

Dávkové spracovanie

Pod pojmom dávka príkazov (batch) sa rozumie množina príkazov, pri ktorých nezáleží na poradí ich spracovania. Dávkovým spracovaním príkazov je možné dosiahnuť zrýchlenie spracovania vďaka možnosti paralelizácie.

Dávka príkazov sa vytvára funkciou [EDA_CreateBatch](#), ktorá vracia identifikátor dávky. Následne sa do dávky pridávajú príkazy na spracovanie a to tak, že sa zavolá funkcia, ktorá podporuje dávkové spracovanie s nastaveným parametrom *batchId*. Takto zavolaná funkcia sa nevykoná hne, ale vloží sa na spracovanie v rámci dávky. Po naplnení dávky príkazmi sa spustí spracovanie funkciou [EDA_ExecuteBatchRec](#).

V prípade potreby je možné spracovanie dávky prerušiť zavolaním funkcie [EDA_CancelBatch](#).

Pretože spracovanie dávky príkazov môže byť asynchrónna operácia a pre užívateľa je potrebné zobrazovať progres spracovania dávky, existuje možnosť zaregistrovať obsluhu informovaných správ z EDA v ESL.

Dávkové spracovanie v EDA definuje dva typy informovaných správ:

- [EDA_BATCH_INFO](#)
- [EDA_BATCH_COMMAND_INFO](#)

Ich obsluha sa v ESL registruje pomocou akcie ON, za ktorou sa napíše typ obsluhovanej správy a RPC procedúra, ktorá sa pri odchycení takejto správy zavola.

Príklad použitia

```
; schema S.EDA_Batch_Test
; v tejto scheme sa vytvorí a spustí dávkové spracovanie
INT _batchId, _retCode
TIME _bt, _et
RECORD NOALIAS (SD.EDA_InsertValuesFromVector_Params_V1) _insParams
RECORD NOALIAS (SD.EDA_ExecuteBatch_Params_V1) _batchParams

; vytvorí batch
CALL %EDA_CreateBatch(_batchId, _retCode)

_inspParams[1]^structVersion := 1
_inspParams[1]^batchId := _batchId
_bt := %StrToTimeEx("2014-01-01 00:00:00", "rrrr-mm-dd hh:mi:ss")
_et := %StrToTimeEx("2015-01-01 00:00:00", "rrrr-mm-dd hh:mi:ss") - 1

; vloží príkazy do batch-u
CALL %EDA_InsertValuesFromVectorRec("vector.read.1", "vector.insert.1", _bt, _et, 0, _insParams, _retCode)
CALL %EDA_InsertValuesFromVectorRec("vector.read.2", "vector.insert.2", _bt, _et, 0, _insParams, _retCode)
CALL %EDA_InsertValuesFromVectorRec("vector.read.3", "vector.insert.3", _bt, _et, 0, _insParams, _retCode)
CALL %EDA_InsertValuesFromVectorRec("vector.read.4", "vector.insert.4", _bt, _et, 0, _insParams, _retCode)

_batchParams[1]^structVersion := 1
_batchParams[1]^keepRunning := @TRUE
_batchParams[1]^infoOptions := 25 ; pošle informáciu (EDA_BATCH_INFO) po spracovaní 25%, 50%, 75% a 100%
príkazov
_batchParams[1]^commandInfo := 1 ; pošle informáciu (EDA_BATCH_COMMAND_INFO) v prípade každého neúspešného
spracovania príkazu

; spustí batch
CALL %EDA_ExecuteBatchRec(_batchId, _batchParams, _retCode)
```

```

; schema S.EDA_Batch_Test_Listener
; v tejto scheme budu odchytene a spracovane informacne spravy o spracovaní davky
RPC PROCEDURE OnEDABatchInfo (IN RECORD NOALIAS (SD.ARR_OBJECT3) _params)
    MESSAGE "Batch id = " + %IToStr(_params[1]^val1) + "; progress = " + %RToStr(100*_params[1]^val2, "###.#")
+ "%; errorCode = " + %IToStr(_params[1]^val3) ON _FROM_HIP
END OnEDABatchInfo

RPC PROCEDURE OnEDABatchCommandInfo (IN RECORD NOALIAS (SD.ARR_OBJECT3) _params)
    MESSAGE "Batch id = " + %IToStr(_params[1]^val1) + "; command idx = " + %IToStr(_params[1]^val2) + ";
errorCode = " + %IToStr(_params[1]^val3) ON _FROM_HIP
END OnEDABatchCommandInfo

BEGIN
; zaregistruje obsluhu sprav o priebehu batch-u
ON "EDA_BATCH_INFO" GOTO OnEDABatchInfo
; zaregistruje obsluhu sprav o spracovaní jednotlivých príkazov batch-u
ON "EDA_BATCH_COMMAND_INFO" GOTO OnEDABatchCommandInfo
END

```

Príkazy, ktoré podporujú dávkové spracovanie:

- [EDA_InsertValuesFromVectorRec](#)
- [EDA_InsertValuesFromCalcFuncRec](#)
- [EDA_InsertValuesToVektorRec](#)
- [EDA_DeleteIntervalFromVektorRec](#)
- [EDA_InsertValuesToPairVektorRec](#)
- [EDA_DeleteIntFromPairVektorRec](#)
- [EDA_CreateVectorRec](#)
- [EDA_UpdateVectorRec](#)
- [EDA_DeleteVectorRec](#)
- [EDA_UnDeleteVectorRec](#)
- [EDA_SetDescriptiveVectorRec](#)
- [EDA_SetIntervalVectorRec](#)
- [EDA_PredPrepocitajRec](#)
- [EDA_SetFunctionRec](#)
- [EDA_CreateVectorUserVersionRec](#)
- [EDA_DeleteVectorUserVersionRec](#)
- [EDA_UpdateVectorUserVersionRec](#)
- [EDA_UnDeleteVectorUserVersionRec](#)
- [EDA_DeleteVectorVersionHistRec](#)
- [EDA_InsertVectorVersionHistRec](#)