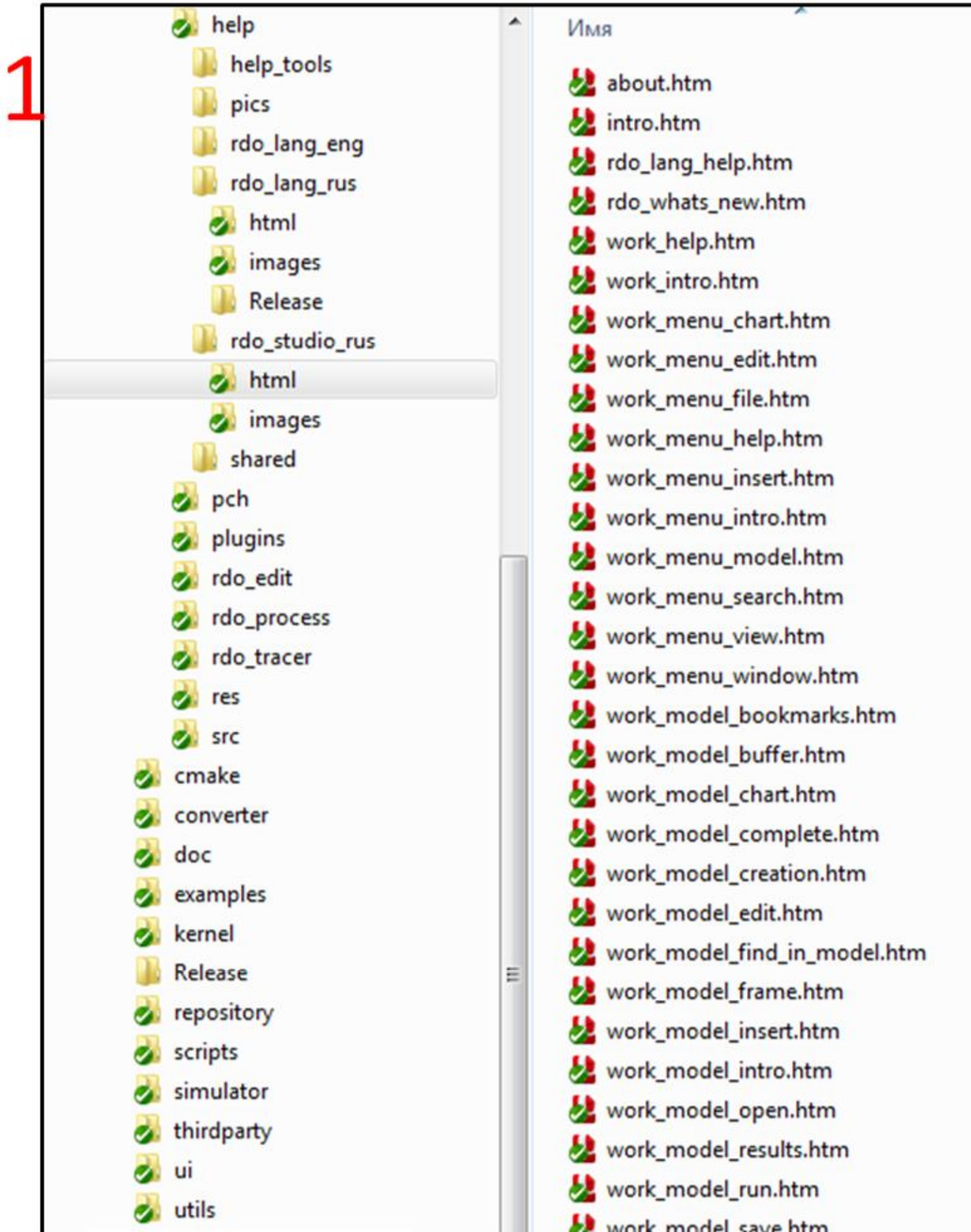
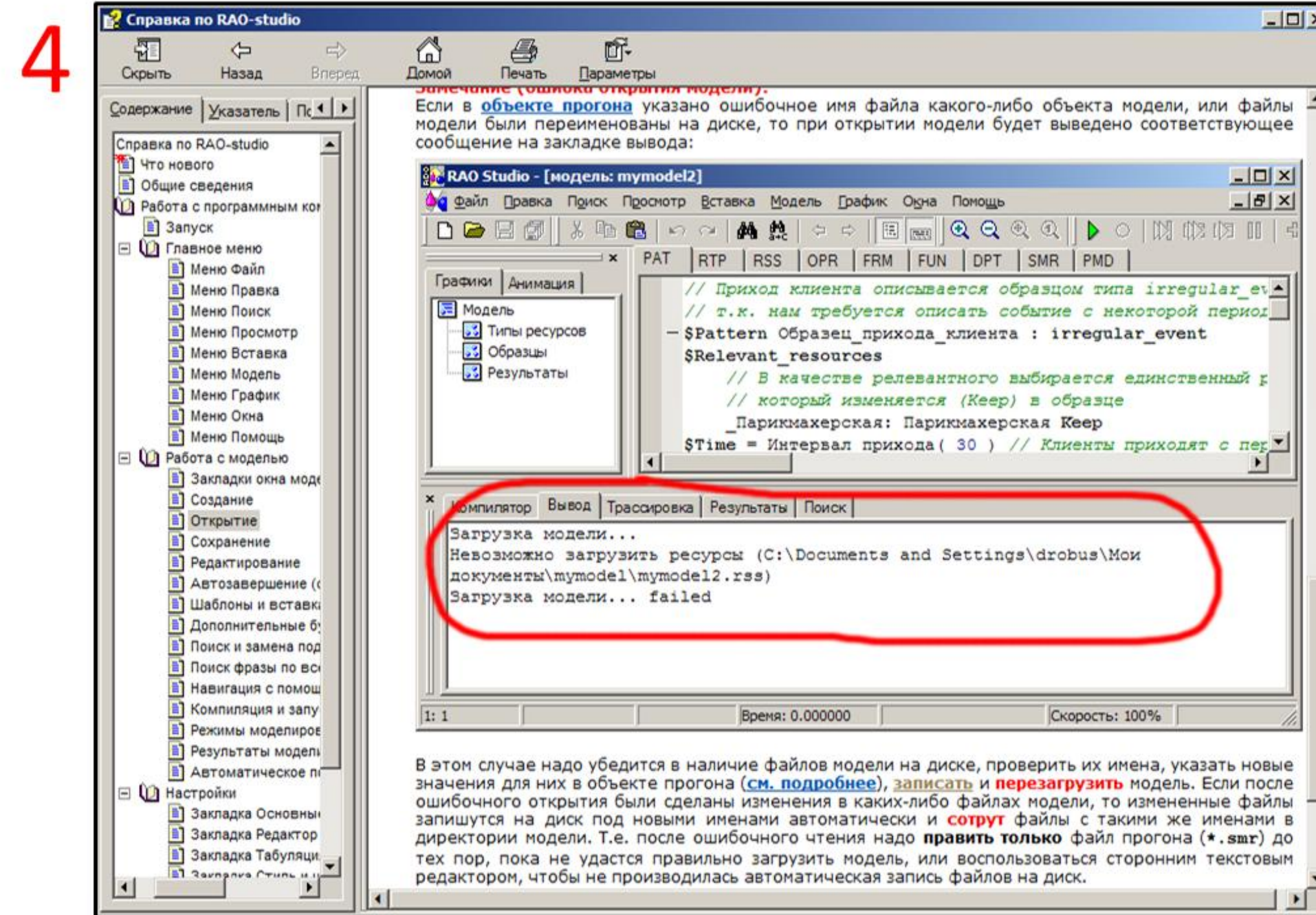


# Постановка задачи

## Плоская файловая структура

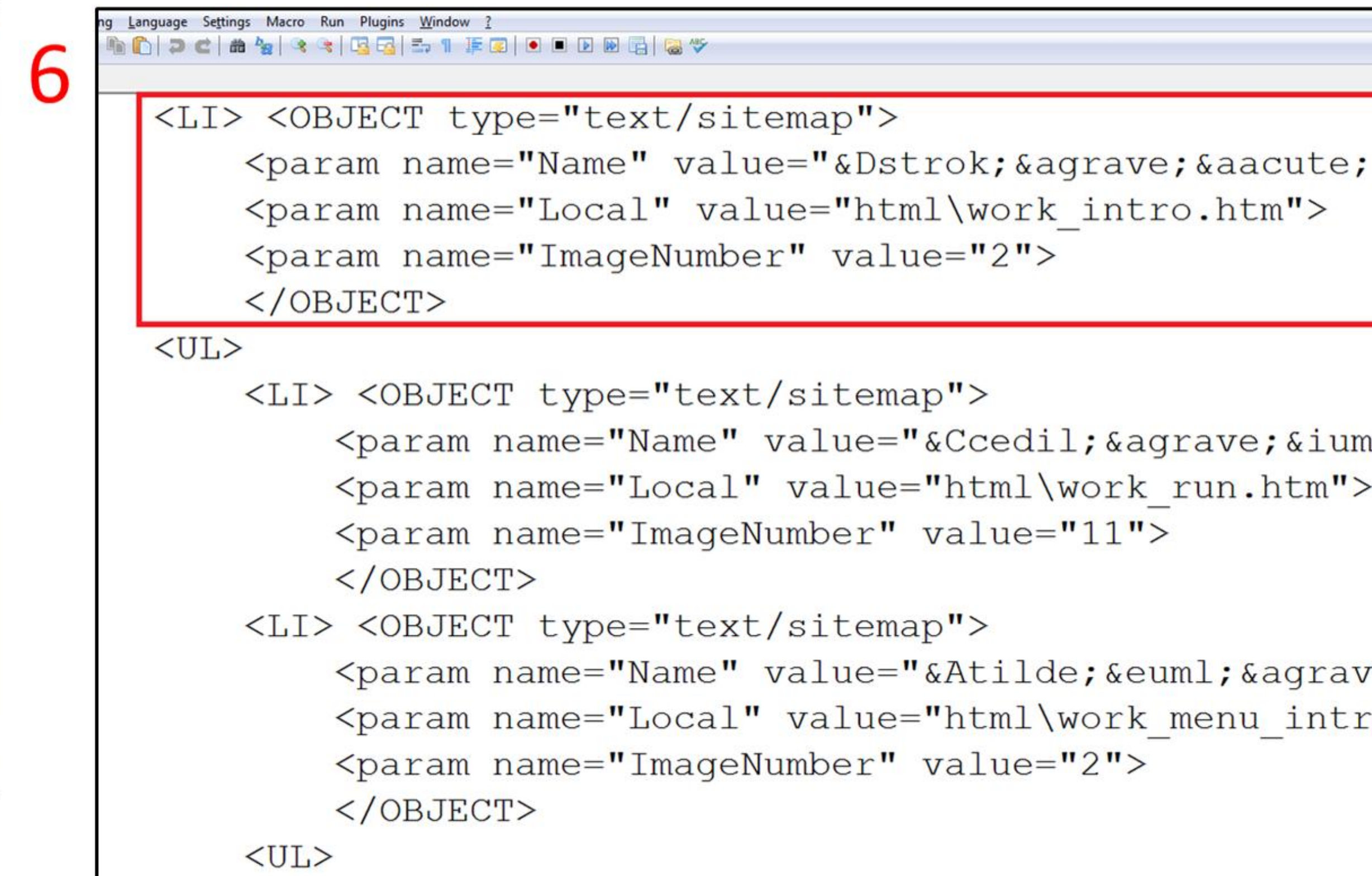


## Тело документации неадекватно текущему состоянию РДО

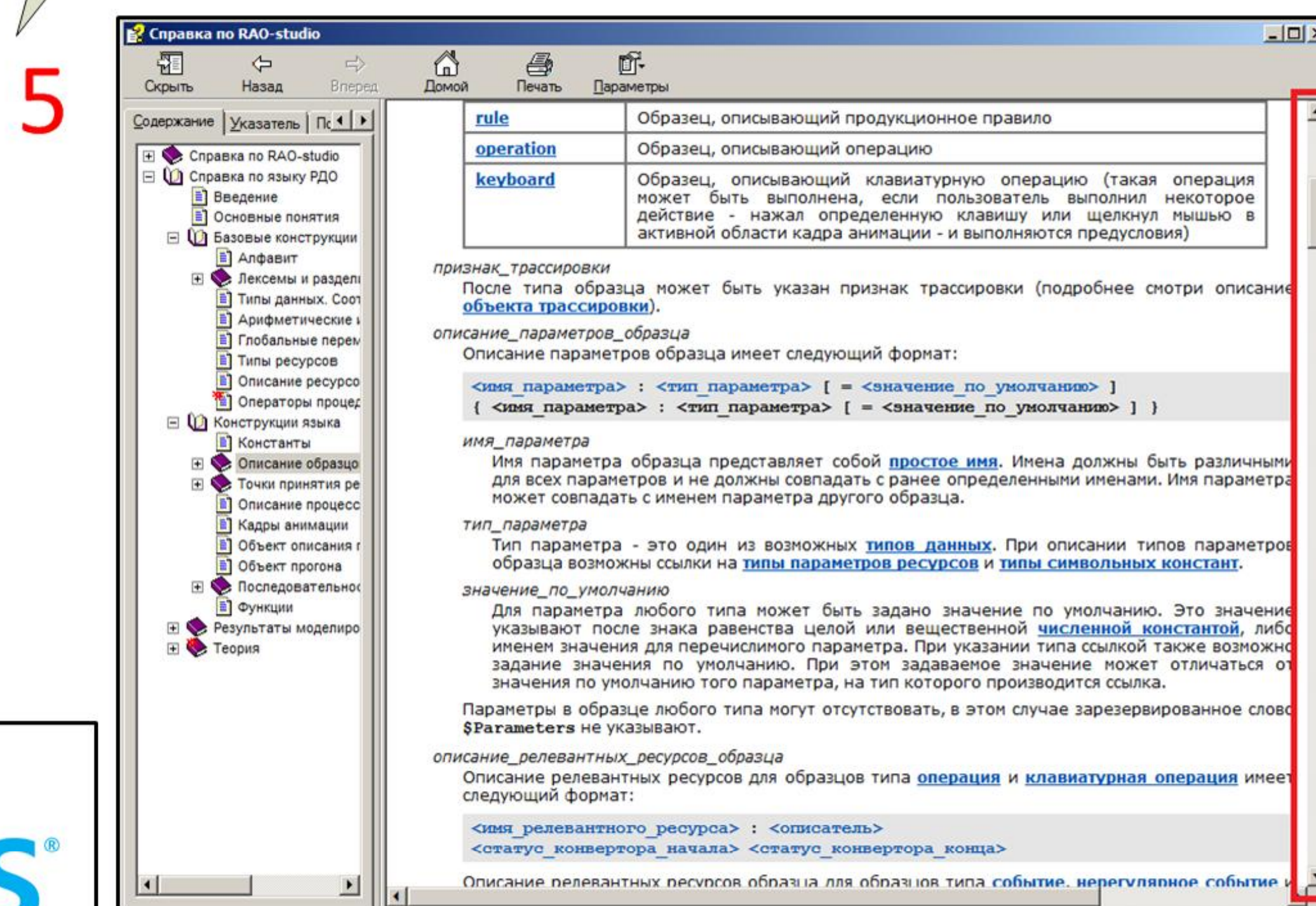


## Неудобство редактирования структуры документации

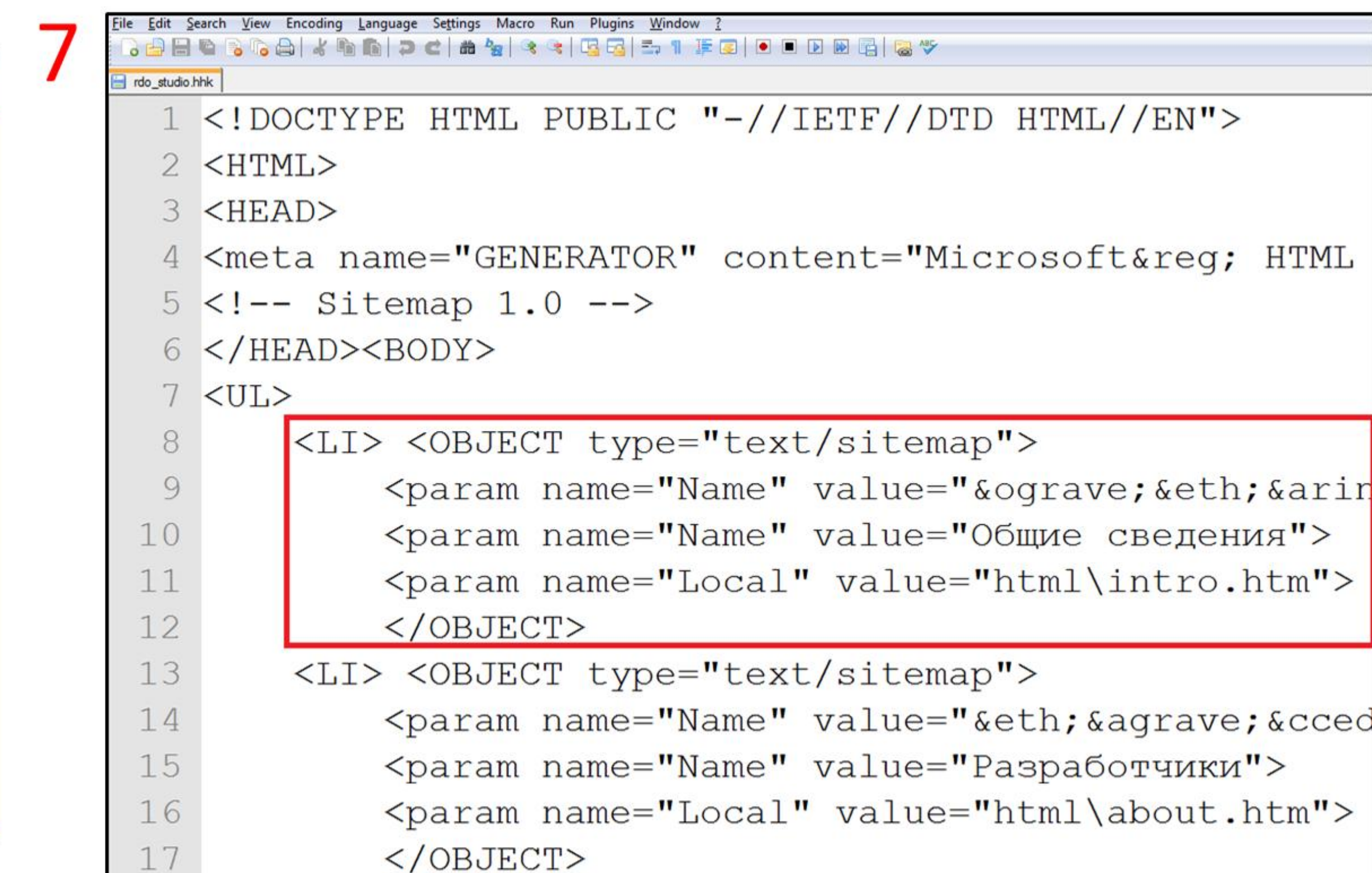
### Содержание



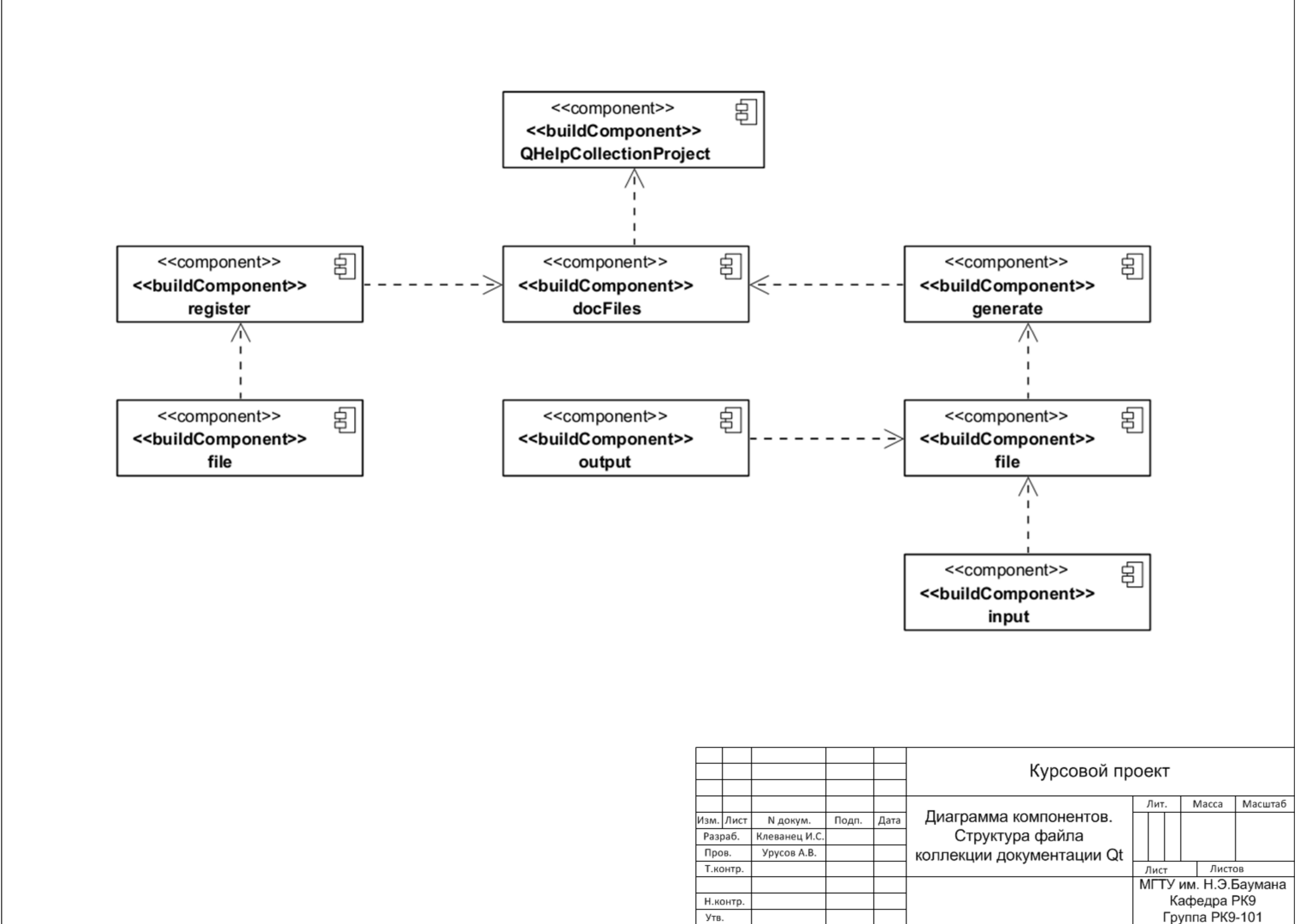
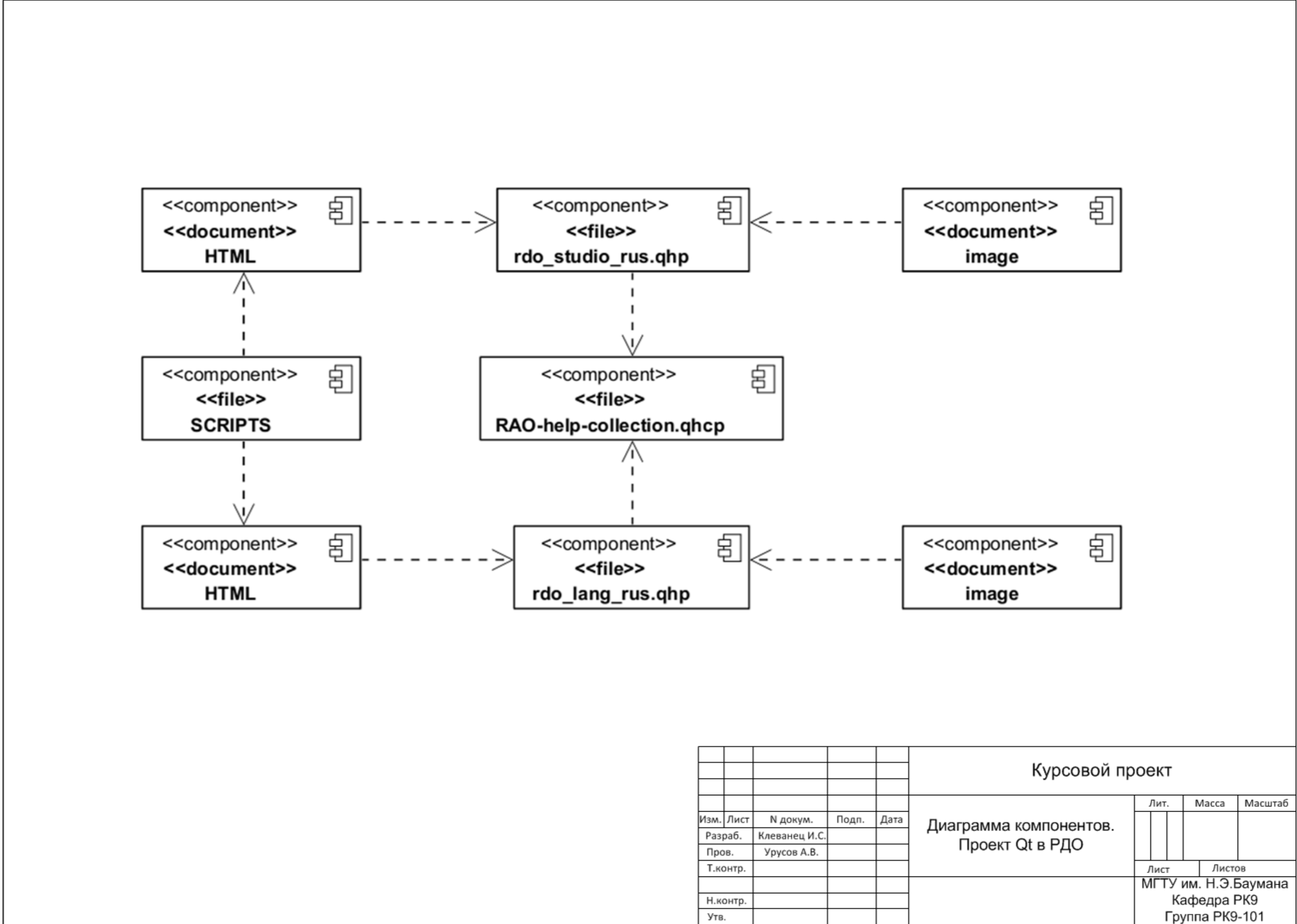
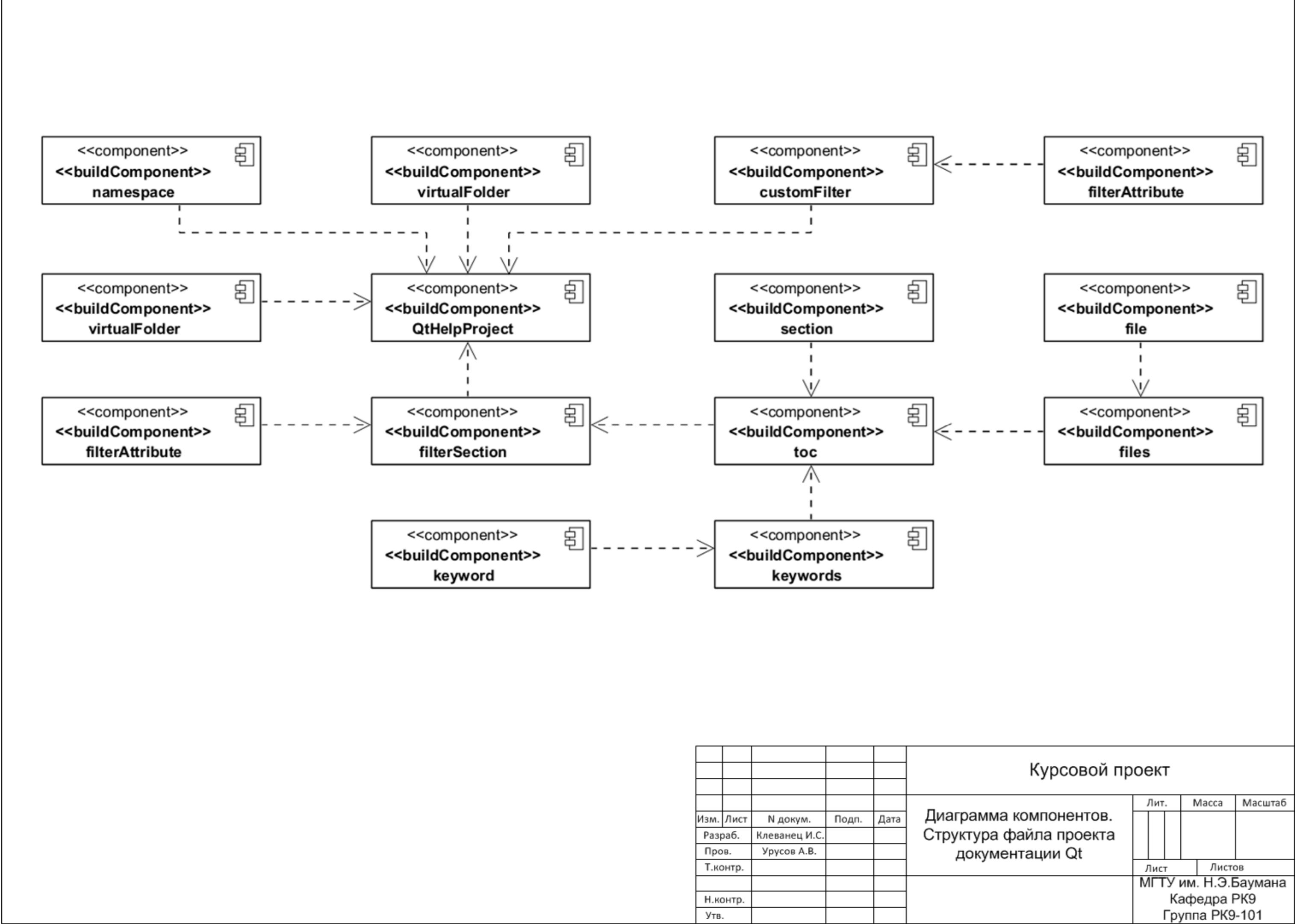
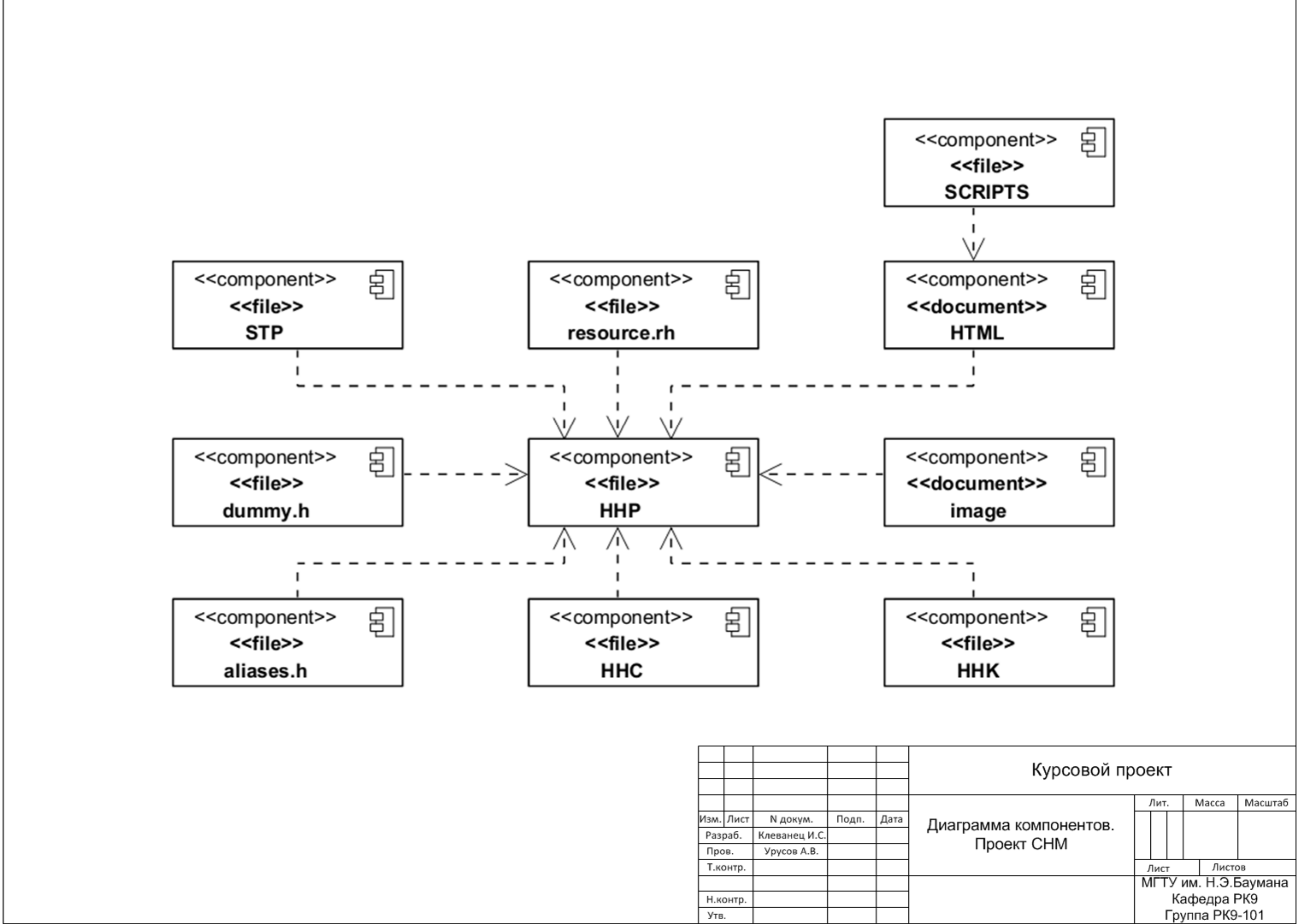
## Большие и неудобные разделы документации



### Список ключевых слов









# Перепроектирование докуменатации

Справка по RAO-studio

выражение времени представляет собой **арифметическое выражение** целого или вещественного типа, по котор времени вместе с зарезервированным словом \$Time и знаком равенства не указывают.

тело\_образца

Тело образца имеет следующий формат:

```
<имя_релевантного_ресурса> <правило_использования>
{ <имя_релевантного_ресурса> <правило_использования> }
```

имя\_релевантного\_ресурса

Имена релевантных ресурсов должны следовать в том порядке, в котором они указаны в описании **релевантных**

правило\_использования

Правило использования должно быть указано для каждого ресурса и в зависимости от типа образца имеет форм для образца типа operation и keyboard:

```
<предусловие> [ Convert_begin <список_инструкций> ] [ Convert_end <список_инструкций> ]
```

для образца типа event:

```
Convert_event <список_инструкций>
```

для образца типа rule:

```
<предусловие> [ Convert_rule <список_инструкций> ]
```

предусловие

Предусловие записывают в следующем формате:

```
Choice from <логическое_выражение> [ <способ_выбора> ]
или
Choice NoCheck [ <способ_выбора> ]
```

логическое\_выражение

**Логическое выражение** определяет предусловие использования ресурса (возможность выполнения опер вообще отсутствием Choice from.

способ\_выбора

Способ выбора задается также, как и **после описания релевантных ресурсов образца**. Если он не ука ресурса после **логического выражения**. Способ выбора может быть задан также в случае, когда опи использовать любой способ выбора: все равно он не будет учитываться, и результат будет одинаковым выбора, если же не удовлетворяет, то также независимо от способа выбора ресурс не может быть исполь Если способ выбора указан после описания релевантных ресурсов образца, в теле образца он должен отс

Разбиение

Qt Assistant

File Edit View Go Bookmarks Help

Content... Ind... Bookma... Sea...

Contents

- Справка по RAO-studio
- Справка по языку РДО
- Введение
- Основные понятия
- Базовые конструкции языка
- Конструкции языка
- Кадры анимации
- Константы
- Образцы
- Операция
- Продукционное правило
- Клавиатурная операция
- События
- Объект описания показателей
- Объект прогона
- Последовательности
- Процессы обслуживания
- Точки принятия решений
- Функции
- Результаты моделирования
- Теория

### ТИП ОБРАЗЦА "ОПЕРАЦИЯ"

Образцы типа **operation** описывают модифицированные продукционные правила, учитывающие временные связи. Опе изменения состояния ресурсов в начале и конце соответствующего действия.

Образец типа **operation** содержит список релевантных ресурсов (то есть ресурсов, участвующих в описываемой образце преобразования образов, участвующих в операции, в ее начале и в конце. В данном типе образцов имеются также выр

Формат образца:

```
$Pattern <имя_образца> : operation [ <признак_трассировки> ]
[ $Parameters <описание_параметров_образца> ]
$Relevant_resources <описание_релевантных_ресурсов_образца>
[ <способ_выбора> ]
$Time = <выражение_времени>
$Body <тело_образца>
$End
```

тело\_образца

Тело событий и образцов имеет следующий формат:

```
<имя_релевантного_ресурса> <предусловие> [ Convert_begin <список_инструкций> ] [ Convert_end <сп
```

имя\_релевантного\_ресурса

Имена релевантных ресурсов должны следовать в том порядке, в котором они указаны в описании **релевантных**

Qt Assistant

File Edit View Go Bookmarks Help

Content... Ind... Bookma... Sea...

Contents

- Справка по RAO-studio
- Справка по языку РДО
- Введение
- Основные понятия
- Базовые конструкции языка
- Конструкции языка
- Кадры анимации
- Константы
- Образцы
- Операция
- Продукционное правило
- Клавиатурная операция
- События
- Объект описания показателей
- Объект прогона
- Последовательности
- Процессы обслуживания
- Точки принятия решений
- Функции
- Результаты моделирования
- Теория

### ТИП ОБРАЗЦА "ПРОДУКЦИОННОЕ ПРАВИЛО"

Образцы типа **rule** описывают обычные продукционные правила. Они служат для изменения состоян вычисления времени операции. Это связано с тем, что действие выполняется мгновенно, как только в

Образец типа rule содержит список релевантных ресурсов, предусловия начала операции (задаваем формат:

```
$Pattern <имя_образца> : rule [ <признак_трассировки> ]
[ $Parameters <описание_параметров_образца> ]
$Relevant_resources <описание_релевантных_ресурсов_образца>
[ <способ_выбора> ]
$Body <тело_образца>
$End
```

тело\_образца

Тело событий и образцов имеет следующий формат:

```
<имя_релевантного_ресурса> <предусловие> [ Convert_rule <список_инструкций> ]
```

имя\_релевантного\_ресурса

Имена релевантных ресурсов должны следовать в том порядке, в котором они указаны в описа

Справка по RAO-studio

### ОПЕРАТОРЫ ПРОЦЕДУРНОГО ПРОГРАММИРОВАНИЯ

Для описания алгоритмов в данной версии языка есть возможность использования

- условный оператор** - if...else;
- оператор объявления и инициализации локальных переменных;**
- оператор цикла** - for;
- оператор прерывания** - break;
- оператор возврата значения** - return;
- оператор выделения области локальной памяти** - { ... };
- оператор вызова подпрограммы;**

Данные операторы можно использовать при описании алгоритмических зависи

#### Условный оператор

Описание условного оператора имеет следующий формат:

```
if (<условие>)
    <первая_ветвь_исполнения>;
{else <вторая_ветвь_исполнения>;}
```

условие

Логическое выражение, определяющее выполнение первой ветви **процеду**

первая\_ветвь\_исполнения

Одна **процедура** или список процедур, заключенный в **оператор выделения**

вторая\_ветвь\_исполнения

Одна **процедура** или список процедур, заключенный в **оператор выделения**

#### Примеры

**Вычисление наибольшего числа** из двух данных:

```
$Parameters
INPUT_1 : integer
INPUT_2 : integer
$Body
if (A>B) return INPUT_1;
esle return INPUT_2;
$End
```

**Проверка состояния канала:**

```
$Parameters
СМО : Тип_СМО
$Body
if (СМО.состояние_канала == Свободен)
    return true;
else
    return false;
$End
```

### Операторы объявления и инициализации локальной переменной

Инкапсуляция

Qt Assistant

File Edit View Go Bookmarks Help

Content... Ind... Bookma... Sea...

Contents

- Справка по RAO-studio
- Справка по языку РДО
- Введение
- Основные понятия
- Базовые конструкции языка
- Алфавит
- Арифметические и логичес...
- Глобальные переменные и ...
- Лексемы и разделители
- Процедурное программиро...
- Инструкции используе...
- Ресурсы
- Типы данных. Соответствие т...
- Типы ресурсов
- Конструкции языка
- Кадры анимации
- Элементы отображения
- Константы
- Образцы
- Операция
- Продукционное правило
- Клавиатурная операция
- События
- Объект описания показате...
- Объект прогона
- Последовательности
- Процессы обслуживания
- Точки принятия решений
- Функции
- Результаты моделирования
- Теория

### ОПЕРАТОРЫ И ИНСТРУКЦИИ ПРОЦЕДУРНОГО ПРОГРАММИРОВАНИЯ

Для описания алгоритмов в данной версии языка есть возможность использования **операторов** процедурного (имп

- условный оператор** - if...else;
- оператор объявления и инициализации локальных переменных;**
- оператор цикла** - for;
- оператор прерывания** - break;
- оператор возврата значения** - return;
- оператор выделения области локальной памяти** - { ... };
- оператор вызова подпрограммы;**

Данные операторы можно использовать при описании алгоритмических зависимостей в теле образцов и функций

#### Условный оператор

Описание условного оператора имеет следующий формат:

```
if (<условие>)
    <первая_ветвь_исполнения>;
{else <вторая_ветвь_исполнения>;}
```

условие

Логическое выражение, определяющее выполнение первой ветви **процедур** условного оператора

первая\_ветвь\_исполнения

Одна **процедура** или список процедур, заключенный в **оператор выделения области локальной памяти**. Дан

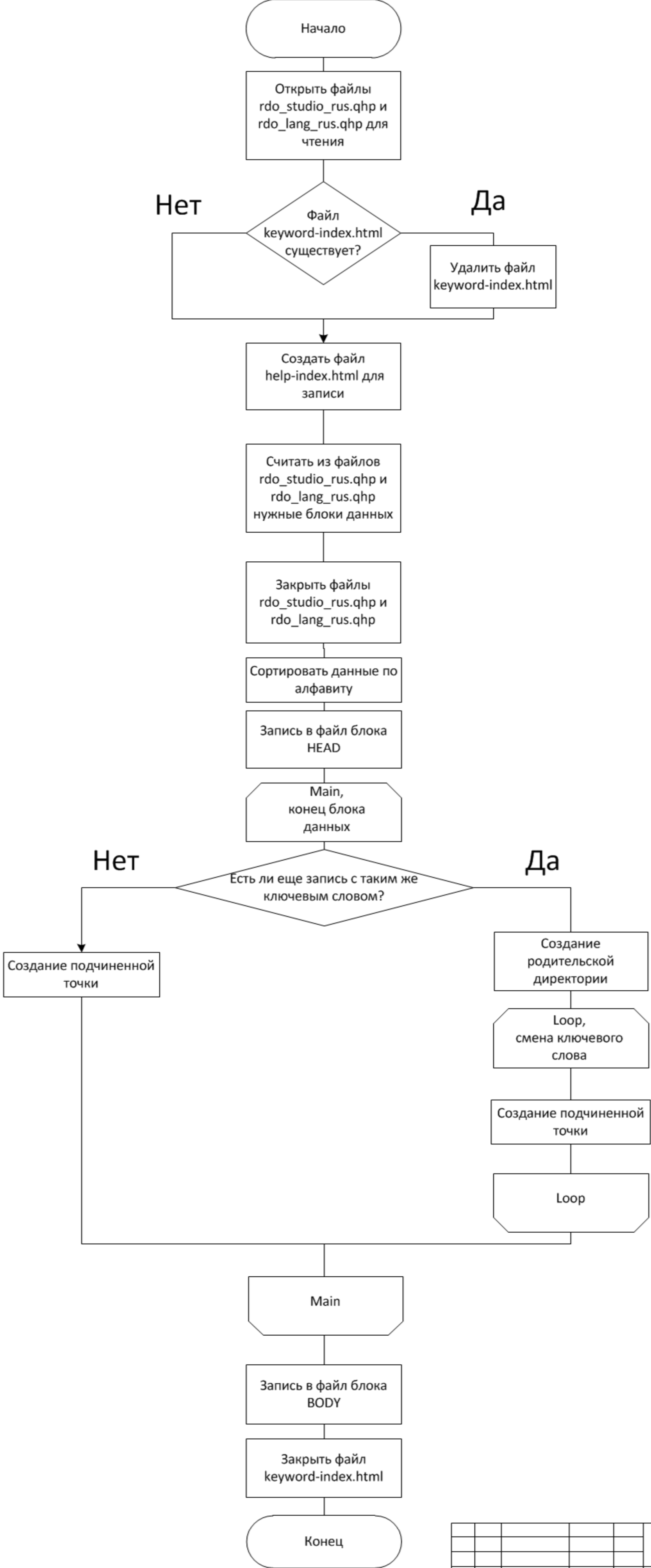
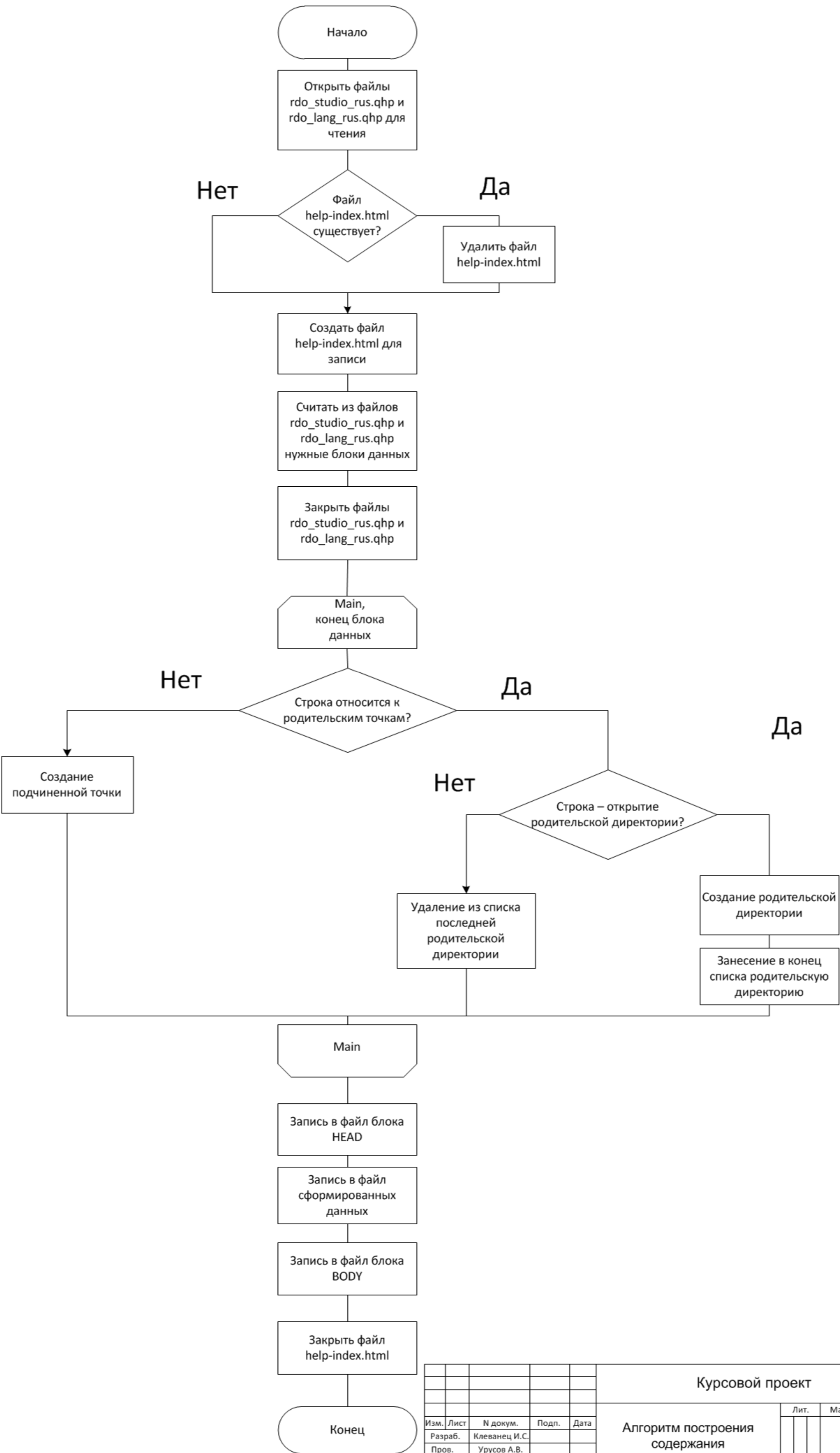
вторая\_ветвь\_исполнения

Одна **процедура** или список процедур, заключенный в **оператор выделения области локальной памяти**. Дан

#### Примеры

### Операторы объявления и инициализации локальной переменной







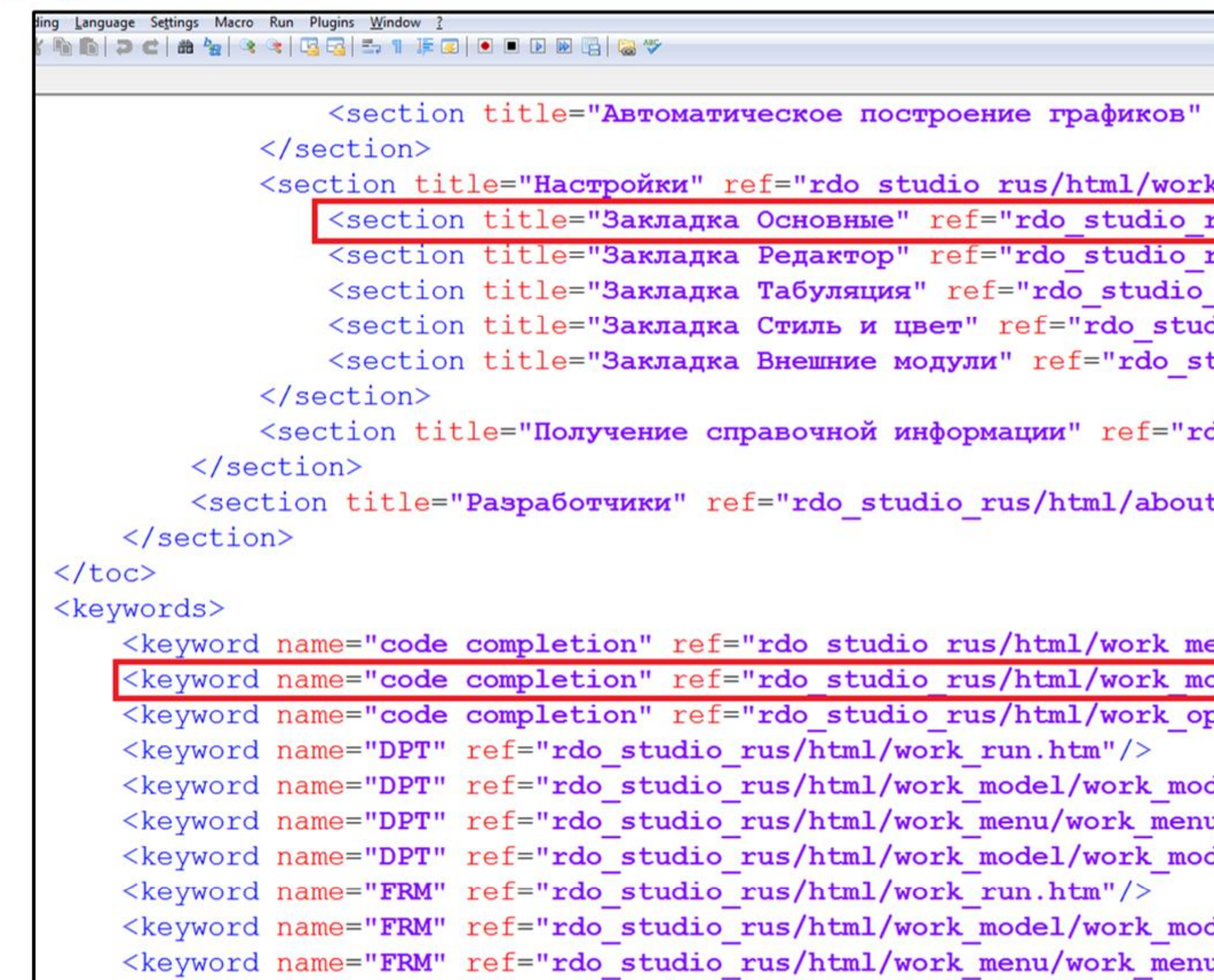
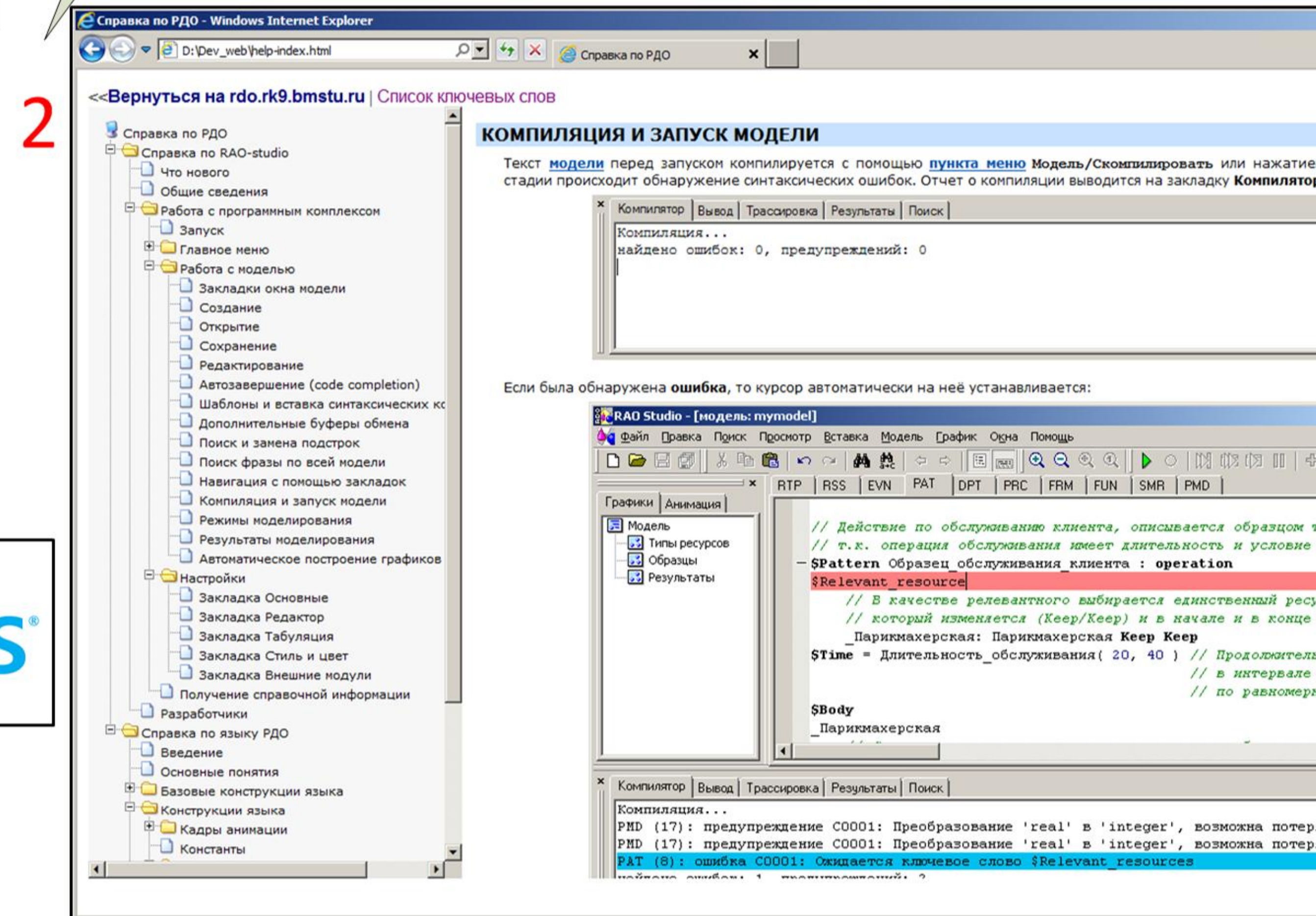
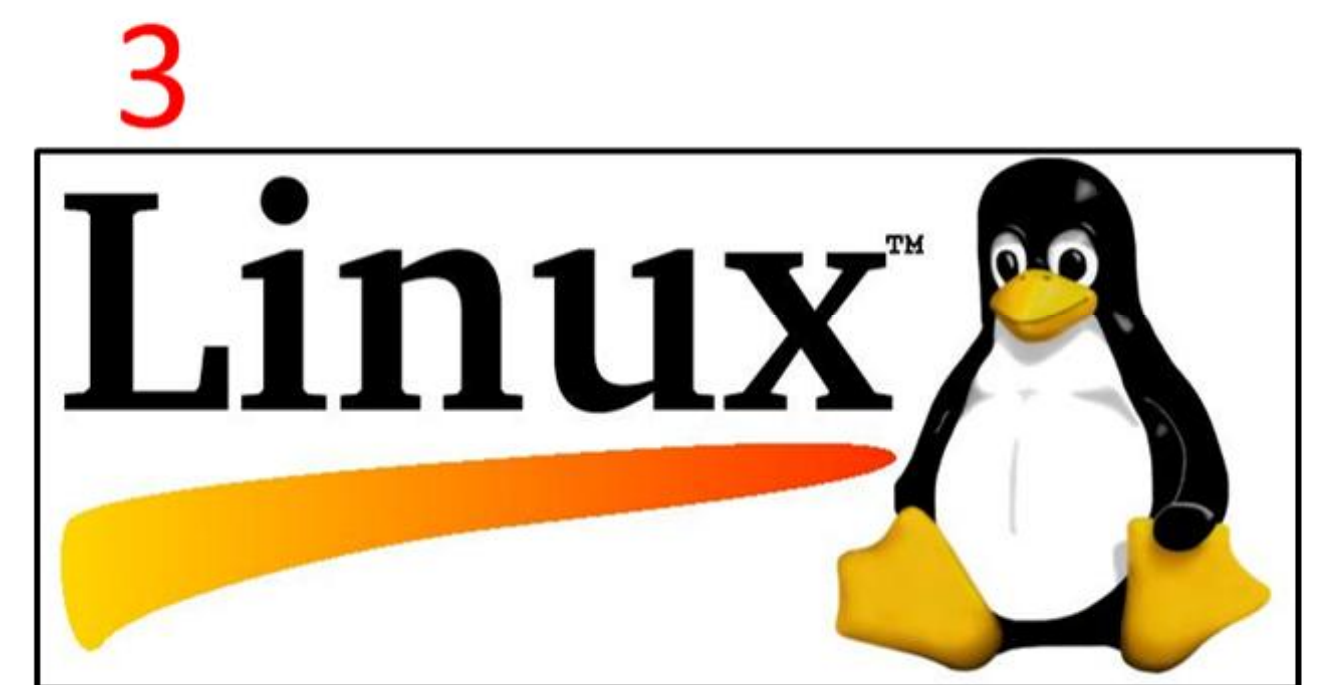
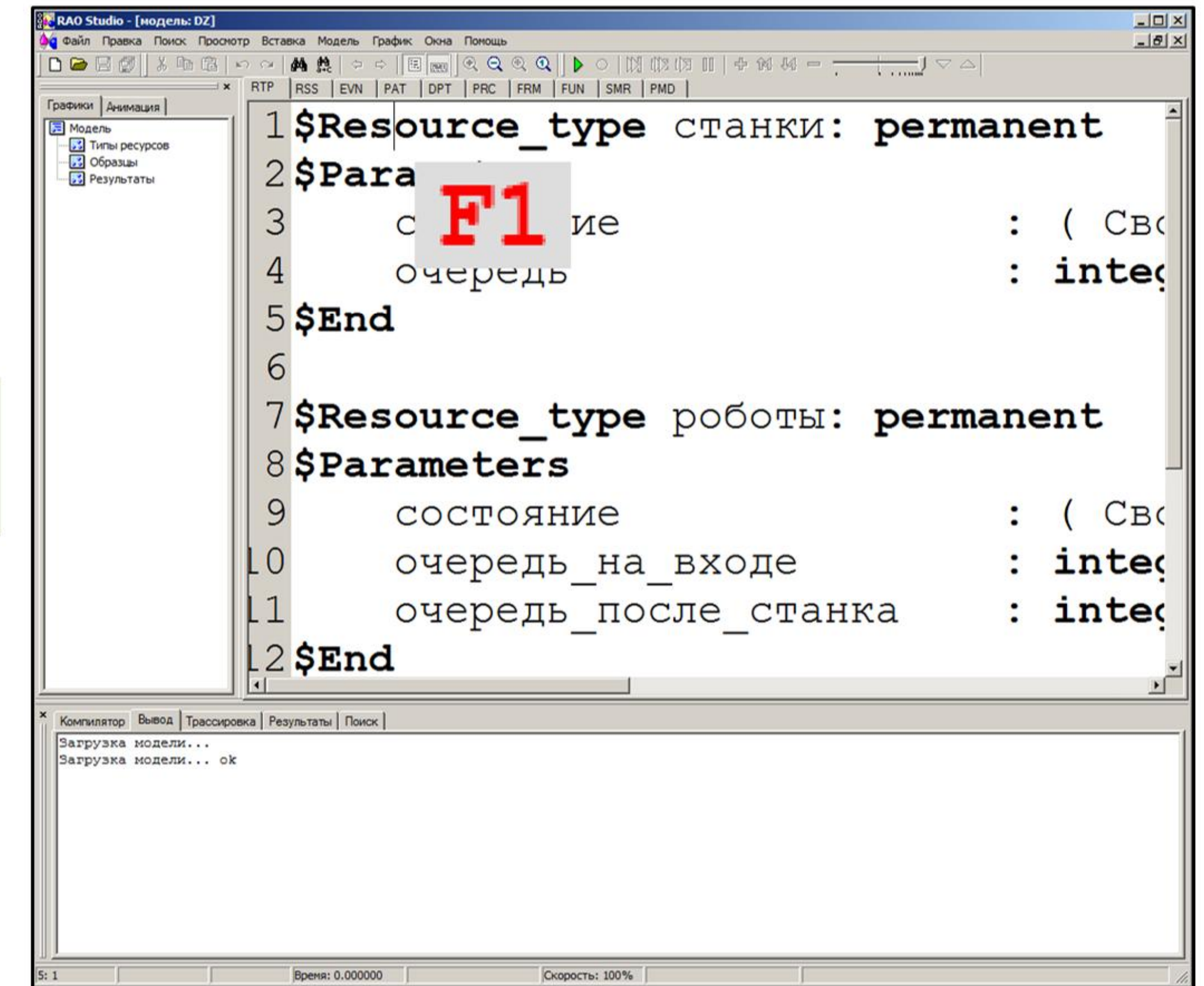
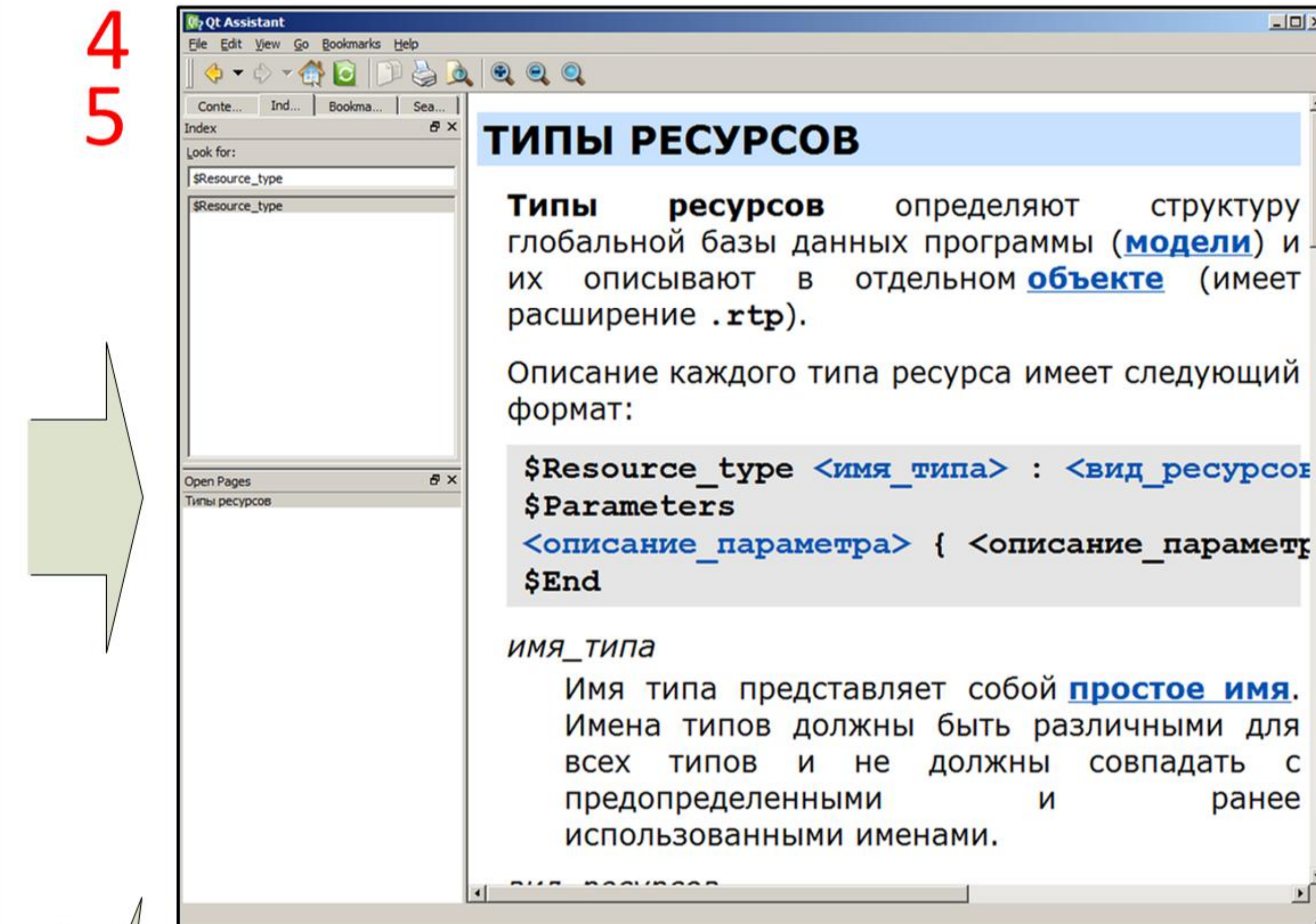
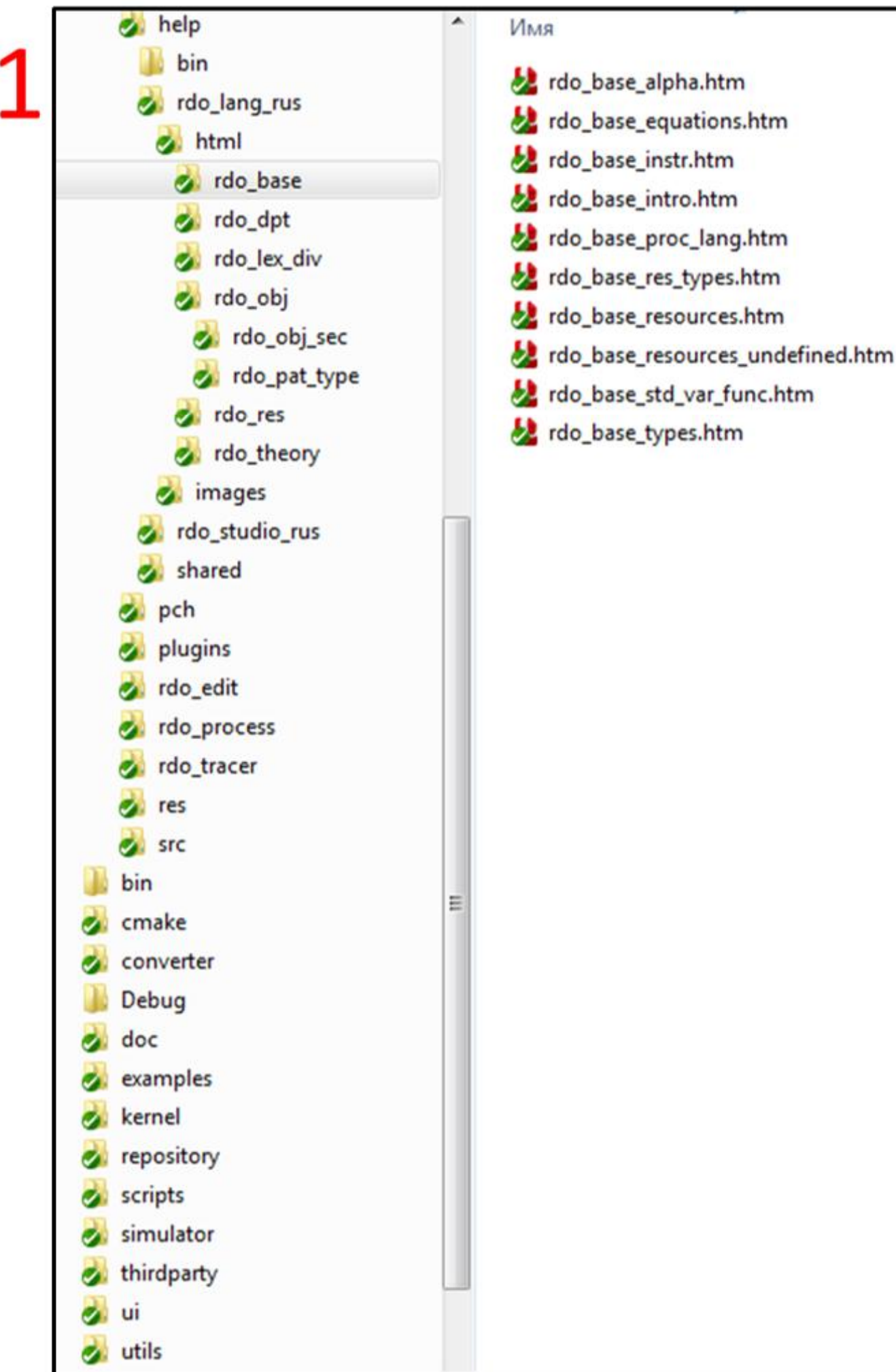
# Результаты

## Единая иерархия

## Контекстный вызов документации

## Сайт

## Удобное ведение документации





					Курсовой проект									
					Постановка задачи					Лит.		Масса	Масштаб	
Изм.	Лист	№ докум.	Подп.	Дата										
Разраб.		Клеванец И.С.												
Пров.		Урусов А.В.												
										Лист		Листов		
										МГТУ им. Н.Э.Баумана				
										Кафедра РК9				
Н.контр.										Группа РК9-101				
Утв.														

					Курсовой проект									
					Результаты					Лит.		Масса	Масштаб	
Изм.	Лист	№ докум.	Подп.	Дата										
Разраб.		Клеванец И.С.												
Пров.		Урусов А.В.												
										Лист		Листов		
										МГТУ им. Н.Э.Баумана				
										Кафедра РК9				
Н.контр.										Группа РК9-101				
Утв.														

					Курсовой проект									
					Перепроектирование документации					Лит.		Масса	Масштаб	
Изм.	Лист	№ докум.	Подп.	Дата										
Разраб.		Клеванец И.С.												
Пров.		Урусов А.В.												
										Лист		Листов		
										МГТУ им. Н.Э.Баумана				
										Кафедра РК9				
Н.контр.										Группа РК9-101				
Утв.														