

**Санкт-Петербургский государственный университет  
Математико-механический факультет**

Кафедра параллельных алгоритмов

**Апробация метода DocLine разработки технической  
документации на промышленном проекте**

Дипломная работа студентки 543 группы  
Дроздовой Татьяны Валерьевны

Научный руководитель  
к.ф.-м.н., доцент  
Д.В. Кознов

Рецензент  
ст. преп.  
К.Ю. Романовский

“Допустить к защите”  
заведующий кафедрой,  
д.ф.-м.н., профессор  
Ю. К. Демьянович

Санкт-Петербург  
2008

# Оглавление

1. Введение .....	3
2. Постановка задачи .....	5
3. Обзор литературы .....	6
3.1. Семейства программных продуктов .....	6
3.2. Метод DocLine и язык DRL .....	7
3.2.1. Процесс .....	7
3.2.2. Обзор графической нотации DRL .....	7
3.2.3. Обзор текстовой нотации DRL .....	9
3.3. Технология DocBook .....	13
3.3.1. Введение .....	13
3.3.2. Особенности технологии .....	14
3.4. Выводы .....	14
4. Перевод промышленной документации в DocLine .....	15
4.1. Краткое описание промышленного проекта .....	15
4.2. Описание документации .....	16
4.3. Структура документации в DocLine .....	17
4.4. Анализ вариативности .....	20
4.4.1. Примеры удалений .....	22
4.4.2. Примеры изменений .....	23
4.4.3. Локальные изменения .....	25
4.4.4. «Параллельные места» .....	26
4.5. Об использовании DRL/PR .....	26
4.6. Возникшие проблемы генерации документации .....	27
5. Предложения по улучшению технологии DocLine .....	29
6. Заключение .....	31
7. Список литературы .....	32
Приложение 1. Фрагмент документации на языке DRL/PR .....	33
Приложение 2. Пример №1 документации в формате PDF .....	39
Приложение 3. Пример №2 документации в формате PDF .....	40

## 1. Введение

В любом IT-проекте возникает необходимость написания документации. Зачастую, объем документации оказывается весьма значительным, а требования к качеству – высокими. При этом требуется своевременно вносить изменения. Необходимы также различные представления документации – справочные системы, печатные документы, интернет-документация и т.д. Таким образом, разработка и поддержка документации оказываются непростой задачей.

Разработка семейств программных продуктов (product line approach) – популярный подход к разработке ПО, позволяющий эффективно переиспользовать различные активы разработки, такие как программные компоненты, модели, требования и т.д. Семейство программных продуктов – это набор программных систем, имеющих общий набор функциональности, удовлетворяющих нуждам конкретного сегмента рынка или конкретной цели и разрабатываемых установленным образом из общего набора повторно используемых активов [1]. Техническая документация различных продуктов имеет много общего, однако, в большинстве случаев, ее разработка для нового продукта семейства ведется либо «с нуля», что очень трудоемко, либо методом «сору-paste», что порождает проблемы при эволюции документации. Например, возникает необходимость в изменении версии продукта во всей документации; добавлении\удалении описания некоторых свойств нового продукта, также более детальная работа с текстом: добавление\удаление пунктов в списках и таблицах.

Существуют различные технологии для разработки технической документации, до некоторой степени поддерживающие повторное использование – DocBook [2], DITA [3], FrameMaker [4]. Однако они имеют ряд недостатков: не поддерживают визуальное проектирование структуры пакетов документации, а это необходимо, так как повторно используемые компоненты могут быть многочисленны и связаны друг с другом значительным количеством связей. Также они не учитывают того факта, что повторно используемые фрагменты зачастую требуют адаптации к использованию в различных документах. А именно, возникает потребность переиспользовать не только крупные блоки текста – главы, параграфы, – но и мелкие – слова, словосочетания. Это необходимо, так как переиспользуемый текст в каждой конкретной документации конечного продукта нуждается в адаптации к контексту использования и называется вариативностью. Таким образом известные подходы к созданию технической документации не предоставляют достаточной гибкости в этом вопросе и основным способом реализации вариативности в них является условное включение блока текста без адаптации к контексту.

В работах [5, 6, 7] представлен метод разработки документации семейств программных продуктов DocLine. Метод включает в себя процесс разработки документации, язык DRL (Document Reuse Language), имеющий графическую (DRL/GR – DRL Graphic Representation) и текстовую (DRL/PR –

DRL Phrase Representation) нотации, и архитектуру инструментальных средств. В отличие от вышеупомянутых подходов, метод DocLine обеспечивает как «крупноблочное» переиспользование фрагментов текста (включение условных блоков, присваивание псевдопеременных и точки расширения), так и «мелкозернистую» адаптацию фрагментов (переиспользование и адаптация к контексту отдельных слов или фрагментов предложений с помощью словарей и каталогов). Кроме того, в языке DRL/PR для полиграфической<sup>1</sup> разметки документации используется формат DocBook, что позволяет разделить содержание документации и форматирование текста и, в дальнейшем, с помощью готовых средств DocBook, позволяет получить пакет документов в различных форматах (PDF, HTML, HTMLHelp и т.д.).

Технология DocLine включает в себя набор инструментальных средств, реализованных в виде модулей расширения среды разработки Eclipse [8]. Графический редактор модели документации обеспечивает создание и редактирование модели документации. Редактор текстов документации обеспечивает создание и редактирование текстов документации. Есть возможность распространения изменений в модели на документацию и наоборот. Документация и модель хранятся на диске в виде XML-файлов, а сборка документов осуществляется в соответствии с параметрами в специализированном информационном продукте и, с помощью инструментальных средств DocBook, получаются итоговые документы в различных форматах.

Данная работа посвящена апробации метода DocLine. Этот метод является новым и пока еще не было произведено полномасштабных апробаций на реальных пакетах документов, лишь небольшие пилотные внедрения [13]. Еще одной целью работы было применение технологии DocLine для выявления проблем внедрения и решения этих проблем, а также предложения улучшения технологии.

---

<sup>1</sup> Полиграфическая разметка текста - выбор типа и разметки шрифта, выравнивание текста и т.д.

## **2. Постановка задачи**

Основной целью данной работы является пилотное внедрение технологии DocLine. Полный пакет пользовательской документации фрагмента семейства телефонных станций ЗАО ЛАНИТ-ТЕРКОМ должен быть перепроектирован в терминах DocLine и доведен до публикации средствами технологии в два формата – PDF и HTML. В ходе выполнения работы должны быть проанализированы различные аспекты технологии, сделаны предложения по ее улучшению и доработке.

Для достижения этой цели были поставлены следующие задачи:

- изучить XML-технологии DocBook, на которой основывается DocLine, технологию DocLine и язык разметки DRL, а также инструментальные средства DocLine;
- проанализировать техническую документацию семейства телефонных станций ЗАО ЛАНИТ-ТЕРКОМ, понять предназначение и функциональные возможности данных программных систем;
- выявить повторно используемые фрагменты документации (информационные элементы и каталоги), создать верхнеуровневую модель повторного использования;
- полностью выполнить в DocLine исследуемый фрагмент документации;
- проанализировать результаты, сделать предложения по улучшению технологии DocLine.

## 3. Обзор литературы

### 3.1. Семейства программных продуктов

Семейство программных продуктов – это набор программных систем, предназначенных для определенного сегмента рынка, схожих между собой и разделяющих много общей функциональности. При этом имеется набор общих ресурсов, которые разрабатываются в контексте целого семейства, и могут быть использованы при разработке конкретного продукта. В качестве переиспользуемых ресурсов могут выступать: архитектура, модели, процессы, планы, требования, документация и т.д. [9].

Каждый продукт семейства может иметь собственного заказчика и уникальный набор требований (хотя заказчик может и не подозревать о существовании семейства продуктов), но конкретный продукт семейства изготавливается из набора общих активов и заложенной в них вариативности, а также новых компонент, необходимых для создания этого продукта.

Разработка семейства программных продуктов – относительно новый подход к разработке программных продуктов, преимуществом которого является: снижение количества разработчиков в штате, повышение производительности труда и качества продуктов, быстрый вывод продуктов на рынок и снижение стоимости разработки.

Согласно [9], процесс разработки семейства продуктов состоит из трех основных видов деятельности:

- разработка общих активов (Core Assets Development);
- разработка продукта (Product Development);
- управление разработкой семейства (Product Management).

При *разработке общих активов* учитываются требования к продуктам, стратегия разработки, производственные ограничения и наличие ресурсов, активов и опыта. Результатами являются: архитектура семейства продуктов, спецификация области, в которой разрабатывается семейство продуктов, метод использования общих активов, привязка требований к архитектуре и сами общие активы.

В процессе *разработки отдельных продуктов* используются данные, полученные при разработке общих активов, а также учитываются требования, выдвигаемые к конкретному продукту. Если при рассмотрении требований замечено, что нужная функциональность не реализована в общих компонентах, выносится решение о реализации некоторых требований отдельно, в рамках разработки общих активов.

*Управление разработкой семейства* программных продуктов осуществляется на двух уровнях: техническом и организационном. Техническое управление контролирует исполнение метода разработки, определённого для семейства, управляет проектами и решает вопросы, связанные с инструментальной поддержкой и конфигурационным

управлением. Организационное управление анализирует рынок, обеспечивает ресурсами, управляет взаимодействиями с внешним миром, разрешает конфликты и осуществляет стратегическое планирование.

Все три вида деятельности имеют выраженный итеративный характер (постепенная эволюция с изначальным планом) и тесно связаны между собой.

### **3.2. Метод DocLine и язык DRL**

Метод DocLine [5, 6, 7] состоит из следующих частей:

- процесс разработки документации;
- язык DRL, имеющий графическую (DRL/GR – DRL Graphic Representation) и текстовую (DRL/PR – DRL Phrase Representation) нотации;
- инструментальные средства.

#### **3.2.1. Процесс**

Процесс разработки документации, как и процесс разработки семейства в целом, состоит из двух частей – разработки документации семейства и разработки документации продуктов. При необходимости допускается вносить изменения в документацию семейства при разработке документации продуктов, что может потребоваться для тонкой настройки повторного использования документации.

Графическая нотация предназначена для «крупноблочного» проектирования и представления документации, а текстовая – соответственно для написания текста. При работе с документацией процесс допускает одновременную работу с этими двумя представлениями и существует возможность перехода от написания текста к редактированию диаграмм и обратно, без нарушения структуры текста.

#### **3.2.2. Обзор графической нотации DRL<sup>2</sup>**

Графическая нотация языка DRL (DRL Graphic Representation) предназначена для визуального проектирования документации семейства и отдельных продуктов. Включает три вида диаграмм:

- главная диаграмма;
- диаграмма вариативности;
- диаграмма продукта.

---

<sup>2</sup> Раздел был написан в соответствии с [5], [6] с дополнениями описания диаграмм.

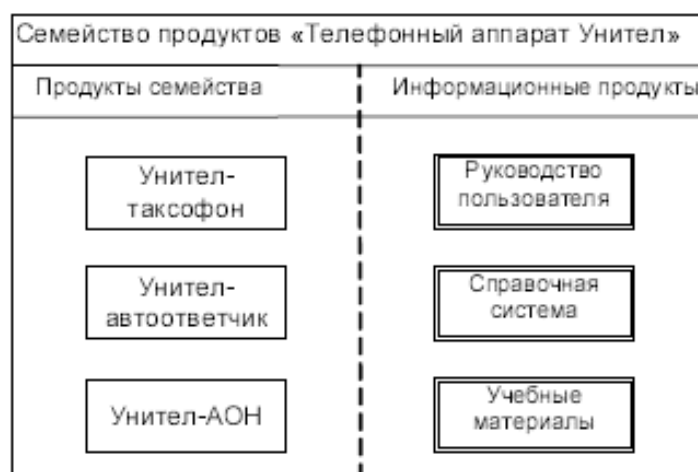


Рис. 1

Главная диаграмма (Рис. 1) позволяет определить все продукты семейства и ИП<sup>3</sup>, и связи между ними.

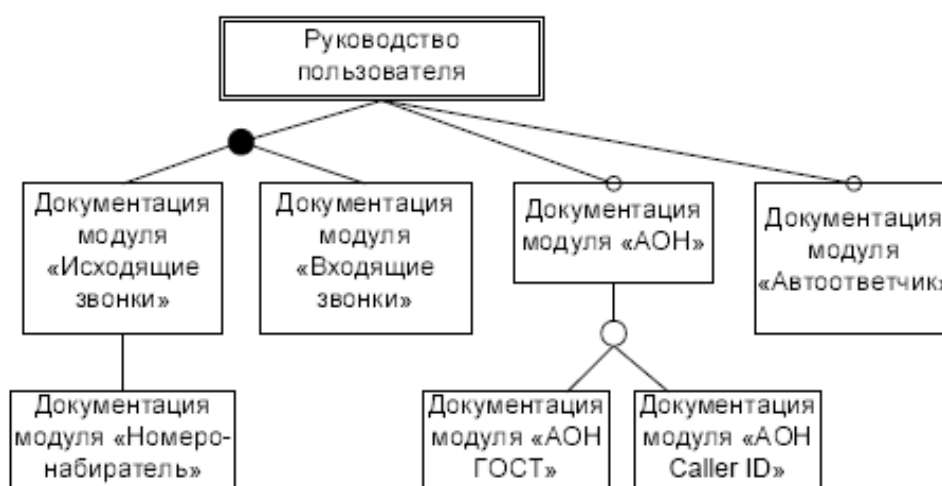


Рис. 2

Каждый ИП задается с помощью диаграммы вариативности (Рис. 2), на которой изображена структура ИП и ИЭ<sup>4</sup>. ИП могут содержать произвольное число ИЭ. А ИЭ, в свою очередь, могут состоять из других ИЭ. В корне диаграммы находится ИП и обозначается прямоугольником с двойными границами, а все остальные – ИЭ. Связи между ИЭ обозначают отношения включения и могут иметь различную семантику: строгое альтернативное включение (XOR), нестрогое альтернативное включение (OR), обязательное и необязательное включение. В диаграмме вариативности XOR-группа обозначается белым кругом, OR-группа – черным кругом, обязательное

<sup>3</sup> Информационный Продукт – документ или набор документов, который является самостоятельным результатом процесса разработки документации [5].

<sup>4</sup> Информационный Элемент – фрагмент документации, предназначенный для использования в одном или нескольких ИП [5].



включение – окружностью в месте соединения с ИЭ. Таким образом, диаграмма вариативности позволяет описать «крупноблочное» переиспользование через ИП, ИЭ и связи между ними.



Рис. 3

Диаграмма продукта (Рис. 3) представляет собой «выборку» из диаграммы вариативности, т.е. для конкретного продукта разрешаются все неопределенности в группах включений ИЭ (в группах XOR, OR) и исключаются неиспользуемые элементы.

### 3.2.3. Обзор текстовой нотации DRL<sup>5</sup>

Текстовая нотация DRL (DRL Phrase Representation) предназначена для непосредственного написания текста документации с последующей генерацией пакета документов в различные форматы.

DRL является XML-языком. Каждая его конструкция представляется XML-тэгом.

Основной конструкцией DRL является конструкция, описывающая схему семейства программных продуктов. Листинг 1 иллюстрирует текстовое представление продуктов семейства:

```

<d:ProductLine name="Семейство телефонных аппаратов «Унител»">
  <d:Product name="Унител-таксофон" id="tax"/>
  <d:Product name="Унител-автоответчик" id="answer"/>
  <d:Product name="Унител-АОН" id="callerid"/>
</d:ProductLine>
  
```

Листинг 1. Схема семейства продуктов

Конструкция `ProductLine` обозначает семейство продуктов, а конструкция `Product` – отдельный продукт семейства. Каждый продукт имеет имя (атрибут `name`) и идентификатор (атрибут `id`) для ссылок.

<sup>5</sup> Раздел была написан в соответствии с [5], [6] с изменениями в листингах, так как DRL/PR изменился с момента написания этих статей.

Текстовая DRL-спецификация документации семейства представлена в Листинге 2. Документация семейства указывает на то, какие документы (ИП) могут входить в семейство и из каких частей (ИЭ) состоят, где и какая настройка повторного использования должна быть сделана. Фактически, ИП можно понимать как шаблон документа, который после указания всех параметров и настроек будет готов к включению в документацию конечного продукта.

```
<d:DocumentationCore>
  <d:InfProduct id="userguide" name="Руководство
пользователя">
    <book>
      <bookinfo>
        <title><d:DictRef dict="main" Entry="productname"/>
        </title>
      </bookinfo>
      <d:InfElemRefGroup id="in_out" modifier="OR"
name="in_out"/>
      <d:InfElemRef id="out_ref" infelemid="Исходящие"
groupid="in_out"/>
      <d:InfElemRef id="in_ref" infelemid="Входящие"
groupid="in_out"/>
      <d:InfElemRef id="aon_ref" infelemid="АОН"
optional="true"/>
    </book>
  </d:InfProduct>
  <d:InfProduct id="help" name="Справочная система"/>
  <d:InfElement id="АОН" name="АОН"/>
</d:DocumentationCore>
```

Листинг 2. Документация семейства

DocumentationCore – это ядро документации, т.е. та ее часть, которая содержит набор повторно используемых компонент. Конструкция InfProduct указывает на ИП, конструкция InfElement – на ИЭ. InfElemRefGroup указывает на группу ИЭ, связанных OR или XOR отношением (см. Графическую нотацию DRL), описанным в параметре modifier. А параметр optional в ссылке на ИЭ указывает на необязательное включение этого ИЭ в конечный продукт документации.

Документация продукта состоит из набора специализированных информационных продуктов (СИП) и описывает, каким образом нужно модифицировать документацию семейства, чтобы получить пакет документов для конкретного продукта семейства. Т.е. каждый СИП соответствует ИП в документации семейства, описывает, какие ИЭ должны войти в документацию конкретного продукта и как их настроить. Также он содержит ссылку на продукт из схемы семейства, к которому полученный документ будет относиться. Документация продукта описана в Листинге 3.

```

<d:ProductDocumentation productid="callerid">
  <d:FinalInfProduct id="UserGuide1" infproductid="userguide">
    <d:Adapter infelemrefid="CallerID_aon_ref">
      <d:Insert-After nestid="Способы индикации номера">
        Также предусмотрена возможность
        звуковой индикации номера абонента.
      </d:Insert-After>
    </d:Adapter>
  </d:FinalInfProduct>
</d:ProductDocumentation>

```

Листинг 3. Документация продукта

Конструкция `ProductDocumentation` задает описание документации одного продукта семейства. Атрибут `productid` задаёт привязку к документу из схемы семейства продуктов. `FinalInfProduct` определяет СИП, идентификатор соответствующего ему ИП задан атрибутом `infproductid`. Теги `Adapter` служат для настройки ИЭ в контексте заданного продукта семейства. Для каждой ссылки в СИП может иметься конструкция адаптер, позволяющая модифицировать соответствующее вхождение ИЭ. Таким образом, в процессе трансляции DRL-текстов ссылка на ИЭ заменяется содержимым самого ИЭ, модифицированным при помощи соответствующего адаптера.

Таким образом, DRL-файлы документации являются XML-файлами, каждому из которых соответствует свой корневой элемент: `ProductLine`, `DocumentationCore` или `ProductDocumentation`.

Возвращаясь к описанию адаптеров, заметим, что атрибут `nestid` указывает на точку расширения в тексте ИЭ, с помощью которой возможна адаптация компонент к требованиям конкретного контекста. Точки расширения задаются конструкцией `Nest` и являются конструкциями для реализации «крупноблочного» повторного использования. Она описана в Листинге 4.

```

<d:InfElement id="АОН CallerID" name="АОН CallerID">
  <d:Nest id="Способы Индикации номера">
    При успешном определении номер отображается на
    встроенном дисплее телефонного аппарата.
  </d:Nest>
</d:InfElement>

```

Листинг 4. Точка расширения

Кроме случаев «крупноблочного» переиспользования нередко требуется организовать повторное использование отдельных слов и словосочетаний, в частности, для адаптации словоформ стандартных терминов к особенностям контекста. Такое переиспользование называется «мелкозернистым» и реализуется в словарях и каталогах.

Словарь – это набор пар имя-значение, обычно используется для описания стандартных терминов. Включение элемента словаря происходит посредством указания конкретного словаря и имени элемента. Листинг 5 иллюстрирует описание словаря (конструкция Dictionary) и включение элемента словаря (конструкция DictRef).

```
<d:Dictionary id="main" name="main_dict">
  <d:Entry id="productname">Название продукта</Entry>
  <d:Entry id="productversion">1.0</Entry>
</d:Dictionary>
<d:InfElement id="About" name="About">
  О программе <d:DictRef dictid="terms" entryid="productname"/>
</d:InfElement>
```

Листинг 5. Словарь

Каталог – используется в том случае, когда необходимо представление какой-либо информации в нескольких «форматах», т.е. информации схожей структуры (например, команды приложения, каждая из которых может быть представлена разными способами и в различных контекстах, Листинг 6).

```
<d:Directory id="commands" name="Команды">
  <d:Entry id="Ab">
    <d:Attr id="name">О программе</d:Attr>
    <d:Attr id="hint">Информация о программе</d:Attr>
    <d:Attr id="icon">about.jpg</d:Attr>
  </d:Entry>
</d:Directory>
<d:DirTemplate id="CommandNameHint" directoryid="commands">
  <d:AttrRef attrid="name"/>( <d:AttrRef attrid="hint"/>)
</d:DirTemplate>
<d:InfElement id="CommandList" name="Command List">
  Команды: <d:DirRef templateid="CommandNameHint" entryid="Ab"/>
</d:InfElement>
```

Листинг 6. Каталог

Конструкция Directory – описание каталога, DirTemplate – шаблон отображения элемента каталога, DirRef – использование элемента каталога. Все элементы имеют идентификатор для ссылок.

Есть также вспомогательные конструкции для повторного использования: условный блок и присваивание псевдопеременной. Листинг 7 описывает эти конструкции.

```
<d:InfProduct id="userguide" name="Руководство пользователя">
  <book>
    <bookinfo>
      <title>Комплекс автоматизации вещания</title>
    </bookinfo>
    <d:InfelemRef id="server" infelemid="Серверная подсистема"/>
    <d:Conditional condition="maintainable=yes">
```

```

        <d:InfelemRef id="maintenance" infelemid="Обслуживание
сервера"/>
    </d:Conditional>
</book>
</d:InfProduct>
<d:FinalInfProduct infproductid="userguide">
    <d:Setvalue name="maintainable">no</Setvalue>
    <d:Adapter infelemrefid="CallerID_aon_ref">
        <d:Insert-After nestid="Способы индикации номера">
            Также предусмотрена возможность звуковой индикации
номера абонента.
        </d:Insert-After>
    </d:Adapter>
</d:FinalInfProduct>

```

Листинг 7. Вспомогательные конструкции

Описанные выше конструкции: соответственно Conditional и Setvalue, – могут использоваться в тексте ИП и ИЭ, а псевдопеременная – и в тексте СИП. Также Листинги 7, 2 иллюстрируют использование кода DocBook внутри элементов DRL. Хотелось бы отметить, что использование в конструкциях d: – говорит о том, что эта конструкция языка DRL, а все остальные конструкции – языка DocBook.

### 3.3. *Технология DocBook*

#### 3.3.1. Введение

Технология DocBook была впервые предложена в 1991г. DocBook – XML-язык полиграфической разметки текстов, уже многие годы являющийся стандартом де-факто для создания сложной технической документации [11].

Первоначально DocBook разрабатывался исходя из нужд написания технической документации. Но он вполне универсален и используется для самых разных целей, например для разработки веб-сайтов и создания презентаций. Наиболее типичное применение DocBook – автоматическая генерация документации из единого исходного представления в разные форматы (HTML, PDF и т.д.).

Исходный документ DocBook представляет собой XML-файл, схема которого стандартизована. Язык DocBook содержит конструкции для описания структуры текста (разделение на главы, параграфы), разметки текста (выделение разными шрифтами, создание таблиц, списков), форматирования. Например, есть конструкции для определения формата документации в целом: <book/>, <article/> – для создания документации в виде книги или статьи соответственно. А в конструкции <book/> есть тэги для определения списка авторов, названия книги и другой информации.

DocBook является стандартом де-факто для документации на свободно распространяемое программное обеспечение. Его используют ведущие IT-

компании и организации: IBM, Microsoft, Hewlett-Packard, Sun, Novell, WWW Consortium.

### **3.3.2. Особенности технологии**

По отношению к традиционным средствам подготовки документации – текстовым процессорам (например, Microsoft Word), настольным издательским системам (например, Adobe PageMaker) и языкам разметки (TeX, HTML) – технология DocBook обладает следующими преимуществами:

- встроенные возможности для работы с различным содержанием;
- сопровождается системой, позволяющей, используя различные инструменты, преобразовывать DocBook в различные выходные форматы – HTML-страницы, PDF-файлы, Microsoft HTMLHelp, и т.д., из единого исходного текста;
- существует набор инструментов, которые также позволяют преобразовывать различные форматы (HTML, Javadoc, Texinfo files и т.д.) в DocBook;
- хорошо документирована (к сожалению, в большинстве случаев не на русском языке);
- полностью открытый (open-source) стандарт;
- прекрасно настраиваемая и расширяемая система;
- поддержка коллективной работы над документацией, возможность повторного использования документов.

### **3.4. Выводы**

Метод разработки семейств программных продуктов – это современный эффективный метод разработки ПО, основанный на переиспользовании доступных ресурсов. Проблеме разработки документации семейства адресован метод разработки документации DocLine.

Основой DocLine является язык разметки документации DRL, позволяющий организовать высокую степень повторного использования текста (от целых параграфов, абзацев до отдельных слов). DRL имеет две нотации – текстовую (DRL/PR) и графическую (DRL/GR). Апробация технологии DocLine на реальном проекте в полном объеме не была произведена. Поэтому применение подхода должно выявить ряд трудностей и привести к предложениям по решению этих проблем, а также по улучшению непосредственно метода. До настоящего момента за рамками апробации осталось применение метода не только в сфере ПО, но и в других областях, что не малоинтересно.

## 4. Перевод промышленной документации в DocLine

### 4.1. Краткое описание промышленного проекта<sup>6</sup>

В 80-х годах коллектив ЛГУ на базе НИИ математики и механики ЛГУ и ЛНПО «Красная заря» начал разработку ПО для АЦТС (Автоматических цифровых телефонных станций). Для коммерциализации и развития этих и других работ была образована компания ЗАО «ЛАНИТ-ТЕРКОМ». В результате было создано ПО для многих телефонных станций специального и гражданского назначения. К настоящему моменту главным продуктом этих разработок является ПО для АЦТС «КВАНТ-Е-СОКОЛ». Около 500 таких станций выпущены и установлены по разным городам и областям России.

Данная дипломная работа посвящена апробации метода DocLine на документации для рабочих мест операторов (РМО) телефонной станции «КВАНТ-Е-СОКОЛ», а именно на полной версии документации для станции гражданского назначения и «урезанной» версии – для станции специального назначения.

Фактически, работа технического писателя над новой документацией начинается с изучения исходной документации и, в первую очередь, области, в которой работали над данным промышленным проектом, в результате чего и появилась необходимость написания пакета документов.

Еще в XIX веке телефонные станции были не автоматизированы. На станции оператор принимал от вызывающего абонента информацию о номере вызываемого абонента, отыскивал на коммутаторе гнездо, в которое включена линия этого абонента, и если она была не занята, соединял между собой линии вызывающего и вызываемого абонента, а после получения от абонентов сигнала отбоя производил разъединение. С ростом числа абонентов приходилось открывать все больше телефонных станций. Чтобы избежать ошибок соединения, нужно было автоматизировать телефонные станции.

Итак, ПО АЦТС «КВАНТ-Е-СОКОЛ» можно условно разбить на три части [12], связанные между собой.

1. Функциональное программное обеспечение (ФПО) – ПО, записываемое в АЦТС на управляющие микросхемы памяти с помощью специального устройства, которое называется «программатор» (такая операция называется «прожиганием»). ФПО отвечает за установление всех видов соединений/разъединений, обеспечение дополнительных услуг, сканеры технических средств, обеспечение технического обслуживания, включая начальный запуск станции.

---

<sup>6</sup> Раздел был написан в соответствии со статьей [12] с дополнениями, касающимися истории возникновения АЦТС и непосредственно данной работы (апробации технологии DocLine).

2. Системное программное обеспечение (СПО) – ПО, «прожигаемое» в АЦТС, которое включает в себя программные средства, обеспечивающие работу ФПО в распределенной вычислительной среде.

3. Программное обеспечение рабочего места оператора (РМО) для технического обслуживания и эксплуатации АЦТС, которое функционирует на непосредственно подключенном к АЦТС или удаленном рабочем месте, в качестве которого используется ПЭВМ. Существенную часть РМО составляет база данных АЦТС. Если ПО станции является в основном универсальным для станций одного типа, то БД индивидуальны для каждой АЦТС. БД формируется при запуске станции в эксплуатацию, после чего может меняться по требованию пользователей и обслуживающего персонала.

Конечный продукт представляет собой интеграцию этих трех компонент ПО, взаимодействующих на разных уровнях. Первые две части ПО являются встроенными, третья часть – интерфейсная.

Существует две модификации телефонной станции «КВАНТ-Е-СОКОЛ». Одна является базовым продуктом (далее будем называть ее (гр)), а вторая – урезанной версией для специальных применений (далее будем называть ее (сп)). При этом, конечно, эта урезанная версия имеет некоторые изменения функциональности, которые были внесены в станцию специально для ее заказчиков. Например, переделано главное окно компоненты «Карта обслуживания абонентских линий».

#### **4.2. Описание документации**

Для выполнения данной работы был взят фрагмент руководства пользователя ПО АЦТС «КВАНТ-Е-СОКОЛ», описывающий РМО, а именно, было выбрано руководство пользователя для РМО АЦТС гражданского и специального назначения. Документация для РМО (сп) «написана» путем извлечения нужной информации из документации для РМО (гр), а именно, содержит описание нужных (для конкретного заказчика) функций телефонной станции. Поскольку функциональность станции специального назначения много меньше функциональности станции гражданского назначения, то заказчик станции (сп) хотел, чтобы в документации для РМО (сп) были описаны только те компоненты, которые реализованы в станции (сп). Кроме того, эти функции немного отличаются друг от друга, что отражено и в документации.

Версия документа для РМО (гр) включает в себя описание реализации, описание задач пользователя, приложения и другие разделы, содержащие информацию о назначении комплекса РМО, о его структуре, правилах установки и настройки, использовании и т.д. Комплекс программных средств РМО позволяет выполнять основные задачи обслуживания телефонной станции: управление встроенным ПО, обслуживание и управление абонентскими линиями и межстанционными каналами, тестирование, диагностику и мониторинг различных программных и аппаратных элементов, а также сбор статистической и тарификационной информации и



т.п. Кроме того, комплекс поддерживает ряд дополнительных возможностей: удаленную работу со станцией через протокол ТСР/IP, авторизацию пользователей, протоколирование действий оператора, выполнение задач по расписанию, синхронизацию времени и т.д.

Пользовательская документация для РМО (сп) содержит описание только основных функций комплекса РМО, таких как описание панели РМО, создание/изменение списка пользователей и групп, назначение прав пользователям и группам, работа с модулями станции, управление характеристиками абонента, просмотр/анализ состояний абонентских линий, просмотр конфигурации и состояния внешних соединительных линий станции.

Базовая документация (для РМО (гр)) включает описания 9 разделов. В руководстве пользователя для РМО (сп) из этих 9 разделов содержится 1 раздел полностью (это «Расшифровка терминов и сокращений») и другой («Приложения») – частично. Опишем, что это значит, более детально.

### **4.3. Структура документации в DocLine**

На рис. 4 представлена диаграмма вариативности для пользовательской документации РМО (гр) и РМО (сп). Повторно используемые разделы документации отмечены жирными прямоугольниками, в тех разделах, где есть точки расширения, указано их количество. На рис. 5. представлена «вырезка» для документации РМО (гр), а на рис. 6 – для РМО (сп). Для разделов, имеющих точки расширения, указано, сколько из них задействовано. Например, запись 6/7 для раздела «Панель состояния тракта» на рис. 5 говорит о том, что из семи точек расширения в случае документации для РМО (гр) задействовано шесть.

Остановимся на диаграмме вариативности (рис. 4). Сначала мы выделили все разделы документаций первого уровня в ИЭ. Далее, изучая эти разделы документации, мы выделили в отдельные ИЭ лишь те подразделы, в которых есть переиспользуемость и вариативность. Таким образом, у нас получилось большое поддерево у раздела «Приложения». Остальные разделы первого уровня или целиком входят в оба варианта документации (как раздел «Расшифровка терминов сокращений»), либо могут быть опущены – но также целиком, без вариаций внутри. Эти разделы мы не стали «раскладывать» далее в ИЭ, так как это усложнило бы диаграмму вариативности, но, собственно, вариативности бы не добавило.

В подразделы раздела «Приложения», а также те их подразделы, которые содержат вариативность, мы также выделили в отдельные ИЭ, и так до самого «низа» - тех подразделов документации, которые уже не дробятся нами на ИЭ либо потому, что у них нет подразделов, либо потому (например, подразделы «Обзор», «Главное окно», «Поиск»), либо которые не содержат внутри себя вариативности, как, например, «Проводник СММ, скрипты РМО». У подраздела «Главное окно» есть три XOR-группы – в них

объединены фрагменты текста, которые не являются подразделами, но входят в разных версиях документа в разные места раздела «Главное окно».

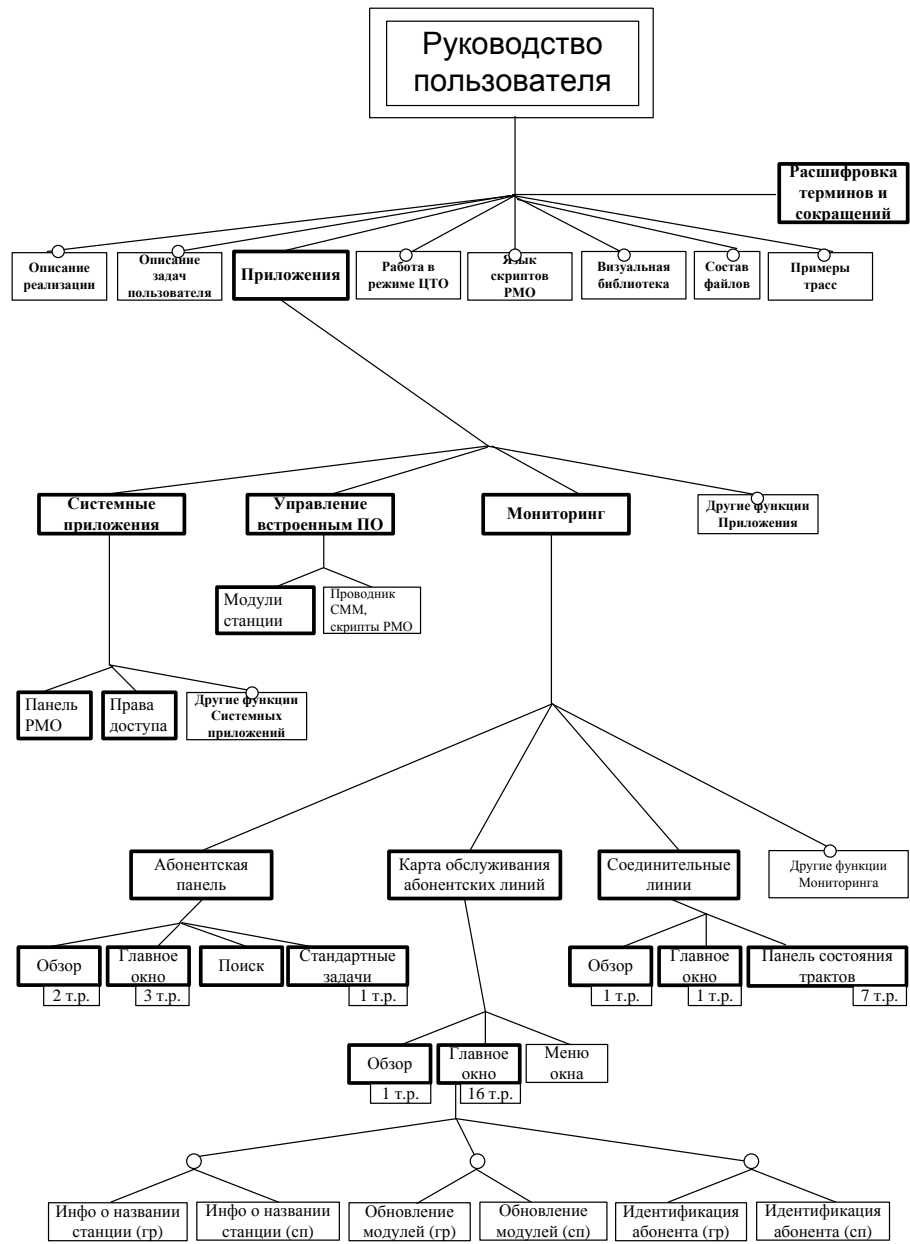


Рис. 4

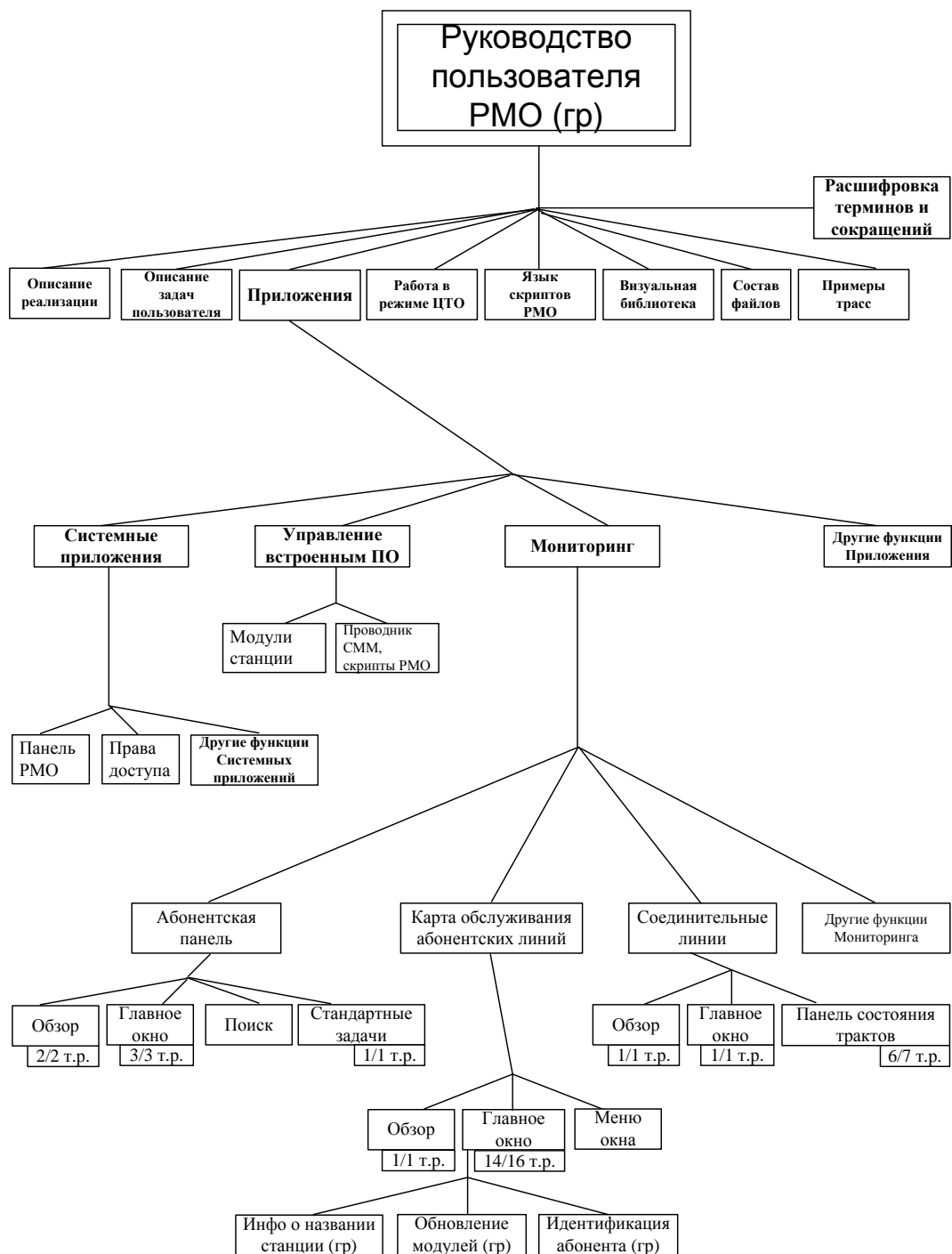


Рис. 5

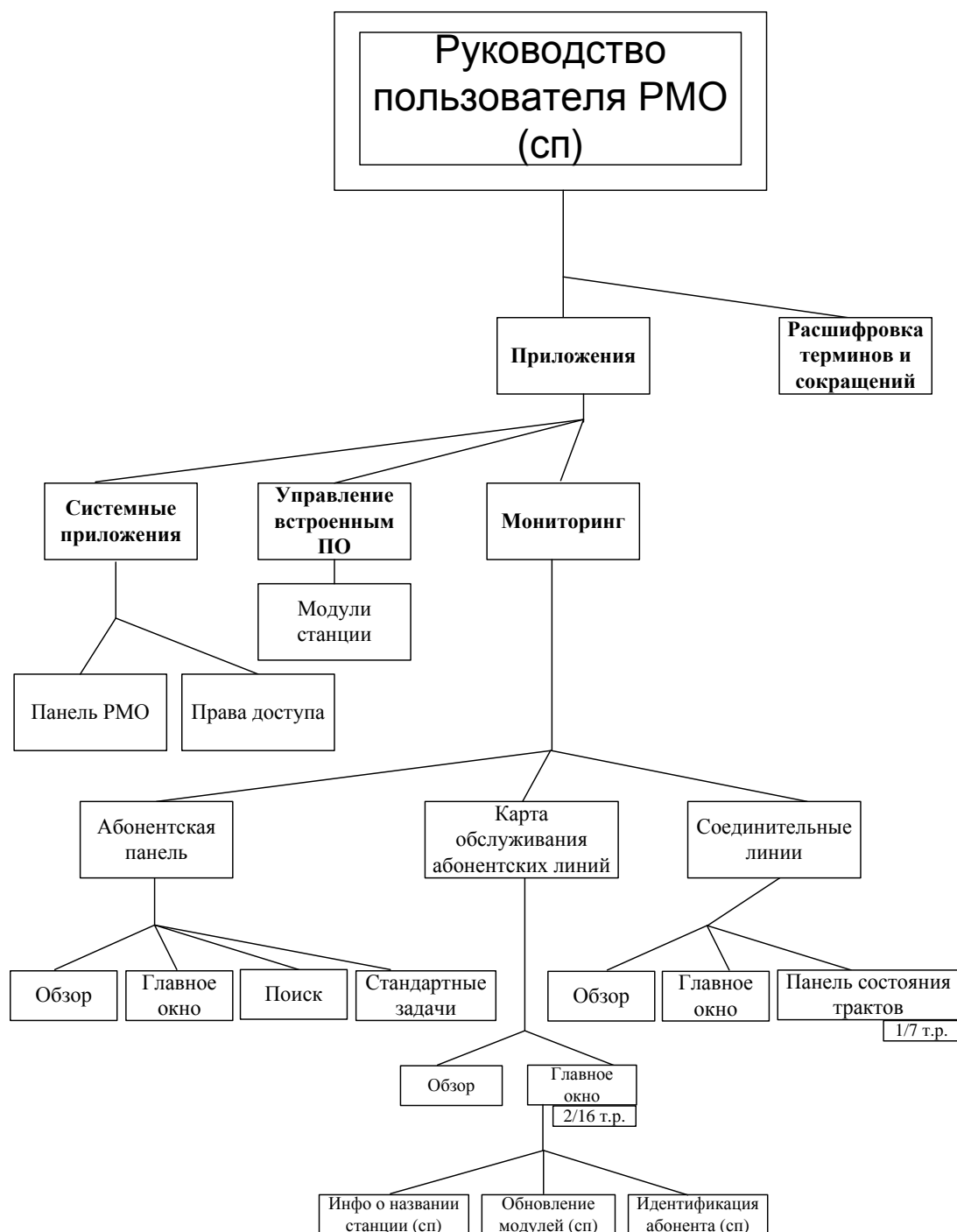


Рис. 6

#### 4.4. Анализ вариативности

Итак, в руководствах пользователя для РМО (гр) и РМО (сп) есть три подраздела – «Абонентская панель», «Карта обслуживания абонентских линий», «Соединительные линии», – которые повторяются в этих руководствах и одновременно содержат ряд отличий, не разрешаемых на уровне условного включения других ИЭ. То есть они содержат мелкие отличия, которые реализуются через точки расширения и механизм адаптации. Классифицируем различные изменения текста документов в этой

ситуации «снаружи», с точки зрения документации, а не с позиций реализации на DRL. Для этого введем несколько определений.

*Удаление* – из базовой версии ПО (в данном случае из РМО(гр)) удаляется какое-либо функциональное свойство, которое не нужно заказчикам новой версии (в нашем случае – РМО(сп)). Соответственно, описание этого свойства удаляется из документации. Важно, что описание этого свойства могло быть «размазано» по нескольким местам документа, то есть такое исправление документа может быть «веерным», и/или удаляется большой фрагмент текста. Последним способом были удалены все главы из документации РМО (гр), описывающие функциональность, не вошедшую в РМО (сп).

*Добавление* – в базовую версию ПО добавляется какое-либо новое функциональное свойство. Соответственно, его описание должно быть добавлено в пользовательскую документацию. Это операция является обратной удалению. Как и там, здесь важно, что изменения текста могут быть веерными. Эта операция не характерна для нашего случая, так как в РМО (сп) не произошло добавления каких-либо новых функциональных возможностей по сравнению с РМО (гр).

*Изменение* – значительно меняется какое-либо функциональное свойство, что для документации выражается в том, что отсутствует описание какого-либо атрибута этого свойства в виде значительного фрагмента подряд идущего текста<sup>7</sup>.

*Локальное изменение* – меняется лишь какой-либо отдельный атрибут какого-либо функционального свойства, например, количество обслуживаемых абонентов. Для пользовательской документации такое изменение функциональности выльется в небольшое локальное изменение текста. Например, при изменении количества обслуживаемых абонентов будет изменено число в сводной таблице функциональных возможностей. В подобных случаях могут быть добавлены или убраны отдельные строки в таблицы, списки с перечислениями, отдельные предложения и т.д.

«*Параллельные места*» – это еще один вид изменений документации, когда в новой версии та же самая информация описывается другими словами. То есть этот фрагмент текста создан не с помощью «copy-paste».

Еще раз подчеркнем, что удаление и добавление особенны тем, что некоторая информация (описание одной функциональной возможности) не локализована в тексте, поэтому ее добавление и удаление требует особой внимательности. Ошибки здесь могут привести к потере корректности текста. Конечно, изменения функциональных возможностей могут также затронуть несколько фрагментов текста.

---

<sup>7</sup> либо изменения в тексте рассосредоточены в разных метках, но в контексте изменений в нашей ситуации таких случаев нет.

Приведем таблицу примерных процентных соотношений тех или иных изменений в документации (Табл. 1), а затем перейдем к примерам изменений текста документов.

Табл. 1

<i><b>Характер изменений</b></i>	<i><b>%</b></i>
Удаление	32
Добавление	0
Изменение	5
Локальное изменение	26
«Параллельные места»	37

Исходя из соотношений, представленных в таблице, можно сделать вывод о том, что в документации для РМО (гр) и РМО (сп) примерно одинаковое количество таких изменений, как удаление и «параллельные места», чуть меньше локальных изменений, очень мало значительных изменений функциональности и совсем нет дополнительной функциональности в станции (сп) по отношению к станции (гр).

Теперь приведем некоторые примеры.

#### ***4.4.1. Примеры удалений***

Главное окно приложения «Карта обслуживания абонентских линий» РМО (сп) не содержит меню, которое было в РМО (гр). В руководстве пользователя соответствующий подраздел раздела «Карта обслуживания абонентских линий», описывающий это меню, будет отсутствовать в варианте документации для РМО (сп).

Также в этой компоненте для РМО (сп) отсутствует возможность сформировать список модулей (модуль – некоторое множество абонентов), информация о состоянии которых будет загружаться из станции. Таким образом, в разделе документации для РМО (сп) нет описания этой функции: в подразделе «Обзор» нет соответствующего предложения (Рис. 7)<sup>8</sup>, а также документация РМО (сп) не содержит подраздела «Окно «Выбор модулей» (Рис. 8).

---

<sup>8</sup> Далее на всех рисунках полужирным курсивом будут выделены различные изменения текста.

### Обзор

Данное приложение предназначено для просмотра и анализа состояний абонентских линий станции.


С его помощью оператор может посмотреть текущую статистику состояний в виде диаграммы.

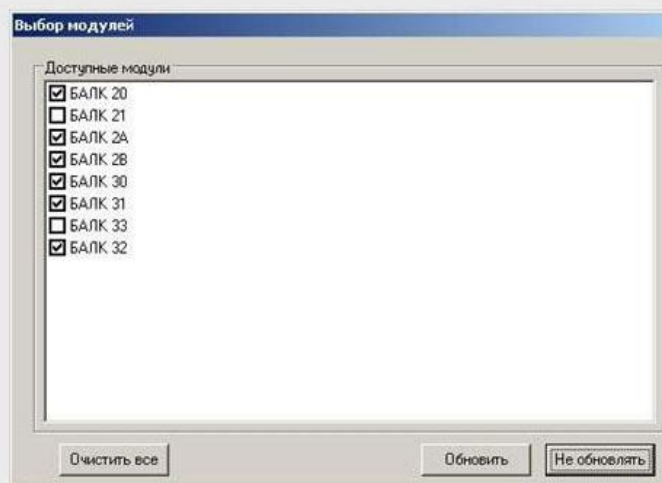
*Приложение позволяет заказать модули, информация о состоянии которых будет загружаться из станции.*

Приложение также включает в себя возможность автоматического обновления текущих состояний через определённые оператором промежутки времени.

Рис. 7

### Окно «Выбор модулей»

Пользователь имеет возможность сформировать список модулей, информация о состоянии которых будет загружаться из станции. Нажатие дополнительной секции кнопки Обновить (  ) в верхней части окна и выбор пункта Модули вызовет появление окна «Выбор модулей»:



Для заказа модулей, информация о состоянии которых будет загружаться из станции, необходимо подвести указатель мыши к полю активации и нажать правую кнопку. Цвет текста названий модулей, отображённых в главном окне «Состояния абонентов», примет соответствующую окраску.

В левом нижнем углу окна находится кнопка групповой операции Выбрать все (Очистить все), нажатие которой вызывает активацию (деактивацию) всех модулей списка «Опрашиваемые модули».

Рис. 8

В других двух компонентах документаций («Абонентская панель» и «Соединительные линии») также есть изменения функциональности (как веерное удаление, так и удаление фрагмента компоненты).

#### 4.4.2. Примеры изменений

Главное окно приложения «Карта обслуживания абонентских линий» РМО (сп) (Рис. 9) и РМО (гр) (Рис. 10) различаются тем, что в главном окне компоненты для РМО (гр) по цвету заголовка и названия модуля можно извлечь дополнительную информацию о состоянии и «заказанности» модуля. То есть у функционального свойства РМО (гр) «Главное окно карты обслуживания абонентских линий» есть дополнительный по сравнению с РМО (сп) атрибут, описанию которого соответствует значительный фрагмент текста – в соответствующем разделе РМО (сп) отсутствует таблица описания цвета заголовка и текста названия модуля главного окна компоненты (Рис. 11).

Главное окно программы выглядит так:

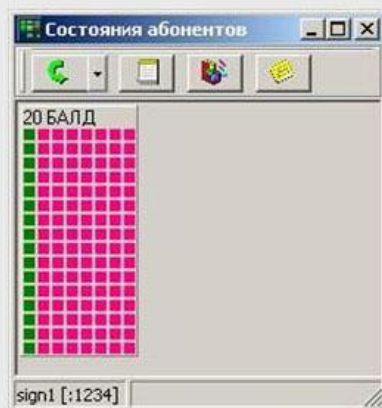


Рис. 9

Главное окно программы имеет следующий вид:

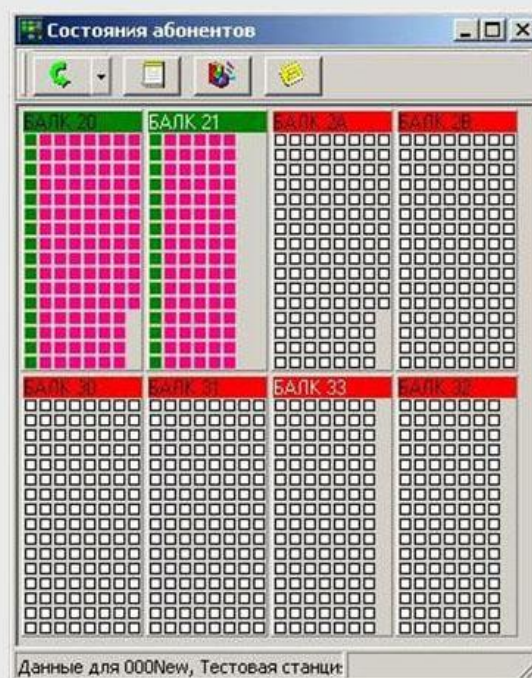


Рис. 10

Цвет заголовка названия модуля соответствует состоянию модуля:

Серый	информация о модуле еще не прочитана, это изначальное состояние.
Зелёный	модуль активен.
Красный	модуль не ответил на запрос.

Цвет текста названия модуля соответствует заказанности модуля:

Черный	модуль заказан в списке опрашиваемых модулей
Белый	модуль не заказан в списке опрашиваемых модулей

Рис. 11



#### 4.4.3. Локальные изменения

Ряд функциональных возможностей в РМО (сп) изменился по сравнению с РМО (гр) локально. Рассмотрим все ту же компоненту «Карта обслуживания абонентских линий». И для РМО (гр), и для РМО (сп) она содержит возможность настроить автоматическое обновление состояний абонентских линий. Однако в РМО (гр) период времени обновления можно установить в 5, 10 или 30 секунд (Рис. 12), а в РМО (сп) – еще и на 1 секунду (Рис. 13).

Чтобы активизировать автоматическое обновление состояний абонентских линий необходимо нажать на правую часть этой кнопки и в появившемся меню выбрать необходимый период времени обновления (из представленных: 5, 10 или 30 секунд). При этом будут обновляться состояния абонентов, **которые заказаны в списке опрашиваемых модулей**. Для обновления состояния всех абонентских модулей станции (в случае если какой-то модуль во время последней загрузки был недоступен), то необходимо выбрать пункт меню Обновить все. Для прекращения автоматического обновления необходимо в том же меню выбрать Стоп. Следует заметить, что после обновления состояний и при открытом окне статистики абонентских линий, последняя обновляется автоматически.

Рис. 12

Чтобы активизировать автоматическое обновление состояний абонентских линий необходимо нажать на правую часть этой кнопки и в появившемся меню выбрать необходимый период времени обновления (из представленных: **1**, 5, 10 или 30 секунд). При этом будут обновляться состояния **только тех** абонентов, которые **были доступны при последней загрузке данных из станции**. Если же хочется обновить состояния по всем абонентским модулям станции (в случае если какой-то модуль во время последней загрузки был недоступен), то необходимо выбрать пункт меню Обновить все. Для прекращения автоматического обновления необходимо в том же меню выбрать Стоп. Следует заметить, что после обновления состояний и при открытом окне статистики абонентских линий, последняя обновляется автоматически.

Рис. 13

В компоненте «Соединительные линии» есть таблица, описывающая все возможные операции над каналами. Для РМО (сп) для некоторых операций есть дополнительное уточнение, что они используются для отладки на предприятии-изготовителе (Рис. 14), и этого нет для РМО (гр).

СПИСОК ВСЕХ ВОЗМОЖНЫХ ОПЕРАЦИЙ НАД КАНАЛАМИ	
Блокировать входящую связь	Для всех выделенных каналов, предоставляющих такую возможность
Разблокировать входящую связь	Для всех выделенных каналов, предоставляющих такую возможность
Сбросить канал(ы)	Послать команду сброс для всех выделенных каналов <b>(используется для отладки на предприятии-изготовителе)</b>
Снять захват	Отменить действие предыдущей команды. На абонентов эта команда не воздействует. <b>Используется для отладки на предприятии-изготовителе</b>
Включить трассу объекта(ов) ЗОВСл	Для всех выделенных каналов <b>(используется для отладки на предприятии-изготовителе)</b>
Выключить трассу объекта(ов) ЗОВСл	Для всех выделенных каналов <b>(используется для отладки на предприятии-изготовителе)</b>
Включить трассу объекта(ов) сигнализации	Для всех выделенных каналов <b>(используется для отладки на предприятии-изготовителе)</b>
Выключить трассу объекта(ов) сигнализации	Для всех выделенных каналов <b>(используется для отладки на предприятии-изготовителе)</b>

Рис. 14

В этом же разделе («Соединительные линии») в таблице с описанием всех возможных типов динамической информации о канале для РМО (сп)

отсутствуют некоторые дополнительные типы динамической информации, имеющиеся для РМО (гр) (Рис. 15).

СПИСОК ВСЕХ ВОЗМОЖНЫХ ТИПОВ ДИНАМИЧЕСКОЙ ИНФОРМАЦИИ О КАНАЛЕ	
<i>Рестарт</i>	<i>Рестарт по каналу (DSS-1)</i>
<i>П/п пр.</i>	<i>Полупостоянное проключение (DSS-1)</i>
<i>ПриоИсх</i>	<i>Флаг приоритета исходящей связи (ОКС №7)</i>
<i>Блокир</i>	<i>Канал заблокирован (DSS-1)</i>
<i>Занят</i>	<i>Канал занят (DSS-1)</i>
<i>Инд.</i>	<i>Индекс пары объектов, выделенных для обслуживания вызова (DSS-1)</i>
АппБлк	Аппаратная блокировка
ИсхБлк	Блокировка исходящей связи (с внешней стороны)
ВхБлк	Блокировка входящей связи
АдмИсхБлк	Административная блокировка исходящей связи (с нашей стороны)
Сост. ЗОВСл	Состояние объекта ЗОВСл
Сост. сигнализации	Состояние объекта сигнализации
ТЗСл	Флаг трассировки объекта ЗОВСл
ТСгн	Флаг трассировки объекта сигнализации

Рис. 15

#### 4.4.4. «Параллельные места»

При описании первого примера предыдущего подраздела («Локальные изменения») также можно заметить изменения, которые не связаны с описанием функциональности (Рис. 12, 13). Они связаны с тем, что описание данной компоненты двух руководств были написаны скорее всего с нуля, потому что описаны одни и те же свойства, но по-разному. К тому же, одинаковые абзацы с незначительными изменениями находятся в разных местах описания компоненты.

#### 4.5. Об использовании DRL/PR

Прежде всего, в статьях о технологии DocLine DRL/PR недостаточно полно и точно описана, а конструкции DRL/PR постоянно изменялись, так как метод является новым, постоянно совершенствуется. Поэтому было трудно использовать ее для написания документации, приходилось тестировать различные модификации конструкций языка.

Всегда существует несколько решений задачи, будь то задача программирования или задача выбора покупок в магазине. Так вот одной из задач, стоявших в данной работе при написании документации, была задача выбора тех или иных конструкций DRL/PR, которые целесообразней использовать в конкретной ситуации. Рассмотрим такие случаи.

Каждое руководство пользователя РМО делится на разделы и подразделы. Применив на начальном шаге метод DocLine, разделы были выделены в отдельные ИЭ. Встает вопрос: создавать ли группу ИЭ или переиспользуемые ИЭ сделать обязательными для включения в продукты семейства, а все остальные – необязательными для включения. Конечно, необходимо смотреть на документацию, её структуру. Так как переиспользуемых ИЭ на верхнем уровне всего два, то смысла создавать группу не было, тем более, конечных документов всего два и в оба документа

входят описания этих разделов. Если же в пакете документов несколько разделов переиспользуются в разных составах в некотором количестве конечных продуктов, отличном от двух и, например, являются схожими функциями, то есть мотив для создания группы ИЭ. А такое очень часто встречается в документации для линейки продуктов.

Рассматривая возможность «крупноблочного» переиспользования в виде точек расширения, были выделены абзацы с незначительными изменениями и добавлениями и предложение, которые использовались в обоих руководствах пользователя, но в разных частях соответствующих подразделов. Такие фрагменты текста можно переиспользовать, если создать XOR-группу ИЭ, чтобы в разных конечных документах сделать включение нужного ИЭ, либо через дополнительные конструкции языка DRL: условный блок и присваивание псевдопеременной. Но, в данном случае решение принято в пользу группы ИЭ, так как, опять же: всего два конечных документа, всего два ИЭ и в конечном продукте должно быть включение единственного ИЭ. Но, в данном случае, выделять абзац или предложение в отдельный ИЭ, – нецелесообразно. Какую-либо информацию извлечь из таких ИЭ сложно, так как абзацы и, тем более предложения, несут маленькую смысловую нагрузку, а на диаграммах вариативности отображаются.<sup>9</sup>

Остается нерешенным вопрос нумерации заголовков. При использовании конструкции `<book/>` и подключении ИЭ обязательно используется конструкция `<chapter/>`, которая в тексте отображается как глава, а внутреннюю нумерацию разделов приходится писать в заголовках разделов. Во всем есть свои плюсы и минусы. Например, при работе над документацией для РМО АЦТС «КВАНТ-Е-СОКОЛ» – автоматическая нумерация была бы, пожалуй, лучшим решением, так как руководство РМО (сп) – это выборка из подразделов руководства РМО (гр). Нумерация нарушается во второй упомянутой документации. Этот вопрос с помощью создания нумерованных списков решить не удастся, редактор выдает ошибку о неправильном контексте использования «внутренних» конструкций. Поэтому лучше нумерацию подразделов не учитывать. Наоборот, если в качестве документации взять пакет документов со специфической нумерацией, автоматическая нумерация была бы лишней.

#### ***4.6. Возникшие проблемы генерации документации***

Перед генерацией документации в различные форматы происходит валидация (проверка) на ошибки группы XML-файлов, которые участвуют в генерации документации для каждого СИП.

---

<sup>9</sup> В таких ситуациях хотелось бы использовать конструкцию для «крупноблочного» переиспользования, которая не отображается в диаграмме вариативности. См. раздел 6. Предложения по улучшению технологии DocLine.

Руководства пользователя РМО АЦТС создавались на языке DRL без шаблонов, поэтому ошибки в написании конструкций были неизбежны, но быстро устранимы.

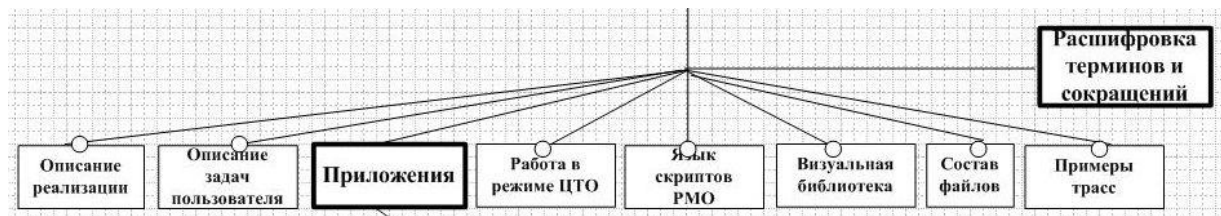
Немало трудностей вызвало использование именно языка DocBook. Особенно затронут вопрос оформления сгенерированной документации в формате .pdf. Настройка с помощью конструкций DocBook была unsuccessful, поэтому приходилось настраивать оформление титульного листа, колонтитулов, нумерацию страниц и т.д. программными средствами (то есть в файле настроек DocBook прописывается оформление заголовков, списков, таблиц и т.д. и их можно менять). Этими настройками занимался разработчик текстового редактора Романовский К.Ю. Таким образом, появилась новая версия текстового редактора.

Затронув вопрос выбора различных конструкций языка DRL в предыдущем разделе, хочется также сказать, что приходится делать выбор и между созданием точки расширения или включением условного блока текста. Ошибка при валидации возникает в том случае, когда есть необходимость в тексте точки расширения использовать конструкции языка DRL для «мелкозернистого» переиспользования (словари, каталоги), что, к сожалению, невозможно на данный момент. Поэтому нужные фрагменты документации, нуждающиеся в изменении с помощью механизмов «мелкозернистого» переиспользования технологии DocLine, выделяются в условные блоки и подключаются в нужном СИП с помощью псевдопеременных. А это так же неудобно, как и в случае с выбором между группой ИЭ и включением условного блока.

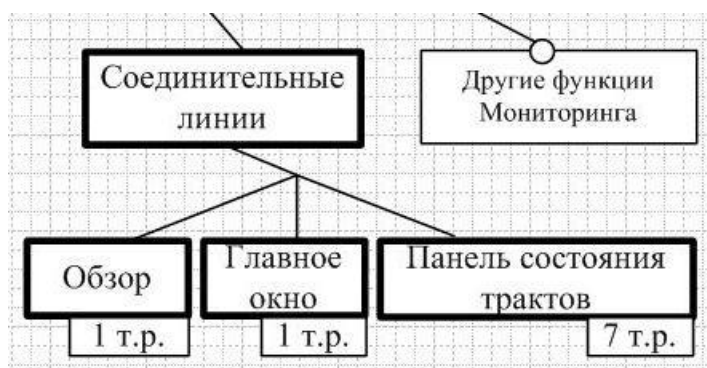
## 5. Предложения по улучшению технологии DocLine

### 1. В DRL/GR

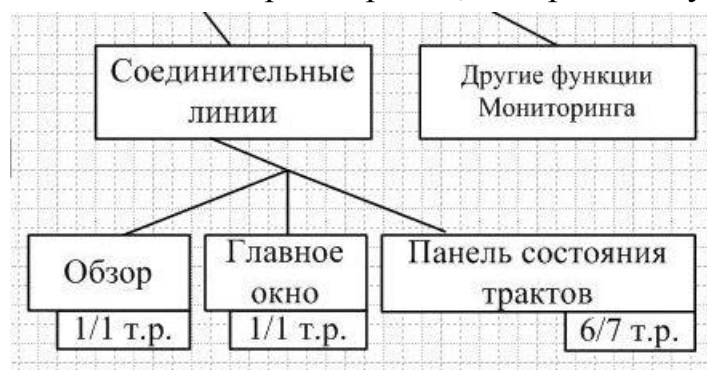
1.1. В диаграмме вариативности для большей наглядности жирной линией выделить те ИЭ, которые переиспользуются:



1.2. В диаграмме вариативности ИЭ нижнего уровня дополнить окошками, в которых указать общее количество точек расширения, которое есть у данного ИЭ:



1.3. В диаграммах продукта ИЭ нижнего уровня дополнить окошками, в которых можно более конкретно указать изменения в конечном продукте: количество точек расширения в конкретном конечном продукте/общее количество точек расширения, которое есть у данного ИЭ:



### 2. В DRL/PR

2.1. Кроме ИЭ и точек расширения, есть необходимость создания возможности «крупноблочного» переиспользования, когда нужно выделить предложение или абзац для повторного использования в одной главе или

параграфе, но для разных СИП, и в то же время, неоправданно в таких случаях записывать эти небольшие фрагменты текста в ИЭ.

Такая же необходимость есть и в том случае, когда в точке расширения нужно использовать элементы DRL (такие как словари и каталоги), что невозможно. Поэтому приходится создавать ИЭ для этого фрагмента текста и через условный блок и присваивание псевдопеременной включать его в конечный документ. Это не рационально.

В таких случаях создаются «лишние» элементы, которые отображаются на диаграммах, хотя эти небольшие фрагменты текста являются лишь отрывком в блоке текста (параграфе, разделе).

2.2. Для использования языка DRL при создании документации в разных областях должна быть возможность включения\выключения и настройки автоматической нумерации заголовков.

3. В редакторе текстов, для удобства создания документации, хорошим дополнением было бы сделать панель с кнопками вставки шаблонов (конструкций) языка DRL.

## 6. Заключение

В рамках данной работы были достигнуты следующие результаты:

- изучены технологии DocBook и DocLine, а также язык разметки DRL и инструментальные средства технологии DocLine;
- проанализирована техническая документация семейства телефонных станций ЗАО ЛАНИТ-ТЕРКОМ;
- создана верхнеуровневая, графическая модель пользовательской документации проекта «РМО семейства телефонных станций “КВАНТ-Е-СОКОЛ”» в терминах DRL/GR;
- разработана полная спецификация документации на DRL/PR и DocBook (около 200 страниц);
- настроены средства DocBook и сгенерированна документация в форматах PDF и HTML;
- предложены улучшения технологии DocLine.

Техническая сторона тестируемой технологии нуждается в улучшении, но, в целом, применение метода оставило хорошее впечатление и есть стимул дальнейшего развития технологии.



## 7. Список литературы

- [1] Clements P. & Northrop L.M. (2003). Software Product Lines. // [http://www.sei.cmu.edu/programs/pls/sw-product-lines\\_05\\_03.pdf](http://www.sei.cmu.edu/programs/pls/sw-product-lines_05_03.pdf) .
- [2] DocBook official site // <http://www.docbook.org> .
- [3] OASIS DITA Technical Committee official site. // <http://www.oasis-open.org/committees/dita> .
- [4] Marques M. Single-sourcing with FrameMaker // TECHWR-L Magazine Online, <http://www.techwr-l.com/techirl/magazine/technical/singlesourcing.html> .
- [5] Романовский К.Ю. Метод разработки документации семейств программных продуктов. // Системное программирование. Вып. 2. Сб. статей / Под ред. А.Н. Терехова, Д.Ю. Булычева. СПб.: Изд-во СПбГУ, 2006. С. 191-218.
- [6] Романовский К.Ю., Кознов Д.В. Язык DRL для проектирования и разработки документации семейств программных продуктов // Вестник СПбГУ. Сер. 10, 2007. Вып. 4. С. 110-122.
- [7] Koznov D.V., Romanovsky K. Ju. DocLine: a method for software product line documentation development // Accepted to published in Programming and Computer Software/ Editor V.P. Ivannikov, Vol. 34, 2008.
- [8] Day D., Priestley M., Shell D.A. Introduction to the Darwin Information Typing Architecture // Toward portable technical information. <http://www.106.ibm.com/developerworks/xml/library/x-dital>.
- [9] Clements, P., Northrop. L. A Framework for Software Product Line Practice. – Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University. <http://www.sei.cmu.edu/productlines/framework.html>, 2006.
- [10] Ari Jaaksi. Developing Mobile Browsers in a Product Line. // IEEE Software, Jul./Aug. 2002, pp. 73-80.
- [11] The DocBook Project. // <http://docbook.sourceforge.net>.
- [12] Кияев В.И., Кищенко Д.М., Окомин И.С. Опыт усовершенствования и стандартизации процесса создания ПО цифровых телефонных станций // Системное программирование. Вып. 2. Сб. статей / Под ред. А.Н. Терехова, Д.Ю. Булычева. СПб.: Изд-во СПбГУ, 2006. С. 219-239.
- [13] Кознов Д.В., Перегудов А.Ф., Романовский К.Ю., Кашин А.А., Тимофеев А.Е. Опыт использования UML при создании технической документации // Системное программирование. Вып. 1. Сб. статей / Под ред. А.Н. Терехова, Д.Ю. Булычева. СПб.: Изд-во СПбГУ, 2006. С. 18-35.



## Приложение 1. Фрагмент документации на языке DRL/PR

### ***ProductLine.drl***

```
<d:ProductLine name="Электронная автоматическая телефонная станция «Квант-Е-Сокол»" xmlns:d="http://math.spbu.ru/drl">
  <d:Product name="РМО 1.5 полная версия" id="complete"/>
  <d:Product name="РМО 1.5 базовая версия" id="based"/>
</d:ProductLine>
```

### ***User\_Manual.drl***

```
<d:DocumentationCore>
  <d:InfProduct name="Руководство пользователя" id="UserManual">
    <book lang="ru">
      <bookinfo>
        <title>
          Руководство пользователя <d:DictRef dictid="main"
entryid="productname"/>
        </title>
        <subtitle>
          Электронная автоматическая телефонная станция «Квант-Е-Сокол»
          Программное обеспечение
          Рабочее место техобслуживания и эксплуатации
          Руководство оператора
          БЯ1.220.0001-01 20 01
        </subtitle>
        <copyright>
          <year>2004</year>
          <holder>ЗАО ЛАНИТ-ТЕРКОМ</holder>
        </copyright>
      </bookinfo>
      <d:InfElemRef id="ref_implementation_specification"
infelemid="implementation_specification" optional="true"/>
      <d:InfElemRef id="ref_users_problem_description"
infelemid="users_problem_description" optional="true"/>
      <d:InfElemRef id="ref_applications" infelemid="applications"/>
      <d:InfElemRef id="ref_rate_CTO" infelemid="rate_CTO" optional="true"/>
    </book>
  </d:InfProduct>
  <d:Dictionary id="main" name="main_dictionary">
    <d:Entry id="productname">РМО</d:Entry>
  </d:Dictionary>
</d:DocumentationCore>
```

### ***Service\_map.drl***

```
<d:DocumentationCore>
  <d:InfElement id="service_map" name="Карта обслуживания абонентских линий">
    <section><title>3.3.2. Карта обслуживания абонентских линий</title>
      <d:InfElemRef id="ref_review_sm" infelemid="review_sm"/>
      <d:InfElemRef id="ref_main_window_sm" infelemid="main_window_sm"/>
      <d:InfElemRef id="ref_menu_sm" infelemid="menu_sm" optional="true"/>
    </section>
  </d:InfElement>

  <d:InfElement id="review_sm" name="Обзор">
    <section><title>Обзор</title>
      <para>Данное приложение предназначено для просмотра и анализа состояний абонентских линий станции.</para>
      <para>С его помощью оператор может посмотреть текущую статистику состояний в виде диаграммы.</para>
    </section>
  </d:InfElement>
```

```

    <d:Nest id="ordered_modules"/>
    <para>Приложение также включает в себя возможность автоматического
обновления текущих состояний через определённые оператором промежутки
времени.</para>
  </section>
</d:InfElement>
<d:InfElement id="main_window_sm" name="Главное окно">
  <section>
    <d:Nest id="main_win">
      <title>Главное окно</title>
      <para>Главное окно программы выглядит так:</para>
      <informalfigure>
        <graphic fileref="F:/runtime-
New_configuration/rmo/Result/illustrations/service_map/service_state_443.jpg"
/>
      </informalfigure>
    </d:Nest>
    <d:InfElemRefGroup id="sent" modifier="XOR" name="Предложение"/>
    <para><d:InfElemRef id="ref_sentence_15_sm" infelemid="sentence_15_sm"
groupid="sent" optional="true"/> На форме этого приложения <d:Nest
id="refl">вы увидите</d:Nest> все абонентские модули станции, каждому из
которых соответствует <d:Nest id="mult">некоторое множество</d:Nest>
квадратиков (как правило 128), расположенных ниже названия этого
модуля.</para>
    <d:Nest id="balk_20"/>
    <para>Каждый из квадратиков символизирует <d:Nest id="soboj"/> абонента
этого модуля, <d:Nest id="color">а его цвет</d:Nest> - состояние этого
абонента (белым цветом отображаются абоненты, состояния которых не удалось
прочитать из станции<d:Nest id="color_info"/>). <d:InfElemRef
id="ref_sentence_443_sm" infelemid="sentence_443_sm" groupid="sent"
optional="true"/></para>
    <d:Nest id="color_name"/>
    <d:InfElemRefGroup id="renewal" modifier="XOR" name="Обновление
абонентов"/>
    <d:InfElemRefGroup id="identif" modifier="XOR" name="Идентификация
абонентов"/>
    <para><d:InfElemRef id="ref_renewal_15_sm" infelemid="renewal_15_sm"
groupid="renewal" optional="true"/> <d:Nest id="renew_ab"/><d:InfElemRef
id="ref_identif_15_sm" infelemid="identif_15_sm" groupid="identif"
optional="true"/></para>
    <d:Nest id="win_legend"/>
    <para>Для просмотра легенды цветов состояний нужно нажать на кнопку
<emphasis>Легенда</emphasis> ( <inlinegraphic fileref="F:/runtime-
New_configuration/rmo/Result/illustrations/service_map/key_legend.jpg"/> ) в
верхней части окна приложения.</para>
    <para><d:Nest id="at_that">При этом</d:Nest> появится окно в котором
каждому цвету квадрата сопоставлено состояние абонента. В появившейся
панели вы увидите список, которым устанавливаются однозначные соответствия :
номер - цвет и цвет - Состояние Абонента.</para>
    <informalfigure>
      <graphic fileref="F:/runtime-
New_configuration/rmo/Result/illustrations/service_map/legend.jpg"/>
    </informalfigure>
    <d:Nest id="win_curstat"/>
    <d:InfElemRef id="ref_identif_443_sm" infelemid="identif_443_sm"
groupid="identif" optional="true"/>
    <para><d:Nest id="sooner"/>Имеется возможность посмотреть диаграмму
состояний абонентских линий и статистику по ним. Для <d:Nest
id="for_it">этого</d:Nest> необходимо нажать на кнопку
<emphasis>Диаграмма</emphasis> верхней части окна приложения, <d:Nest
id="whereupon">после чего</d:Nest> появится панель с диаграммой, на которой

```

будет изображена диаграмма по цветам состояний и количеству абонентских линий, находящихся в соответствующем состоянии.

```

</informalfigure>
<graphic fileref="F:/runtime-
New_configuration/rmo/Result/illustrations/service_map/current_statistics.jpg
"/>
</informalfigure>
<para>Для обновления состояний абонентских линий необходимо нажать
кнопку Обновить в верхней части окна.</para>
<d:InfElemRef id="ref_renewal_443_sm" infelemid="renewal_443_sm"
groupid="renewal" optional="true"/>
<d:Nest id="choise_module"/>
</section>
</d:InfElement>
<d:InfElement id="sentence_15_sm" name="предложение15">
В нижней части окна приложения есть информация о названии станции.
</d:InfElement>
<d:InfElement id="renewal_443_sm" name="обновление443">
<para>Чтобы активизировать автоматическое обновление состояний
абонентских линий необходимо нажать на правую часть этой кнопки и в
появившемся меню выбрать необходимый период времени обновления (из
представленных: 1, 5, 10 или 30 секунд). При этом будут обновляться состояния
только тех абонентов, которые были доступны при последней загрузке данных из
станции. Если же хочется обновить состояния по всем абонентским модулям
станции (в случае если какой-то модуль во время последней загрузки был
недоступен), то необходимо выбрать пункт меню Обновить
все. Для прекращения автоматического обновления необходимо в том
же меню выбрать Стоп. Следует заметить, что после
обновления состояний и при открытом окне статистики абонентских линий,
последняя обновляется автоматически.</para>
</d:InfElement>
<d:InfElement id="identif_15_sm" name="идентификация15">
Для удобства идентификации абонента есть возможность посмотреть его номер
и расшифровку состояния. Для этого подведите указатель мыши к
соответствующему абоненту, вызовется всплывающая подсказка, которая содержит
указанную выше информацию. Для получения более полной информации по
интересующему вас абоненту имеется возможность перейти в приложение
Абонентская панель (см. разд. «3.3.1 Абонентская панель»). Переход в
приложение Абонентская панель осуществляется нажатием правой кнопки мыши на
интересующем абоненте и в появившемся меню нажать на пункт
Абонент.
</d:InfElement>
<d:InfElement id="menu_sm" name="Меню окна">
<section><title>Меню</title>
<para>Главное меню окна «Состояния
абонентов».</para>
<para>Главное меню окна «Состояния абонентов» представлено
пиктограммами в верхней части окна.</para>
<para><command>Главное меню окна «Состояния абонентов»</command></para>
<informaltable>
<tgroup cols="2" align="left" colsep="1" rowsep="1">
<colspec colnum="2" colname="c2"/>
<thead>
<row><entry>Элемент
меню/подменю</entry><entry>Описание</entry></row>
</thead>
<tbody>
<row>
<entry>Обновить -> Обновить все</entry>
<entry>Обновление состояний всех модулей</entry>
</row>
<row>

```

<div> <div>Обновить -&gt; Модули...</div> <div>Открытие окна «Выбор модулей»</div> </div>
---

<div> <div>Легенда</div> <div>Открытие окна «Легенда»</div> </div>
--

<div> <div>Всплывающие меню</div> <div>окна</div> <div>«Состояния абонентов».</div> </div>
--

Окно «Состояния абонентов» имеет два всплывающих меню: «Модуль» и «Абонент». Всплывающее меню «Модуль» активизируется при установке указателя мышки на название выбранного модуля и нажатии правой кнопки. Всплывающее меню «Абонент» активизируется при установке указателя мышки квадратик абонента и нажатии правой кнопки

## Decoding.drl

```

<d:DocumentationCore>
  <d:InfElement id="decoding" name="Расшифровка терминов и сокращений">
    <chapter>
      <title>Расшифровка терминов и сокращений</title>
      <para id="TCPIP" xreflabel="TCP/IP"><d:DirRef templateid="term"
entryid="TCPIP"/></para>
      <para id="XML" xreflabel="XML"><d:DirRef templateid="term"
entryid="XML"/></para>
      <para id="ДВО" xreflabel="ДВО"><d:DirRef templateid="term"
entryid="ДВО"/></para>
      <para id="ПО" xreflabel="ПО"><d:DirRef templateid="term"
entryid="ПО"/></para>
      <para id="PMO" xreflabel="PMO"><d:DirRef templateid="term"
entryid="PMO"/></para>
      <para id="config" xreflabel="конфиг"><d:DirRef templateid="term"
entryid="config"/></para>
    </chapter>
  </d:InfElement>
  <d:Directory id="terms" name="Термины и сокращения">
    <d:Entry id="TCPIP">
      <d:Attr id="abbr">TCP/IP</d:Attr>
      <d:Attr id="decod">Семейство протоколов связи, применяемых в Интернет
и в локальных сетях передачи данных, основанных на базовом протоколе
IP</d:Attr>
    </d:Entry>
    <d:Entry id="ДВО">
      <d:Attr id="abbr">ДВО</d:Attr>
      <d:Attr id="decod">Дополнительные виды обслуживания. Сервисные
услуги, предоставляемые абонентам АТС помимо основной – доступа к ТфОП. Это
могут быть такие услуги, как будильник, конференц-связь, передача АОН и
т.п.</d:Attr>
    </d:Entry>
    <d:Entry id="PMO">
      <d:Attr id="abbr">PMO</d:Attr>
      <d:Attr id="decod">Рабочее место оператора</d:Attr>
    </d:Entry>
    <d:Entry id="config">
      <d:Attr id="abbr">конфиг</d:Attr>
      <d:Attr id="decod">Совокупность статических данных всех программных
объектов модуля</d:Attr>
    </d:Entry>
  </d:Directory>

```

```

</d:Directory>
<d:DirTemplate id="term" directoryid="terms">
  <d:AttrRef attrid="abbr"/> - <d:AttrRef attrid="decod"/>
</d:DirTemplate>
</d:DocumentationCore>

```

### ***UserManual\_based.drl***

```

<d:ProductDocumentation>
  <d:FinalInfProduct id="UserManual_based" infproductid="UserManual">
    <d:Adapter infelemrefid="ref_state_sections_b">
      <d:Insert-After nestid="where_use">
        <para>(используется для отладки на предприятии-изготовителе)</para>
      </d:Insert-After>
    </d:Adapter>
    <d:Adapter infelemrefid="ref_main_window_sm">
      <d:Insert-After nestid="soboj">собой</d:Insert-After>
      <d:Insert-After nestid="sooner">Как уже говорилось ранее:</d:Insert-
After>
    </d:Adapter>
    <d:Adapter infelemrefid="ref_sentence_443_sm"/>
    <d:Adapter infelemrefid="ref_renewal_443_sm"/>
    <d:Adapter infelemrefid="ref_identif_443_sm"/>
  </d:FinalInfProduct>
  <d:Dictionary id="main" name="main_dictionary">
    <d:Entry id="productname">ПМО П-443</d:Entry>
  </d:Dictionary>
</d:ProductDocumentation>

```

### ***UserManual\_complete.drl***

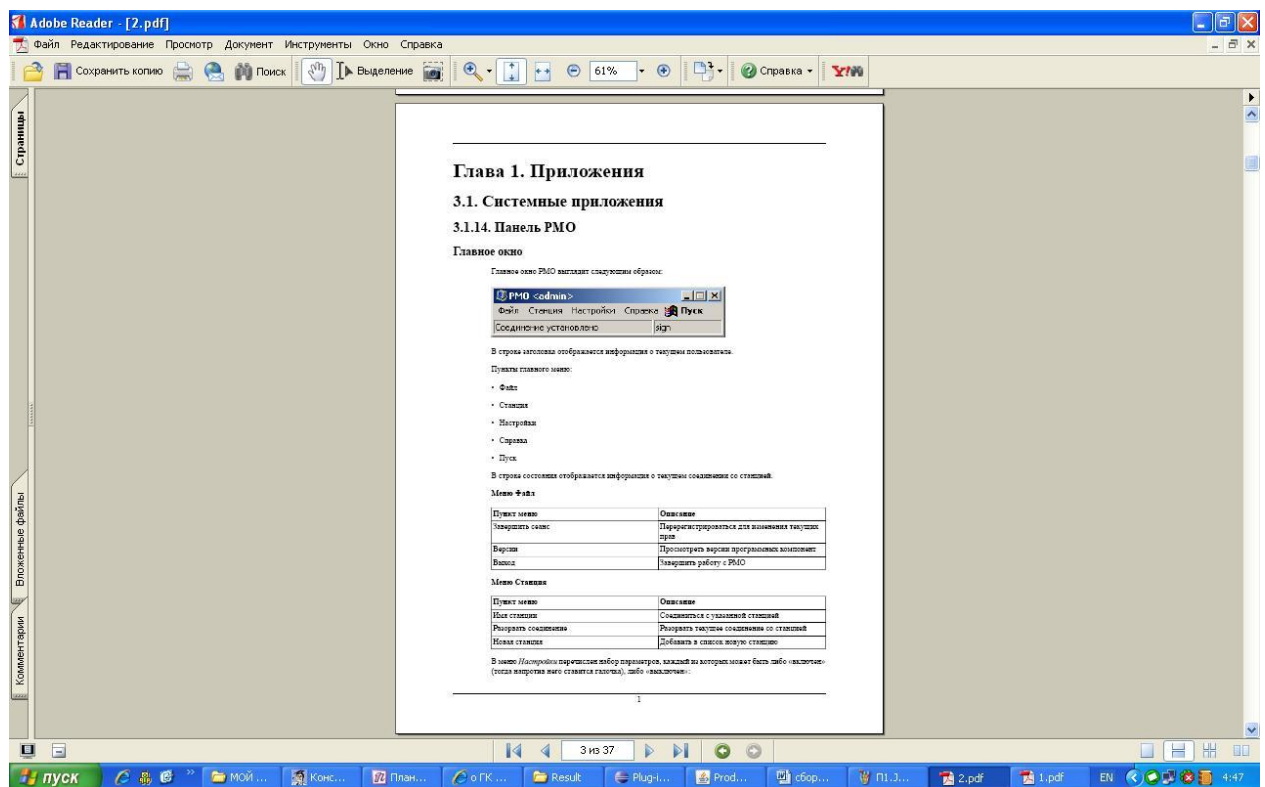
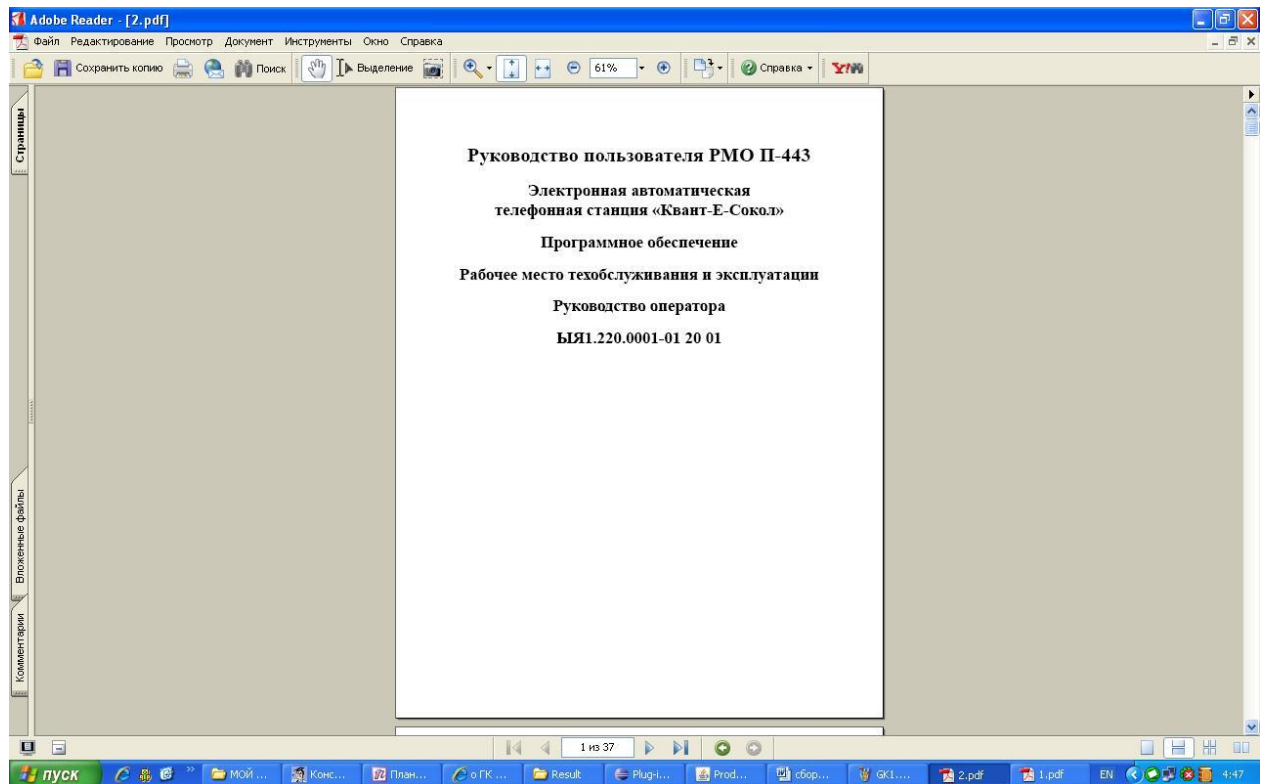
```

<d:ProductDocumentation>
  <d:FinalInfProduct id="UserManual_complete" infproductid="UserManual">
    <d:Adapter infelemrefid="ref_implementation_specification"/>
    <d:Adapter infelemrefid="ref_users_problem_description"/>
    <d:Adapter infelemrefid="ref_rate_CTO"/>
    <d:Adapter infelemrefid="ref_join_in_applications"/>
    <d:Adapter infelemrefid="ref_join_in_sysappl"/>
    <d:Adapter infelemrefid="ref_join_in_embsoft"/>
    <d:Adapter infelemrefid="ref_join_in_monitoring"/>
    <d:Adapter infelemrefid="ref_review_sp"/>
    <d:Adapter infelemrefid="ref_review_b"/>
    <d:Adapter infelemrefid="ref_state_sections_b"/>
    <d:Adapter infelemrefid="ref_main_window_sm">
      <d:Replace-Nest nestid="main_win">
        <title>Главное окно "Состояния абонентов"</title>
        <para>Главное окно программы имеет следующий вид:</para>
        <informalfigure>
          <graphic fileref="F:/runtime-
New_configuration/rmo/Result/illustrations/service_map/service_state_15.jpg"/
>
        </informalfigure>
      </d:Replace-Nest>
      <d:Replace-Nest nestid="refl">отображены</d:Replace-Nest>
      <d:Replace-Nest nestid="mult">набор</d:Replace-Nest>
      <d:Replace-Nest nestid="color">цвет квадрата</d:Replace-Nest>
      <d:Insert-After nestid="color_info">, полная информация о расцветках
отражена в окне «Легенда»</d:Insert-After>
      <d:Insert-After nestid="win_legend"><para><emphasis>Окно
"Легенда"</emphasis></para></d:Insert-After>
      <d:Replace-Nest nestid="at_that">После нажатия на кнопку</d:Replace-
Nest>

```

```
<d:Insert-After nestid="win_curstat"><para><emphasis>Окно "Текущая
статистика по абонентским линиям"</emphasis></para></d:Insert-After>
<d:Replace-Nest nestid="for_it">просмотра</d:Replace-Nest>
<d:Replace-Nest nestid="whereupon">нажатие вызовет</d:Replace-Nest>
</d:Adapter>
<d:Adapter infelemrefid="ref_sentence_15_sm"/>
<d:Adapter infelemrefid="ref_renewal_15_sm"/>
<d:Adapter infelemrefid="ref_identif_15_sm"/>
<d:Adapter infelemrefid="ref_menu_sm"/>
</d:FinalInfProduct>
</d:ProductDocumentation>
```

## Приложение 2. Пример №1 документации в формате PDF



## Приложение 3. Пример №2 документации в формате PDF

