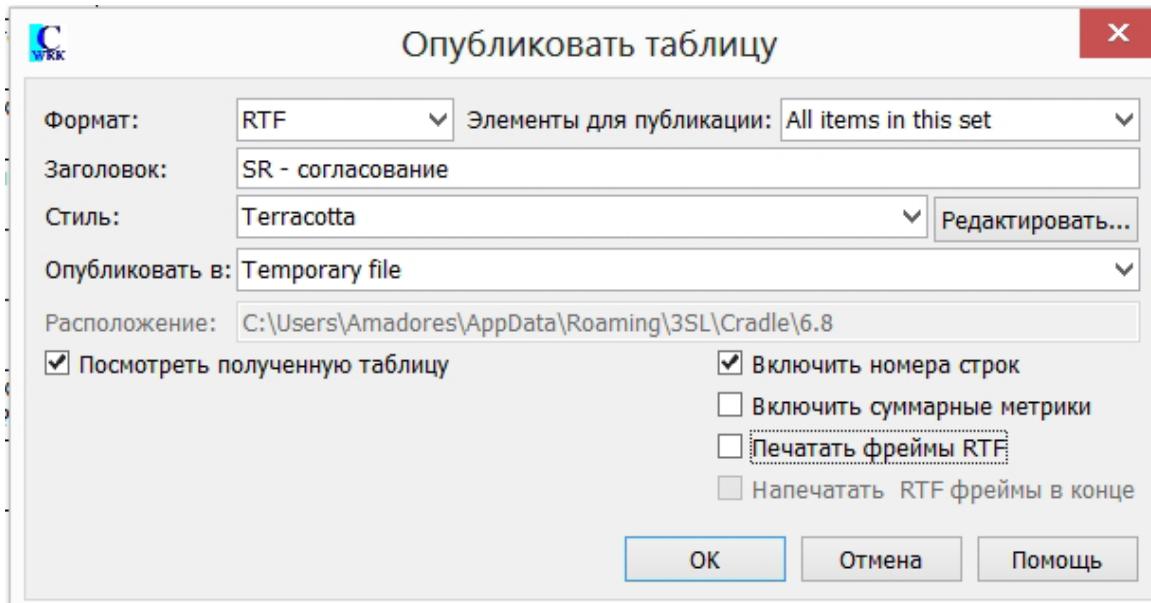
Запрос: SR - согласование			
SR - согласование			
	Ид	Вопрос	Ответ
Предыдущий...			
1	53	Что означает хорошие продукты?	
2	54	Что означает свежие продукты?	
3	55	Что такое систематическая доставка?	
4	56	Что означает постоянная клиентская база?	

Теперь можно опубликовать это представление в формате RTF, чтобы взять с собой на совещание. Для этого:

7. В контекстном меню выбрать **Опубликовать таблицу**



8. Выбрать следующие настройки



Обратите внимание, что вы можете публиковать не все элементы запроса, а выбрать лишь некоторые из них. Для этого в разделе Элементы для публикации выберите Selected items:

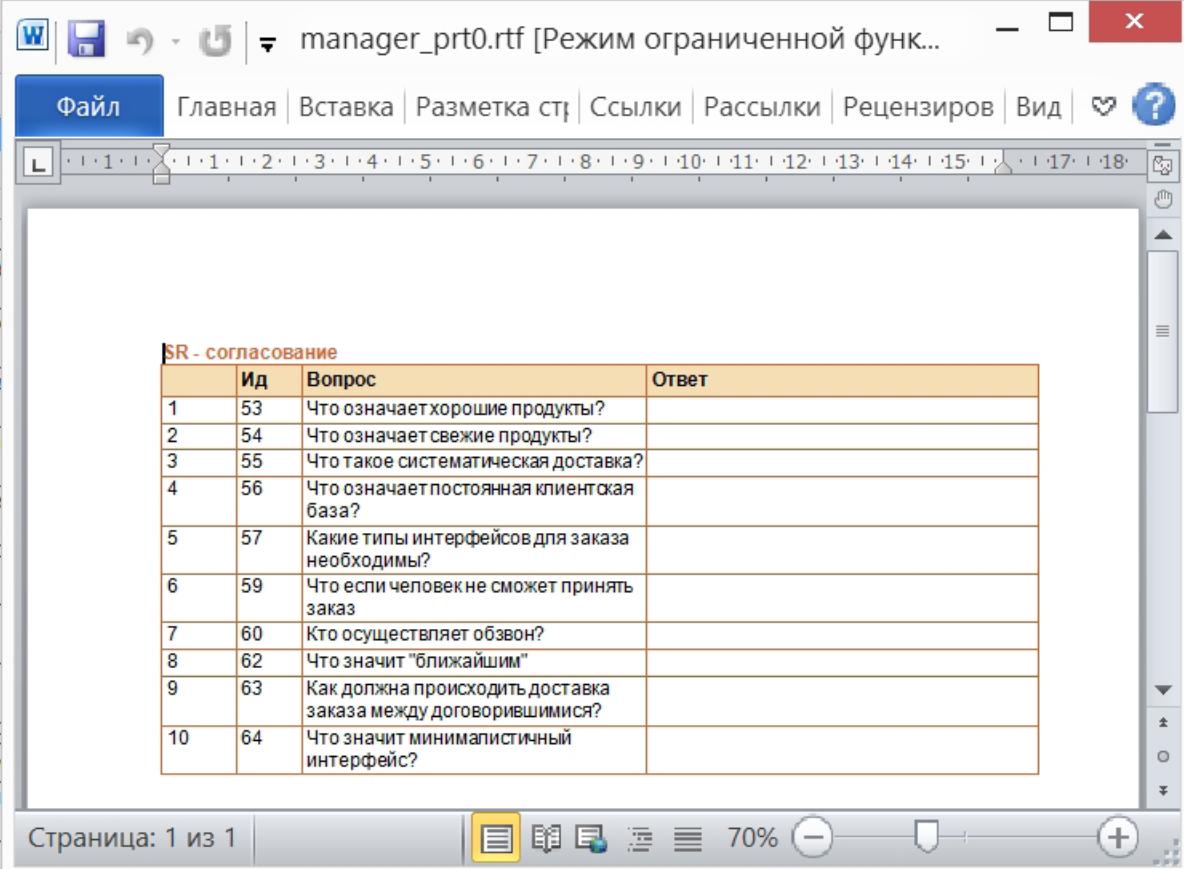


В разделе Стиль, кнопка Редактировать, вы можете сделать дополнительные настройки форматирования, если вам не подходят предустановленные стили.

#### 9. Нажать OK для публикации шаблона

В результате мы получим файл RTF, который лучше сразу сохранить в формате Word для последующей загрузки в Cradle результатов совещания.

Обратите внимание, что в отчете появилось дополнительное поле, отражающее порядковый номер вопроса. Это может понадобится, чтобы не оперировать идентификаторами элементов в ходе совещания. Однако сами идентификаторы должны присутствовать в этой таблице, т.к. это понадобится нам для обратной загрузки ответов.



manager\_prt0.rtf [Режим ограниченной функции]

Файл Главная Вставка Разметка стиля Ссылки Рассылки Рецензиров Вид ?

1 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

SR - согласование

	Ид	Вопрос	Ответ
1	53	Что означает хорошие продукты?	
2	54	Что означает свежие продукты?	
3	55	Что такое систематическая доставка?	
4	56	Что означает постоянная клиентская база?	
5	57	Какие типы интерфейсов для заказа необходимы?	
6	59	Что если человек не сможет принять заказ	
7	60	Кто осуществляет обзвон?	
8	62	Что значит "ближайшим"	
9	63	Как должна происходить доставка заказа между договорившимися?	
10	64	Что значит минималистичный интерфейс?	

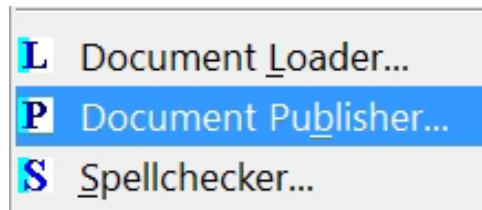
Страница: 1 из 1

Вы можете вручную добавить в этот простой отчет все остальные реквизиты повестки, такие как наименование рабочей встречи, участники, дата проведения и др. Если же такие совещания проводятся довольно часто, то мы рекомендуем настроить шаблон для генерации повестки с помощью Document Publisher.

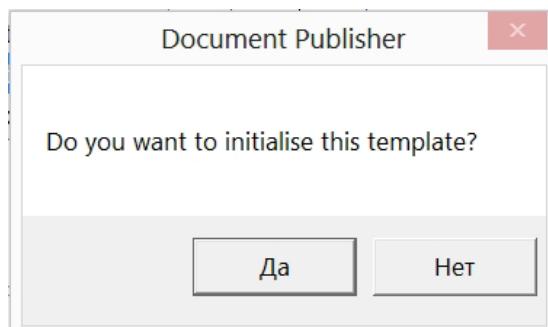
## Настроить корпоративный шаблон для генерации повестки

Ранее мы создали задачу «Рабочая встреча №1» и соединили с ней все вопросы, которые мы хотим согласовать в ходе этой рабочей встречи. Разрабатываем шаблон повестки встречи, который будет выводить в табличном виде все вопросы, связанные с данной задачей, а также другую атрибутику повестки. Для этого:

1. Открыть Document Publisher из меню Инструменты



2. Перейти на вкладку Define template
3. Нажать кнопку New и выбрать формат - doc или docx
4. Не проводить инициализацию шаблона, ответив НЕТ на вопрос:



После этого будет открыт MS Word в котором вы можете обычным образом добавить все атрибуты корпоративного шаблона повестки рабочей встречи - колонитулы, номера страниц и др. Если в вашей компании уже существует такой шаблон, просто скопируйте из него все необходимые разделы.

После того, как все стандартные, не изменяющиеся элементы внесены, можно разметить динамическое содержание повестки, которое будет формироваться на основании данных проекта, а именно:

- название рабочей встречи и дату проведения, которые мы возьмем из задачи, соответствующей рабочей встрече
- таблицу вопросов, связанных с этой задачей

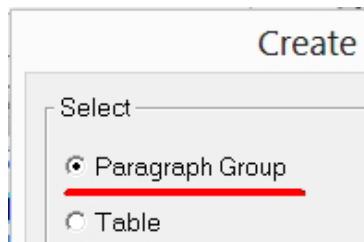
Для этого:

5. Установить курсор в месте, где должны быть выведены дата и время проведения встречи

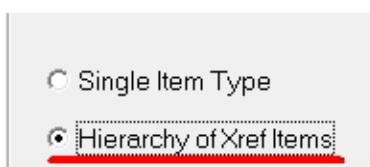
Дата проведения: ¶

Место проведения: ¶

6. На панели Document Publisher нажать кнопку Create и выбрать Paragraph Group



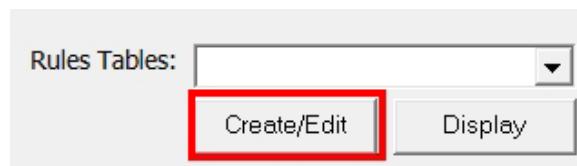
7. В открывшемся диалоге Item Selection выбрать режим Hierarchy of Xref Items



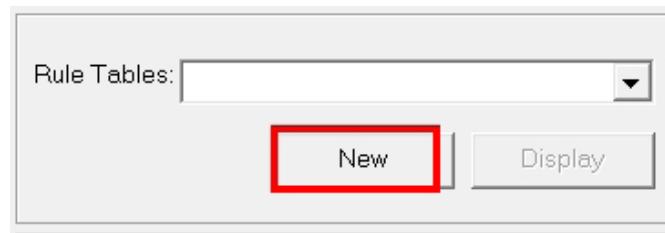
Это означает, что мы хотим работать не с линейным списком элементов одного типа, а с иерархией связанных элементов.

Теперь нам необходимо задать ту иерархию, с которой мы хотим работать. Для этого надо создать таблицу правил.

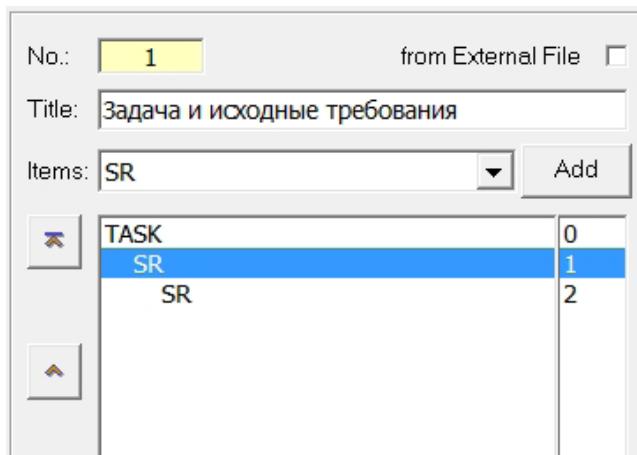
8. Нажать кнопку Create/Edit в разделе Rules Table



9. Нажать кнопку New



**10. Создать следующую иерархию:**



При этом настройки связей (справа) для каждого уровня должны быть следующими:

Для обоих переходов - TASK-SR и SR-SR



Не забывайте сохранять настройки переходов, нажимая кнопку



после каждого редактирования.

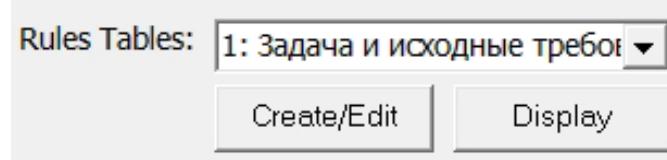
**11. После выполнения всех настроек таблицы правил сохранить результат,**

нажав:

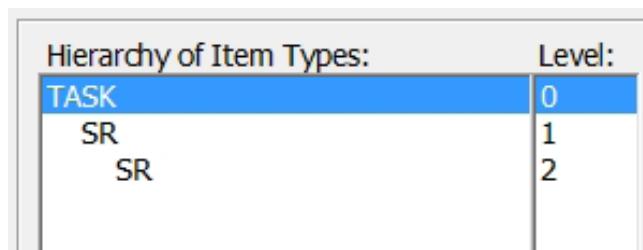


12. Вернуться в предыдущий диалог, нажав Return

13. В выпадающем списке Rules Table выбрать созданную таблицу и нажать Display



Созданная иерархия отразится на панели слева:



14. В поле Identifier ввести знак вопроса:



В этом поле указывается идентификатор элемента, начиная с которого необходимо выбирать иерархию для вывода. Это поле может принимать значение \*, что означает - брать все элементы, может быть конкретным идентификатором элемента, от которого строится иерархия или знаком вопроса, тогда мы будем иметь возможность указать идентификатор при публикации. Таким образом, наш шаблон будет параметрическим и мы сможем использовать его для публикации повесток всех рабочих встреч.

15. Сохранить настройки, нажав Save tag и Save template

16. Закрыть диалог

Выполнив эти настройки мы указали структуру, с которой мы будем работать далее. Таких структур в одном комплексном документе может быть довольно-

но много, но в рамках этой задачи нам будет достаточно только этой структуры.

Теперь нам необходимо указать конкретные поля, которые необходимо выводить для каждого уровня структуры в каждом конкретном случае. Для вывода даты проведения встречи из этой структуры нам понадобится только верхний уровень, а именно поле Start Date задачи, соответствующей встречи.

17. В шаблоне Word установить курсор внутрь созданного на предыдущем этапе тега Paragraph Group

Дата·проведения<1::ParagraphGroup>¶

Место·проведения::¶

18. На панели инструментов Document Publisher нажать кнопку Create

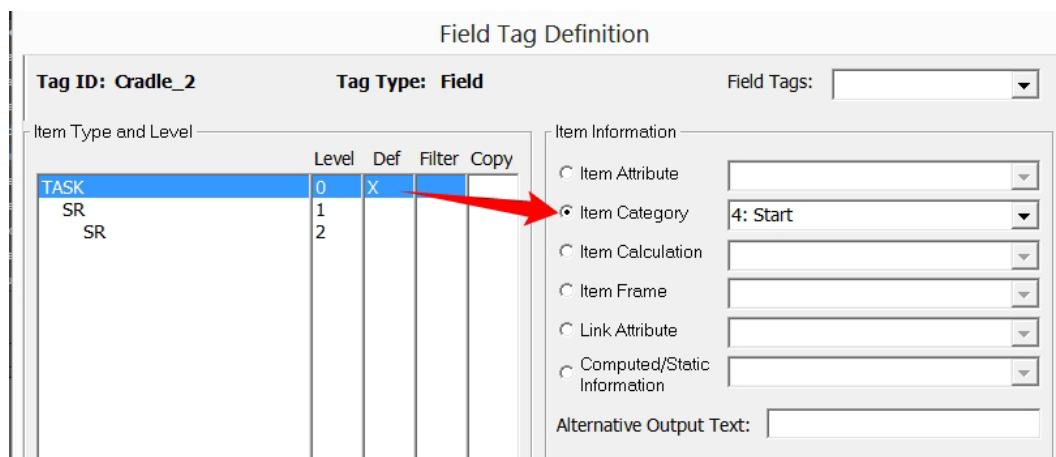
Теперь диалог, который будет нам предложен будет совершенно иной. В рамках тега Paragraph group мы можем вставить отображение только поля.



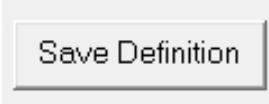
Это нам сейчас и необходимо.

19. В появившемся диалоге Field tag Definition выбрать первый уровень иерархии - TASK

20. Выбрать раздел Category и установить значение Start



21. Сохранить этот раздел, нажав Save Definition



Save Definition

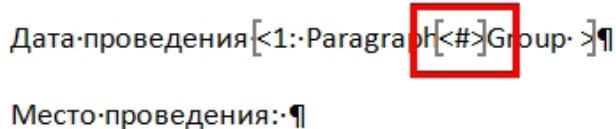
22. Сохранить тег, нажав внизу диалога Save Tag



Save Tag

23. Закрыть диалог, нажав Close

Теперь поле data будет выглядеть следующим образом:



Дата·проведения<1·:·Paragraph>[<#>Group·]<¶>  
Место·проведения:·¶

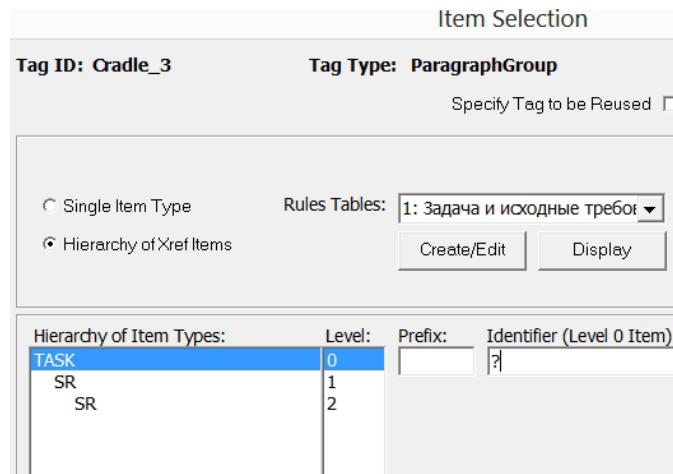
В основном теге Параграфа включен тег конкретного поля, который необходимо вывести для заданного уровня.

Обратите внимание, что к данным тегам можно применять любое форматирование Word.

Далее по аналогии выведем наименование встречи, взяв его из поля Name соответствующей задачи и отформатируем его как название документа, для этого:

24. Создать тег Параграфа в месте, где должно быть наименование повестки

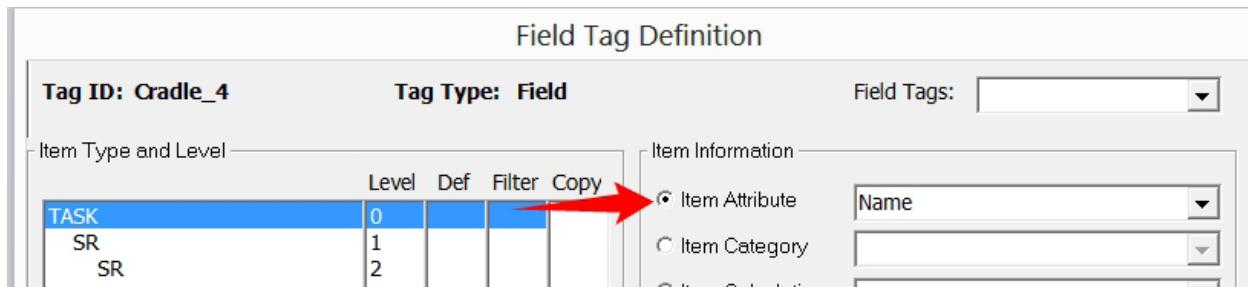
25. В настройка выбрать ту же таблицу правил и установить знак вопроса в раздел Identifier



26. Сохранить шаблон (Save Tag и Save Template) и закрыть этот диалог

27. Установить курсор внутрь созданного тега Параграфа и снова нажать Create на панели инструментов.

28. Создать тег Поля для которого установить следующие настройки:



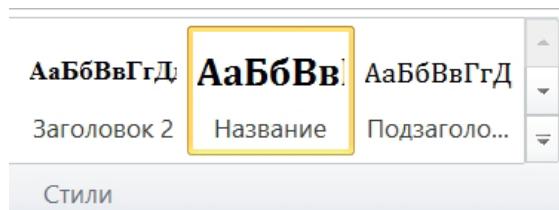
29. Сохранить тег нажав Save Definition, а затем Save Tag

30. Закрыть диалог настройки поля, нажав Close

31. В MS Word выделить созданный тег Параграфа с тегом Поля внутри

11  
 Дата·проведения[<1::Paragraph[<#>Group- >]¶  
 Место·проведения:¶  
 ¶  
 [<3::Paragraph[<#>Group- >]·¶  
 ¶

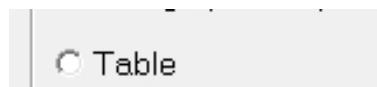
32. Применить стиль форматирования Название или любой другой, который подходит для вашего корпоративного шаблона



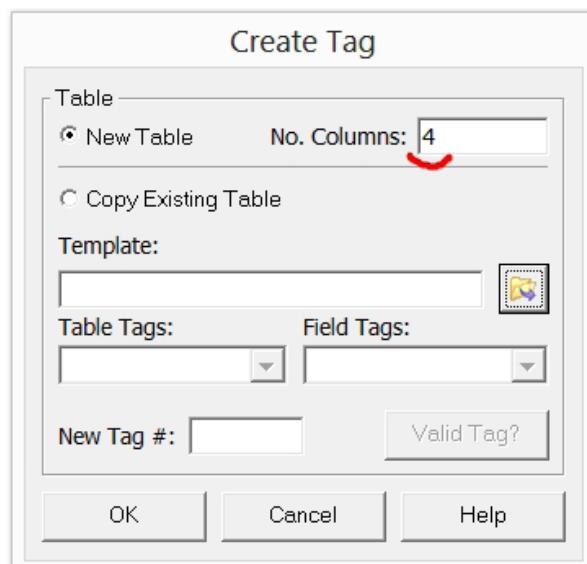
Теперь, когда в это место при публикации будет выводиться название задачи, оно будет отформатировано выбранным вами стилем.

Далее настроим вывод таблицы вопросов рабочей встречи, взяв их из исходных требований, соединенных с задачей рабочей встречи.

1. Установить курсор на новую строку в шаблоне документа MS Word под ранее созданным тегом заголовка документа.
2. Нажать Create на панели инструментов
3. Выбрать тип тега - Table



4. В следующем диалоге установить число столбцов таблицы равным 4



5. Аналогично предыдущим настройкам выбрать таблицу правил и установить знак вопроса в идентификатор элемента верхнего уровня

Single Item Type      Rules Tables: 1: Задача и исходные требований

Hierarchy of Item Types:

	Level:	Prefix:	Identifier (Level 0 Item):
TASK	0		?
SR	1		
SR	2		

Create/Edit      Display

## 6. Сохранить тег и закрыть это диалоговое окно.

После создания тега таблицы наш шаблон таблицы будет выглядеть следующим образом:

ColumnTitle_1	ColumnTitle_2	ColumnTitle_3	ColumnTitle_4
1			

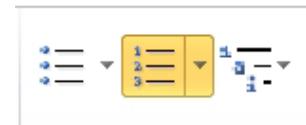
Теперь необходимо поименовать заголовки и указать, какие поля каких элементов из таблицы правил выводить в соответствующие колонки.

Сначала просто отредактируем заголовки следующим образом:

№	Ид	Вопрос	Ответ
1			

А затем сделаем «набивку» полей.

## 7. Установить курсор в ячейку под заголовок № и нажать на иконку нумерованного списка Word

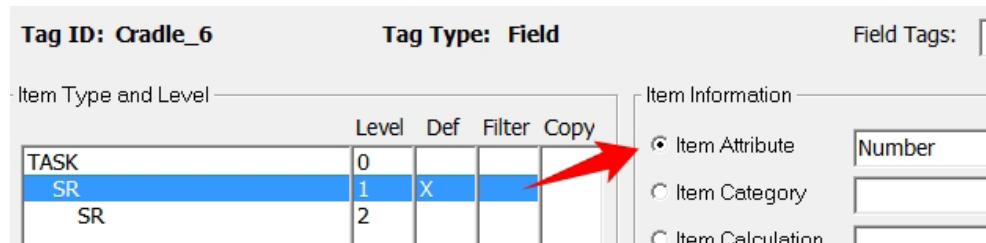


Тем самым мы используем автоматическую нумерацию ячеек таблицы для удобства работы на совещании.

## 8. Установить курсор в следующую ячейку и нажать Create на панели инструментов

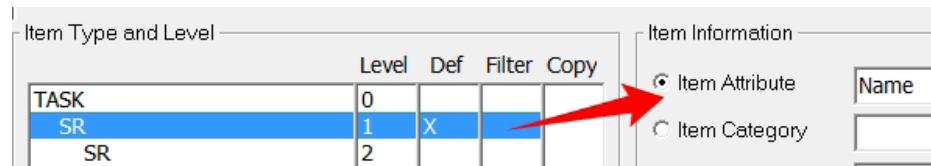
ментов Document Publisher

9. Создать тег типа Field со следующими настройками:

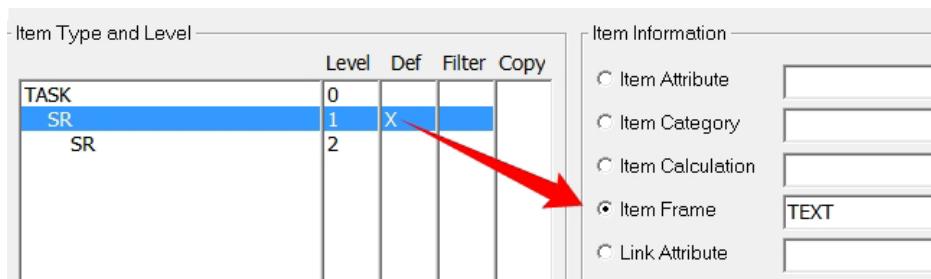


Тем самым мы выведем во вторую колонку идентификаторы SR, которые подлежат согласованию.

10. Выделить третью ячейку и создать тег поля для вывода имени SR с настройками:



11. Выделить четвертую ячейку и создать тег поля для вывода текста SR с настройками:

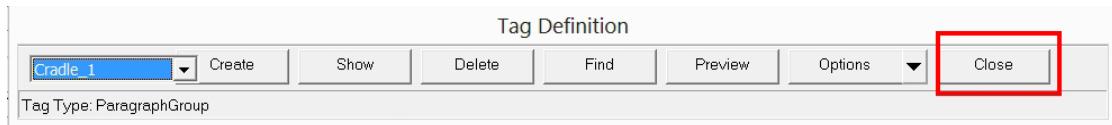


В итоге в шаблоне получится следующая картина:

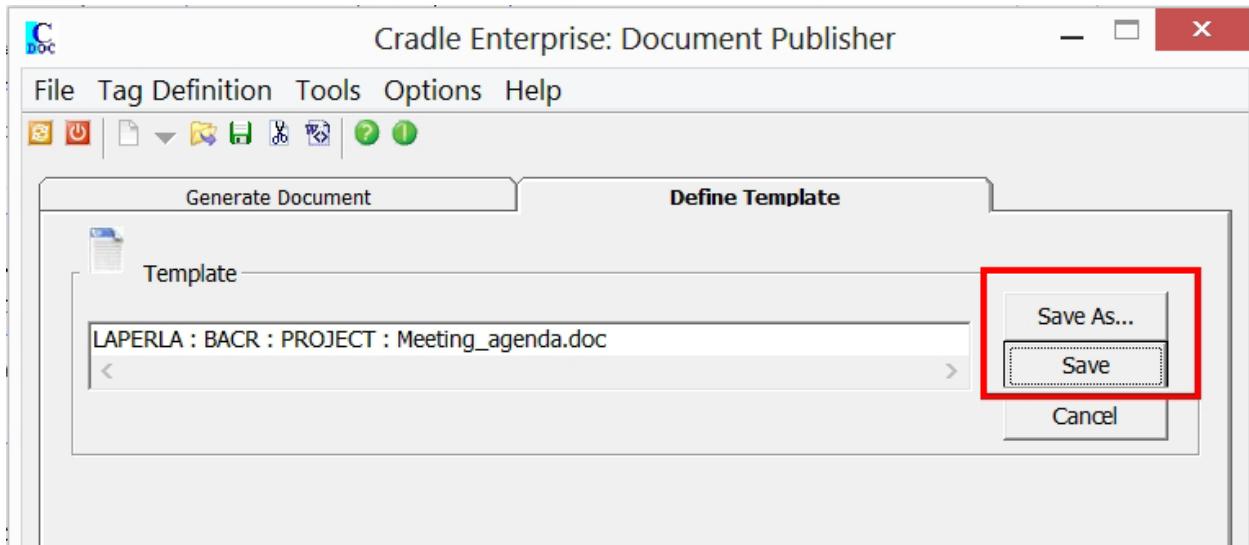
№	Идя	Вопрос	Отвєт
1	[#]·¤	[#]¤	[#]¤

Все, наш шаблон готов, теперь мы можем сохранить его. Для этого

12. Нажать кнопку Close на панели инструментов Document Publisher



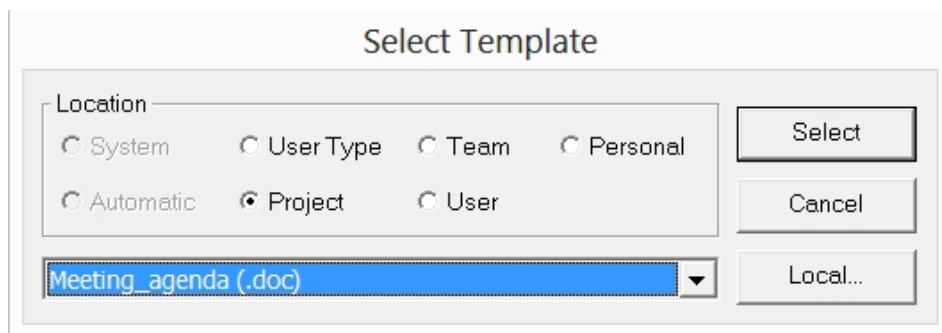
### 13. Нажать кнопку Save на вкладке Define template



## Опубликовать повестку рабочей встречи

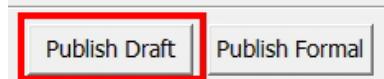
Чтобы опубликовать повестку по сделанному шаблону

1. Запустить Document Publisher
2. Перейти во вкладку Generate Document
3. Нажать кнопку Select в разделе Template
4. Выбрать сохраненный шаблон



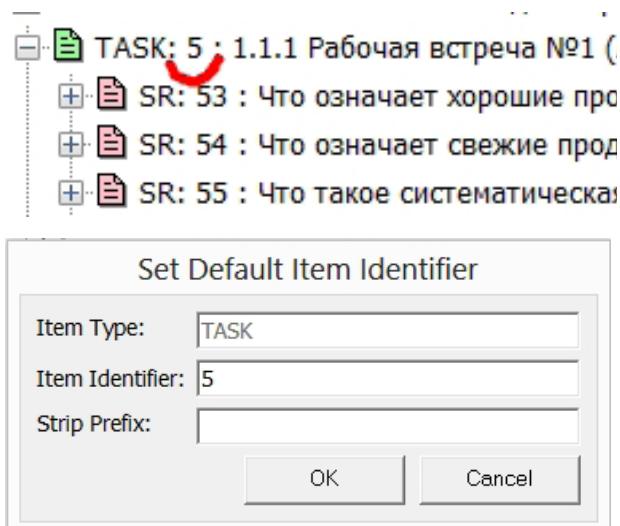
5. Нажать кнопку Publish Draft для публикации черновика документа без

присоединения его к базе данных Cradle



6. После запуска публикации система спросит вас об идентификаторе задачи, на основании которой будет строится публикация.

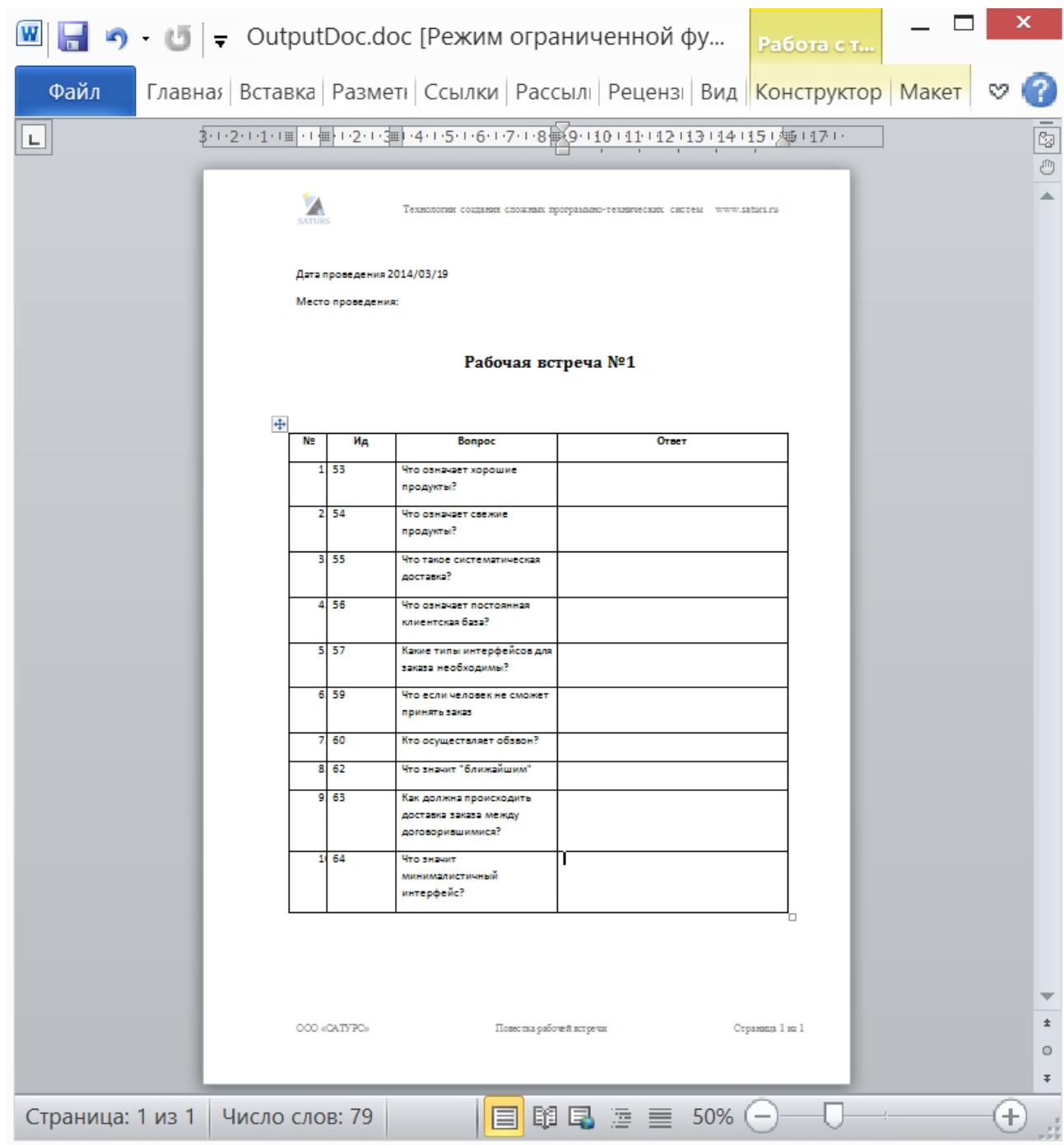
Введите номер задачи, с которой у вас связаны требования, подлежащие согласованию и нажмите OK.



По окончанию публикации будет выведено сообщение:



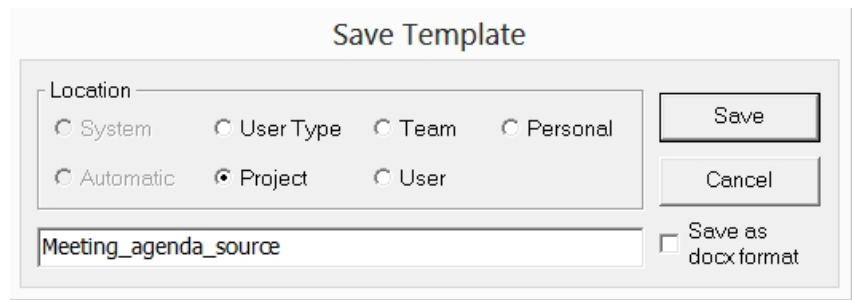
Если вы все сделали правильно, то опубликованный документ будет выглядеть следующим образом



Отлично! Теперь мы можем использовать этот шаблон для публикации любых повесток рабочих встреч. Однако мы можем сделать его еще немного информативнее и вывести текст исходных требований, с которыми были связаны наши вопросы.

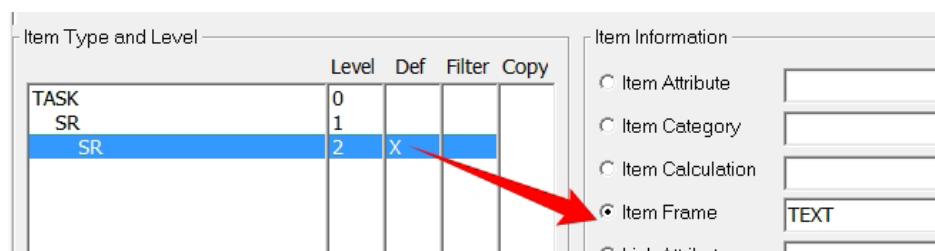
## Расширить шаблон для вывода связанных исходных требований

1. Открыть созданный ранее шаблон на вкладке Define Template
2. Нажать на панели инструментов кнопку Close, чтобы сохранить его под новым именем



Это позволит нам в дальнейшем использовать оба шаблона, выбирая подходящий для каждой конкретной ситуации.

3. Открыть сохраненную копию шаблона
4. Удалить все теги полей, связанных с таблице, устанавливая курс в соответствующий тег и нажимая последовательно кнопки Find, Delete
5. Удалить тег таблицы
6. Создать новый тег таблицы, расчитанной на пять колонок.
7. Первые четыре колонки назначить аналогично тому, как мы настраивали для первого шаблона.
8. В пятую колонку вывести текст, связанных с вопросами SR для этого создать тег со следующими настройками:



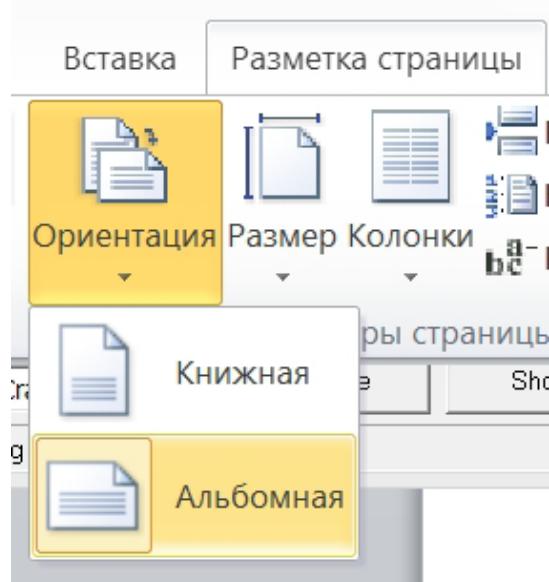
9. Выделить созданный тег и установить для него более мелкий шрифт

В итоге у вас должна получиться следующая картина таблицы:

<b>[&lt;3::Paragraph[&lt;#&gt;Group-]&gt;]</b>				
Ном	Идн	Вопрос	Ответ	Связан-с-я
1.→	[<#>]	[<#>]	[<#>]	[<#>]

Для этого шаблона повестки мы рекомендуем использовать альбомную ориентацию листа. Для этого

10. Перейти в раздел меню Word Разметка страницы и выбрать Альбомная:



11. Закрыть и сохранить шаблон

После публикации повестки по данному шаблону вы должны получить результат, представленный на картинке ниже. Последний столбец повестки заполнен текстом исходных требований, с которым связан обсуждаемый вопрос. Это очень удобно.

При желании вы можете вывести текст исходных требований на отдельный лист, оставив в таблице только ссылки на их номера. Это может быть удобно, если есть много вопросов, связанных с одним и тем же исходным требованием и позволит избежать многократного вывода его текста.

Рабочая встреча №1

№	Ид	Вопрос	Ответ	Связан с
1.	53	Что означает хорошие продукты?		Приобрести хорошие продукты для семьи в супермаркетах мегаполисов становится все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выживания» им не хватает организованности для того, чтобы объединиться с другими хозяйствами и решить вопросы систематической доставки продуктов в мегаполис из пригорода. Группа инициативных клиентов (далее Группа) решает взять на себя эти организационные функции, чтобы прежде всего обеспечить себя свежими продуктами на постоянной основе. Для этого им необходимо набрать постоянную клиентскую базу, чтобы организовать отовую закупку, т.к. ради двух кг мяса никто не будет организовывать забой скотины.
2.	54	Что означает свежие продукты?		Приобрести хорошие продукты для семьи в супермаркетах мегаполисов становится все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выживания» им не хватает организованности для того, чтобы объединиться с другими хозяйствами и решить вопросы систематической доставки продуктов в мегаполис из пригорода. Группа инициативных клиентов (далее Группа) решает взять на себя эти организационные функции, чтобы прежде всего обеспечить себя свежими продуктами на постоянной основе. Для этого им необходимо набрать постоянную клиентскую базу, чтобы организовать отовую закупку, т.к. ради двух кг мяса никто не будет организовывать забой скотины.

ООО «САТУРС» Повестка рабочей встречи Страница 1 из 3

Страница: 1 из 3 Число слов: 571 русский 70%

## Настроить панель фаз для генерации повестки

Чтобы каждый раз, когда нам необходимо сгенерировать повестку встречи не тратить время на запуск Document Publisher мы можем вынести вызов этой операции на панель Фаз и/или на стартовую страницу. Для этого

1. Открыть Workbench
2. Войти в меню Установки проекта, раздел Фазы проекта
3. Добавить новую фазу «Подготовка к совещаниям»
4. В эту фазу добавить вызов публикации отчета по первой и второй форме.

Результатирующие настройки должны выглядеть следующим образом:

Установки проекта

Заголовок проекта: Базовый курс Cradle

Настройки: Фазы проекта

Фазы проекта

	Определение	Действие
Управление проектом		
Подготовка к совещанию		
<b>D Опубликовать повестку - вопросы</b>	Meeting_agenda	
<b>D Опубликовать повестку - вопросы-исходные требования</b>	Meeting_agenda_source	
Анализ исходных требований		
Оценка проекта		

Свойства фазы

Имя: Опубликовать повестку - вопросы

Тип: Document (.doc)

Документ: Meeting\_agenda\_source

Добавить  
Копировать  
Удалить  
Наверх  
Вниз  
<->

5. Сохранить настройки

6. Перейти на панель Фаз и нажать на любой из вызовов публикации:

Фазы

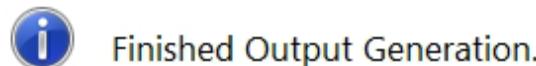
- Управление проектом
- Подготовка к совещанию
  - D Опубликовать повестку - вопросы**
  - D Опубликовать повестку - вопросы-исходные требования**
- Анализ исходных требований
- Оценка проекта

Как и при публикации из Document Publisher будет запрошен номер задачи:

Set Default Item Identifier

Item Type:	TASK
Item Identifier:	5
Strip Prefix:	
OK Cancel	

А по окончании публикации выведено сообщение:

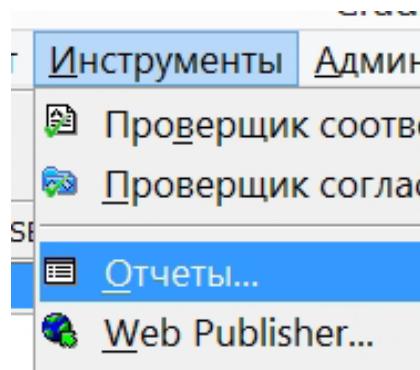


После этого необходимо немного подождать и в MS Word откроется результатирующий документ повестки рабочей встречи.

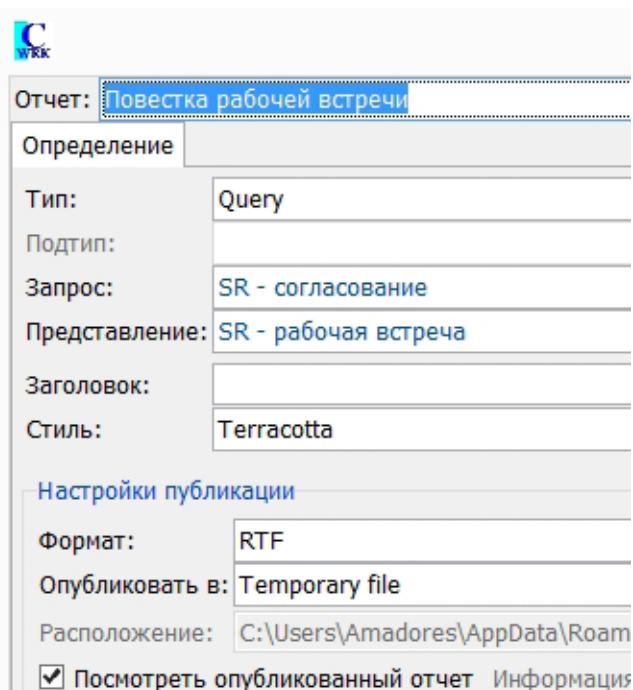
Можно идти на встречу!

Мы также можем вывести на панель фаз вызов простого отчета, который выводил таблицу вопросов. Преимущество этого подхода в том, что не требуется лицензий на модуль DOC. Для этого:

1. Перейти в меню Инструменты - > Отчеты

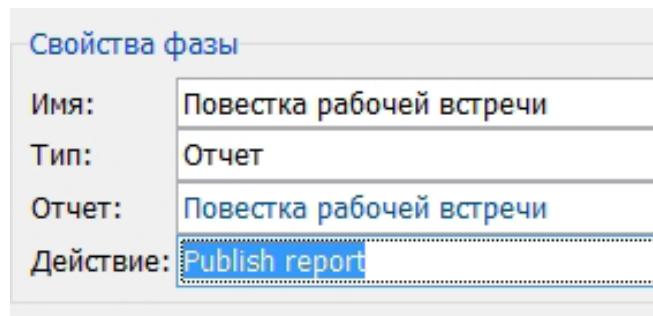


2. Выбрать следующие настройки

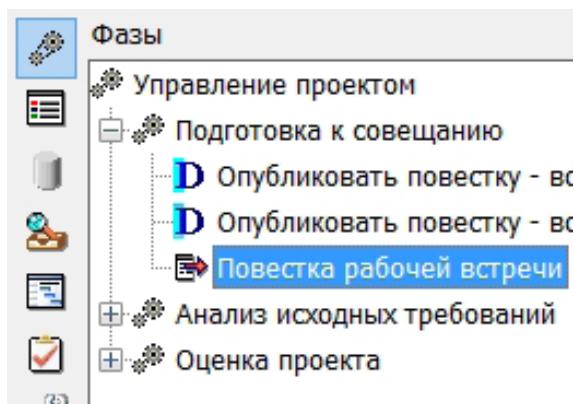


Отчет:	Повестка рабочей встречи
Определение	
Тип:	Query
Подтип:	SR - согласование
Запрос:	SR - рабочая встреча
Представление:	SR - рабочая встреча
Заголовок:	
Стиль:	Terracotta
Настройки публикации	
Формат:	RTF
Опубликовать в:	Temporary file
Расположение:	C:\Users\Amadores\AppData\Roam
<input checked="" type="checkbox"/> Посмотреть опубликованный отчет	

3. Сохранить отчет под именем «Повестка рабочей встречи»
4. Добавить вызов отчета на панель Фаз с настройками:



Теперь нажимая на соответствующую фазы вы получите таблицу вопросов в RTF



Так поэтапно будет формироваться типовая структура задач, часто используемых в вашей организации. Со временем донастроек систем понадобится все меньше и меньше, а технология будет транслироваться из проекта проект вместе с типовой схемой настроек Cradle, зафиксированных в схеме проекта.

## Подготовиться к оценке области влияния

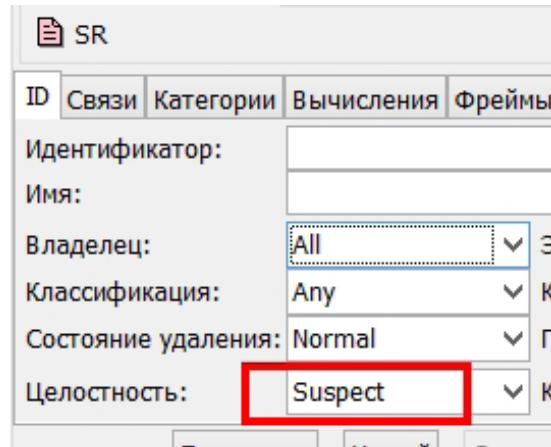
Чтобы после загрузки ответов успешно воспользоваться функциями подсветки области изменений необходимо включить режим подсветки области изменений.

1. Выбрать меню Workbench Редактировать-> Настройки, раздел Управление интерфейсом
2. Включить Показывать подозрения на нарушение целостности в режим

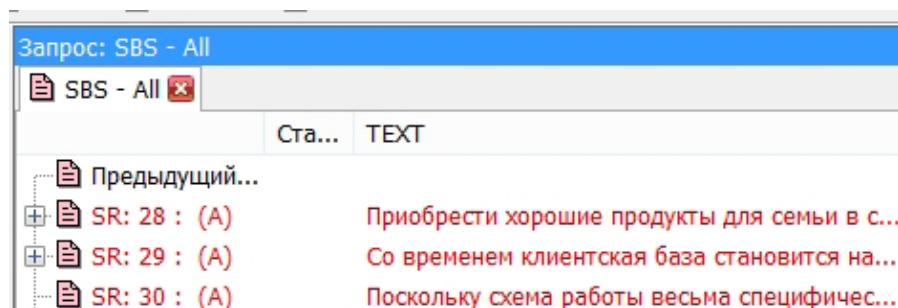


Поскольку мы не включили сразу эту функцию, то в ходе различных изменений могло накопиться довольно много элементов, подозреваемых на изменение. Очистим эту информацию, чтобы далее она нам не мешала. Для этого:

3. Создать запрос поочередно ко всем типам элементов, который выбирал бы элементы, подозреваемые на изменение, установив поле Целостность = Suspect:



Выбранные таким запросом элементы будут отображены красным цветом, так отображаются в системе элементы, подозреваемые на изменение (при включенном режиме подсветки).



4. Выделить выбранные элементы

5. В контекстном меню Еще выбрать Подтвердить целостность иерархии или подтвердить целостность элемента

Обновить запрос и проделать тоже самое до тех пор, пока результат запрос не окажется пустым для всех типов элементов.

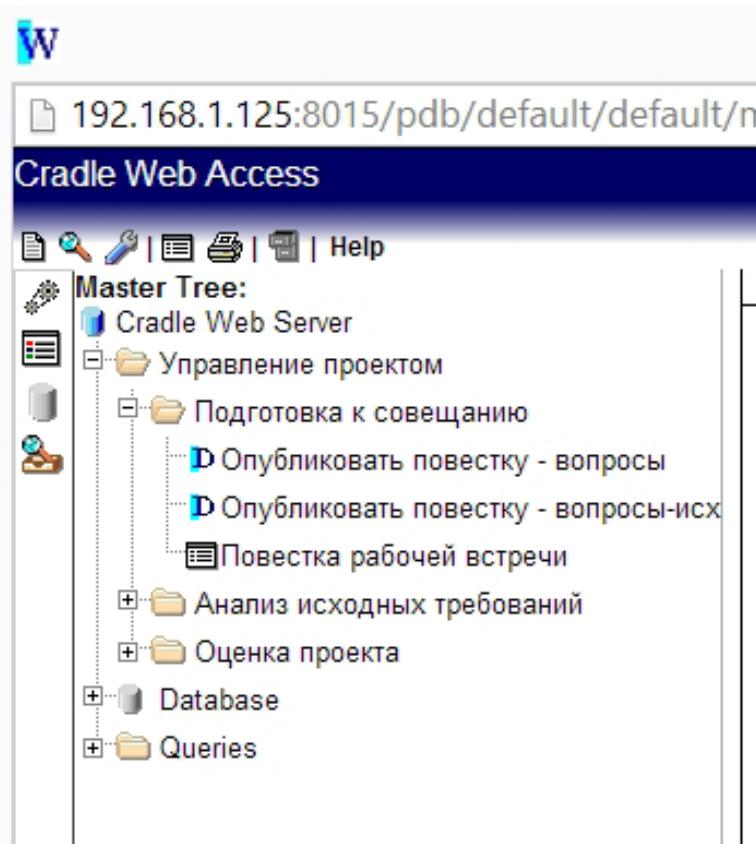
**Провести встречу и зафиксировать результаты**

Воспользуемся готовой повесткой и зафиксируем в колонке Ответ результаты совещания. Это выполняется использованием стандартных функций MS Word.

Если же у Вас есть возможность сразу писать ответы в Cradle, то воспользуйтесь этим, тогда по результатам совещания, нажав на фазу генерации повестки, вы автоматически получите протокол совещания и следующие два этапа будут не нужны:

- загрузить результаты встречи
- объединить ответы с вопросами

Помните, что в ходе проведения совещания вы также можете воспользоваться веб-интерфейсом для ввода ответов на вопросы.



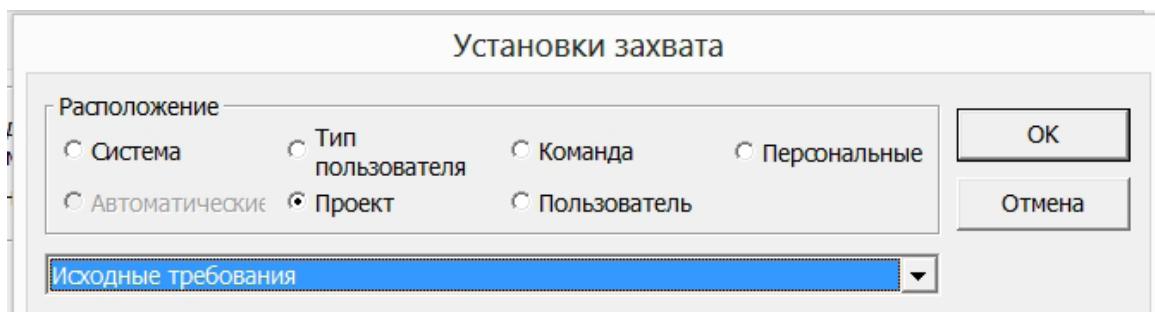
Master Tree:

- Cradle Web Server
- + Управление проектом
- Database
- Current Query
  - Previous...
  - SR: 53 : Что означает хорошие продукты? (A)
  - SR: 54 : Что означает свежие продукты? (A)
  - SR: 55 : Что такое систематическая доставка
  - SR: 56 : Что означает постоянная клиентская
  - SR: 57 : Какие типы интерфейсов для заказа
  - SR: 59 : Что если человек не сможет принять
  - SR: 60 : Кто осуществляет обзвон? (A)
  - SR: 62 : Что значит "ближайшим" (A)
  - SR: 63 : Как должна происходить доставка з
  - SR: 64 : Что значит минималистичный интерфейс
  - Next...

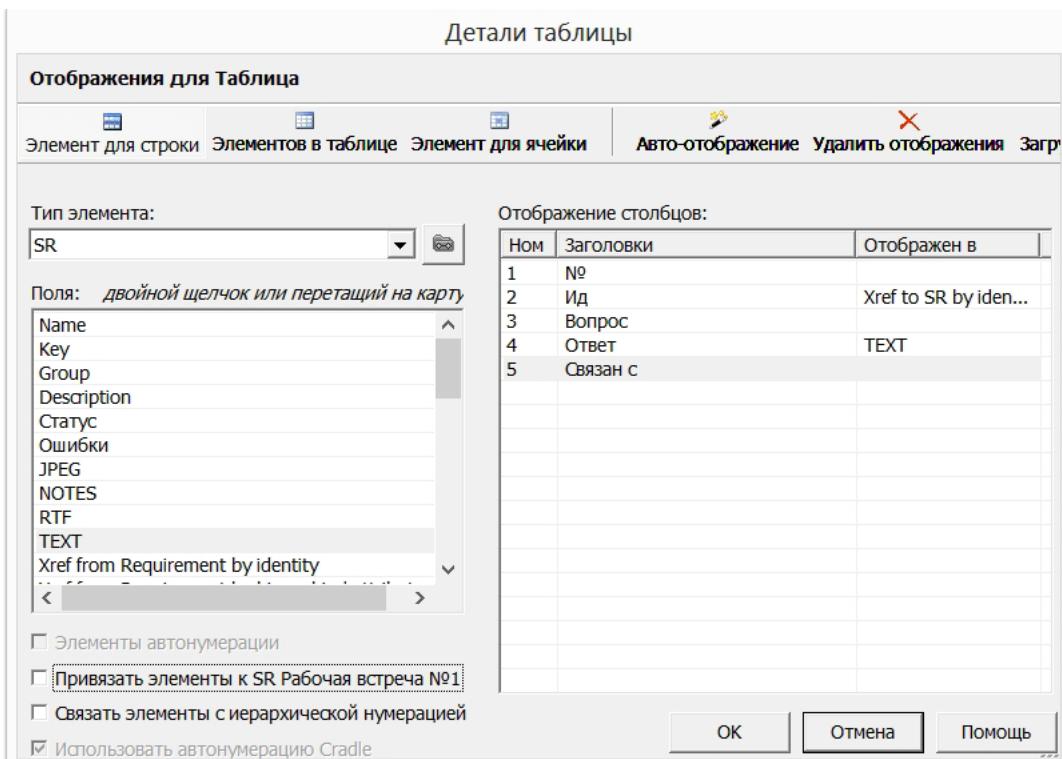
### Загрузить результаты встречи

Если вы не писали ответы на вопросы сразу в проектную базу, а заполняли генерированную повестку встречи, то далее можно автоматизированно загрузить ответы обратно в базу. Для этого:

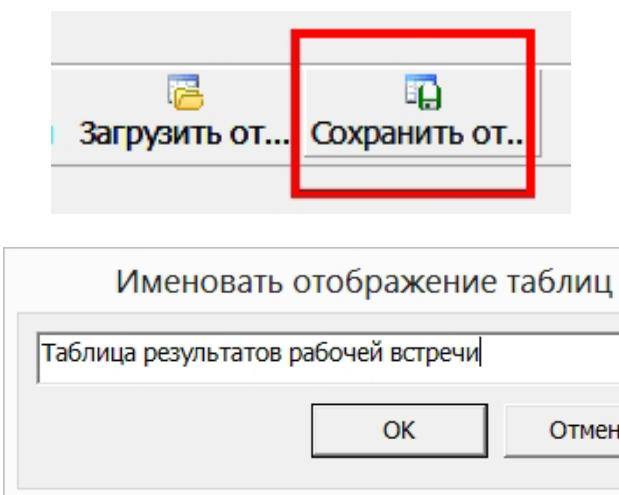
1. Открыть Document Loader
2. Открыть файл повестки с ответами установками захвата, которые мы сохранили для загрузки исходных данных:



3. Выделить таблицу и указать для нее следующие настройки отображения



4. Сохранить настройки отображения таблицы, для того, чтобы использовать повторно в будущем:



5. Нажать OK

6. Нажать Захватить документ и установить настройки

**Захватить документ**

**Детали исходного документа**

Имя:	MA
Версия:	01
Заголовок:	Meeting_agenda_source_answers
Тип документа:	▼
Классификация:	UNCLASSIFIED ▼
Выпуск:	1
Дата выпуска:	мар 2014
Ссылка:	1
Комментарии:	П д б
Расположение:	C:\3SL_Cradle_base_book\CradleProject\Ф <

**Элемент верхнего уровня**

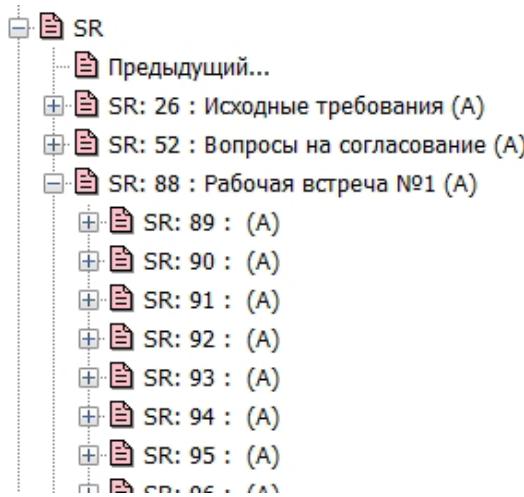
Тип элемента:	SR	▼
Индекс:	<auto generated>	
Имя:	Рабочая встреча №1	

**7. Нажать Захватить**

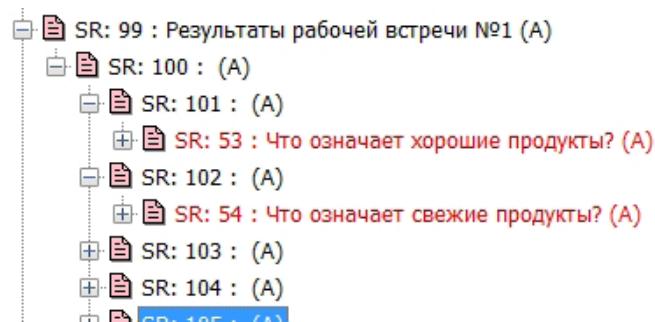
**8. Проверить отчет и убедиться, что все захвачено без ошибок и все необходимые связи созданы:**

✓ Создать элемент	SR 98 (A)
✓ Добавить фрейм	TEXT фрейм SR 98
✓ Создать связь	От Source Statement к SR 98
✓ Создать связь	От SR 89 (A) к SR 98
✓ Создать связь	От SR 90 (A) к SR 98
✓ Создать связь	От SR 91 (A) к SR 98
✓ Создать связь	От SR 92 (A) к SR 98
✓ Создать связь	От SR 93 (A) к SR 98
✓ Создать связь	От SR 94 (A) к SR 98
✓ Создать связь	От SR 95 (A) к SR 98
✓ Создать связь	От SR 96 (A) к SR 98

В Workbench в панеле База данных захваченные элементы будут выглядеть следующим образом:



Аналогично тому, как мы загружали исходные требования, результаты совещания прогружены в поле текст и потом не имеют заголовков. Однако каждый из них связан с исходным вопросом:



При этом сработала функция подсветки изменений и вопросы теперь отражены красным цветом.

Мы можем оставить эти требования как есть и далее обработать аналогично другим исходным требованиям. А можем объединить с исходными вопросами, чтобы сократить количество элементов.

## Объединить ответы с вопросами

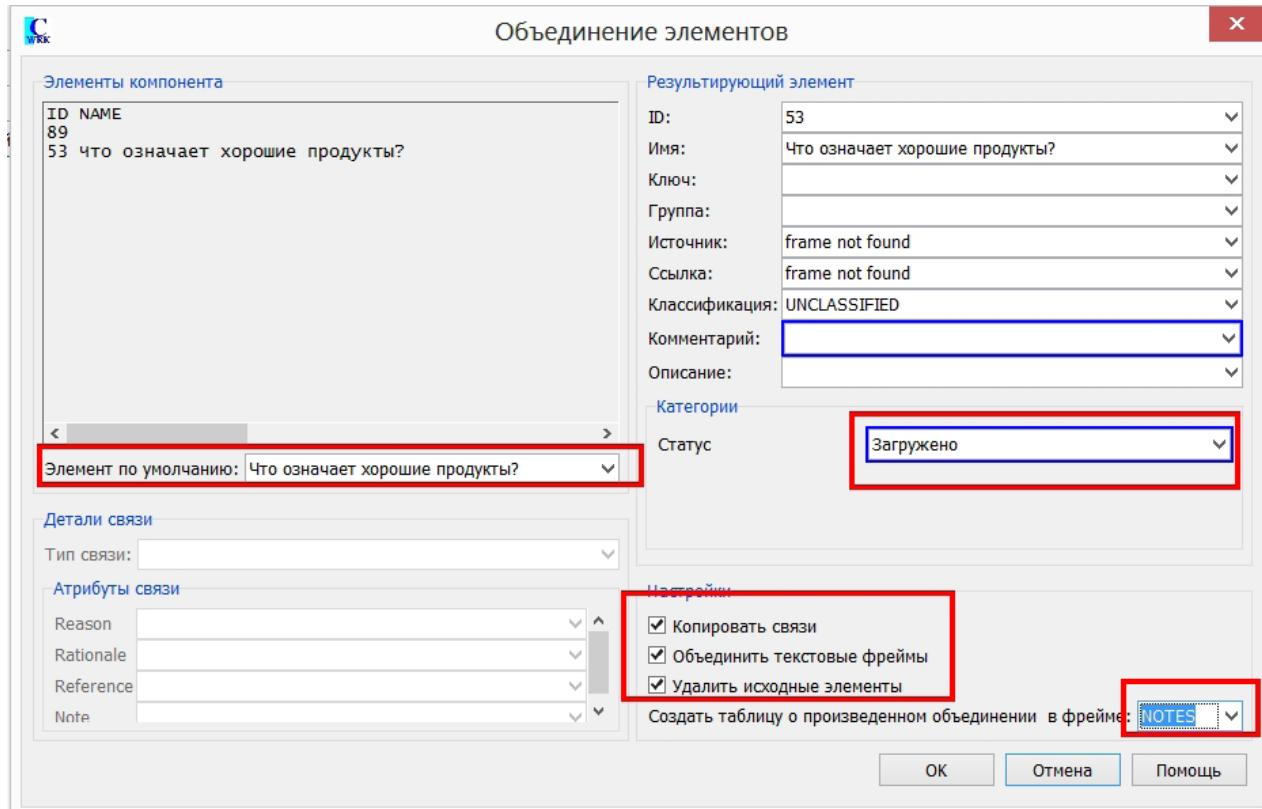
Чтобы объединить вопрос и ответ,

9. Выбрать оба элемента

10. В контекстном меню выбрать Еще -> Объединить

## 11. Установить следующие настройки объединения

Выбрать в качестве элемента по умолчанию тот, что соответствует вопросу и выставить остальные настройки как указано на рисунке.



После выполнения операции объединения мы получим следующий результат:

В поле Name будет сохранено имя вопроса, а в поле текст отразится ответ на этот вопрос. Статус будет изменен с Согласование на Загружен, что соответствует обычному статусу только что загруженного исходного требования, подлежащего обработке, а в поле NOTES будет запись об объединении двух элементов.

Проделайте это для всех вопросов и ответов - это выполняется очень быстро и потренируйтесь объединять элементы.

SR: 53 (A)

53 (A) X

ID: 53	Key: <input type="text"/>	Name: Что означает хорошие продукты?	Created By: MANAGER	On: 27/02/14
Security: UNCLASSIFIED		Changed By: MANAGER On: 08/03/14		
Owner: MANAGER		Status: <input type="checkbox"/> Baseline: <input type="text"/>		

Note details:

Group:  Comment:  Description:

Text:

Хорошие продукты – это продукты без химических добавок, которые необходимы для ежедневного питания. В первую очередь – это молоко и мясо без антибиотиков, фрукты, овощи, ягоды с допустимым содержанием нитритов и нитратов. Зерно, мука, сухофрукты, крупы, яйца. Все это должно быть выращено без применения удобрений, наносящих вред человеку. Кисломолочные продукты, масло, сыр, творог

Categories

Статус Загружено

Ошибки

Discussions

Text frames

RTF

Other frames

JPEG

JPEG ▾

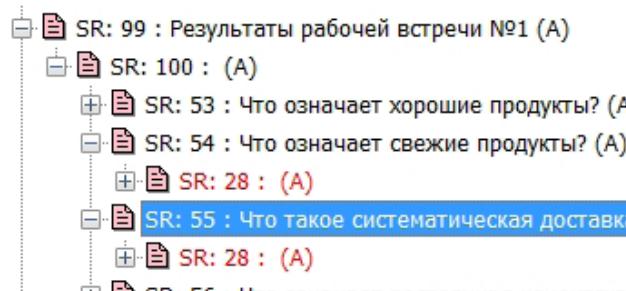
Notes:

Requirement created by merging requirements:

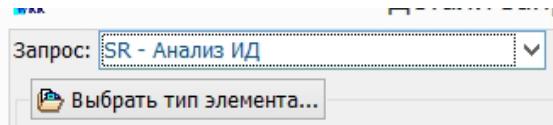
Number	Key	Origin	Reference
53			

## Оценить область влияния

После объединения ответов и вопросов, подозреваемыми на изменение стали исходные требования, которые были связаны с вопросами, теперь они подсвечиваются красным во всех представлениях:



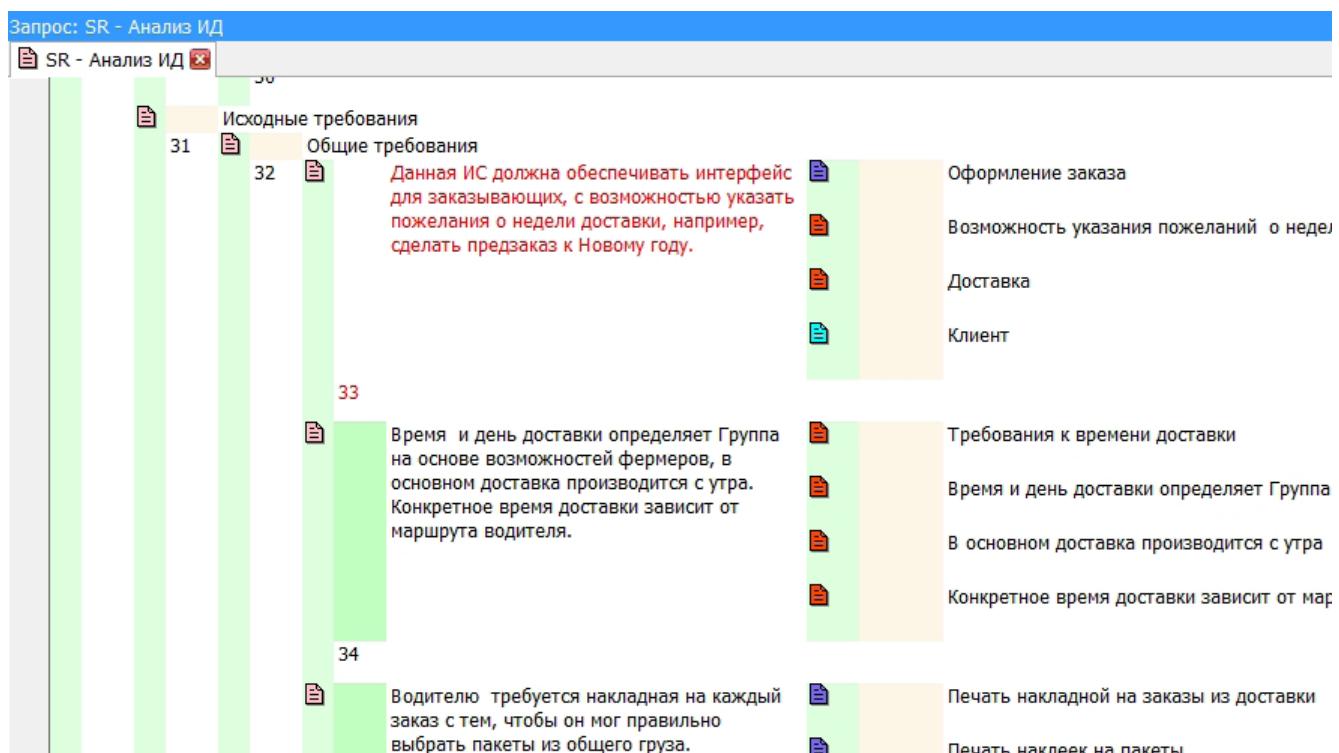
Мы можем открыть запрос SR - анализ ИД



применить к нему представление SR - Анализ с которым мы работали в начале проекта

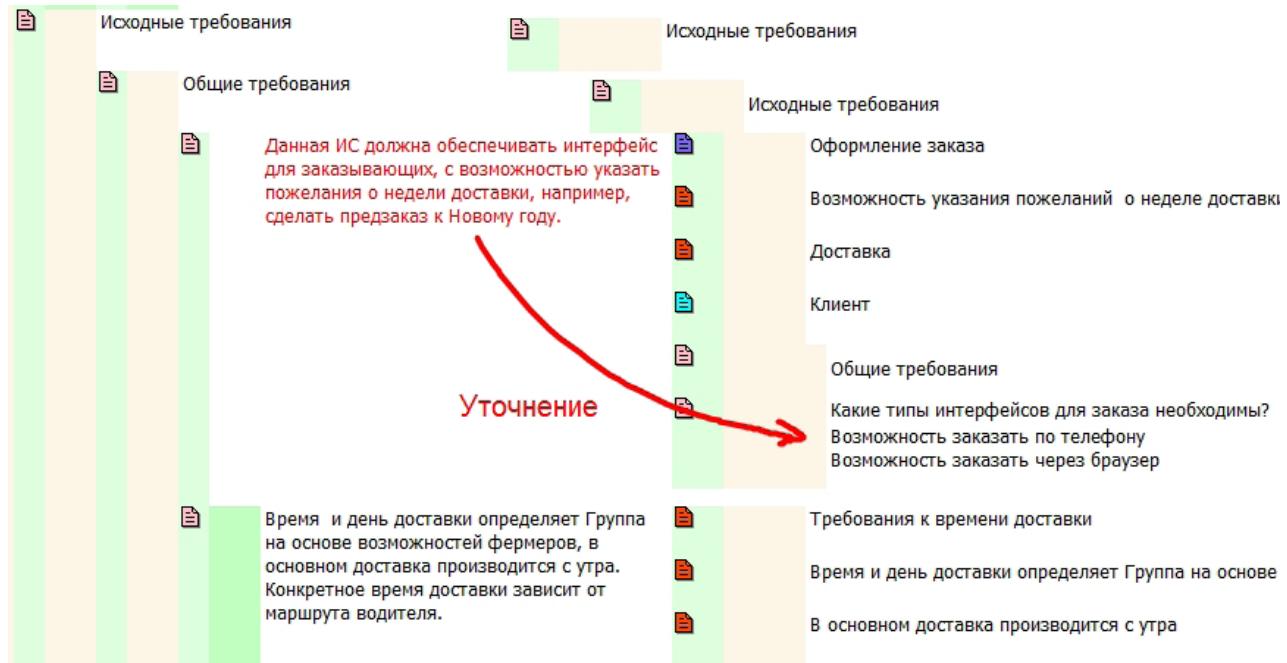


и увидеть, что теперь в данном представлении посвящаются исходные требования, которые были дополнены ответами на вопросы



А это означает, что производные требования, связанные с этим исходным требованием, возможно подлежат уточнению.

Чтобы вывести тут же и сами ответы на вопросы воспользуемся представлением SR - анализ и вопросы, которое мы также ранее настраивали.



## Формализовать новые ИТ - результаты рабочей встречи

Аналогично тому, как мы проводили анализ исходных требований далее необходимо формализовать и разложить на элементарные требования полученные в ходе рабочей встречи ответы на вопросы.

Откройте сохраненную сессию «Анализ ИД», чтобы организовать рабочие области удобным для этой задачи способом.

Также должны быть учтены ошибки, конфликты, возможно появятся новые вопросы. Этот процесс будет итеративно сходиться в процессе выполнения проекта.

Постройте основной запрос для анализа SR, к узлу - Вопросы на согласование. Это позволит вам сконцентрироваться только на новых поступивших требованиях. При этом исходные требования, связанные с вопросами также попадут в представление SR-Анализ по навигации Downwards, которая используется в этом представлении.

Скриншот программы Cradle Enterprise: WorkBench - Базовый курс Cradle - MANAGER - (Анализ ИД) - (28 непрочитанных оповещений)

Файл Редактировать Представление Элемент Запрос Инструменты Администрирование Окно Помощь

Запрос: SR - Вопросы

SR - Вопросы Предыдущий...

1 52 Хорошие продукты – это продукты без химических добавок, которые необходимы для ежедневного питания. В первую очередь – это молоко и яйца без антибиотиков, фрукты, овощи, ягоды, зерновые сорта, зерноглазаты и китроты. Зерно, мука, сухофрукты, крупы, яйца. Все это должно быть выражено без применения удобрений, наносящих вред человеку.

Кисломолочные продукты, масло, сыр, творог

53 Требование для семян в супермаркетах неподъемное становиться все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выживания» им не хватает организованности для того, чтобы организовать доставку продукции в супермаркеты. Для этого им необходимо набрать постоянную клиентскую базу, чтобы организовать оптовую закупку, т.к. ради двух кг мяса никто не будет организовывать забой скотины.

28 Молоко – утреннее или вечернее Яйца – не более трех дней Мясо – вечерний забой Овощи и фрукты должны храниться не дольше, чем 2-3

54 Требование для семян в супермаркетах неподъемное становиться все сложнее, в тоже время небольшие фермерские хозяйства страдают из-за отсутствия организованного рынка сбыта – в режиме «выживания» им не хватает организованности для того, чтобы объединиться с другими хозяйствами и решить вопросы систематической доставки

Запрос: NFR - Top-Level

NFR - Top-Level Предыдущий...

NFR: 1 : Цель создания системы - обеспечить себя (инициативную группу)

NFR: 15 : Требование к информационной модели (A)

NFR: 66 : Правила управления заказами (A)

NFR: 67 : Правила управления доставкой (A)

Следующий...

Запрос: FUNCTION - Bottom-Level

FUNCTION - Top-Level FUNCTION - Bottom-Level Identity Ver Dft Name Предыдущий...

1 6 1 А Сортировка заказов

2 5 2 А Планирование вязерации

3 4 4 А Печать накладной доставки

4 5 5 А Печать наклекки

5 6 6 А Выбрать пакеты и

6 7 7 А Указать применение мяса

7 8 8 А Разделка мяса

8 12 12 А Позвонить клиенту согласования доставки

9 13 13 А Задфиксировать результат звонка по согласованию

Следующий...

Запрос: SBS - Top-Level

SBS - Top-Level Identity Ver Dft Name Key

Предыдущий...

SBS: 1 : Фермерское хозяйство (A) 1 А Фермерское хозяйство

SBS: 2 : Группа инициативных клиентов (A) 2 А Группа инициативных клиентов

SBS: 6 : Клиент (A) 6 А Клиент

Следующий...

# *Проектирование системы*

## **Понять задачи проектирования**

С точки зрения работы в системе дальнейший процесс практически ничем не отличается от того, что мы выполняли ранее - мы также должны создавать новые элементы и отражать их связи, контролируя общую целостность модели.

После того, как мы обработали исходные требования процесс сдвигается в область работы над формализованными требованиями - определение необходимых функций, компонентов системы и нефункциональных требований.

В ходе определения функций не должны быть забыты нефункциональные требования, которые во многом определяют не только качественные характеристики функций, но и могут приводить к изменению их состава.

Поэтому в рамках этого раздела книги мы посмотрим какие еще, упрощающие анализ и контроль, средства может предложить инструмент, а именно: матрицы трассировки, диаграммы HID.

На фазе проектирования часто требуется проработать функциональные модели не только на структурном уровне, но и отразить временной аспект, поэтому далее мы изучим возможности 3SL Cradle по разработке поведенческих моделей в нотации UML Activity Diagram или, по просту, блок-схемы алгоритма.

Если это не первая разрабатываемая вами система, то безусловно у вас уже будет накоплен опыт и база требований/моделей, которые возможно использовать в этом проекте повторно, поэтому мы посмотрим как экспортить и импортировать требования для их повторного использования.

И наконец, если Заказчик поменяет часть исходных требований (что бывает довольно часто) - мы будем к этому готовы, если изучим как загрузить новую версию документа с исходными требованиями, оценить дельту и объем изменений.

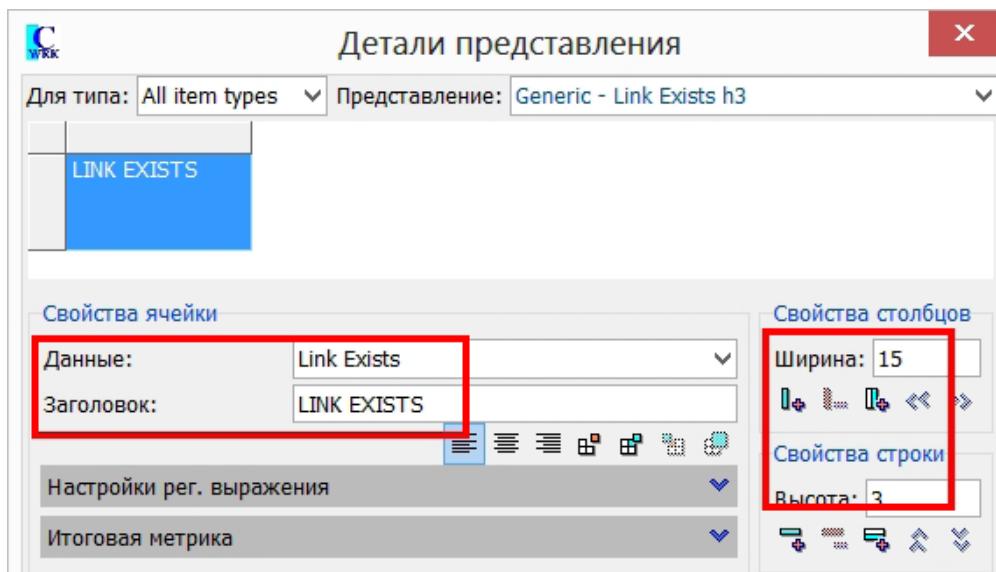
## Настроить матрицы трассировки

Один из важных этапов проектирования - это определение компонентной архитектуры системы и распределение (allocation) функций по компонентам. Есть ряд правил, позволяющих построить сбалансированную архитектуру системы, которые мы обычно преподаем на курсах ООО «САТУРС», в рамках данного раздела мы рассмотрим только инструментальную поддержку анализа архитектуры.

Чтобы построить симпатичную матрицу трассировки нам понадобится два вспомогательных представления:

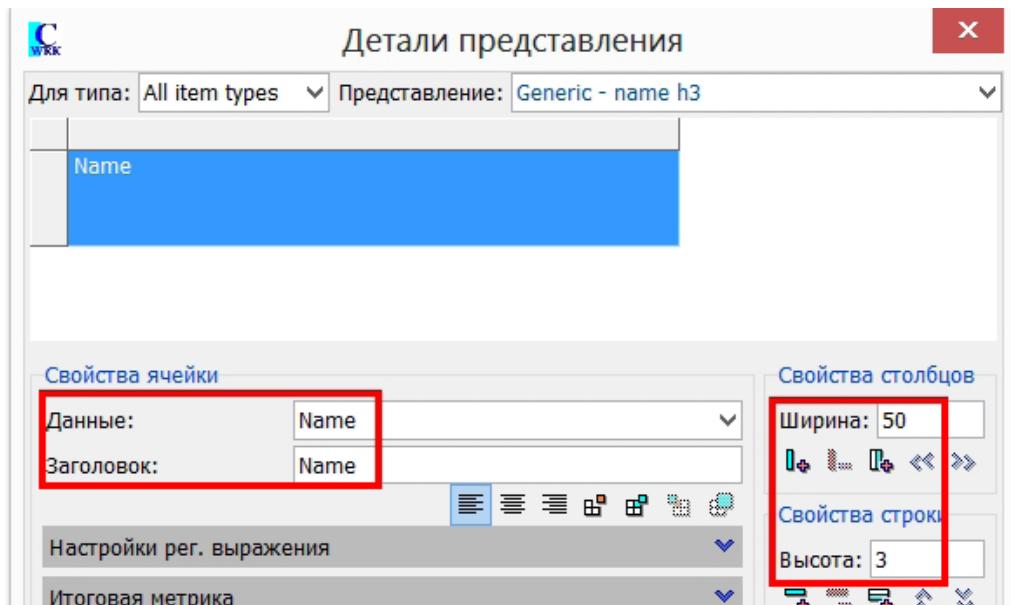
### 1. Создать представление типа Generic со следующими настройками

Это представление будет использоваться для вывода значков связей в матрице трассировки.



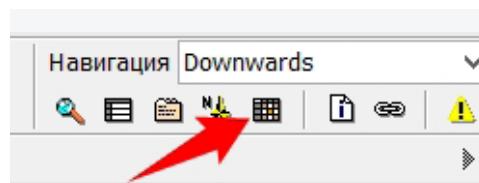
### 2. Сохранить представление под именем Generic - Link Exists h3

### 3. Создать второе вспомогательное представление типа Generic со следующими настройками:

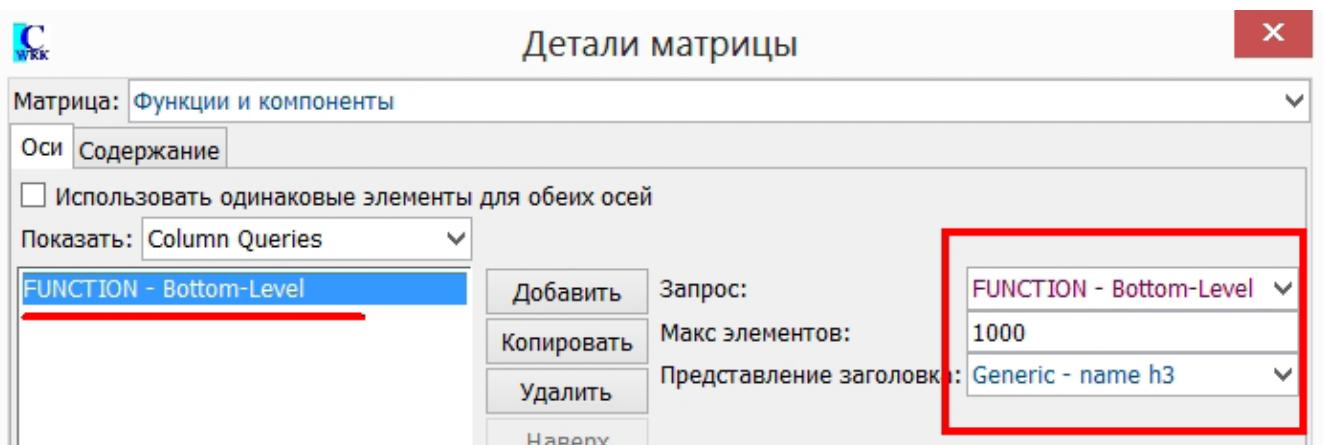


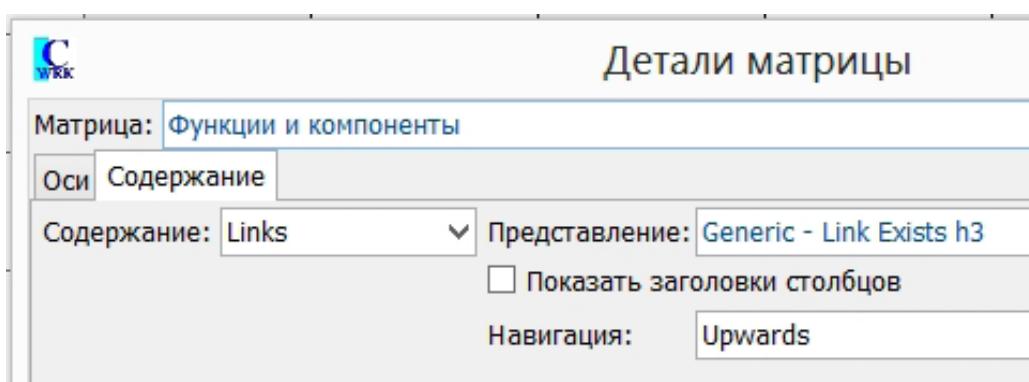
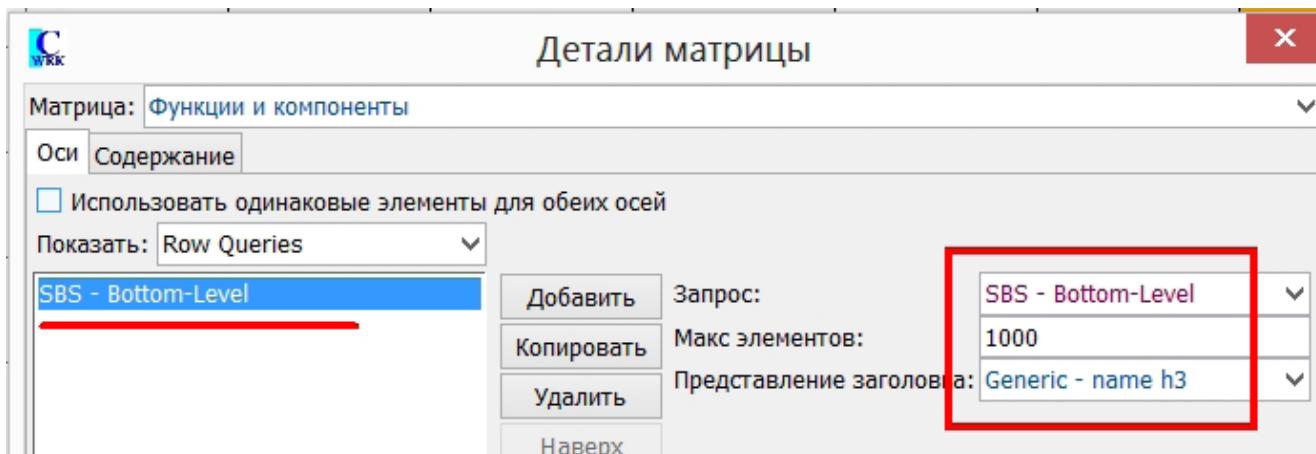
4. Сохранить представление под именем Generic - name h3

5. Вызвать диалог Детали матрицы



6. Настроить отображение колонок, столбцов и содержания ячеек как указано на следующих трех изображениях:





7. Сохранить матрицу трассировки под именем Функции и компоненты
8. Нажать кнопку Применить на диалоге Детали матрицы

В итоге у вас должна получиться примерно следующая картина (см. рисунок ниже).

Эта матрица трассировки полезна тем, что по ней легко увидеть

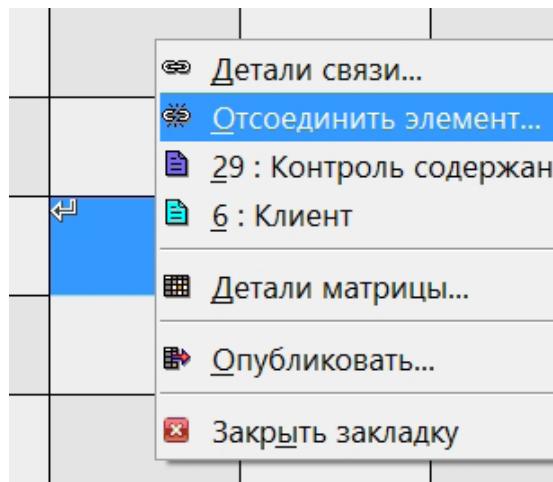
- Какие из компонентов не имеют связанных с ними функций
- Какие из компонентов перегружены
- Какие из функций не имеют ни одного реализующего их компонента

При этом, такая оценка не требует напряжения, вчитывания в текст и легко строится на основе общего взгляда на представление.

Прямо в этой матрице трассировки вы можете проводить распределение функций по компонентам, отмечать кто из ролей какую функцию выполняет. Для этого достаточно щелкнуть на пересечение функции и компонента, после

чего откроется диалог создания связи.

Если необходимо удалить созданную связь - воспользуйтесь контекстным меню ячейки



В качестве тренировки вы можете построить обратную матрицу, где наверху будут компоненты системы, а в качестве строк - функции. Попробуйте и посмотрите, какая из двух вам будет удобнее для работы.

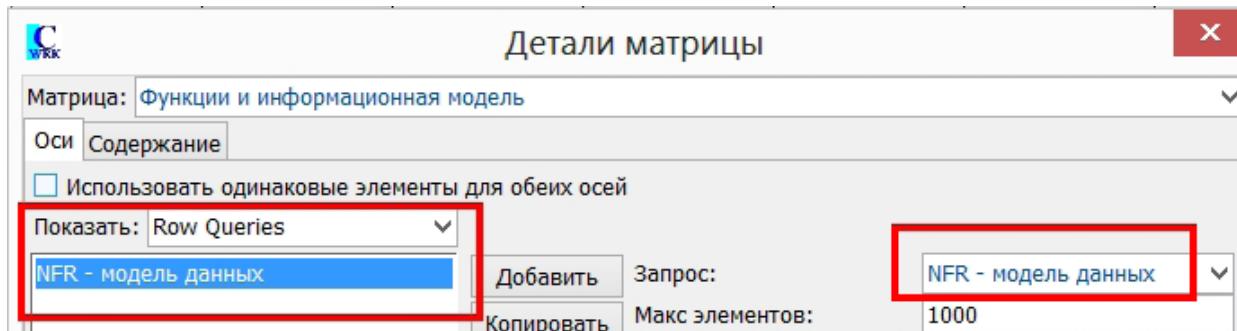
Матрица: Функции и компоненты							
Функции и компоненты		Функции и компоненты					
	Печать накладной на заказы из	Печать наклеек на пакеты	Выбрать пакеты из общего груза	Указать примечания по разделке мяса	Разделка мяса	Позвонить клиенту для согласования	Зафиксировать результаты звонка по
Фермерское хозяйство							
Водитель			↙				
Мясник				↙	↙		
Клиент				↙	↙		
Главный менеджер					↙	↙	↙
Телефон							

Вторая матрица трассировки, которая может быть нам очень полезна на данном этапе - это трассировка элементов информационной модели к функциям. Фактически это представление, которое позволит нам легко увидеть какие поля в каких функциях используются. Чтобы ее настроить:

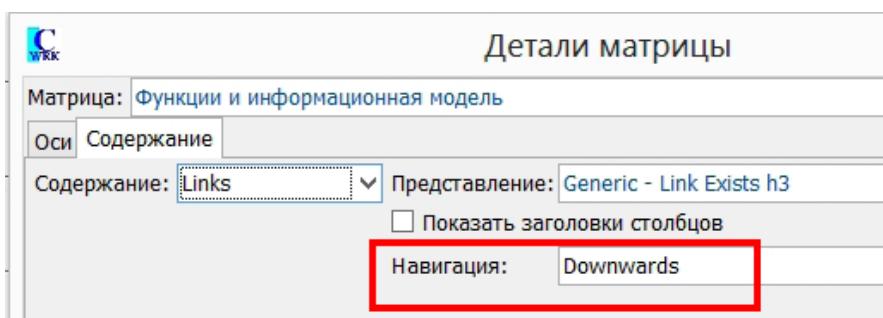
1. Откройте предыдущую матрицу - Функции и компоненты и сохраните под

именем Функции и модель данных.

2. В разделе Row Queries замените запрос SBS - Bottom-Level на запрос NFR - модель данных (который мы строили ранее)



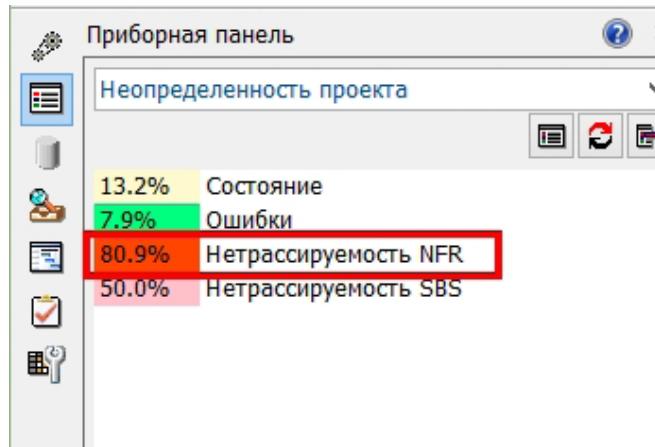
3. В разделе Содержание замените навигацию на Downwards



4. Сохраните и примените матрицу.

Матрица: Функции и информационная модель						
	Сортировка заказов	Планирование взаимодействия с фермерами	Печать накладной на заказы из	Печать наклеек на пакеты	Выбрать пакеты из общего груза	Управление
Номер накладной			↑	↑		
Общий вес пакетов						
Дата накладной						

На данном этапе матрица сильно разрежена, поскольку мы еще не прописали связи между функциями и моделью данных, о чем нам также говорит наша панель индикаторов:



При разработке системы созданные нами матрицы трассировки могут оказаться довольно большими и не всегда удобными в работе, поэтому вы можете создать другие запросы, которые будут выбирать, например, не всю информационную модель, а лишь ту часть, над которой работаете именно вы (по признаку OWNER) или по любому другому признаку.

В матрицах трассировки вы можете выводить не только информацию о наличии связей, но и данные элементов, через которые связаны колонки и строки матрицы. Например, возможно построить матрицу которая выводит элементы, посредством которых связаны строки и столбцы матрицы. Для этих элементов можно выводить все обычные поля, которые используются в представлениях, и, конечно, использовать подсветку и регулярные выражения.

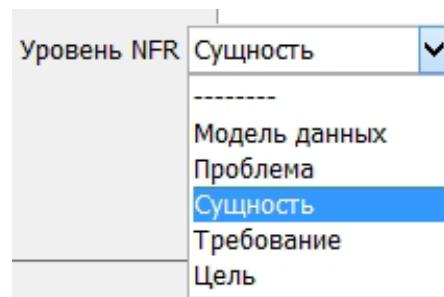
Матрица: Информационная модель и компонент				
	Накладная	Продукт заказа	Элемент заказа - Номер п/п	
Фермерское хозяйство				
Водитель	Выбрать пакеты из общего груза			
Мясник				

Комбинируя настройки, определяющие состав выборки для колонок и строк, а также то, что выводить в ячейки можно получать крайне полезные и в то же время компактные представления, например, как следующее, которое в строках отражает Сущности информационной модели системы, в столбцах - функции,

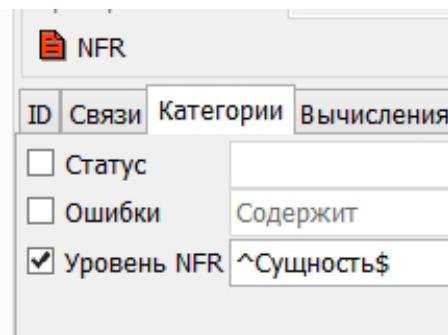
а в ячейках те конкретные поля, принадлежащие сущности и обрабатываемые функцией.

		Сортировка заказов	Планирование взаимодействия с фермерами	Печать накладной на заказы из доставки
Заказ	Район доставки заказа			Номер накладной
	Желаемая дата доставки			
	Запланированная дата доставки			
Доставка				
Справочник продуктов			Допустимые характеристики продуктов	
Фермерское хозяйство				
Настройки системы				

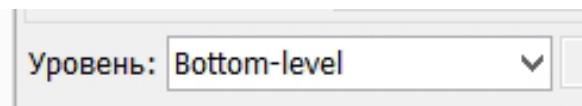
Чтобы построить эту матрицу мы ввели атрибут «Сущность» в уровне NFR и отметили этим значением все элементы информационной модели, которые являются самостоятельными сущностями, а не их полями.



Это позволило выбрать их запросом со следующей настройкой:



Для выборки функций мы использовали стандартный запрос - Function-Bottom-Level, который выводит все элементы функций нижнего уровня:



Этот запрос не надо было создавать, т.к. он относится к автоматическим - запросам, которые создаются при создании нового типа элемента.

В настройках самой матрицы мы установили следующие значения:

**Детали матрицы**

Матрица: Функции и информационная модель2

Оси Содержание

Использовать одинаковые элементы для обеих осей

Показать: Column Queries

FUNCTION - Bottom-Level

Добавить

Запрос:

Макс элементов: 1000

Представление заголовка: Generic - name h3

Копировать

Удалить

Оси Содержание

Использовать одинаковые элементы для обеих осей

Показать: Row Queries

NFR - модель данных верхний уровень

Добавить

Запрос:

Макс элементов: 1000

Представление заголовка: Generic - name h3

Копировать

Удалить

Матрица: Функции и информационная модель2

Оси Содержание

Содержание: Linked Items Тип: NFR

Представление: Generic - name hort h3

Показать заголовки столбцов

Навигация столбца: Upwards

Навигация строки: Transitive Down

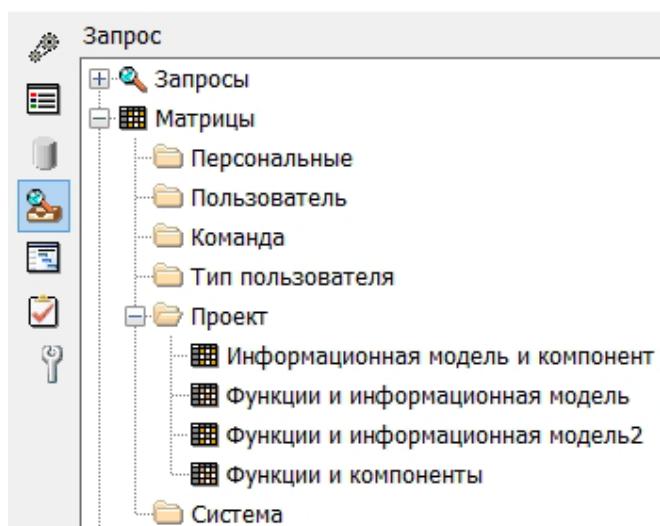
Обратите внимание на выбранные навигации, в них используется навигация Transitive Down, эта навигации позволяет выбрать все элементы, связанные не только напрямую, но и опосредованно.

Представление, используемое для ячеек, - «Generic - name short h3» выводит наименование и статус элемента.

Удобно также то, что эти матрицы являются интерактивными, вы можете щелкнуть на любой элемент ячейки и сразу же перейти к этому элементу, чтобы уточнить или отредактировать его другие атрибуты.

## Вызвать сохраненные матрицы трассировки

Для того, чтобы вызвать сохраненные ранее матрицы трассировки перейдите на панель Запросов. Так же как запросы, представления и другие настройки, матрицы распределены по областям доступа - от системных до персональных.

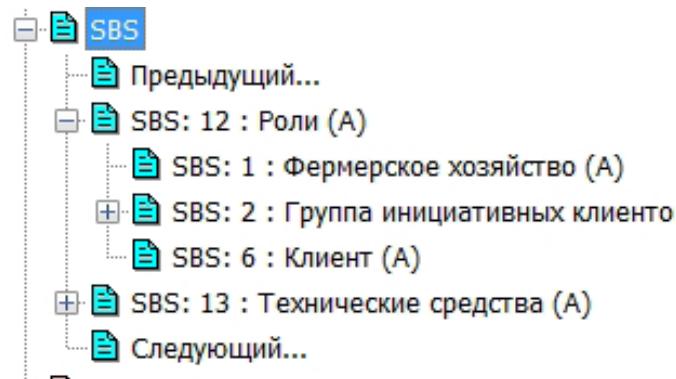


## Настроить диаграммы иерархий (HID)

Диаграммы иерархий - это отличное представление структуры требований для согласования и обсуждения «на бумаге». С ее помощью можно вывести организационную структуру ролей проектируемой АС или, например, функциональную архитектуру системы.

Диаграмму иерархий возможно построить от любого выбранного элемента. Построим ее на примере ролей АС. Для этого

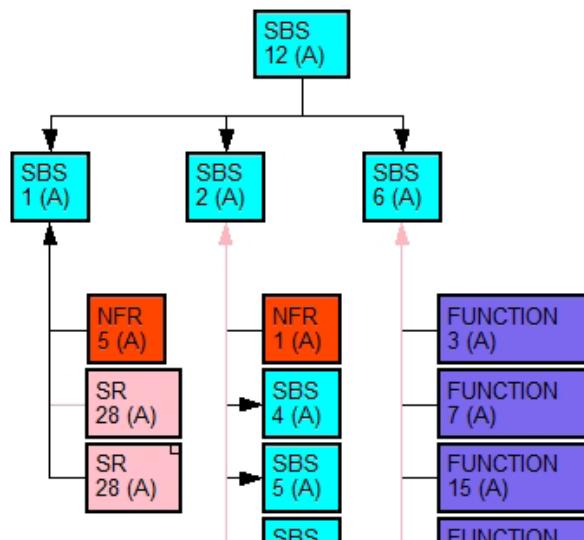
1. Сгруппировать все роли под одним корневым элементом:



2. Выделить элемент верхнего уровня, от которого необходимо строить иерархию. В данном случае - Роли.

3. В контекстном меню выбрать Связи -> Диаграмма иерархии (HID)

После этого мы получим следующее представление:



На диаграмме будут отражены все связи выбранного элемента с элементами других типов. Для организационной структуры это избыточно, поэтому ограничим вывод одним типом элементов - SBS:

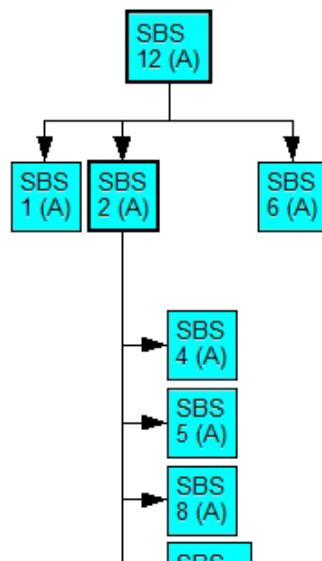
#### 4. В разделе Связи установить настройки

К типу:	SBS
Тип связи:	<any>
Направление связи:	Only Down
Владелец:	All
Экземпляр:	Latest Accessible



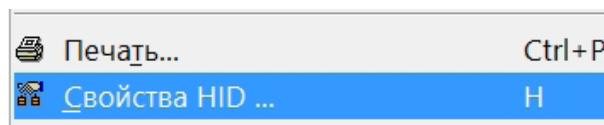
#### 5. Обновить HID, нажав кнопку

Вы должны получить следующий результат:

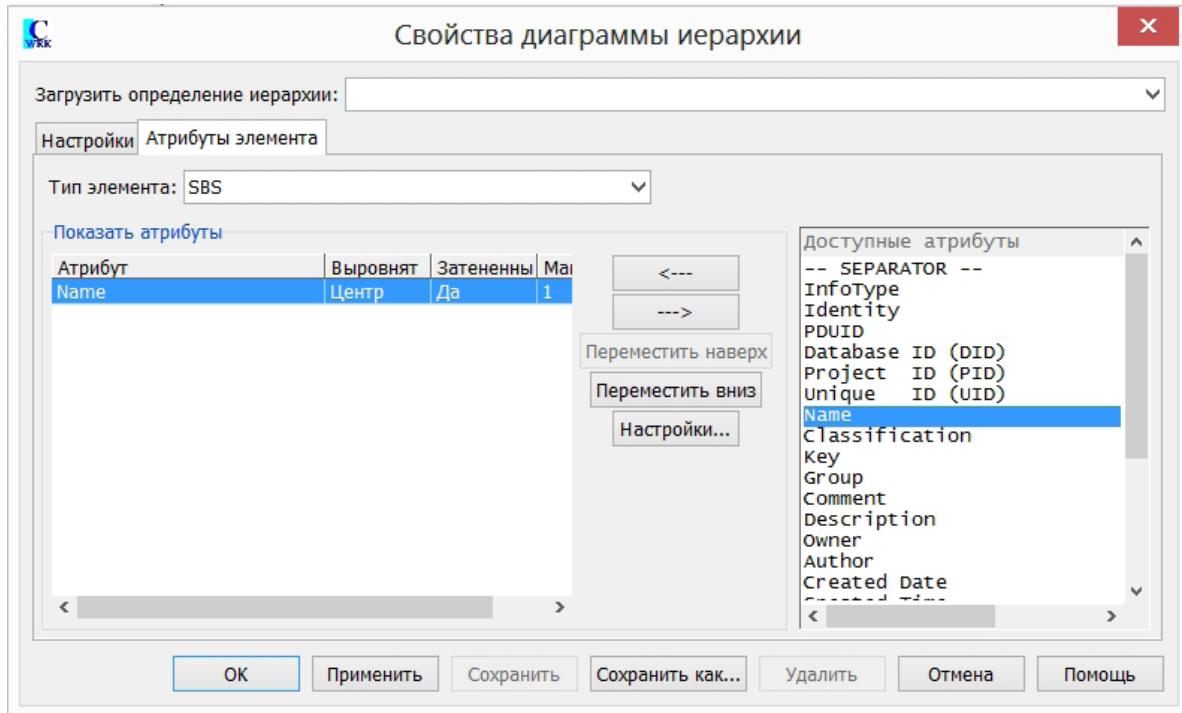


Теперь заменим идентификаторы, которые нам пока ни о чем не говорят, на поле Name. Для этого

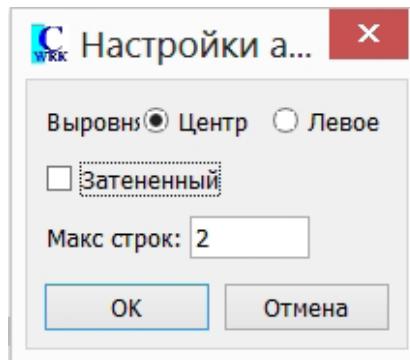
#### 6. Открыть контекстное меню Свойства HID



7. Выбрать Тип элемента = SBS
8. Удалить все исходные поля, заменив их на поле Name



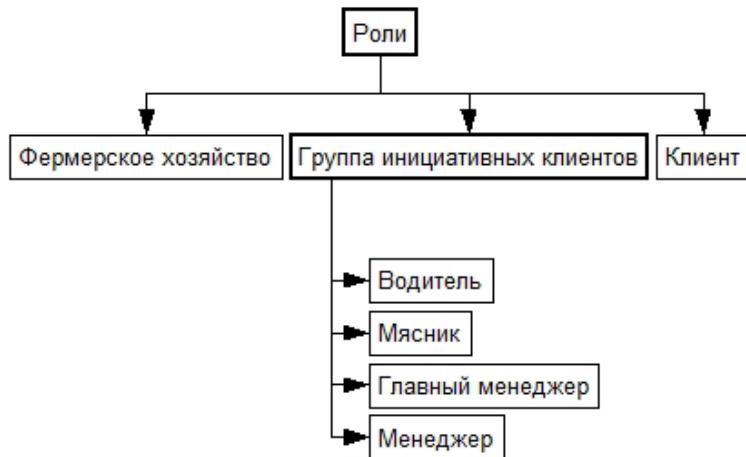
9. Нажать кнопку Настройки, убрать галку с поля Затененный и увеличить до двух максимальное число строк.



10. Сохранить настройки под именем Простая иерархия SBS и применить представление

Результат должен быть аналогичным тому, что представлен на изображении ниже.

Диаграммы иерархий имеют массу настроек, попробуйте изменять их значения и посмотрите на результат.



Особенно важной является возможность «Следовать заданной иерархии», которая позволяет очень точно указать, какие элементы по каким связям выводить на каждом уровне иерархии.

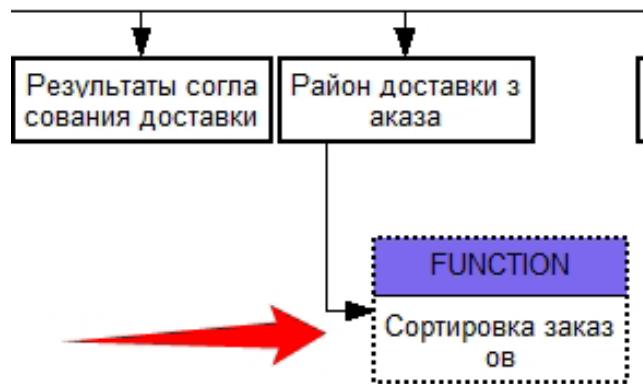
Диаграммы иерархий являются интерактивными в части и элементов и связей. Вы можете

- щелкнуть на любое элемент или связь и получить доступ ко всем их атрибутам.
- вы можете закрыть или открыть отдельную ветку (Раскрыть/Свернуть в контекстном меню выделенного элемента)

Если полученная диаграмма оказалась слишком крупной и вы хотите исследовать лишь какую-то ее часть, то воспользуйтесь функцией контекстного меню элемента:

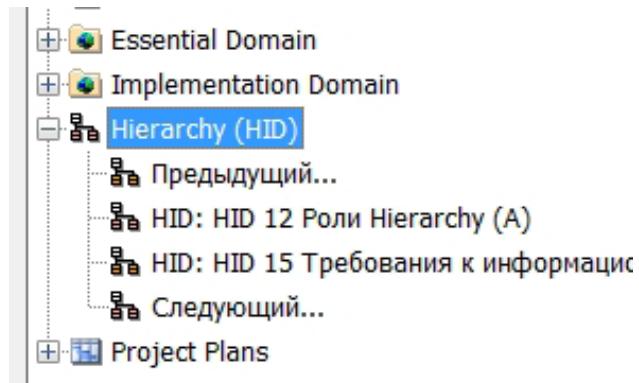
Перерисовать как источник

При включенном режиме отображения элементов, подозреваемых на изменение, такие элементы отображаются на HID с помощью пунктирных линий:



После того, как вы получили желаемый вид HID сохраните ее или напечатайте в RTE.

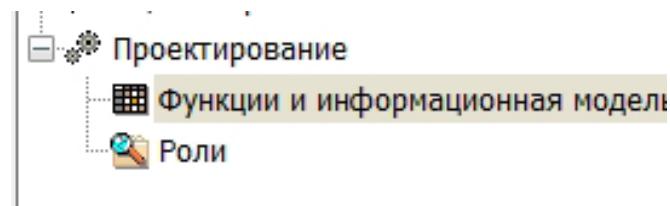
Получить доступ к сохраненным HID можно, например, из панели База данных:



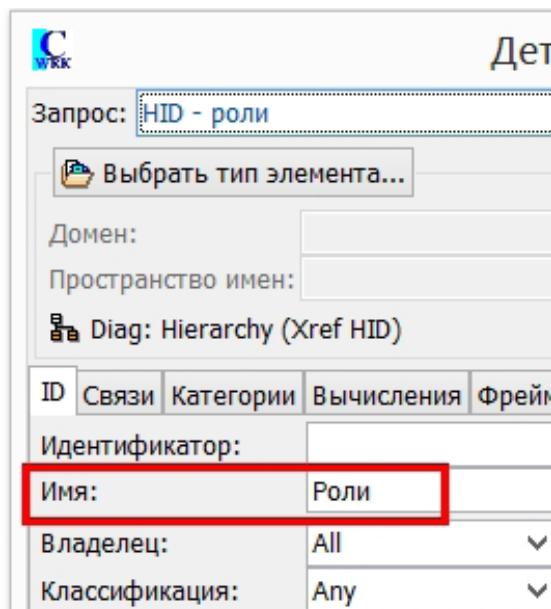
## Дополнить панель фаз или стартовую страницу

Если вы создали представления, которые планируете часто использовать, то вынесите их на панель фаз - для всей команды или на стартовую страницу - для себя лично.

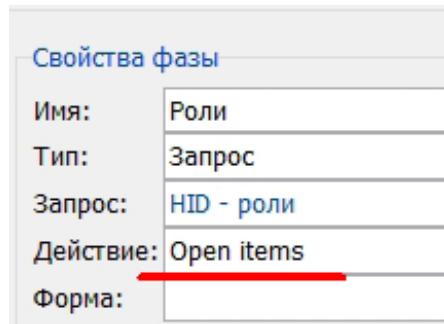
Из тех представлений, что мы настроили в последнем разделе удобно будет иметь под рукой матрицу трассировки функций и информационной модели (в ее компактном последнем представлении), а также HID-диаграмму, отражающую иерархию ролей в системе.



Чтобы диаграмма иерархии открывалась по щелчку с панели Фаз необходимо построить запрос, результатом которого будет нужная вам диаграмма



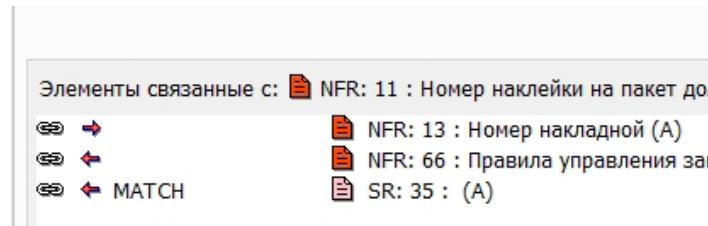
и в настройках фазы указать вызов этого запроса в режиме Открыть элементы (Open items):



## Быстро посмотреть все связанные элементы

Что необходимо быстро понять, с какими элементами связан данный или, например, найти его источник в исходном документе с требованиями. Это можно сделать без построения запросов и представлений несколькими способами.

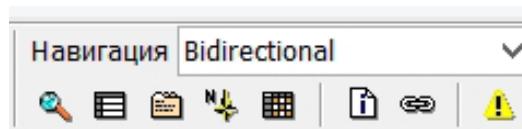
Способ 1. Выделить элемент и в контекстном меню выбрать - Связи -> Связанные элементы все. После чего откроется диалоговое окно, в котором отразятся все связанные элементы



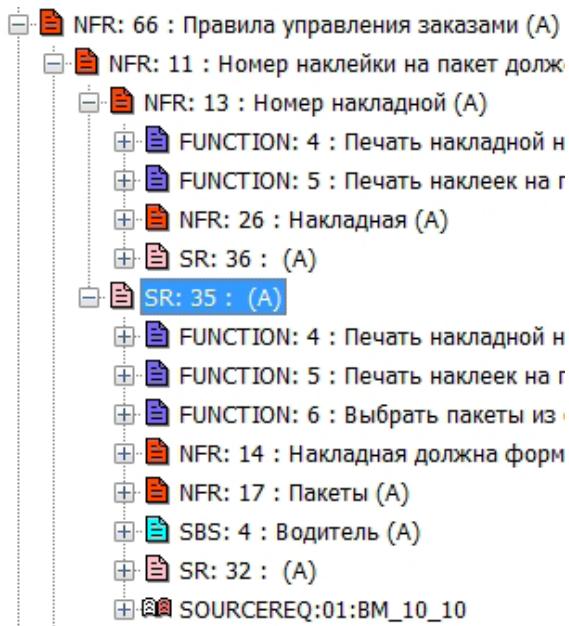
вы можете щелкнуть на любой элемент и он откроется в форме.

Этот вариант работает в любом представлении - будь то дерево, открытая форма элемента или элемент на HID диаграмме.

Способ 2. Если у вас открыто древовидное представление элементов, то вы можете установить навигацию Bidirectional



чтобы проследить связи со всеми элементами во всех направлениях. Иногда это удобно, чтобы проверить свою догадку у связанных элементов или быстро найти необходимый элемент с той точки, в которой вы работаете.



## Повторно использовать требования из другого проекта

Это одна из наиболее важных задач, позволяющих здорово сэкономить на проектировании. Один из возможных способов повторного использования требований из других проектов - это экспорт/импорт.

Обязательная практика - выполнять резервное копирование проекта перед любым импортом. Для этого достаточно просто скопировать папку проекта в раздел резервного копирования.

Импорт требований из другого проекта значительно упрощается, если оба проекту имеют одинаковую схему.

## Разработать модели

Существует подход, именуемый MBSE, - Model Based Systems Engineering. Это подход к проектированию систем, который предполагает, что проектирование выполняется с использованием формальных моделей на протяжении всего жизненного цикла системы - от сбора требований до модернизации.

Формальные модели могут иметь или не иметь визуальную нотацию. Однако чаще всего MBSE ассоциируется именно с визуальным моделированием.

На практике применение визуального моделирования в процентном соотношении к объему проектов весьма не велико т.к. разработка визуальных моделей довольно дорогое занятие.

Вся та формализация, которую мы выполняли ранее также относится к модель-ориентированному подходу, однако требует меньше накладных расходов.

В то же время, эта формализация отражает в большей степени структурный аспект системы, нежели ее динамику. Поэтому для проработки особенно важных разделов могут понадобиться и визуальные нотации, такие как eFFBD, Activity Diagram, Sequence Diagram и другие. К счастью, Cradle позволяет не только создавать визуальные модели, но и связывать их с элементами требований, не имеющих графического отображения, что позволяет поддерживать целостность всей системы.

## Понять принципы моделирования в Cradle

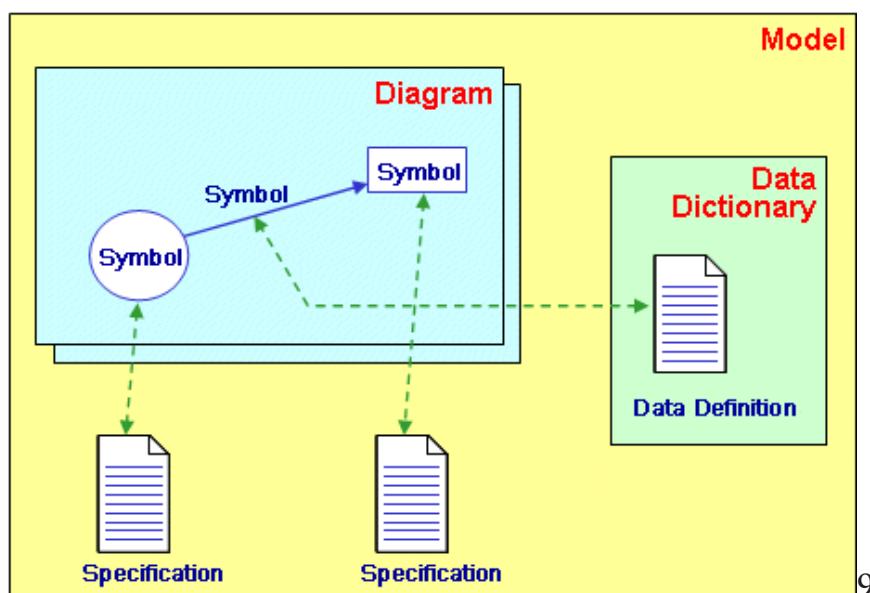
Каждая база данных Cradle (проект) предоставляет две области моделирования, которые называются соответственно:

- Essential — для анализа и моделирования, независимого от реализации.
- Implementation — для анализа и моделирования, зависимого от реализации.

Каждая из этих областей может содержать любое количество моделей, которые могут быть организованы в одну или несколько иерархий. Каждая модель — это набор связанных диаграмм, созданных с определенной целью. При этом модель может включать диаграммы, созданные в разных нотациях (например,

Process Flow , Function Flow Block или Use Case).

Диаграммы состоят из символов. Каждый символ, кроме имени и визуального представления, имеет определение (definition), которой содержит его полное описание. Такие описания хранятся в виде отдельных элементов в БД Cradle — спецификаций (specification) и определений данных (data definition). Спецификации индивидуальны для каждой диаграммы, в то время как определения данных являются общими для всех диаграмм в рамках модели. Определения данных также называются сущностями Словаря Данных (Data Dictionary entries), кратко — DD entries и все вместе называются Словарем Данных (Data Dictionary).

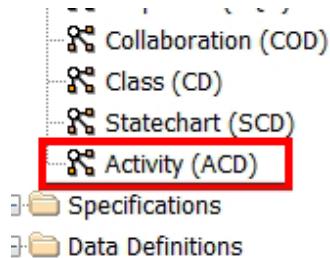


9

Не вдаваясь глубоко в принципы моделирования, заложенные в 3SL Cradle, так как эта тема требует отдельной книги, посмотрим как создать модели и связать их с текстовыми требованиями, созданными нами ранее.

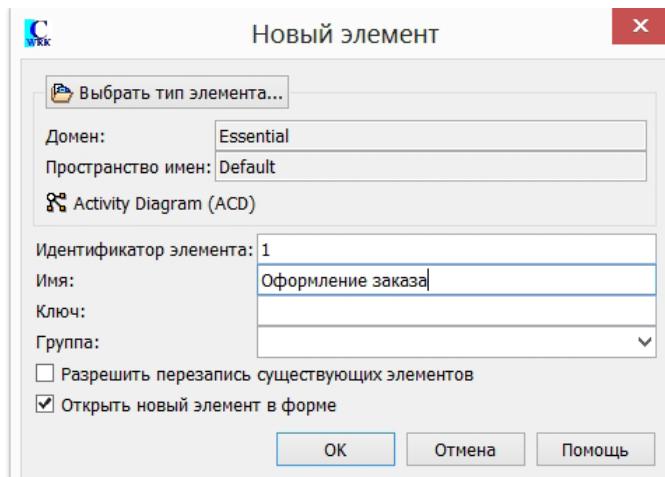
### Разработать Activity Diagram

1. Перейти в раздел Базы данных Essential Domain ->Default->Diagrams
2. Выбрать тип диаграммы Activity (ACD)

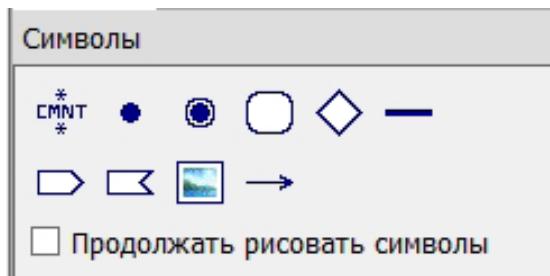


3. В контекстном меню выбрать Новый->Элемент

4. Ввести идентификатор и наименование диаграммы



5. Создавать символы, выбирая их на панели инструментов и щелкая по рабочей области:



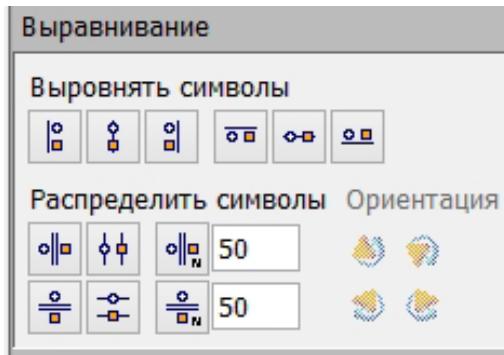
Стрелки создаются методом двух кликов - первый на исходный объект, второй - на конечный.

Чтобы добавить изгиб стрелки при ее создании необходимо делать промежуточные клики, а после создания - щелкнуть в место добавления точки, нажав SHIFT.

Чтобы удалить промежуточную точку на стрелке необходимо нажать SHIFT и кликнуть в эту точку. Эти же операции можно вызвать из контекстного меню

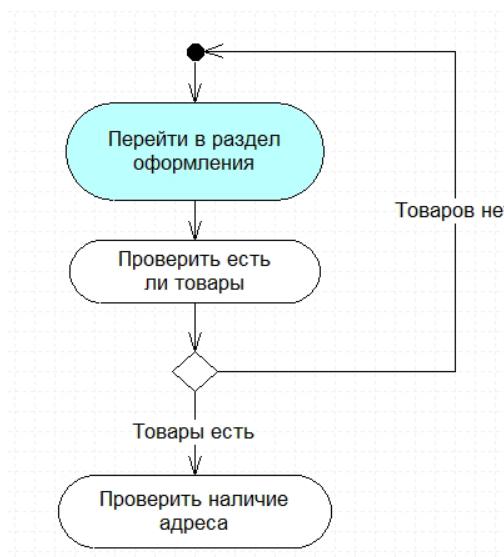
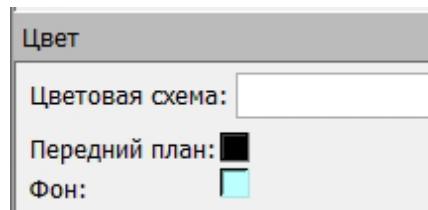
Точка излома.

Выделив группу символов, вы можете применить к ним операции выравнивания и равномерного распределения:



Если вы случайно установили не тот символ, то вы можете его заменить на другой, прямо в том месте, где он находится, не удаляя и не создавая новый, для этого воспользуйтесь контекстным меню символа Заменить.

Вы можете установить цвета символов с помощью панели инструментов Цвет

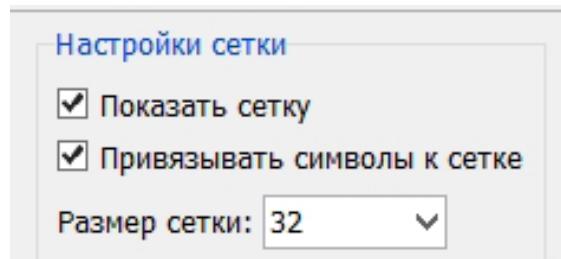


Чтобы сэкономить время на выравнивании диаграмм включите сетку, для этого

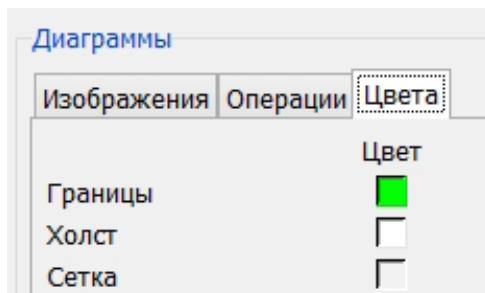
1. Вызвать меню Настройки и перейти в раздел Диаграммы



2. Включить настройки сетки



3. На вкладке Цвета откорректировать цвет сетки



Мы рекомендуем использовать для сетки светло-серый цвет.

При переименовании символов, спецификаций и определений данных необходимо помнить, что связь символа с определением данных осуществляется по имени, а связь со спецификацией - по номеру. Таким образом, изменение имени определения данных (например, потока) приведет к отсоединению этого элемента от своего символа и от диаграммы, аналогично изменение номера (идентификатора) спецификации приведет к отсоединению от диаграммы.

Чтобы переименовать символ

1. Выделить символ и нажать правую кнопку мыши.
2. В контекстном меню выбрать Переименовать.

Обратите внимание, что при переименовании символа определения данных, например, потока, и сохранении соответствующей диаграммы для этого символа создается новый поток, при этом поток со старым именем отсоединяется от

диаграммы, но не удаляется (т.к. он может быть использован в другом месте).

Для того, чтобы переименовать поток необходимо выбрать в его контекстном меню (правая кнопка мыши) Еще->Переименовать

При переименовании потока он отсоединяется от своего старого символа (т.к. связь осуществляется по имени)

При этом, если символ остался без потока, то при следующем сохранении диаграммы поток для данного символа будет создан.

Связь между функциями, диаграммами и соответствующими символами осуществляется не по названию, а по номеру, поэтому переименование символа/ спецификации приводит к автоматическому переименованию спецификации/ символа, при этом связи не изменяются (в случае спецификаций связи изменяет перенумерация)

## **СПЕЦИАЛЬНЫЕ ТЕМЫ**

# Использование горячих клавиш

## Редактирование информации

<b>Tab</b>	- переход к следующему полю в форме
<b>Shift-Tab</b>	- переход к предыдущему полю в форме
<b>Home</b>	- переход к начальному полю / параграфу
<b>End</b>	- переход к последнему полю / параграфу
<b>Cursor keys</b>	- переход влево, вправо, вверх и вниз (где применим) внутри полей или текста
<b>Shift-cursor</b>	- начало или расширение выделения
<b>Shift-ctrl-cursor</b>	- начало или расширение выделение по слову (для лево/право) или построчно / по параграфу (для вверх/вниз)
<b>Ctrl-A</b>	- выбрать все
<b>Ctrl-C</b>	- скопировать
<b>Ctrl-S</b>	- сохранить
<b>Ctrl-V</b>	- вставить
<b>Ctrl-W</b>	- закрыть панель с формой
<b>Ctrl-X</b>	- вырезать
<b>Delete</b>	- удалить
<b>Ctrl-Z/Ctrl-Y</b>	- отменить, повторить. Есть два стека за каждой ячейкой в представлении и также каждым полем в каждой форме

## Навигация по основному меню

<b>Alt-F</b>	- открыть меню Файл
<b>Alt-E</b>	- открыть меню Редактировать
<b>Alt-V</b>	- открыть меню Представление
<b>Alt-I</b>	- открыть меню Элемент (при выбранном элементе)
<b>Alt-S</b>	- открыть меню Символ, в ходе редактирования диаграммы
<b>Alt-T</b>	- открыть меню Инструменты
<b>Alt-A</b>	- открыть меню Администрирование
<b>Alt-W</b>	- открыть меню Окно
<b>Alt-H</b>	- открыть меню Помощь
<b>Cursor down</b>	- перейти к следующему пункту меню
<b>Cursor up</b>	- перейти к предыдущему пункту меню
<b>Cursor right</b>	- открыть каскадное меню для текущего пункта
<b>Cursor left</b>	- закрыть каскадное меню
<b>Enter</b>	- выбрать текущий пункт

## Другие привязки

<b>Ctrl-A</b>	- выделить все элементы в представлении
<b>Ctrl-D</b>	- отменить выделение всех элементов в текущем представлении
<b>Ctrl-F</b>	- найти элементы с заданным текстом в представлении
<b>Ctrl-I</b>	- инвертировать текущее выделение

<b>Ctrl-O</b>	- открыть элемент
<b>Ctrl+</b>	- увеличить (только для графических представлений)
<b>Ctrl--</b>	- уменьшить (только для графических представлений)
<b>Ctrl-&lt;</b>	- создать связь от текущего элемента
<b>Ctrl-&gt;</b>	- создать связь к текущему элементу
<b>Ctrl-1</b>	- изменить стиль представления на Список
<b>Ctrl-2</b>	- изменить стиль представления на Таблица
<b>Ctrl-3</b>	- изменить стиль представления на Документ
<b>Ctrl-4</b>	- изменить стиль представления на Дерево
<b>F1</b>	- открыть справку
<b>Ctrl-F4</b>	- открыть диалог Оповещения
<b>Ctrl-F5</b>	- открыть диалог Информации об элементе для текущего элемента
<b>Ctrl-F7</b>	- открыть диалог Детали связи для связи текущего элемента
<b>Ctrl-F8</b>	- открыть диалог Детали запроса
<b>Ctrl-F9</b>	- открыть диалог Детали представления
<b>Ctrl-F10</b>	- открыть диалог Детали формы
<b>Ctrl-F11</b>	- открыть диалог Детали навигации
<b>Ctrl-F12</b>	- открыть диалог Детали матрицы

# Разработка Use Cases

3SL Cradle предоставляет возможность выбора технологии разработки Use Cases удобной для конкретного коллектива. Ниже рассматриваются основные технологические подходы к организации разработки вариантов использования в 3SL Cradle.

## Немного о Use Cases.

Алистер Коберн «Современные методы описания функциональных требований к системам»:

*Вариант использования фиксирует соглашение между участниками системы о ее поведении. Вариант использования описывает поведение системы при ее ответах на вопрос одного из участников, называемого основным действующим лицом.*

*Варианты использования представлены большей частью в текстовой форме. Вариант использования, как форма описания, стимулирует обсуждение проектируемой системы в группе разработчиков.*

Структура описания варианта использования может быть адаптирована к задачам конкретного коллектива, но как правило фиксируются следующие разделы:

- Основное действующее лицо
- Область действия
- Участники и интересы
- Предусловия
- Минимальные гарантии
- Гарантия успеха
- Основной сценарий
- Расширения

## Технологии разработки Use Cases в 3SL Cradle

### 1. Способ первый. Один элемент — один вариант использования — одно поле.

Основная единица управления в базе данных Cradle — элемент (item). Вы можете определить любой набор типов элементов, например, требования, тесты, функции, классы, оборудование... Также мы можем определить тип элемента Use Case и в поле ТЕХТ данного элемента вводить полное описание варианта использования.

- *Преимущества подхода:* простота настройки, легкость сору/past из документов, обеспечение трассируемости варианта использования в целом к другим элементам, в том числе к другим вариантам использования, изменение структуры Use Case не влияет на технологию.
- *Недостатки подхода:* сложность анализа и сравнения объемных вариантов использования.

### 2. Способ второй. Один элемент — один вариант использования — каждому разделу варианта использования свое поле.

Данный способ отличается от предыдущего тем, что для элемента типа Use Case настраиваются поля в соответствии с выбранной структурой Use Case. Один раздел — одно поле. Таким образом, будут настроены поля: основное действующее лицо, область действия, участники и интересы и др.

- *Преимущества подхода:* легче анализировать варианты использования, сравнивать разделы разных вариантов использования, обеспечение трассируемости варианта использования в целом к другим элементам, в том числе к другим вариантам использования.
- *Недостатки подхода:* при изменении структуры Use Case необходимо будет добавить поля (легко) и откорректировать созданные представления (не сложно, визуальный редактор), пораздельное копирование из документов (при необходимости перевода ввода из документов).

The screenshot shows the WinCradle WorkBench interface. On the left, there is a tree view of 'Фазы' (Phases) including 'Требования с процессом изготовления' (Requirements with manufacturing process), 'Определение структуры' (Definition of structure), 'Структура узла' (Structure of assembly), 'Детали узла' (Assembly parts), 'Разработка определений данных' (Development of data definitions), 'Определения данных (потоки)' (Data flow definitions), 'Спецификации (функции)' (Function specifications), 'Проверка диаграмм' (Diagram verification), and 'Формирование отчета' (Report generation). The main area shows a Use Case diagram with a central node 'Spec - Essential Function - Yours' and several associated nodes. Below the diagram is a table titled 'Вопрос: Spec - Essential Function - Yours' with columns: Identity, Name, TEXT, MAIN FLOW, PRE CONDITIONS, POST CONDITIONS, ALTERNATE FLOWS, and EXCEPTION F. The table contains 15 rows of requirements, each with a unique ID (1-15), a description, and various conditions and flows.

Identity	Name	TEXT	MAIN FLOW	PRE CONDITIONS	POST CONDITIONS	ALTERNATE FLOWS	EXCEPTION F
Предыдущий...							
1	3 Требования к процессу построения макета робота	Цель: проверить полноту макета Точка зрения: проектировщик макета					
2	3.1 Построить макет робота	Построить макет однозвездного робота с платформой	Подготовить платформу макета, собрать руку манипулятора и соединить приводной вал электродвигателя с валом редуктора. Закрепить	Имеются в наличии детали для	Построен макет робота	нет	В случае обнаружения
3	3.1.1 Подготовить платформу	Подготовить платформу для крепления манипулятора	Соединить приводной вал электродвигателя с валом редуктора. Закрепить	Имеются в наличии следующие детали: корпус платформы, корпус платформы	Подготовлена платформа макета робота	нет	В случае обнаружения механических
4	3.1.1.1 Соединить электродвигатель с редуктором	Соединить электродвигатель с редуктором	Соединить приводной вал электродвигателя с валом редуктора	Имеется в наличии электродвигатель и редуктор	Редуктор соединен с электродвигателем	нет	В случае обнаружения механических
5	3.1.1.2 Установить сборку 3.5-3.6 на корпус платформы	Установить электродвигатель на корпус платформы	Закрепить электродвигатель с редуктором на корпусе платформы	Имеется в наличии корпус платформы макета и сборка	Электродвигатель с редуктором закреплен на корпусе платформы	нет	В случае обнаружения механических
6	3.1.1.3 Подключить к двигателю управляющую плату	Подключить к двигателю управляющую плату	Подключить управляющую плату к управляющему входу двигателя	Имеется в наличии управляющая плата, подключена к управляющему входу	Управляющая плата	нет	В случае обнаружения механических
7	3.1.2 Собрать руку манипулятора	Произвести сборку руки манипулятора из одного звена с шарнирами	Собрать звено руки манипулятора с шарнирами и установить датчик положения на	Имеются в наличии детали для руки манипулятора и	Собрана рука робота	нет	В случае обнаружения механических
8	3.1.2.1 Установить звено руки с шарнирами	Присоединить к звено руки робота шарниры крепления	Установить звено руки манипулятора шарниры крепления (для крепления звена	Имеются в наличии звено руки робота, шарниры крепления и	Установлены шарниры крепления на звено руки робота	нет	В случае обнаружения механических
9	3.1.2.2 Установить датчик положения на звено руки робота с шарнирами	Установить датчик положения на звено руки робота с шарнирами	Установить звено руки манипулятора с шарнирами и звено датчик положения на звено руки робота с шарнирами	Имеется в наличии звено руки робота с шарнирами	Установлен датчик положения на звено руки робота	нет	В случае обнаружения механических
10	3.1.2.3 Установить электродвигатель на шарнир устройства захвата	Установить электродвигатель на шарнир устройства захвата	Установить звено руки манипулятора с шарнирами и звено датчик положения на звено руки робота с шарнирами	Имеется в наличии звено руки робота с шарнирами	Установлен датчик положения на звено руки робота	нет	В случае обнаружения механических
11	3.1.3 Подготовить устройство захвата	Подготовить устройство захвата для подключения к руке манипулятора	Подготовить устройство захвата для подключения к руке манипулятора	Имеется в наличии устройство захвата	Подготовлено устройство захвата	нет	В случае обнаружения механических
12	3.1.3.1 Установить пневмоцилиндр на устройство захвата	Установить пневмоцилиндр на устройство захвата	Установить пневмоцилиндр для работы привода пальцев на устройство захвата	Имеется в наличии пневмоцилиндр и устройство захвата	Установлен пневмоцилиндр на устройство захвата	нет	В случае обнаружения механических
13	3.1.3.2 Подключить пневмоцилиндр к сборке 3.4-3.7	Подключить пневмоцилиндр к устройству захвата	Подключить пневмоцилиндр к пневмоцилинду привода пальцев устройства захвата	Имеется в наличии сборка из пневмоцилиндра и	Подключен пневмоцилиндр к пневмоцилинду	нет	В случае обнаружения механических
14	3.1.3.3 Установить датчик захвата на звено руки	Установить датчик захвата на звено руки	Установить датчик захвата на звено руки	Имеется в наличии сборка из пневмоцилиндра	Установлен датчик захвата	нет	В случае обнаружения механических
15	3.1.4 Собрать манипулятор	Собрать манипулятор	Соединить устройство захвата с	Имеется в наличии	Собран манипулятор	нет	В случае

### 3. Способ третий. Один вариант использования — иерархия элементов с атрибутом, отражающим тип раздела варианта использования.

Данный способ предполагает занесение каждого раздела Use Case в отдельный элемент (item), при этом можно ввести атрибут — тип раздела, что упростит анализ и выборки. Каждый вариант использования будет выглядеть как иерархия элементов, где элемент верхнего уровня соответствует данному варианту использования в целом, а подчиненные элементы — его разделам.

- Преимущества подхода:* возможна большая детализация при трассировке — трассировать не к Use Cases целиком, а только к одному из разделов, повторное использование отдельных разделов, что может значительно сократить время разработки всего набора вариантов использования, пораздельное управление правами доступа, построение пораздельных выборок, например, выбрать всех действующих лиц и т.п.
- Недостатки подхода:* необходимость создавать и поддерживать связи не только между отдельными Use Cases, но и связи между элементами разделов (не сложно при технологии drag&drop).

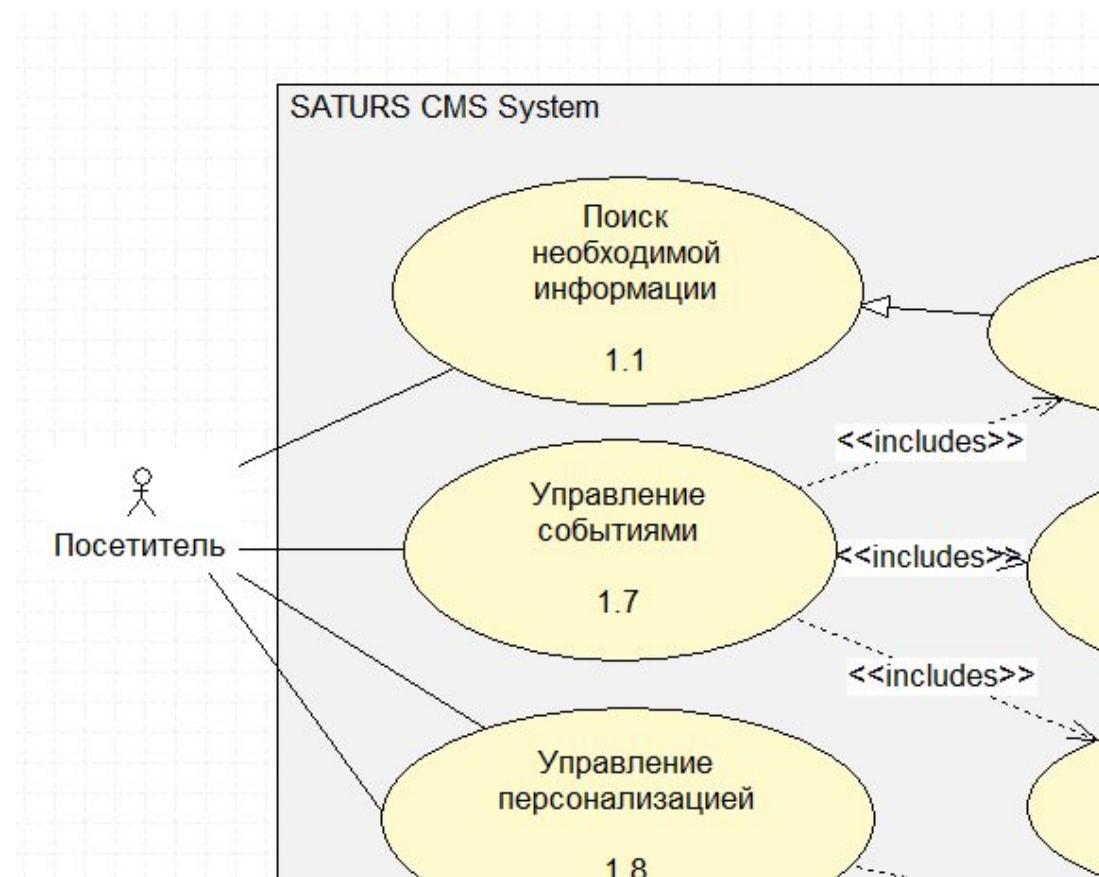
#### 4. Способ четвертый. Каждому разделу варианта использования — свой тип элементов.

Данный способ предполагает, что для каждого раздела варианта использования создаются соответствующие типы элементов. Один из вариантов соответствующей модели трассировки описан в статье «Зоопарк трассировки», в разделе Альбом моделей трассировки требований.

- *Преимущества подхода:* дополнительно к предыдущему — возможность использовать автоматически созданные для каждого типа запросы и представления.
- *Недостатки подхода:* аналогично предыдущему.

Практика показывает, что выбирать стоит из второй, третьей и четвертой технологии разработки Use Cases. Окончательный выбор зависит от общей технологии работы, модели трассировки и задач, которые решаются с помощью трассировки требований.

#### 5. Способ пятый. Вы можете использовать диаграммы Use Cases для графического представления вариантов использования.



## *Смена парадигмы*

Переход к разработке требований с использованием Cradle аналогичен переходу от двумерного рисования чертежей к трехмерному моделированию и автоматическому получению необходимых плоских срезов. Это смена парадигмы.

Если вчера мы рисовали плоские эскизы от руки, а сегодня делаем тоже самое в Автокаде, это не требует перестройки мышления. Также, если вчера мы делали заметки на полях, а сегодня быстро орудуем копипастом в Ворд, мы не меняем подход к решению задачи. Иное дело — переход к объемному моделированию, как геометрических форм, так и системных требований — когда до разговора о формах и пространственных связях еще не дошли и связи носят причинно-следственный окрас. Это совершенно другой подход.

Но действительно ли он новый для нас? Нет. Рисуя плоские чертежи, мы все равно вертим трехмерный образ детали в голове. Анализируя требования, мы все равно идем по ассоциативным связям. Поэтому суть лишь в том, что теперь инструменты ближе к поддержке естественного процесса мышления. И смена парадигмы лишь в том, чтобы дать мозгу возможность работать наиболее естественным для него образом, а не так как его научили в ограниченных условиях. А это иногда оказывается гораздо сложнее, чем еще больше ограничит свой мозг...