

КПО

Документирование кода Автоматизация документирования

Лекция №5
(версия 1.0)

Руководство пользователя

Типичное руководство пользователя содержит:

- Аннотацию, в которой приводится краткое изложение содержимого документа и его назначение
- Введение, содержащее ссылки на связанные документы и информацию о том, как лучше всего использовать данное руководство
- Страницу содержания
- Главы, описывающие, как использовать, по крайней мере, наиболее важные функции системы
- Глава, описывающая возможные проблемы и пути их решения
- Часто задаваемые вопросы и ответы на них
- Где ещё найти информацию по предмету, контактная информация
- Глоссарий и, в больших документах, предметный указатель

Руководство пользователя

РД 50-34.698-90 (п.п. 3.4 Руководство пользователя)	ГОСТ 19.505-79 Руководство оператора
Введение	
Область применения	
Описание возможностей	
Уровень подготовки пользователя	
Перечень эксплуатационной документации, с которыми необходимо ознакомиться пользователю	
Назначение и условия применения	
Виды деятельности и функции, для автоматизации которых предназначена программа	Назначение программы
Условия, при которых обеспечивается применение программы	Условия выполнения программы
Подготовка к работе	Выполнение программы

Руководство пользователя

Состав и содержание дистрибутивного носителя данных	
Порядок загрузки данных и программ	Порядок загрузки, запуска и завершения программы
Проверка работоспособности	
Описание операций	Описание функций
Описание всех выполняемых функций, задач, комплексов задач, процедур	
Описание операций технологического процесса обработки данных, необходимых для выполнения функций, комплексов задач (задач), процедур	
Аварийные ситуации (действия в случае длительных отказов технических средств, обнаружении несанкционированного вмешательства в данные, действия по восстановлению программ или данных, и т.п.)	Сообщения оператору
Рекомендации по освоению	

Руководство пользователя

USER GUIDE

Название системы

версия системы

Название руководства

версия руководства

Логотип / Авторство

Руководство пользователя

- Титульная страница
- Предисловие
- Содержание
- Введение
- Требования к системе
- Подготовка к запуску
- Знакомство с системой
- Основные бизнес-процессы

Руководство пользователя

- Печать документов
- Экспорт / Импорт данных
- Системные сообщения
- Справочники
- Глоссарий
- Предметный указатель
- Форма для сообщения об ошибке, отзыве

Руководство администратора

ADMIN GUIDE

- Титульная страница
- Предисловие
- Содержание
- Введение
- Требования к системе
- Автоматическая установка системы
- Ручная установка системы
- Подготовка к запуску
- Знакомство с системой

Руководство администратора

- Конфигурирование системы
- Разграничение прав доступа к системе
- Обслуживание системы / регламентные работы / аудит
- Архивирование системы
- Восстановление после сбоев
- Системные сообщения
- Экспорт / импорт данных

Руководство администратора

- Справочники
- Глоссарий
- Предметный указатель
- Форма для сообщения об ошибке, отзыве

Пример кода

```
1 #!/usr/bin/perl
2
3 use CGI qw(:param);
4 use CGI::Carp qw(fatalsToBrowser);
5 use Mysql;
6
7 $dbh = Mysql->connect('localhost','students2','dekanat','') or die("Проблема подключения к базе данных: $!");
8
9 print <<END_of_text;
10 Content-type: text/html; charset-windows-1251
11 Expires: Thu, Jan 1 1970 00:00:00 GMT
12 Pragma: no-cache
13 Cache-Control: no-cache
14
15 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
16 <html><head><title>Форма выплаты степендий</title><style type="text/css">TD,P,SELECT,INPUT { font-size: 8pt;
17 font-family: verdana,arial }</style></head>
18 <body bgcolor-white link-white vlink-white>
19 END_of_text
20
21 $studyform-param('study');
22 $term="archived=0 and studyform=".$studyform;
23 if (defined(param('spec')) and param('spec') ne ''){ $special_id-param('spec'); $term.-" and
24 special_id=".$special_id;}
25 else{ $special_id-1;}
26 if (defined(param('course')) and param('course') ne ''){ $course_id-param('course'); $term.-" and
27 course_id=".$course_id;}
28 else{ $course_id - 1;}
29 if (defined(param('group')) and param('group') ne ''){ $group_id-param('group'); $term.-" and group_id=".$group_id;
30 else{ $group_id - 1;}
31 if(defined(param('order')) and param('order') ne ''){ $order-param('order');} else { $order-1;};
32
33 $in = $dbh->query("select special_id,special_short from specials") or die("Проблема чтения из базы данных: $!");
34 while(@rows=$in->fetchrow)
35 {
36   $specials[$rows[0]]=$rows[1];
37 }
38
39 $in=$dbh->query("select type1,type2,type3,type4,type5,type6 from tarifes where mode=0 and year=$year_id and
40 semestr=$semestr_id") or die("Проблема чтения базы данных: $!");
41 ($perc1,$perc2,$perc3)-$in->fetchrow;
42 $in=$dbh->query("select concat(lastname,' ',firstname,'
```

Пример кода

```
47 @spec=$in->fetchcol(1);
48 @course=$in->fetchcol(2);
49 @group=$in->fetchcol(3);
50 @sheet=$in->fetchcol(4);
51 @paying=$in->fetchcol(5);
52 # - - - Обратная процедура сортировки - - -
53 if($order--1 || $order--2)
54 {
55   @x = sortmass($nr, ([@lns], [@spec], [@course], [@group], [@sheet], [@paying]));
56   @lns = @x[0..$nr-1];
57   @spec = @x[$nr..$nr*2-1];
58   @course = @x[$nr*2..$nr*3-1];
59   @group = @x[$nr*3..$nr*4-1];
60   @sheet = @x[$nr*4..$nr*5-1];
61   @paying = @x[$nr*5..$nr*6-1];
62 }
63 # - - - - -
64 for($i=0;$i<$nr;$i++)
65 {
66   if(param('type') eq 'bank')
67   {
68     $hold1=&new_int($paying[$i]*$perc2)/100;
69     $hold2=&new_int(($paying[$i]-$hold1)*$perc3)/100+&new_int($perc1*$perc3)/100;
70     print "<tr><td>".($i+1)."</td><td>".($sheet[$i] ne ''?$sheet[$i]:"&nbsp;")."</td><td>$lns[$i]</td>\n";
71     print
72     " <td>$specials[$spec[$i]]$course[$i]$group[$i]</td><td>".($paying[$i]+$perc1-$hold1-$hold2)."</td></tr>\n";
73   }
74   else
75   {
76     $hold1=&new_int($paying[$i]*$perc2)/100;
77     $hold2=&new_int(($paying[$i]-$hold1)*$perc3)/100+&new_int($perc1*$perc3)/100;
78     print "<tr><td>".($i+1)."</td><td>$lns[$i]</td><td>$paying[$i]</td><td>$perc1</td>\n";
79     print "<td>$hold1</td><td>$hold2</td>\n";
80     print "<td>".($hold1+$hold2)."</td>\n";
81     print "<td>".($paying[$i]+$perc1-$hold1-$hold2)."</td><td>&nbsp;</td></tr>\n";
82   }
83 }
84 print "</table></body></html>";
85
86 sub sortmass() {
87   my ($arg, @s1, @s2, @s3, @s4, @s5, @s6);
88   my $i = 0;
89   mv ($num.@xx)-(@ );
```

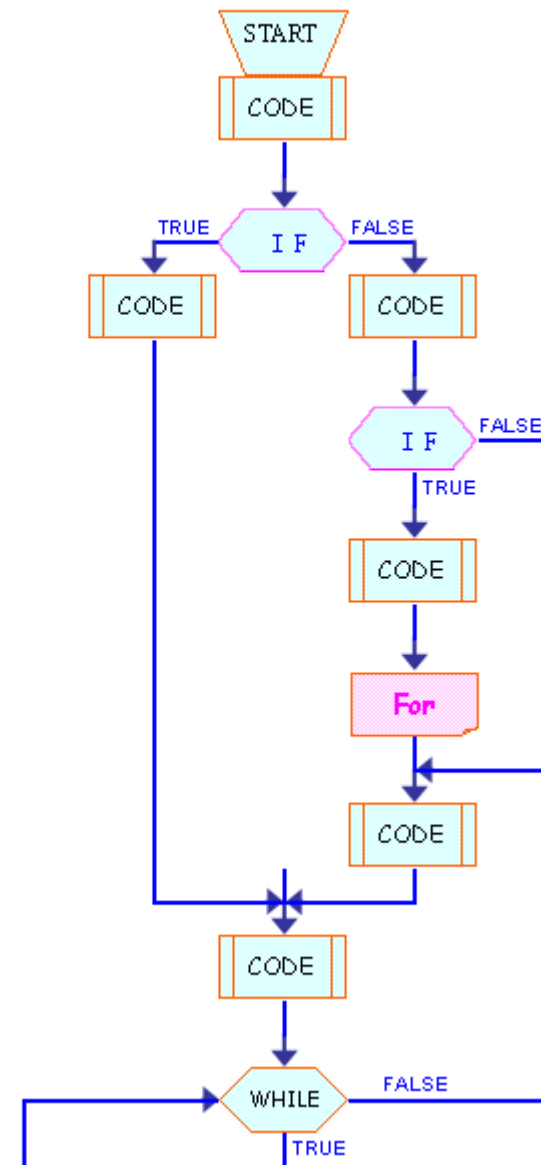
Документирование кода

1. Блок-схемы
2. Псевдокод
3. Самодокументирование

Построение блок-схем

Code Visual to Flowchart

```
Flowchart Export Print Help Sample BIG 3 PH Flowchart -1 Flowchart -2  
c++.cpp login.php php.php  
("class_db.inc");  
  
global $obj_db;  
  
//echo $sol_data."\n";  
//echo $filepath."\n";  
$dbh=ydbf_r_open($filepath);  
if($dbh == false){  
    echo "FAIL\n";  
}else(  
    echo "OK\n";  
    $n=ydbf_num_rows($dbh);  
    printf("Record count : %d\n", $n);  
    //ydbf_table_arch($dbh);  
  
    if ($insert_tablename<>' -1'){  
        $fields=ydbf_table_fields($dbh);  
        //echo $fields . "\n";  
        ydbf_sync_field($dbh);  
  
        for ($num=0;$num<$n;$num++){  
            $row=ydbf_get_row($dbh, $num);  
            if ($row==false){  
                //break;  
            }  
        }  
    }  
}
```



Самодокументирование

POD

Command Paragraph

A command paragraph is used for special treatment of whole chunks of text, usually as headings or parts of lists.

All command paragraphs (which are typically only one line long) start with "=", followed by an identifier, followed by arbitrary text that the command can use however it pleases. Currently recognized commands are

```
=pod
=head1 Heading Text
=head2 Heading Text
=head3 Heading Text
=head4 Heading Text
=over indentlevel
=item stuff
=back
=begin format
=end format
=for format text...
=encoding type
=cut
```

Самодокументирование

```
/*
=pod

=head2 NameStudGroup_PO

=over 1

=for troff

Функция NameStudGroup_PO($group_id) выдает название группы по полученному id группы.
Входные переменные - $group_id группы.
Результатом функции является - имя группы (name);

Автор: Ливенцева Т.В.
Дата : 01.10.2005

=back

=cut

*/

function NameStudGroup_PO($group_id){
    global $conn2;
    if($id_group == 99 || $id_group == 100){
        $ltv_query      = "SELECT name FROM stud_groups WHERE id='".$id_group.'" ";
    }
    $ltv_res   = $conn2->Execute($ltv_query);
    $ltv_line  = $ltv_res->FetchRow();
    $group_name = $ltv_line['name'];
    return $group_name;
}
```


Самодокументирование

- [func_dat](#) - файл с функциями по работе с датой
 - [getnweek](#)
- [func_dekan](#) - файл с функциями по деканату, декану, ректорату и сессии
 - [asu_SpisokSpec](#)
 - [asu_SpisokGroups](#)
 - [asu_SpisokStudents](#)
 - [asu_SearchStudent](#)
 - [asu_StudentInfo](#)
 - [asu_FacName](#)
 - [asu_SpecName](#)
 - [asu_GetVariable](#)
 - [asu_Photo](#)
 - [asu_PhotoUpdate](#)
 - [asu_SpisokFacultet](#)
 - [asu_InsertQuery](#)
 - [asu_GroupInfo](#)
 - [asu_DolzInfo](#)

Самодокументирование

asu_StudentWithoutLogin

Функция `asu_StudentWithoutLogin` выдает список учащихся студентов не имеющих
Выходными данными является массив:
`stud_id(id студента),`
`fio(фio студента),`
`group_name(название группы),`
`nzk(номер зачетной книжки),`
`gr_id (id группы).`

Автор: Ливенцева Т.В.
Дата : 8.09.2006

asu_isStudent

Функция `asu_isStudent(nds_login)` проверяет является ли данный пользователь
Входными данными является логин в сети (более 5 символов).
На выходе: 0 (не студент) или id студента
Или сообщение об ошибке.

Автор: Арбатский Е.В.
Дата : 19.02.2007

asu_isPrepod

Функция `asu_isPrepod(nds_login)` проверяет является ли данный пользователь
Входными данными является логин в сети (более 5 символов).
На выходе: 0 (не преподаватель) или id преподавателя
Или сообщение об ошибке.

Автор: Арбатский Е.В.
Дата : 19.02.2007

Стандарты на код

Положительные моменты:

- Программисты могут прочесть код и легко в нем разобраться;
- Новые программисты быстрее вписываются в проект;
- Новые люди избавлены от необходимости разрабатывать свой стиль и отстаивать его;
- Позволяет избегать типичных ошибок «НОВИЧКОВ».

Стандарты на код

Отрицательные моменты:

- Стандарты – никому не нужный мусор;
- Стандарт – это не то, что я хочу;
- Стандарты понижают творчество;
- Все равно люди не следуют стандартам.

Верификация кода

Методики и мероприятия верификации программного кода:

- статический анализ;
- метрики.

Статический анализ – это изучение предоставленных исходных кодов программных модулей. Статический анализ является формой инспектирования кода – это технология обнаружения тех ошибок, которые могут быть пропущены другими технологиями (например, тестированием). К методам статического анализа кода относятся: анализ указателей, устранение мертвого кода, минимизация количества переменных, обнаружение типичных ошибок.

Инструменты анализа

Класс инструментов, предназначенный для вычисления метрик программного обеспечения, называют **Software Estimation**.

«**Locmetrics**» — очень простой бесплатный продукт с минималистским интерфейсом. В числе поддерживаемых языков — C/C++, C#, Java, SQL — возможно вычисление не только метрики SLOC и ее разновидностей, но и цикломатической сложности.

Инструменты анализа

«**USC Codecount**» – бесплатный продукт с открытыми исходными кодами на языке ANSI C, разработанный Университетом Южной Калифорнии (University of Southern California, USC) – той же организацией, в которой были созданы COCOMO/COCOMO II. Является официальным инструментом для подсчета метрики SLOC при использовании указанных моделей. В число поддерживаемых языков входят C/C++, C#, Java, JavaScript, SQL, Perl, XML. Методика расчета соответствует принятой SEI для моделей CMM/CMMI. Вычисляет количество логических и физических SLOC, пустых строк, комментариев, директив компилятора, описаний данных, исполняемых инструкций по файлам проекта по отдельности и суммарно.

Инструменты анализа

«**Code Counter Pro**» – коммерческий продукт (\$25 за одну лицензию). В отличие от предыдущих имеет развитый графический интерфейс. Поддерживаются следующие языки программирования: Java, JSP, C/C++, VB, PHP, HTML, Delphi/Pascal, ASM, XML, COBOL. Несмотря на то, что программа хорошо справляется со своей задачей и даже позволяет строить детальные отчеты, чего не может предложить, скажем, Locmetrics, она уступает рассмотренным открытым аналогам по количеству вычисляемых показателей (только число физических строк кода, комментариев, пустых строк, а также суммарные значения).

Инструменты анализа

«**Verisoft Complexity Measures Tool**» – коммерческий продукт (1200 евро). Поддерживаются только языки C/C++ и Java. Рассчитывает следующие метрики: SLOC, цикломатическую сложность, метрики Холстеда, индекс сопровождаемости (вычисляется на основе предыдущих). Имеет графический интерфейс (с возможностью работы в режиме командной строки), позволяет формировать отчеты в текстовой форме или HTML.

Инструменты анализа

«Eclipse Metrics Plugin» – представляет собой подключаемый модуль для популярной IDE Eclipse. Eclipse – свободно распространяемая среда программирования для языка Java, разработанная компанией IBM. Вычисляет SLOC, количественные метрики классов, цикломатическую сложность, метрики сложности классов (LOCOM1, LOCOM2, LOCOM3, WMPC, NORM, индекс специализации), метрики связности, уровень абстракции и некоторые другие. Достаточно функциональный продукт, который вполне может дать фору многим коммерческим аналогам.

Инструменты анализа

Из рассматриваемых инструментов наиболее универсальным средством является «**SLOCCount**» – бесплатный продукт, разработанный Дэвидом Вилером (David A. Wheeler), поставляется в виде исходных кодов на языке C по лицензии GNU GPL. В число поддерживаемых языков входят Ada, Assembler, awk, Bourne shell (включая производные: bash, ksh, zsh, pdksh), C, C++, C#, C shell (включая tcsh), COBOL, Expect, Fortran (включая Fortran 90), Haskell, Java, lex (включая flex), LISP (включая Scheme), make-файлы,

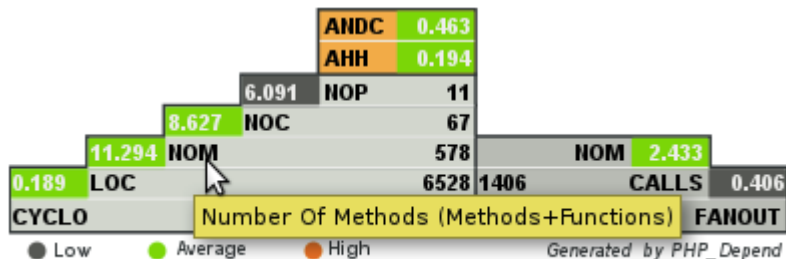
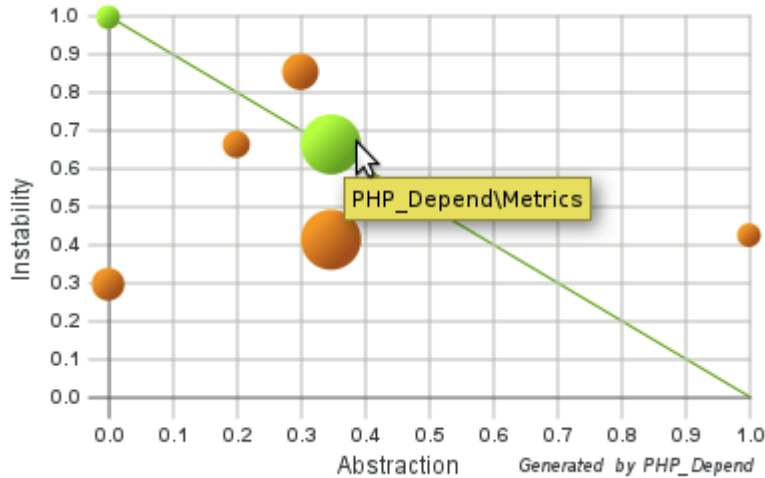
```
Total Physical Source Lines of Code (SLOC) = 72
Development Effort Estimate, Person-Years (Person-Months) - 0.01 (0.15)
(Basic COCOMO model, Person Months = 2.4 * ((SLOC**1.05)))
Schedule Estimate, Years (Months) = 0.10 (1.22)
(Basic COCOMO model, Months = 2.5 * (person-months**0.38))
Estimated Average Number of Developers (Effort/Schedule) - 0.12
Total Estimated Cost to Develop = $ 1.705
(average salary = $56,286/year, overhead = 2.10).
```

Инструменты анализа

Из нового (2010):

PMD — анализатор Java

PHP Depend + PHP MD —
анализатор PHP



Инструменты улучшения

Beautifier — инструмент для создания «красивого» кода. Приводят исходный код к определенному оформлению. При этом логика кода не меняется.

Инструменты улучшения

```
33          31          <table width=100% height=100% bgcolor="#EEFFFF" >
34          32          <!--<tr><td valign="top" width 20% bgcolor="#DBDBDB" align=
solid #050AB3"> solid #050AB3"><p style="font-size:10pt;font-weight:bold;color:#050AB3;">Дата:</td>
35          33          <tr><td valign="top" width 20% align="center" </td> вставит
36          34          <tr><td bgcolor="#DBDBDB" align="center" style="border-bott
ont-size:10pt;font-weight:bold;color:#050AB3;">Проект (<? echo $k['kol']; ?>):</p><
37          </tr> 35          </tr>
38          36
39 <? 37 <?
40 $query = "SELECT 38 $query = "SELECT * FROM project order by name";
41 $results = mysql_ 39 $results = mysql_query("$query");
42 $num_results = m 40 $num_results = mysql_num_rows($results);
43 ////////// 41 //////////
44 if (!isset($M_beg 42 if (!isset($M_beg) || ($M_beg == 0)) {
45 $M_beg=0; 43 $M_beg = 0;
46 } 44 } else {
47 else 45 for ($i = 0;$i < $M_beg;$i++) {
48 { 46 $data = mysql_fetch_assoc($results);
49 for ($i = 0; $i . 47 }
50 $data = mysql_fe 48 }
51 } 49 $page = ceil($num_results / 14);
52 } 50 //////////
53 $page = ceil($nu 51 if ($num_results != 0) {
54 ////////// 52 for ($i = 0;$i < 14;$i++) {
55 53 $data = mysql_fetch_assoc($results);
56 if ($num_results 54 echo "<tr><td><p class=\"mar0\"><a href=\"index.php?beg=" . $M_beg . "&
id_p' ] . "\">" . $data['name'] . "</a></td></tr>";
57 { 55 }
58 for ($i=0; $i < : 56 }
59 $data = mysql_fe 57 //////////
60 echo "<tr><td><p 58 echo "<tr height=100% align=left valign=bottom><td align=left valign=bottom>";
['name'] . "</a></td>
```

Инструменты улучшения

- * HTML : <http://www.digital-mines.com/htb/>
- * HTML : <http://www.datacomm.ch/mwoog/software/perl/beautifier.html>
- * HTML : http://www.watson-net.com/free/perl/s_fhtml.asp

- * SQL : <http://www.netbula.com/products/sqlb>
- * Oracle PLSQL : <http://www.revealnet.com>

- * GPL <http://www.geocities.com/~starkville/vancbj.html>
- * GPL <http://kevinkelley.mystarband.net/java/dent.html>
- * Free <http://www.tiobe.com/jacobe.htm>
- * Free <http://www.mmsindia.com/JPretty.html>
- * Free <http://members.magnet.at/johann.langhofer/products/jxbeauty/overview.html> (has JBuilder support)
- * Free <http://www.semdesigns.com/Products/Formatters/JavaFormatter.html>
- * Commercial \$24.99 <http://smartbeautify.com>
- * Commercial \$129 <http://www.jindent.com>
- * Google http://directory.google.com/Top/Computers/Programming/Languages/Java/Development_Tools/Code_Beautifiers/?tc=1

- * Java, SQL, HTML, C++ : <http://www.semdesigns.com/Products/DMS/DMSToolkit.html>
- * Java JIndent <http://home.wtal.de/software-solutions/jindent>
- * Java Pat <http://javaregex.com/cgi-bin/pat/jbeaut.asp>
- * Java JStyle <http://www.redrival.com/greenrd/java/jstyle>
- * Java JPrettyPrinter <http://www.epoch.com.tw/download/ms/java/java.htm>
- * Java JxBeauty <http://members.nextra.at/johann.langhofer/download/jxbeauty> and the JxBeauty Home
- * Java beautify percolator
- * Java list <http://www.java.about.com/compute/java/library/weekly/aa102499.htm>
- * Java html present [VasJava2HTML](#)
- * Java code colorifier and beautifier <http://www.mycgiserver.com/~lisali/jccb>

- * Perl : <http://www.consultix-inc.com/www.consultix-inc.com/talk.htm>
- * Perl : http://www.consultix-inc.com/www.consultix-inc.com/perl_beautifier.html

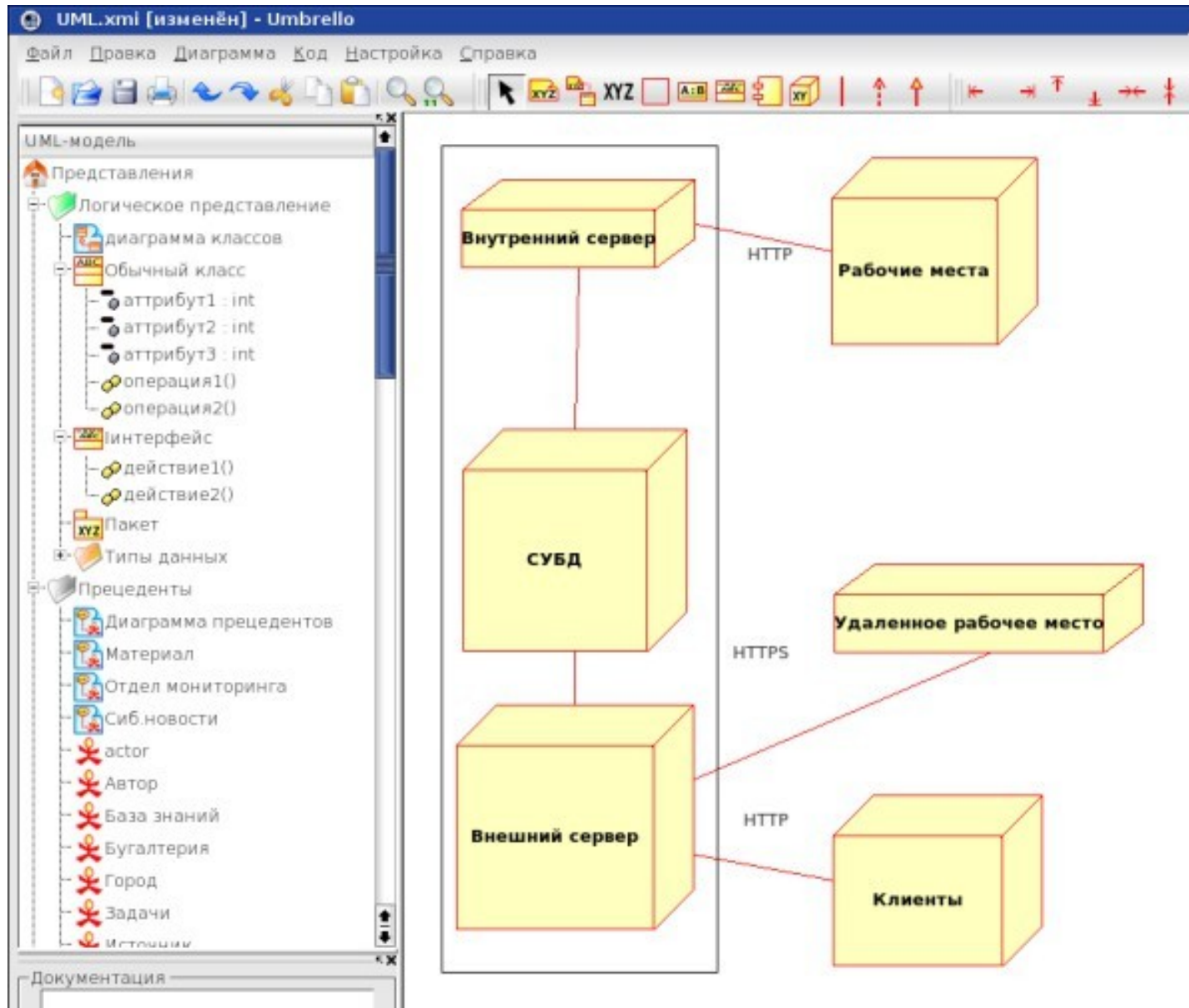
- * Fortran beautifier : <http://www.aeem.iastate.edu/Fortran/tools.html>

- * C++ : BCPP site is at <http://dickey.his.com/bcpp/bcpp.html> or at <http://www.clark.net/pub/dickey>. BCPP ftp site is at <ftp://dickey.his.com/bcpp/bcpp.tar.gz>
- * C++ : <http://www.consultix-inc.com/c++b.html>
- * C : <http://www.chips.navy.mil/oasys/c/> and mirror at Oasys
- * C++, C, Java, Oracle Pro-C Beautifier <http://www.geocities.com/~starkville/main.html>
- * C++, C beautifier <http://users.erols.com/astronaut/vim/ccb-1.07.tar.gz> and site at http://users.erols.com/astronaut/vim/#vimlinks_src
- * GC! GreatCode! is a powerful C/C++ source code beautifier Windows 95/98/NT/2000 <http://perso.club-internet.fr/cbeudet>
- * C++ beautifier 'SourceStyler' <http://www.ochre.com.au>

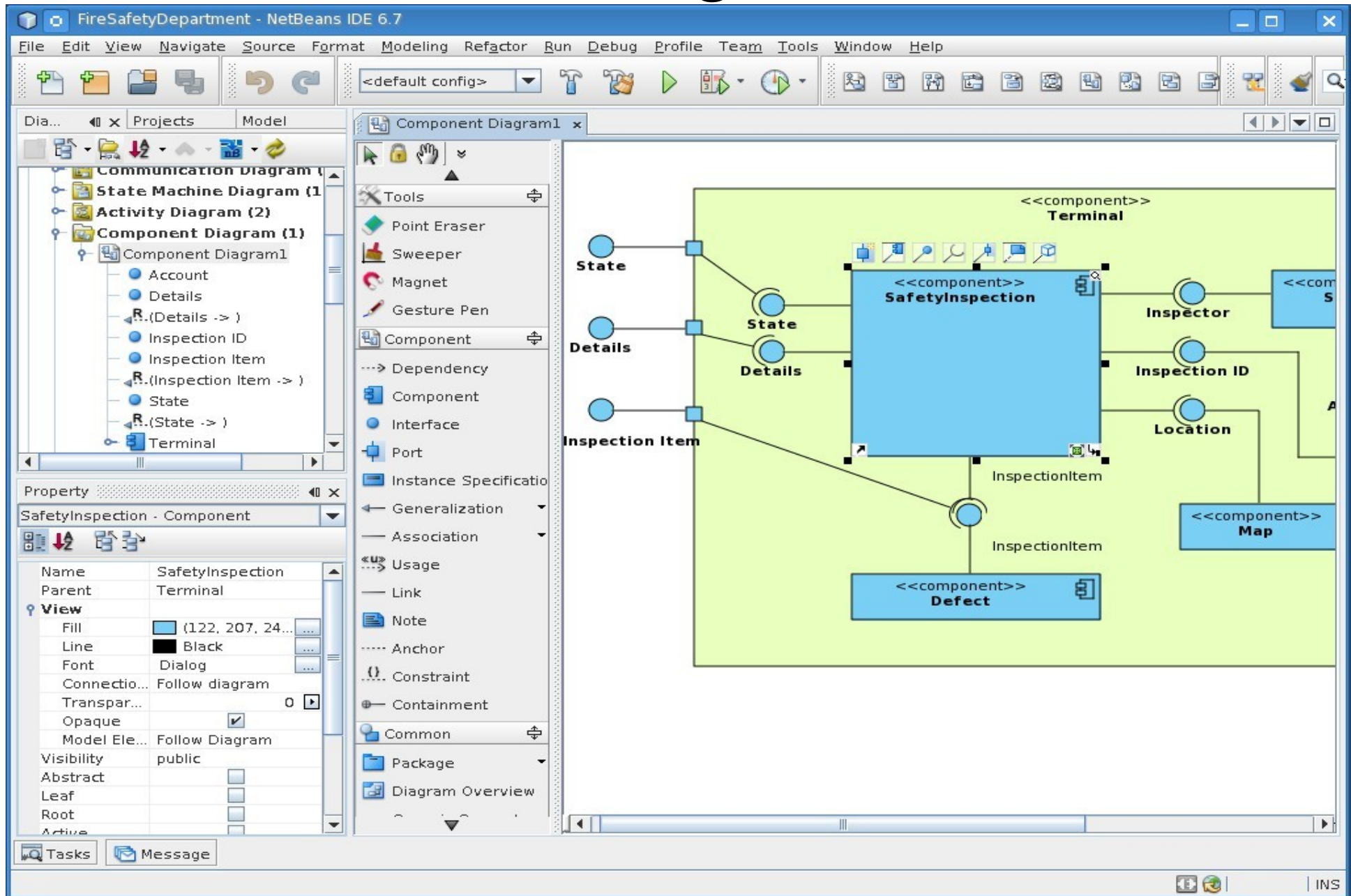
CASE средства для UML

- IBM Rational Rose
- Microsoft Visio 2003 и выше
- Umbrello, Dia ...
- Visual Paradigm for UML
- StarUML, MagicDraw, ArgoUML

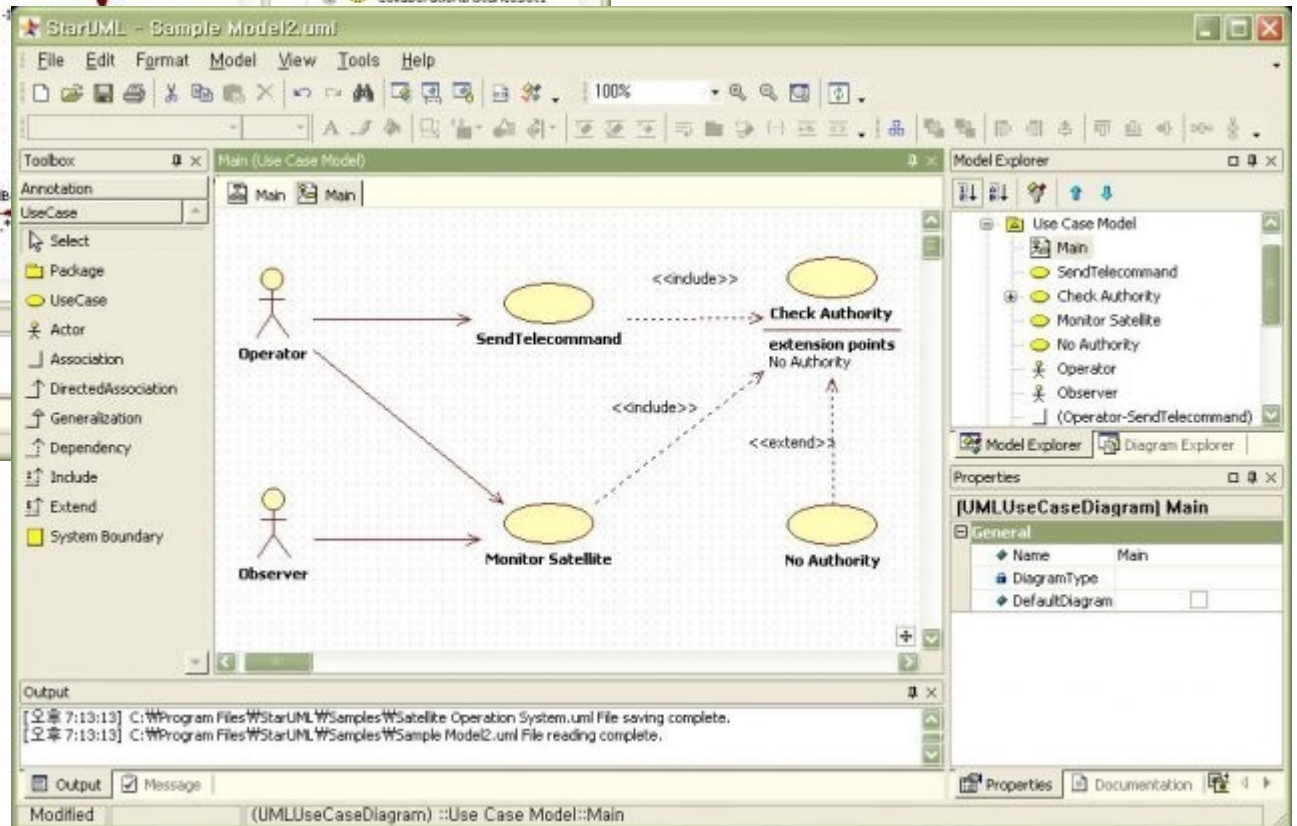
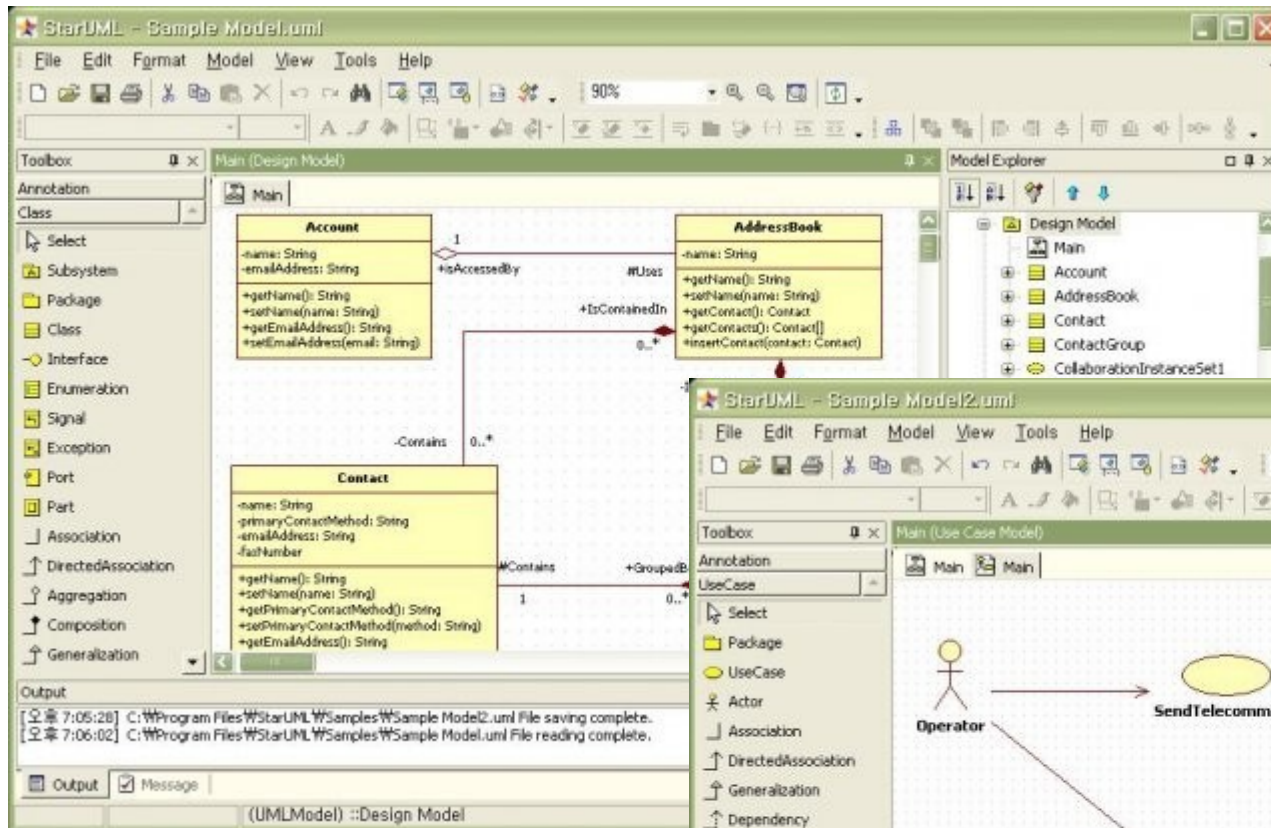
Umbrello



Visual Paradigm for UML



StarUML



ArgoUML

