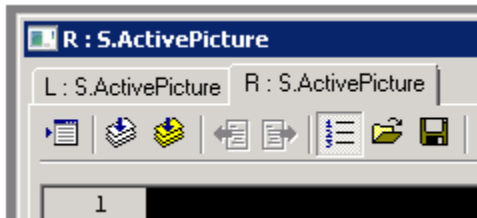


# HIS Server

**HIS Server** je vlastnosť, ktorá umožňuje rozdelenie implementácie ESL skriptov aktívnych schém (PICTURE) a objektov typu [Event](#) (EVENT), otvorených na strane klienta, na dve časti. Časť, ktorá sa vykonáva na strane klienta (lokálny skript) a časť, ktorá je vykonávaná na strane servera (remote skript). Toto rozdelenie umožňuje aplikovanému programátorovi rozdeliť funkčnosť na dva celky podľa toho, s ktorou časťou systému spolupracujú intenzívnejšie. Je predpoklad, že skript na strane klienta bude intenzívne spolupracovať práve s užívateľským rozhraním prostredníctvom rýchleho komunikačného kanála a interakciu so serverom obmedzí na minimum. A naopak, skript na strane servera bude intenzívne spolupracovať s časťami aplikácie na strane servera a interakciu s užívateľským rozhraním obmedzí na minimum. Toto rozdelenie je v rukách aplikovaného programátora.

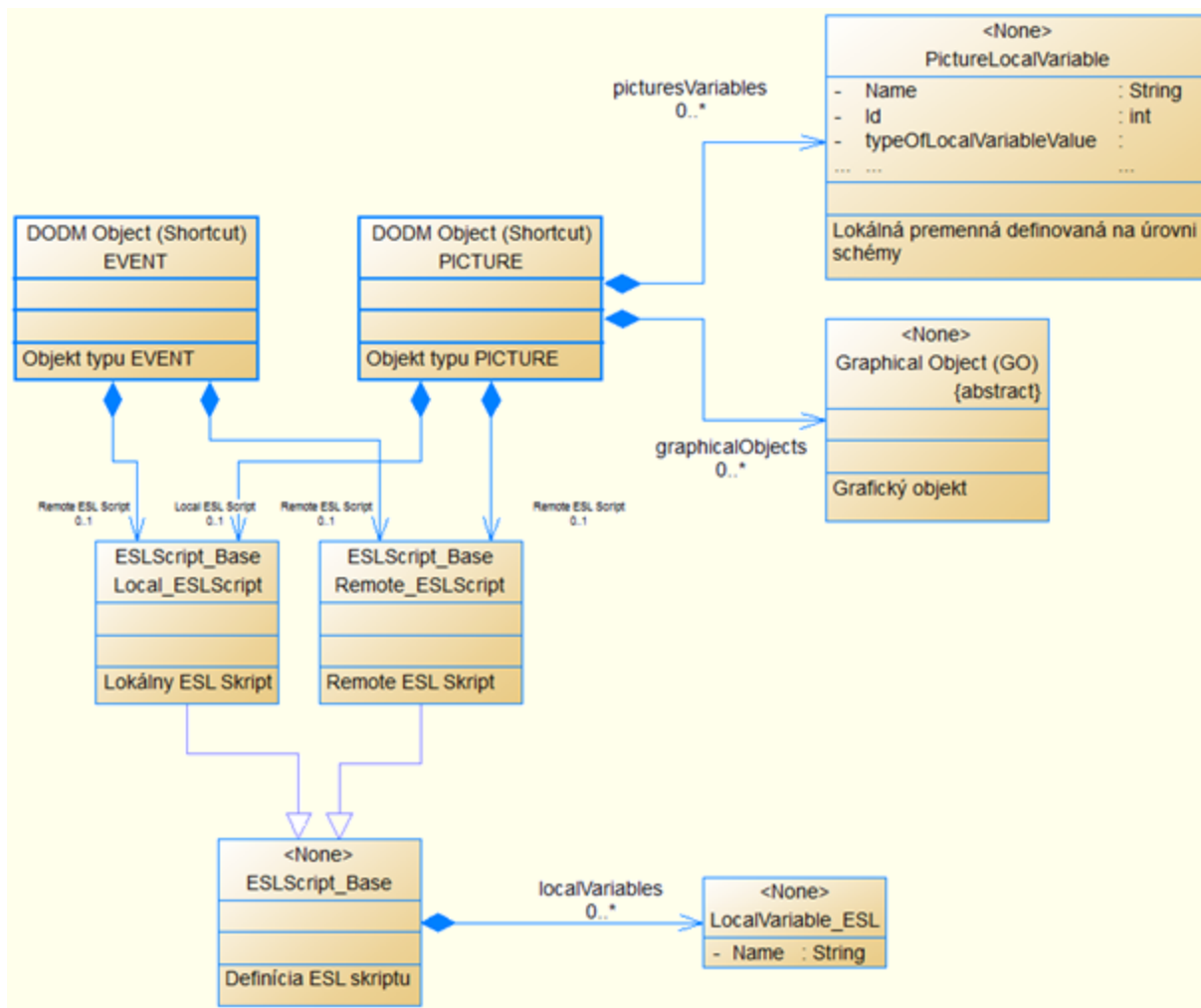
Pre ladenie výkonu a odozvy jednotlivých akcií je určený nástroj [ESL Profiler](#). Časti sú reprezentované dvoma ESL skriptami, ktoré sú súčasťou konfigurácie príslušného objektu. Z používateľského (aplikovaného) hľadiska je táto skutočnosť vo veľkej miere transparentná a neprináša do konfigurácie systému ďalšiu neprimeranú zložitosť.

Vizuálne sú ESL skripty v editore rozdelené nasledujúcim spôsobom:



L: predstavuje lokálny skript  
R: predstavuje remote (vzdialený) skript

Zjednodušený dátový model schémy a eventu:



Vysvetlivky:

- EVENT – objekt typu [Event](#)

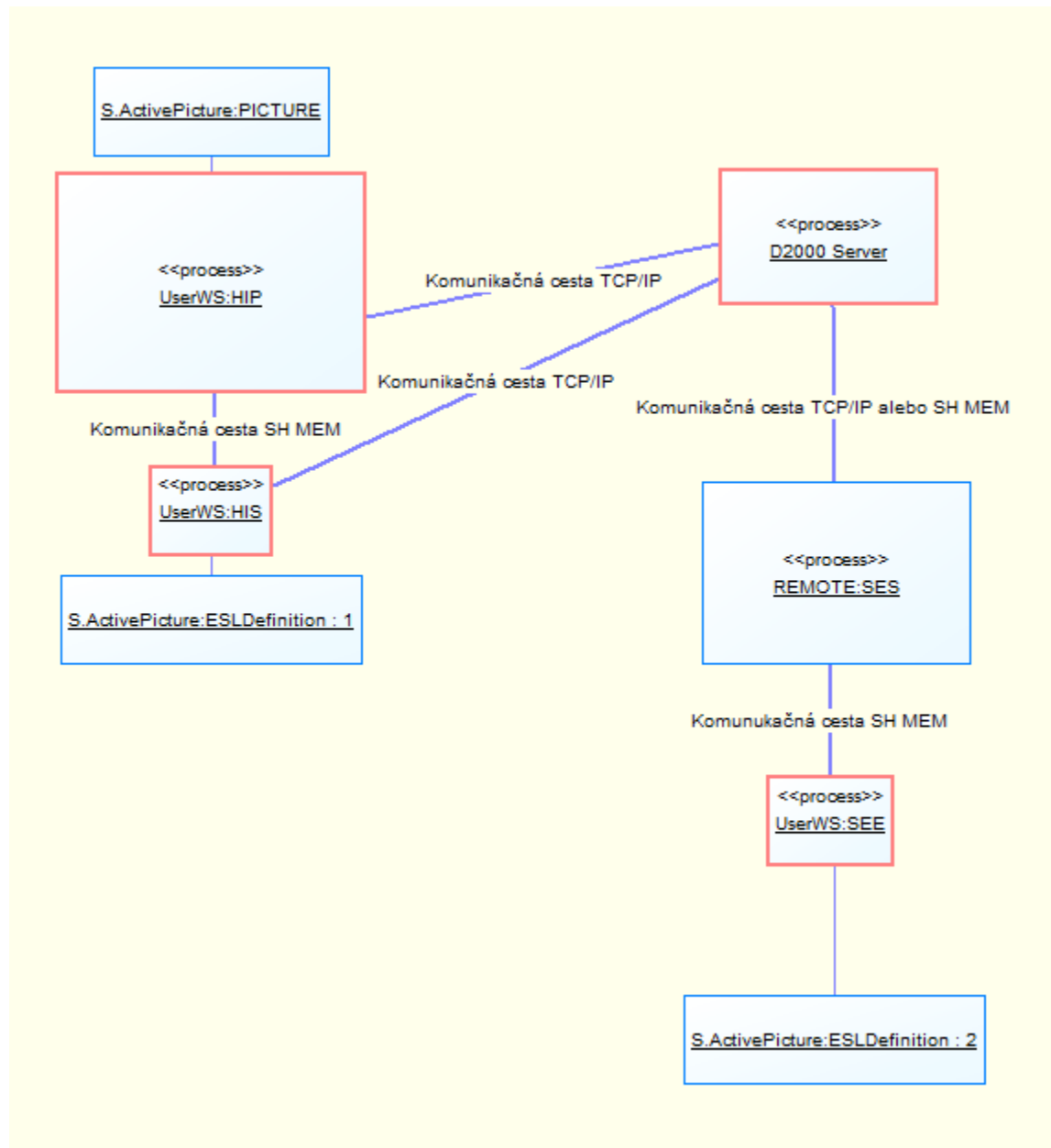
- PICTURE – objekt typu [Schéma](#)
- Local\_ESLScript – ESL skript vykonávaný na strane klienta
- Remote\_ESLScript – ESL skript vykonávaný na strane servera
- Graphical Object (GO) – [grafický objekt](#) na schéme
- PictureLocalVariable – [lokálna premenná](#) definovaná na úrovni schémy
- LocalVariable\_ESL – [lokálna premenná](#) definovaná v rámci ESL skriptu

Funkcionalitu **lokálneho ESL skriptu** (na strane klienta) zabezpečuje proces HIS (meno fyzického procesu je obyčajne odvodené od mena príslušného procesu HIP), ktorého implementácia je zapuzdrená do knižnice *event.dll*.

Funkcionalitu **remote ESL skriptu** (na strane servera) zabezpečuje virtuálny proces **SEE** (**S**cript **E**xecution **E**ngine). Procesy **SEE** obsluhuje proces **SES** (**S**cript **E**xecution **S**erver). Jeho implementácia je zapuzdrená do procesu **event.exe**.

Procesy HIS a SEE v súinnosti vykonávajú požiadavky klienta, ktorý je v tomto prípade proces HIP. Za týmto účelom je medzi nimi vytvorený virtuálny komunikačný kanál, prostredníctvom ktorého komunikujú. Z pohľadu klienta nie je dôležité fyzické umiestnenie procesu (SEE). Z pohľadu vykonávania sú oba procesy samostatné a pracujú paralelne.

Nasledujúci obrázok schematicky znázorňuje jednotlivé zúčastnené komponenty.



Proces	Popis
D2000 Server	Server proces systému D2000
UserWS.HIP	Proces D2000 HI – konzola užívateľa

UserWS_*.HIS	ESL Interpret na strane Klienta
UserWS_*.SEE	ESL Interpret na strane Servera – Script Execution Engine
REMOTE.SES	Remote server obsluhujúci procesy *.SEE.

Procesy HIS a SEE sú dynamické a ich mená sú odvodené od príslušného procesu HIP a doplnené náhodne generovaným íslom, ktoré zabezpečuje jednoznanos mena v rámci aplikácie. Proces HIP predstavuje užívateľskú konzolu, kde je otvorená schéma **S.ActivePicture**, ktorá využíva funkcionality lokálneho a vzdialeného ESL skriptu pri zaškrtnutí parametra *Server Script* v [Parametroch schémy](#).

Obrázok alej zobrazuje umiestnenie jednotlivých astí konfigurácie schémy, ktoré sú znázornené ako objekty.

Objekt	Popis
S.ActivePicture:PICTURE	Definícia schémy potrebná pre vizualizáciu.
S.ActivePicture:ESLDefinition:1	Lokálny ESL skript.
S.ActivePicture:ESLDefinition:2	Vzdialený ESL skript.

Rozloženie jednotlivých procesov je obyčajne nasledujúce:

- D2000 Server a SES bežia na strane aplikovaného servera,
- HIP a HIS bežia na strane klienta.

Prepojenia medzi jednotlivými procesmi predstavujú komunikačné kanály. V závislosti od "vzdialenosti" procesov je použitá komunikácia prostredníctvom Shared Memory alebo TCP/IP. Shared Memory je rýchly a TCP/IP naopak pomalý komunikačný kanál (táto skutočnosť sa významne prejavuje pri TCP/IP spojení v rozsiahlych sieťach, kde je prenosová rýchlosť obyčajne dostatočná, ale naopak **latencia je vysoká**, rádovo niekoľko milisekúnd). Slúžia vo všeobecnosti na prenos správ medzi procesmi, ktoré zabezpečujú funkcionality schémy ako celku podľa jej konfigurácie.

Správy je možné rozdeliť do dvoch kategórií:

#### 1. Správy pre riadenie vizualizácie

Vznikajú použitím ESL funkcií, typicky %HI\_Get\*, ...

Podľa umiestnenia príkazov, ktoré ich iniciujú, sú prenášané komunikačnými kanálmi:

- HIS <-> HIP  
Rýchly komunikačný kanál.
- SEE <-> D2000 Server <-> HIP  
Latentný komunikačný kanál.

#### 2. Správy pre interakciu so systémom

V tomto prípade systém predstavuje proces D2000 Server, ktorý vystupuje ako "router" pre vybavenie požiadaviek ESL skriptu, ktoré sú alej spracovávané jednotlivými procesmi podľa ich charakteru. Vznikajú použitím ESL akcií, typicky DB\_READ, DBS\_READ, SQL\_\*, GETARCHDATA, GETARCHARR, ...

Podľa umiestnenia príkazov, ktoré ich iniciujú, sú prenášané komunikačnými kanálmi:

- SEE <-> D2000 Server  
Rýchly komunikačný kanál.
- HIS <-> D2000 Server  
Latentný komunikačný kanál.

Vzhľadom na existenciu synchronných akcií alebo funkcií v ESL skripte je latencia mimoriadne nepríjemná, lebo významne predlžuje interaktivitu aplikácie. Tomuto sa dá významne predísť vhodným rozmiestnením jednotlivých astí ESL skriptu na stranu klienta (lokálny ESL skript) alebo servera (vzdialený ESL skript).

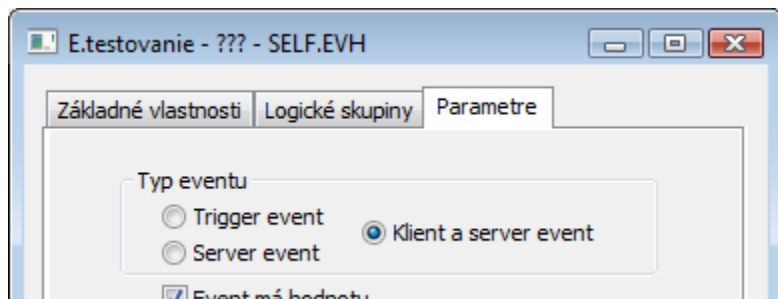
Prípadné ladenie výkonu je možné prostredníctvom nástroja [ESL Profiler](#). Ten umožňuje podrobne analyzovať vykonávanie jednotlivých akcií ESL skriptu z pohľadu rýchlostí. Pre každú akciu poskytne informáciu o ase jej vykonávania. V prípade, že ide o synchronnú akciu (dokončenie vykonania akcie je podmienené potvrdením zo strany vykonávateľa akcie) je informácia rozšírená o as, ktorý akcia strávia pasívnym akáním na potvrdenie.

Napríklad celkový as íťania z databázy akciou [DB\\_READ](#) je daný ako súčet asov.

1. Vyhodnotenie parametrov a odoslanie požiadavky pre íťanie z databázy.
2. akánie na vykonanie požiadavky.
3. Prijem odpovede a zverejnenie načítaných dát.

Bod 2 predstavuje z pohľadu vykonávania akcie DB\_READ pasívne akánie, ktorého as trvania ovplyvňuje latencia komunikačnej cesty (samozrejme aj as vykonania íťania z databázy).

Podobné princípy platia aj pre objekt typu Event otvorený na strane klienta (prostredníctvom akcie [OPENEVENT](#)). Rozdelenie ESL skriptov na lokálny a vzdialený je možné nastavením konfigurovaného parametra "Typ eventu" na hodnotu "*Klient a server event*".



Volanie RPC procedúr medzi lokálnym a remote ESL skriptom je zabezpečené volaním akcie [CALL](#). Ak sa nachádzajú v skriptoch (lokálny a remote) duplicitné deklarácie procedúr, kompilátor vyhlási chybu ("Duplicitná deklarácia procedúry") a nedovolí takúto konfiguráciu ESL skriptov uložiť (vi [Rozšírenia ESL pre HIS Server](#)).

Obsluhu všetkých udalostí (okrem definovaných výnimiek) je možné definovať v oboch skriptoch. Pri vzniku príslušnej udalosti systém zabezpečí volanie oboch obslúh. Výnimkou sa rozumie synchronná udalosť (skript aká na návratovú hodnotu). Ak sú takéto obsluhy udalostí použité v oboch skriptoch, kompilátor takúto konfiguráciu vyhodnotí s chybou ("Duplicitná deklarácia entry") a nedovolí takúto konfiguráciu ESL skriptov uložiť (vi [Rozšírenia ESL pre HIS Server](#)).

Pre zachovanie konfigurácie Remote ESL skriptu pri zmene typu eventu z "*Klient a server event*" na iný, je implementovaná vlastnosť prekopírovania zakomentovaného Remote skriptu do Lokálneho ESL skriptu. Tým je zabezpečené zachovanie konfigurácie. Analogicky to platí aj pri grafických schémach po odškrtnutí parametra "Server Script", ktorý sa nachádza v [dialógovom okne](#), ktoré sa otvorí po výbere položky **Parametre schémy** z ponuky **Nastavenia** v [D2000 GrEditor](#).



#### Súvisiace stránky:

[D2000 SES Proces](#)  
[Rozšírenia ESL pre HIS Server](#)  
[Ladenie skriptov](#)