

ESL Unit Event


ESL UNIT predstavuje nový typ objektu **EVENT**. Umožňuje programátorovi nakonfigurovať univerzálne skripty s procedúrami a premennými a tieto využívať v ESL skriptoch (ako knižnice). Sprístupnené sú iba procedúry označené vymenovaným slovom **PUBLIC**. V Unite je možné využívať všetky dostupné ESL [akcie](#) a [funkcie](#). Jediný rozdiel od ESL skriptu je v tom, že v Unite nemôžu byť definované RPC procedúry a [ESL Interface](#).

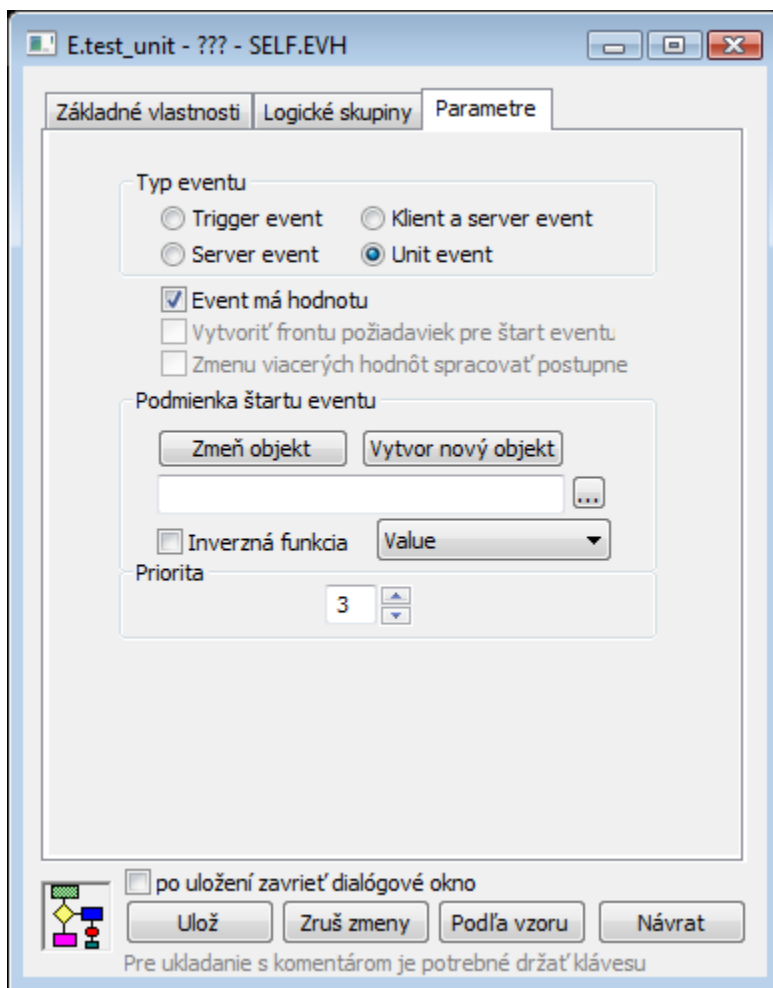
Každou deklaráciou UNITu v skripte vzniká v systéme nový samostatný objekt so svojimi lokálnymi premennými a procedúrami. Je to aj hlavný rozdiel od `#include` v C, kde `#include` vkladá do skriptu celý text knižnice, ktorý obsahuje deklarácie funkcií. Táto vlastnosť UNITu umožňuje v skripte zadefinovať viac ESL UNITov rovnakého typu, pričom každý môže mať rôzne hodnoty svojich lokálnych premenných (každý UNIT má svoj vlastný životný cyklus).

Životný cyklus UNITu je obmedzený dĺžkou životného cyklu skriptu v systéme, ktorý daný UNIT deklaruje a používa. Životný cyklus UNITu začína jeho deklaráciou v skripte a koní zánikom skriptu v systéme, v ktorom bol UNIT deklarován.

Adresovanie Public procedúr UNITu znamená zavolanie procedúr konkrétneho UNITu pomocou lokálnej premennej, ktorá ho reprezentuje. Volanie procedúr UNITu je dovolené iba z ESL skriptu, ktorý daný UNIT deklaruje.

Konfigurácia

Konfigurácia prebieha v [konfiguračnom nástroji D2000 CNF](#). UNIT sa vytvorí ako nový objekt typu EVENT. Pri vytváraní nového eventu sa automaticky otvorí [editor skriptov](#). V ňom sa po stlačení tlačidla  **Parametre** otvorí konfiguračné okno. Na záložke **Parametre** vyberte typ eventu **Unit event**.



Rodiom UNITu môže byť ubovený proces typu *.EVH. Event typu UNIT sa pri štarte procesu nezane vykonáva. Konfigurácia skriptu v ESL UNITe je rovnaká ako pri iných typoch eventov. Využívajú sa všetky vlastnosti ESL jazyka.

Výnimky:

- Rozšírenie o kúové slovo **PUBLIC**.
- Zakázanie RPC procedúr a **ESL Interface**.

Inicializácia skriptu (as medzi **BEGIN** - **END**) sa vykoná vždy. Akcie skriptu UNITu sa vykonávajú v kontexte inštancie ESL skriptu, v ktorom je UNIT deklarován. K eventu typu UNIT sa nedá pripojiť pre ladenie. Popis ladenia je popísaný v sekcii [Ladenie skriptov typu UNIT](#).

Príklad:

```
;*****
; DESCRIPT: Unit1 - Counter
;
;
; AUTHOR: Programmer
; LAST CHANGE:
;*****

INT _iLocal
;
PUBLIC PROCEDURE Make
    _iLocal := _iLocal + 1
END Make

PUBLIC PROCEDURE GetValue(INT _iValue)
    _iValue := _iLocal
END GetValue

BEGIN
    _iLocal := 0
END
```

Deklarácia a použitie Unitu

```
UNIT (Nazov_eventu_typu_unit) _unit1
```

UNIT	Typ lokálnej premennej.
(Nazov_eventu_typu_unit)	Typ UNIT eventu, napr E.unitCounter.
_unit1	Meno deklarovanej lokálnej premennej, ktorá bude daný UNIT reprezentova.

Pri použití UNITu v ESL skripte musí by deklarácia lokálnej premennej typu UNIT v ESL skripte vždy na zaiatku skriptu. Pri štarte skriptu, ktorý obsahuje deklarácie UNIT-ov, sa postupne vykonajú inicializané asti týchto UNIT-ov v poradí ako sú deklarované. Ak sa v deklaráciách lokálnych premenných nachádza viac UNITov rovnakého typu, vykonajú sa inicializané asti všetkých. Implementane je podporené deklarovanie UNITu v UNITE. Týmto je umožnené využí vlastnosti UNITu v inom UNITE. Pre túto vlastnos je podporená detekcia uviaznutia UNITov.

Volanie Public procedúr je zabezpeené akciou [CALL](#), za ktorou musí by zadaný adresát (_unit1), ktorý obsahuje danú PUBLIC procedúru.

```
CALL [_unit1] Make
CALL [_unit1] GetValue(_value)
```

CALL	Akcia CALL.
(Nazov_eventu_typu_unit)	Adresovanie požadovaného UNITu.
_unit1	Meno procedúry + parametre.

Vykonanie sa zabezpeí prepnutím kontextu, vykoná sa volaná procedúra a kontext sa prepne spä na vykonávanie pôvodného skriptu.

Poznámka: Adresát [_unit1] musí by vždy zadaný menom lokálnej premennej, ktorá bola použitá pri deklarácii Unitu.

Príklad:

```

;*****
; DESCRIPT: Unit Caller
;
;
; AUTHOR: Programmer
; LAST CHANGE:
;*****

UNIT (E.Unit1) _unit1
UNIT (E.Unit1) _unit12
UNIT (E.Unit2) _unit2

RPC PROCEDURE CheckValue(BOOL _bOk)
INT _iValue
INT _iValue2

CALL [_unit1] GetValue(_iValue)
CALL [_unit12] GetValue(_iValue2)

_bOk := _iValue # _iValue2

END CheckValue

BEGIN

CALL [_unit1] Make
CALL [_unit12] Make
CALL [_unit12] Make

END

```

V jednom skripte je dovolené deklarovať 1 až n UNITov rovnakého typu, napr. *_unit1* a *_unit12* sú rovnakého typu *E.Unit1*. Táto vlastnosť je zabezpečená tým, že každý deklarovaný UNIT má svoj vlastný životný cyklus v systéme. To znamená, že UNITY rovnakého typu po rôznych operáciách môžu mať rôzne hodnoty lokálnych premenných.

Ladenie skriptov typu Unit

ESL Unit sa ladí spoločne so skriptom, v ktorom je deklarovaný.

Ak použijeme možnosť ladenia [Step Into](#) (F8 - Krokovanie s vnáraním sa), ESL editor automaticky zabezpečí otvorenie záložky UNITu so skriptom a nastaví aktuálne vykonávaný riadok. Pri ladení skriptu, ktorý obsahuje deklarácie UNITov rovnakého typu, ESL editor otvorí potrebné množstvo záložiek so skriptami UNITov.

Po ukončení vykonania požadovanej časti skriptu UNITu sa prepne záložka späť do záložky skriptu, ktorý volal UNIT. Ukončením ladenia skriptu, ESL editor automaticky pozatvára záložky skriptov UNITov, ktoré používal pri ladení.



Súvisiace stránky:

[Eventy](#)

[Štartovacie podmienky](#)

[Eventy - položky konfigurácie okna](#)