МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

Новгородский государственный университет имени Ярослава Мудрого (НовГУ) Кафедра «Электронных и информационных систем»

**Простейший интерфейс на естественном языке с использованием ключевых слов**

Лабораторная работа по дисциплине:

«Системы искусственного интеллекта»

Студент гр. 1095

Золотов Э. В.

(подпись)

«\_\_\_»\_ 2024 г.

Преподаватель

\_\_Михайлов Д.В

(подпись)

« » 2024 г.

г. Великий Новгород

-2024-

# ЦЕЛЬ РАБОТЫ

Целью работы является изучение лингвистических возможностей языка Пролог. Подсистема понимания Естественного Языка реализуется на основе стратегии с использованием ключевых слов. Для формирования пользователем запроса на русском языке в произвольной форме в состав системы должен быть включен текстовый редактор.

# МОДЕЛЬ ПРЕДСТАВЛЕНИЯ ЗНАНИЙ

Предметная область вирусы.

|  |  |  |
| --- | --- | --- |
| Группа | Вирус | Атрибуты |
| Вирус\_маскировщик | троян | Перегрузка\_сети  Повреждение\_внутренней\_памяти  Нагрузка\_процессора |
| вирус\_чернобля | Подвисание\_системы  Перегрузка\_сети |
| Вирус\_шпион | сламмер | Подвисание\_системы  Перегрузка\_сети  Нагрузка\_видеокарты |
| нимда | Повреждение\_внутренней\_памяти  Нагрузка\_процессора  Нагрузка\_видеокарты |
| мелисса | Перегрузка\_сети  Повреждение\_внутренней\_памяти  Нагрузка\_накопителя |

***Таблица 1. Описание предметной области***

# ОПИСАНИЕ ХАРАКТЕРИСТИК РАЗРАБОТАННОЙ ЭС

Наиболее известными являются следующие подходы к решению

задач ЕЯ-общения конечного пользователя с ЭВМ:

— Прагматический анализ - наиболее сложный, связан с изучением смысла предложения в связи с внеязыковой действительностью. Анализ ключевых слов — метод анализа ЕЯ-высказываний на предмет наличия ключевых слов, которые становятся значениями аргументов предикатов. При этом компьютер одинаково реагирует на различные варианты входного текста, наличие грамматической правильности предложений не является обязательным, роль играет лишь наличие ключевых слов. Применение -построение ЕЯ-интерфейсов к Базам Данных (БД).

— Грамматический анализ: контекстно-свободный и контекстно-зависимый. Контекстно-Свободный (КС) анализ — ЕЯ-фразы классифицируются в зависимости от их внутренней структуры вне зависимости от контекста в соответствии с грамматическими правилами, задающими порядок следования допустимых языком символов (слов). Здесь следует отметить синтаксический анализ предложений.

— Прагматический анализ — наиболее сложный, связан с изучением смысла предложения с учетом связи с внеязыковой действительностью.

В этой лабораторной интерфейс на естественном языке реализован с помощью ключевых слов. Введённая фраза разбивается на слова, после чего из этих слов выбираются перовое и последнее. Если первое слово начинается на “вых” или “вый” независимо от второго слова происходит выход из программы. Если первое слово начинается на “най”, а последнее слово – “спорт” делается вывод что пользователь хочет найти спорт и запускается процесс поиска по БЗ. Если ни одно из предыдущих условий не выполнилось пользователь получает сообщение об ошибке в формате “Введённые вами ключевые слова ПЕРВОЕ\_СЛОВО\_ИЗ\_ФРАЗЫ и ПОСЛЕДНЕЕ\_СЛОВО\_ИЗ\_ФРАЗЫ не известны системе.”.

# ТЕСТОВЫЕ НАБОРЫ ДАННЫХ

# *Удачный набор данных:*

Найти вирус\_шпион

Да

Да

Да

Это вирус:сламмер

# *Неудачный набор данных:*

Найти вирус\_маскировщик

Нет

Да

Введенные вами ключевые слова: найти и вирус\_маскировщик не известны системе

# ВЫВОД

В ходе выполнения лабораторной работы я научился использовать лингвистические возможности языка «Visual Prolog» для построения интерфейса на естественном языке для взаимодействия с пользователем.

# ПРИЛОЖЕНИЕ A

# Код программы

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

include "logic\_es\_win.inc"

include "logic\_es\_win.con"

include "hlptopic.con"

%BEGIN\_WIN Task Window

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Event handling for Task Window

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

predicates

task\_win\_eh : EHANDLER

**rev**(word\_list,word\_list,word\_list).

**reverse**(word\_list,word\_list).

nondeterm **rule**(rule\_number,category,category,conditions)

nondeterm **cond**(cond\_number,condition)

nondeterm **topic**(condition)

assert\_database

do\_consulting

% nondeterm info

erase

clear

**do\_answer**(cond\_number,integer)

% keyword(category)

**do\_right\_form**(string,category)

**func\_keyword**(string)

**last**(word\_list, string)

**member**(char,char\_list).

nondeterm **symbol\_counter**(string,integer).

nondeterm **del\_front\_space**(string,string).

nondeterm **fronttoken\_cyr**(string,string,string).

nondeterm **convers**(string,word\_list)

nondeterm **upper\_lower\_cyr**(string,string).

nondeterm **upper\_lower\_cyr\_convers**(char\_list,char\_list).

nondeterm **str\_char\_list**(string,char\_list).

**pack**(char\_list,string).

**goes**(string,category)

nondeterm **go**(history,category)

nondeterm **check**(rule\_number,history,conditions)

**ask\_question**(cond\_number,condition)

constants

%BEGIN Task Window, CreateParms, 21:26:22-22.12.2007, Code automatically updated!

task\_win\_Flags = [wsf\_SizeBorder,wsf\_TitleBar,wsf\_Close,wsf\_Maximize,wsf\_Minimize,wsf\_ClipSiblings]

task\_win\_Menu = **res\_menu**(idr\_task\_menu)

task\_win\_Title = "LOGIC"

task\_win\_Help = idh\_contents

%END Task Window, CreateParms

clauses

**assert\_database**:-

**rule**(Rule\_number,Category,Type\_of\_Virus,Conditions),

**assertz**(**d\_rule**(Rule\_number,Category,Type\_of\_Virus,Conditions)),fail.

**assert\_database**:-

**cond**(Cond\_number,Condition),

**assertz**(**d\_cond**(Cond\_number,Condition)),fail.

**assert\_database**:-

**topic**(Condition),

**assertz**(**d\_topic**(Condition)),fail.

**assert\_database**:-!.

**cond**(**1**,"вирус\_маскировщик").

**cond**(**2**,"вирус\_шпион").

**cond**(**3**,"подвисание\_системы").

**cond**(**4**,"перегрузка\_сети").

**cond**(**5**,"повреждение\_внутренней\_памяти").

**cond**(**6**,"нагрузка\_процессора").

**cond**(**7**,"нагрузка\_видеокарты").

**cond**(**8**,"нагрузка\_накопителя").

**topic**("вирус\_маскировщик").

**topic**("вирус\_шпион").

**rule**(**1**,"вирус","вирус\_маскировщик",[**1**]).

**rule**(**2**,"вирус","вирус\_шпион",[**2**]).

**rule**(**3**,"вирус\_маскировщик","троян",[**4**,**5**,**6**]).

**rule**(**4**,"вирус\_маскировщик","вирус\_чернобля",[**3**,**4**]).

**rule**(**5**,"вирус\_шпион","сламмер",[**3**,**4**,**7**]).

**rule**(**6**,"вирус\_шпион","нимда",[**7**,**5**,**6**]).

**rule**(**7**,"вирус\_шпион","мелисса",[**4**,**5**,**8**]).

**do\_consulting**:-

**goes**(**\_**,**\_**),!.

% go([],First\_keyword),!.

**do\_consulting**:-

not(dummy),

**dlg\_Error**("Данные о вирусе не найдены."),

clear.

**ask\_question**(Virus\_cond,Text):-

**concat**("Вопрос : ",Text,Temp),

**concat**(Temp," ",Temp1),

**concat**(Temp1,"?",Quest),

Response1=**dlg\_Ask**("Выполнить консультацию",Quest,["Да","Нет"]),

Response=Response1+**1**,

**do\_answer**(Virus\_cond,Response).

**rev**([],Init,Init).

**rev**([H|T],Init,Res):-

**rev**(T,[H|Init],Res).

**reverse**(Arg,Res):-

**rev**(Arg,[],Res).

**member**(Head,[Head|**\_**]):-!.

**member**(Elem,[**\_**|T]):-

**member**(Elem,T).

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\32',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\10',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\13',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\33',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\34',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\35',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\36',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\40',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\41',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\44',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\45',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\46',**\_**),!.

**symbol\_counter**(Str,**0**):-

**frontchar**(Str,'\59',**\_**),!.

**symbol\_counter**("",**0**).

**symbol\_counter**(Str,Number):-

**frontchar**(Str,**\_**Char,Rest\_of\_string),

**symbol\_counter**(Rest\_of\_string,Number1),

Number=Number1+**1**.

**fronttoken\_cyr**(Str,Token,Rest\_of\_string):-

**symbol\_counter**(Str,Number),

**frontstr**(Number,Str,Token,Rest\_of\_string).

**del\_front\_space**("","").

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\32',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\10',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\13',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\33',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\34',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\35',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\36',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\44',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\40',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\41',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\45',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\46',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Res):-

**frontchar**(Arg,Char,Res1),Char='\59',!,

**del\_front\_space**(Res1,Res).

**del\_front\_space**(Arg,Arg):-**frontchar**(Arg,Char,**\_**),

not(**member**(Char,['\32','\10','\13','\33',

'\34','\35','\36','\40','\41',

'\44','\45','\46','\59'])).

**convers**("",[]):-!.

**convers**(Str,[Head1|Tail]):-

**fronttoken\_cyr**(Str,Head,Str2),

**upper\_lower\_cyr**(Head,Head1),

**del\_front\_space**(Str2,Str1),

**convers**(Str1,Tail).

**upper\_lower\_cyr**(InString,OutString):-

**str\_char\_list**(InString,Char\_List\_for\_InString),

**upper\_lower\_cyr\_convers**(Char\_List\_for\_InString,Char\_List\_for\_OutString),

**pack**(Char\_List\_for\_OutString,OutString).

**upper\_lower\_cyr\_convers**([],[]).

**upper\_lower\_cyr\_convers**([Char|Char\_List],[Char1|Char\_List1]):-

**char\_int**(Char,ASCII\_code),

ASCII\_code>=**192**,ASCII\_code<=**223**,!,

ASCII\_code\_new=ASCII\_code+**32**,

**char\_int**(Char1,ASCII\_code\_new),

**upper\_lower\_cyr\_convers**(Char\_List,Char\_List1).

**upper\_lower\_cyr\_convers**([Char|Char\_List],[Char1|Char\_List1]):-

**char\_int**(Char,ASCII\_code),

ASCII\_code=**168**,!,

ASCII\_code\_new=ASCII\_code+**16**,

**char\_int**(Char1,ASCII\_code\_new),

**upper\_lower\_cyr\_convers**(Char\_List,Char\_List1).

**upper\_lower\_cyr\_convers**([Char|Char\_List],[Char1|Char\_List1]):-

**char\_int**(Char,ASCII\_code),

ASCII\_code>=**65**,ASCII\_code<=**90**,!,

ASCII\_code\_new=ASCII\_code+**32**,

**char\_int**(Char1,ASCII\_code\_new),

**upper\_lower\_cyr\_convers**(Char\_List,Char\_List1).

**upper\_lower\_cyr\_convers**([Char|Char\_List],[Char|Char\_List1]):-

**upper\_lower\_cyr\_convers**(Char\_List,Char\_List1).

**str\_char\_list**("",[]).

**str\_char\_list**(Word,[Char|Char\_List]):-

**frontchar**(Word,Char,WordRest),

**str\_char\_list**(WordRest,Char\_List).

**pack**([],"").

**pack**([H|T],Res):-

**str\_char**(Str\_H,H),

**pack**(T,Res1),

**concat**(Str\_H,Res1,Res).

% keyword(Keyword):-

% rule(\_,Keyword,\_,\_),!.

% keyword(Keyword):-

% rule(\_,\_,Keyword,\_),!.

**do\_right\_form** (Kname, Lname):-

**func\_keyword** (Kname),

**rule** (**\_**,Lname,**\_**,**\_**),

**go**([],Lname), !.

**do\_right\_form** (Kname,**\_**):-

**frontstr** (**3**, Kname, Word, **\_**),

**upper\_lower\_cyr** (Word, Key),

Key="вый",!,

exit.

**do\_right\_form** (Kname,**\_**):-

**frontstr** (**3**, Kname, Word, **\_**),

**upper\_lower\_cyr** (Word, Key),

Key="вых",!,

exit.

**do\_right\_form** (Kname,**\_**):- **upper\_lower\_cyr** (Kname, Word),

Word="exit", !,

exit.

**do\_right\_form** (Kname,**\_**):-

**upper\_lower\_cyr** (Kname, Word), Word="quit", !,

exit.

**do\_right\_form** (Kname, Lname):-

**concat** ("Введенные Вами ключевые слова : ", Kname, Temp),

**concat** (Temp, " и ",Temp1),

**concat** (Temp1, Lname, Temp2),

**concat** (Temp2, " не известны системе. ", Msg),

**dlg\_Note** (Msg), !.

**func\_keyword** (Word):-

**frontstr** (**3**, Word, Key, **\_**),

**upper\_lower\_cyr** (Key, KeyTr),

KeyTr="най", !.

% first\_keyword\_in\_sentence([Head|\_],Head):-

% keyword(Head),!.

% first\_keyword\_in\_sentence([\_|Tail],First\_keyword):-

% first\_keyword\_in\_sentence(Tail,First\_keyword).

% first\_keyword\_in\_sentence([],\_):-!,info,fail.

**last**([X], X):-!.

**last**([**\_**Head|Tail], Element):-

**last**(Tail, Element).

**goes**(Mygoal,First\_keyword):-

Mygoal=**dlg\_GetStr**("Консультация","Введите запрос : ","Текст запроса"),

**convers**(Mygoal,Word\_list),

/\* do\_right\_form (Word\_list, First\_keyword), \*/

**last**(Word\_list, Lname),

**last**(Word\_list, First\_keyword),

**reverse**(Word\_list, RWrod\_list),

**last**(RWrod\_list, Kname),

**do\_right\_form**(Kname, Lname),!.

% first\_keyword\_in\_sentence(Word\_list,First\_keyword),!.

**go**(**\_**,Mygoal):-

not(**rule**(**\_**,Mygoal,**\_**,**\_**)),!,

**concat**("Это вирус: ",Mygoal,Temp),

**concat**(Temp,".",Result),

**dlg\_Note**("Экспертное заключение : ",Result).

**go**(History,Mygoal):-

**rule**(Rule\_number,Mygoal,Type\_of\_Virus,Conditions),

**check**(Rule\_number,History,Conditions),

**go**([Rule\_number|History],Type\_of\_Virus).

**check**(Rule\_number,History,[Virus\_cond|Rest\_Virus\_cond\_list]):-

**yes**(Virus\_cond),!,

**check**(Rule\_number,History,Rest\_Virus\_cond\_list).

**check**(**\_**,**\_**,[Virus\_cond|**\_**]):-

**no**(Virus\_cond),!,fail.

**check**(Rule\_number,History,[Virus\_cond|Rest\_Virus\_cond\_list]):-

**cond**(Virus\_cond,Text),

**ask\_question**(Virus\_cond,Text),

**check**(Rule\_number,History,Rest\_Virus\_cond\_list).

**check**(**\_**,**\_**,[]).

**do\_answer**(Cond\_number,**1**):-!,

**assertz**(**yes**(Cond\_number)).

**do\_answer**(Cond\_number,**2**):-!,

**assertz**(**no**(Cond\_number)),fail.

**erase**:-**retract**(**\_**),fail.

erase.

**clear**:-**retract**(**yes**(**\_**)),**retract**(**no**(**\_**)),fail,!.

clear.

%BEGIN Task Window, e\_Create

**task\_win\_eh**(**\_**Win,**e\_Create**(**\_**),**0**):-!,

%BEGIN Task Window, InitControls, 21:26:22-22.12.2007, Code automatically updated!

%END Task Window, InitControls

%BEGIN Task Window, ToolbarCreate, 21:26:22-22.12.2007, Code automatically updated!

**tb\_project\_toolbar\_Create**(**\_**Win),

**tb\_help\_line\_Create**(**\_**Win),

%END Task Window, ToolbarCreate

ifdef use\_message

**msg\_Create**(**100**),

enddef

!.

%END Task Window, e\_Create

%MARK Task Window, new events

%BEGIN Task Window, id\_file

**task\_win\_eh**(**\_**Win,**e\_Menu**(id\_file,**\_**ShiftCtlAlt),**0**):-!,

erase,

assert\_database,

do\_consulting,!.

%END Task Window, id\_file

%BEGIN Task Window, id\_help\_contents

**task\_win\_eh**(**\_**Win,**e\_Menu**(id\_help\_contents,**\_**ShiftCtlAlt),**0**):-!,

**vpi\_ShowHelp**("logic\_es\_win.hlp"),

!.

%END Task Window, id\_help\_contents

%BEGIN Task Window, id\_help\_about

**task\_win\_eh**(Win,**e\_Menu**(id\_help\_about,**\_**ShiftCtlAlt),**0**):-!,

**dlg\_about\_dialog\_Create**(Win),

!.

%END Task Window, id\_help\_about

%BEGIN Task Window, id\_file\_exit

**task\_win\_eh**(Win,**e\_Menu**(id\_file\_exit,**\_**ShiftCtlAlt),**0**):-!,

**win\_Destroy**(Win),erase,

!.

%END Task Window, id\_file\_exit

%BEGIN Task Window, e\_Size

**task\_win\_eh**(**\_**Win,**e\_Size**(**\_**Width,**\_**Height),**0**):-!,

ifdef use\_tbar

**toolbar\_Resize**(**\_**Win),

enddef

ifdef use\_message

**msg\_Resize**(**\_**Win),

enddef

!.

%END Task Window, e\_Size

%END\_WIN Task Window

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Invoking on-line Help

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**project\_ShowHelpContext**(HelpTopic):-

**vpi\_ShowHelpContext**("logic\_es\_win.hlp",HelpTopic).

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Main Goal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

goal

ifdef use\_mdi

**vpi\_SetAttrVal**(attr\_win\_mdi,b\_true),

enddef

ifdef ws\_win

ifdef use\_3dctrl

**vpi\_SetAttrVal**(attr\_win\_3dcontrols,b\_true),

enddef

enddef

**vpi\_Init**(task\_win\_Flags,task\_win\_eh,task\_win\_Menu,"logic\_es\_win",task\_win\_Title).

%BEGIN\_TLB Project toolbar, 18:45:04-21.12.2007, Code automatically updated!

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Creation of toolbar: Project toolbar

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

clauses

**tb\_project\_toolbar\_Create**(**\_**Parent):-

ifdef use\_tbar

**toolbar\_create**(tb\_top,**0**xC0C0C0,**\_**Parent,

[**tb\_ctrl**(id\_file\_new,pushb,idb\_new\_up,idb\_new\_dn,idb\_new\_up,"New;New file",**1**,**1**),

**tb\_ctrl**(id\_file\_open,pushb,idb\_open\_up,idb\_open\_dn,idb\_open\_up,"Open;Open file",**1**,**1**),

**tb\_ctrl**(id\_file\_save,pushb,idb\_save\_up,idb\_save\_dn,idb\_save\_up,"Save;File save",**1**,**1**),

separator,

**tb\_ctrl**(id\_edit\_undo,pushb,idb\_undo\_up,idb\_undo\_dn,idb\_undo\_up,"Undo;Undo",**1**,**1**),

**tb\_ctrl**(id\_edit\_redo,pushb,idb\_redo\_up,idb\_redo\_dn,idb\_redo\_up,"Redo;Redo",**1**,**1**),

separator,

**tb\_ctrl**(id\_edit\_cut,pushb,idb\_cut\_up,idb\_cut\_dn,idb\_cut\_up,"Cut;Cut to clipboard",**1**,**1**),

**tb\_ctrl**(id\_edit\_copy,pushb,idb\_copy\_up,idb\_copy\_dn,idb\_copy\_up,"Copy;Copy to clipboard",**1**,**1**),

**tb\_ctrl**(id\_edit\_paste,pushb,idb\_paste\_up,idb\_paste\_dn,idb\_paste\_up,"Paste;Paste from clipboard",**1**,**1**),

separator,

separator,

**tb\_ctrl**(id\_help\_contents,pushb,idb\_help\_up,idb\_help\_down,idb\_help\_up,"Help;Help",**1**,**1**)]),

enddef

true.

%END\_TLB Project toolbar

%BEGIN\_TLB Help line, 18:45:04-21.12.2007, Code automatically updated!

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Creation of toolbar: Help line

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

clauses

**tb\_help\_line\_Create**(**\_**Parent):-

ifdef use\_tbar

**toolbar\_create**(tb\_bottom,**0**xC0C0C0,**\_**Parent,

[**tb\_text**(idt\_help\_line,tb\_context,**452**,**0**,**4**,**10**,**0**x0,"")]),

enddef

true.

%END\_TLB Help line

%BEGIN\_DLG About dialog

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Creation and event handling for dialog: About dialog

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

constants

%BEGIN About dialog, CreateParms, 16:03:46-31.5.2021, Code automatically updated!

dlg\_about\_dialog\_ResID = idd\_dlg\_about

dlg\_about\_dialog\_DlgType = wd\_Modal

dlg\_about\_dialog\_Help = idh\_contents

%END About dialog, CreateParms

predicates

dlg\_about\_dialog\_eh : EHANDLER

clauses

**dlg\_about\_dialog\_Create**(Parent):-

**win\_CreateResDialog**(Parent,dlg\_about\_dialog\_DlgType,dlg\_about\_dialog\_ResID,dlg\_about\_dialog\_eh,**0**).

%BEGIN About dialog, idc\_ok \_CtlInfo

**dlg\_about\_dialog\_eh**(**\_**Win,**e\_Control**(idc\_ok,**\_**CtrlType,**\_**CtrlWin,**\_**CtrlInfo),**0**):-!,

**win\_Destroy**(**\_**Win),

!.

%END About dialog, idc\_ok \_CtlInfo

%MARK About dialog, new events

**dlg\_about\_dialog\_eh**(**\_**,**\_**,**\_**):-!,fail.

%END\_DLG About dialog