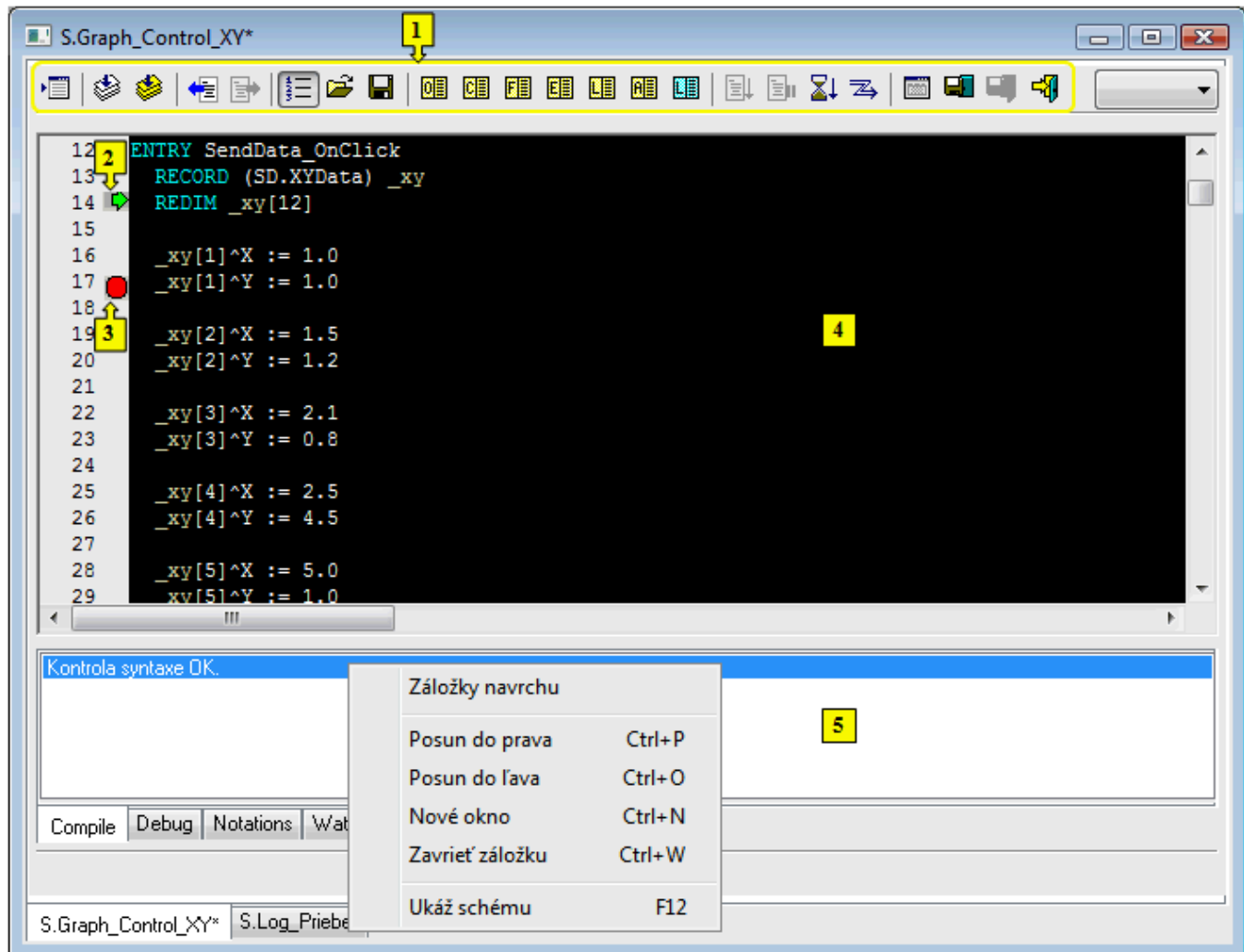




# Editor skriptu

Obrázok znázorňuje lenenie pracovnej plochy editora skriptov. Pre každý novo otvorený skript sa automaticky vytvorí nová záložka.



1 [Nástrojový panel](#) (toolbar).


2 Aktuálna akcia pri ladení skriptu .

3 Break point pri ladení skriptu .

4 Zdrojový text editovaného skriptu - jednotlivé akcie tvoriace skript.

5 as okna v záložkách zobrazujúca:

- *Compile* - chyby pri preklade (syntaktická, sémantická kontrola),
- *Debug* - ladiace výpisy,
- *Notations* - poznámky nachádzajúce sa v skripte,
- *Watch* - hodnoty lokálnych premenných.

Pomer medzi touto asou okna a plochou pre editovanie textu je možné meniť po zmene tvaru myši na  (po prechode myši na ich rozhranie) a so súasnym stlačením avého tlačidla myši a potiahnutím okraja do želanej pozície.

Nad záložkami je možné vyvola nasledovné popup menu:

✓	Záložky navrchu	
	Posun do prava	Ctrl+P
	Posun do ľava	Ctrl+O
	Nové okno	Ctrl+N
	Zavrieť záložku	Ctrl+W
	Ukáž schému	F12

- **Záložky navrchu** - ak je voba zaškrtnutá, záložky sa nachádzajú v hornej asti, inak sú v dolnej asti.
- **Posun doprava** - presunie záložku o jednu pozíciu doprava.
- **Posun doľava** - presunie záložku o jednu pozíciu doľava.
- **Nové okno** - otvorí záložku v samostatnom okne.
- **Zavrieť záložku** - umožní zavrieť záložku, nad ktorou bolo vyvolané popup menu (v prípade klávesovej skratky Ctrl+W zatvára aktuálnu záložku).
- **Ukáž schému** - vyvolá editovanú schému a minimalizuje ESL skript. Voba je aktívna len pre skript schémy, pre skript eventu je neaktívna.

#### Poznámky:

Titulok okna obsahuje meno aktívnej záložky. Toto meno sa súčasne presúva aj do titulu CNF, ak je okno v rámci CNF maximalizované. Ak je okno záložiek minimalizované alebo skryté a pribudne záložka (prípadne je otvorená existujúca záložka), okno záložiek sa opäť zobrazí. Na aktívnej záložke má fókus pole pre editáciu skriptu.

Pri zatvorení editora skriptov sa zatvoria všetky záložky a okno sa skryje.

Pri zatváraní záložiek platia tieto pravidlá:

- skript schémy (otvorený v [D2000 GrEditor](#)) nezobrazuje dotaz, i sa má uložiť, ak boli na skripte prevedené zmeny (skript je vo vlastníctve rozeditovanej schémy)
- skript eventu (otvorený v [D2000 CNF](#) alebo [D2000 GrEditor](#)) zobrazuje pri zatváraní záložky dotaz, i sa má uložiť, ak boli na skripte prevedené zmeny

Použitie klávesových skratiek:

Kombinácia kláves	Akcia
CTRL+N	Otvorenie záložky v samostatnom okne.
CTRL+O	Presun záložky o jednu pozíciu doľava.
CTRL+P	Presun záložky o jednu pozíciu doprava.
CTRL+W	Zatvorenie aktuálnej záložky.
F12	Vyvolá schému k danému skriptu a skript minimalizuje.

#### Vlastnosti editora:

- Automaticky farebne odlišuje kúové slová: **END**.
- Automaticky farebne odlišuje správne zapísané texty: **"Text"**, chybné texty: **Text**.
- Po úspešnej [Kontrola syntaxe](#):
  - Za menom objektu typu [Štruktúrovaná premenná](#) alebo menom lokálnej premennej typu **RECORD** sa po zápise znaku '^' zobrazí zoznam stpcov štruktúry s íslom stpca a jeho popisom. Názvy stpcov v zozname sú abecedne zoradené.

```
RECORD (SD.FIND_FILES) _test
REDIM _test[1]
_test[1]^
```

AccessTime	[3]	Čas, kedy bolo k súb...
bFile	[1]	@TRUE, ak ide o súbo...
CreateTime	[2]	Čas, kedy bol súbor ...
FileName	[6]	Meno súboru.
FileSize	[5]	Veľkosť súboru v baj...
FullAttrib	[7]	Atribúty súboru (adr...
LastWriteTime	[4]	Čas, kedy bolo do sú...

- Umiestnenie kurzora myši nad lokálnu premennú alebo položku štruktúrovanej premennej spôsobí zobrazenie informácií o jej type a mieste deklarácie (v ladiacom režime je zobrazovaná hodnota).
- Po stlačení pravého tlačidla myši nad identifikátorom (lokálna premenná, meno procedúry) sa zobrazí miestna ponuka obsahujúca jedinú položku **Go to definition** - po jej stlačení sa kurzor automaticky presunie na miesto, kde je identifikátor deklarovaný.
- Po neúspešnej **Kontrola syntaxe** alebo **Kompilácii**, dvojitým stlačením myši v zozname chýb (obrázok oblasť 5) sa kurzor presunie na výskyt chyby.
- Stlačenie F1 (help), keď je kurzor umiestnený v rámci mena funkcie/akcie, vyvolá help pre danú funkciu/akciu/udalosť.
- Automatické zobrazovanie popisu zadanej funkcie aj s jej typmi parametrov. Po zadaní názvu funkcie a prvej zátvorky v editore sa zobrazí okno s popisom funkcie a jej typy parametrov. Pri prechode na nový parameter je aktuálny parameter označený tmavomodrou farbou. To isté sa deje aj pri aktualizácii parametrov už zadanej funkcie.

Príklad:

```

%SetItemValue(
END
SetItemValue

Funkcia nastaví hodnotu položky danú číslom riadku a stĺpca

Deklarácia :

%SetItemValue(HBJ, INT, INT, NONE)

```

- Automatické zobrazovanie zadanej akcie s jej parametrami. Po zadaní názvu akcie a stlačení klávesy SPACE sa v editore zobrazí okno s jej deklaráciami. Postupne pri dopínaní jej parametrov sa zobrazia aktuálne možnosti.

Príklady:

```

DB READ BLOB handleIdent Int, BlobColNameIdent Str, blobFileNameIdent Str,
DB READ BLOB handleIdent Int, blobColNameIdent Str, blobFileNameIdent Str, retCodeIdent Int, idKeyIdent
DB READ BLOB handleIdent Int, blobColNameIdent Str, blobFileNameIdent Str, retCodeIdent Int [ WHERE strExpression Str ]

```

```

DB READ BLOB handleIdent Int, BlobColNameIdent Str, blobFileNameIdent Str, retCodeIdent Int,
DB READ BLOB handleIdent Int, blobColNameIdent Str, blobFileNameIdent Str, retCodeIdent Int, idKeyIdent

```

```

DB READ BLOB handleIdent Int, BlobColNameIdent Str, blobFileNameIdent Str, retCodeIdent Int WHERE
DB READ BLOB handleIdent Int, blobColNameIdent Str, blobFileNameIdent Str, retCodeIdent Int WHERE strExpression Str

```

- Po zadaní identifikátora objektu (identifikátor objektu schémy alebo systémového server eventu) sa zobrazí zoznam RPC procedúr a interface, ktoré sú pre daný objekt definované. Po vybratí položky zo zoznamu sa definícia procedúry alebo interface doplní do ESL Editoru. Ponuka sa zobrazí po zadaní znaku "]".

Príklad:

```

TEXT _msg
_msg := " ... "
CALL [E.MSGServer]
I.MsgServer*RegisterClient(IN TEXT _clientName, IN INT _clientHObj, IN INT _clientProcessHObj, IN INT _clientUID)
I.MsgServer*UnRegisterClient(IN INT _clientUID)
I.MsgServer*GetClientList(RECORD NOALIAS (SD.prva) _clients)
I.MsgServer*SendMessage(IN INT _srcClientUID, IN INT _dstClientUID, IN TEXT _msg, IN INT _retCode)
GetClientNumber(IN INT _ahoj)
SetTime(IN TIME _time)

```

- Po zadaní znaku "^" za názvom interface sa zobrazí zoznam procedúr, ktoré sú pre daný interface definované. Po vybratí položky zo zoznamu sa definícia procedúry doplní do ESL Editoru.

Príklad:

```

TEXT _msg
_msg := " ... "
I.MsgServer
RegisterClient(IN TEXT _clientName, IN INT _clientHObj, IN INT _clientProcessHObj, IN INT _clientUID)
UnRegisterClient(IN INT _clientUID)
GetClientList(RECORD NOALIAS (SD.prva) _clients)
SendMessage(IN INT _srcClientUID, IN INT _dstClientUID, IN TEXT _msg, IN INT _retCode)

```

- Zobrazenie popisu RPC procedúr volaných v ESL skripte. Táto vlastnosť umožňuje užívateľovi vytvárať popisy RPC procedúr, ktoré sa zobrazia po umiestnení kurzora myši nad názvom volanej RPC procedúry v inom skripte ako je daná RPC procedúra deklarovaná. V ESL editore sa zobrazia komentár a parametre danej RPC procedúry. Popis môže byť jednoriadkový alebo viaciadkový. Pri viaciadkovom popise je dôležité, aby sa medzi riadkami nenachádzal prázdny riadok. Taktiež je dôležité, aby za popisom nasledovala deklarácia procedúry (bez prázdnych riadkov). Popis začína znakom ";", ako v prípade komentára.

Príklad popisu:

```

16
17 ;Procedura vrati pocet klientov
18 RPC PROCEDURE GetClientNumber(INT _nrClient)
19 END GetClientNumber
20
21
22 ;procedura nastavi cas
23 ;hodnota musi byt validna
24 RPC PROCEDURE SetTime(IN TIME _time)
25 END SetTime

```

Príklad zobrazenia:

```

60
61 CALL [E.MSGServer] SetTime(_time) ON _FROM_HIP
62
63 CALL [E.MSGServer] GetClientNumber(_a) ON _FROM_HIP
64
65 [E.MSGServer] GetClientNumber
66 -----
67 Procedura vrati pocet klientov
68
69 Parameters :
70 _nrClient : INT
71

```

- Automatické odsadenie

Pri písaní ESL skriptu a prechode na nový riadok, ESL editor automaticky odsadí aktuálny riadok (poda prvého nenulového riadku) a nastaví kurzor na danú pozíciu.

Vlastnosti automatického odsadzovania:

1. Pri prechode na nový riadok po vybraných akciách (napr. RPC, PROCEDURE, PUBLIC, FOR, DO\_LOOP, IF... ) v ESL editore sa automaticky odsadí text o 2 znaky doprava od predchádzajúceho riadka.
2. Pri prechode na nový riadok po vybraných akciách (napr. END\_LOOP, ENDIF... ) v ESL editore sa automaticky odsadí text o 2 znaky doava od predchádzajúceho riadka.
3. Pri stlačení kombinácií kláves Ctrl + i sa vyznaený text naformátuje poda predchádzajúcich vlastností. Pre túto akciu je smerodajné odsadenie prvého riadku vo vyznaenej množine riadkov.

Zapnutie/Vypnutie automatického odsadenia je možné v dialógu [Nastavenia](#) prístupného z menu **Systém**.

- Kombinácie kláves:

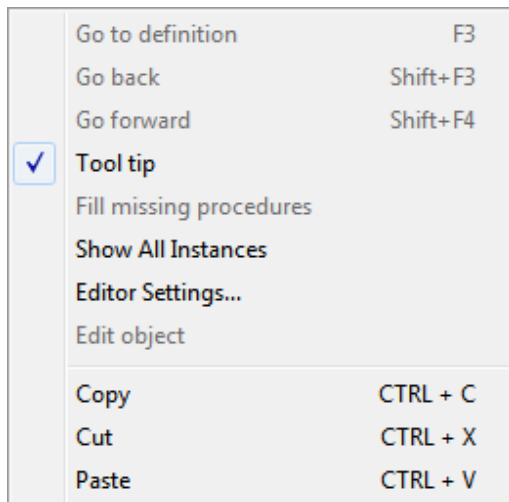
Kombinácia kláves	Akcia
CTRL+K	Vytvorenie poznámky na aktuálnom riadku alebo skupine riadkov pri vyznaenom bloku.
CTRL+SHIFT+K	Odsadenie poznámky na aktuálnom riadku alebo skupine riadkov pri vyznaenom bloku.
CTRL+F	Zobrazenie dialógu pre vyhádavanie alebo nahrádzanie textu v skripte.
CTRL+G	Prechod na daný riadok v texte.
CTRL+i	Odsadenie oznaenej asti skriptu. Smerodajné odsadenie pre túto akciu je odsadenie prvého riadku vo vybranej asti skriptu.
CTRL+S	Pre skript aktívnej schémy - uloženie Schémy. Pre skript objektu typu Event - uloženie.
CTRL+1	Otvorenie <a href="#">zoznamu objektov</a> systému D2000.
CTRL+2	Otvorenie <a href="#">zoznamu preddefinovaných konštánt</a> .
CTRL+3	Otvorenie <a href="#">zoznamu funkcií</a> .
CTRL+4	Otvorenie <a href="#">zoznamu akcií</a> .
CTRL+5	Otvorenie <a href="#">zoznamu lokálnych premenných</a> .
CTRL+6	Otvorenie <a href="#">zoznamu atribútov</a> hodnoty objektu alebo lokálnej premennej.
CTRL+~	ESL editor automaticky doplní rozpísané slovo poda najbližšieho vhodného slova od pozície kurzora a neskôr poda jeho typu (funkcia, konštanta, akcia...) opakovaným stláaním postupne dopa alšie zodpovedajúce slová.

CTRL+>

Zobrazenie tool tipu v ESL editore funguje aj pri vypnutom automatickom tool tipe.

## Popup menu

Nad zdrojovým textom skriptu (as 4 na obrázku) je možné vyvola popup menu (kliknutím pravým tlačidlom myši alebo stlačením klávesy **Menu** na klávesnici) s týmito položkami:



- **Go to definition** (F3) - kurzor sa presunie na miesto, kde je deklarovaný identifikátor. Pri použití "Go to definition" na vzdialenú procedúru ([RPC/PUBLIC](#)), ESL editor automaticky otvorí skript, v ktorom sa nachádza definícia procedúry a nastaví kurzor na definíciu.
- **Go back** (Shift + F3)- kurzor sa presunie na miesto, odkiaľ bolo volané "Go to definition". Pamätá si posledných 50 spätných volaní "Go to definition".
- **Go forward** (Shift + F4) - kurzor sa presunie na miesto, na ktorom bol pred volaním "Go back". "Go back" a "Go forward" je podporené aj medzi rôznymi ESL skriptami. ESL editor automaticky vyberie / otvorí ESL skripty.
- **Tool tip** - povolí / zakáže zobrazenie tooltipu.
- **Fill missing procedures** - doplní chýbajúce implementácie procedúr (s prázdny telom), ktoré sú definované v ESL rozhraní deklarovanom v ESL skripte. Ak sa menu vyvolá kliknutím pravým tlačidlom myši a kurzor sa nachádza nad deklaráciou použitého ESL rozhrania, doplnia sa len jeho procedúry. Inak sa doplnia procedúry všetkých ESL rozhraní deklarovaných v ESL skripte.
- **Show All Instances** - zobrazí zoznam všetkých bežiacich inštancií editovaného [ESL skriptu](#).
- **Editor Settings...** - zobrazí dialóg nastavení ESL editora (písmo a farby).
- **Edit object** - ak je text pod kurzorom myši platné meno objektu, vobou ho môžeme otvoriť na editáciu.
- **Copy** (CTRL+C) - skopíruje vybraný obsah do schránky,
- **Cut** (CTRL+X) - odstráni vybraný obsah,
- **Paste** (CTRL+V) - prilepí obsah zo schránky.

## Farbenie ESL skriptu

ESL editor pre zlepšenie prehľadnosti ESL skriptu farebne rozlišuje jednotlivé slová podľa ich typu. Prednastavené parametre farbenia je možné upraviť pomocou dialógu nastavení ESL editora.



ESL Editor v rámci *Stringu* rozpoznáva odkazy do *slovníka*. V prípade, že bol identifikovaný odkaz do slovníka (kú), ktorý nie je definovaný, bude farbený ako *Chyba* (vi dialóg vyššie).

*Príklad:*

Kú D2\_ActAlarm v slovníku existuje a D2\_ActAlarmAAAAA v slovníku neexistuje.

```

16  _t := "D2_ActAlarm"
17  _t := ".... {!D2_ActAlarm} . text."
18  _t := ".... {!D2_ActAlarmAAAAA} . text."

```