

Archív - ladenie a debugovanie

[Archív - ladenie výkonu](#)

[Archív - debugovanie](#)

[Archív - debugovanie pre vývojárov](#)

Archív - ladenie výkonu

Táto sekcia obsahuje tipy na ladenie výkonu archívu.

- Systémová štruktúra [SV_System_ArchivPerformance](#) je určená na monitorovanie výkonu archívu - vi [popis](#) jednotlivých stpcov. Odporúčame archiváciu stpcov PendingDbRequest, PendingStatRequest, PerformedDbRequest a PerformedCalcRequest (s vhodným filtrom) a prípadne vytvorenie archívnych štatistík nad týmito hodnotami (napr. 1-minútový a hodinový sumár).
- Tell príkaz [PENDING_REQUESTS](#) skopíruje frontu požiadaviek archívu do textového súboru. Ten je možné následne analyzovať a hľadať v ňom anomálie.
- Tell príkaz [STATISTICS](#) umožňuje zistenie množstva hodnôt, ktoré boli pre jednotlivé archívne objekty uložené do databázy za posledných N hodín. Výstupom príkazu je textový súbor, ktorý je možné následne importovať napr. do Excelu a zoradiť podľa posledného stpca. Následne je možné optimalizovať archívy s najväčším potom hodnôt (nastavenie filtrov, prípadne obmedzenie počtu výpotov pre vypočítané archívy).
- Príkaz [STATISTICS](#) nezachytí, pokiaľ bola jedna hodnota menená viackrát (napríklad vkladanie hodnôt zo skriptu). Preto existujú ladiace kategórie [DBG.ARCHIV.MANUAL_INSERT_VALUE](#) a [DBG.ARCHIV.MANUAL_UPDATE_VALUE](#) nastavené pomocou procesu [D2000 System Console](#) v okne [Debug info](#). Tieto kategórie zapínajú zobrazovanie výpisov o akciách [INSERTARCHARR](#) resp. [UPDATEARCHVAL](#) (a o runej modifikácii hodnôt z [D2000 HI](#)). Zobrazené sú ID archívnych objektov a vkladané hodnoty (v prípade akcie [INSERTARCHARR](#) prvá a posledná hodnota z vkladaneho zoznamu).
- ďalším problémom môže byť, že hodnoty opakovane vkladane zo skriptu spôsobujú veľké množstvo následných prepotov. V takom prípade je vhodné pred vkladanim zo skriptu najskôr vykonať [GETARCHVAL](#) a otestovať, či hodnota už v archíve nie je uložená. K dispozícii je aj ladiaca kategória [DBG.ARCHIV.MANUAL_READ_BEFORE_IU](#), ktorá zapína túto funkciu priamo v obsluhu akcií [INSERTARCHARR](#) resp. [UPDATEARCHVAL](#) a ladiaca kategória [DBG.ARCHIV.MANUAL_READ_BEFORE_IU.DBG](#), ktorá aktivuje výpisy, keď sa zistí, že vkladaná hodnota sa už v archíve nachádza. Pomocou týchto ladiacich výpisov je možné zistiť, pre ktoré objekty vkladane zo skriptu má zmysel implementovať íťanie pred zápisom. Výpisy majú tvar
Insert value \$XXX (NAME) Count= N First value=VALUE_DETAILS Last value=VALUE_DETAILS
Update value \$XXX Action=MANARCHINSERT VALUE_DETAILS
- Ladiaca kategória [DBG.ARCHIV.MONITOR.MV](#) zapína výpisy najstarších hodnôt, ktoré sú vkladane akciami [INSERTARCHARR](#) resp. [UPDATEARCHVAL](#). Výpisy majú tvar
ManualValues delay (Update) XXX sec for ID \$YYY: VALUE_DETAILS
resp.
ManualValues delay (Insert) XXX sec for ID \$YYY: VALUE_DETAILS
Pomocou tejto kategórie (zapnutie, vypnutie) je možné zisťovať i sa do archívu nevkladajú zo skriptu staré hodnoty, ktoré následne vyvolávajú prepoty závislých archívnych objektov.
- Ladiaca kategória [DBG.ARCHIV.MONITOR.NV](#) zapína výpisy najstarších hodnôt, ktoré prichádzajú do archívu ako hodnoty objektov, ktoré idú do primárnych archívov. Výpisy majú tvar
NewValues delay XXX sec for ID \$YYY
Pomocou tejto kategórie (zapnutie, vypnutie) je možné zisťovať i do archívu nechodia (napr. z komunikácie) hodnoty so starou časovou značkou, ktoré následne vyvolávajú prepoty závislých archívnych objektov.
Pozn: po vypnutí ladiacej kategórie sa zobrazia najdlhšie dosiahnuté oneskorenia.
- Ladiaca kategória [DBG.ARCHIV.MONITOR.OV](#) zapína výpisy najstarších hodnôt, ktoré prichádzajú do archívu ako staré hodnoty (z komunikácie, z gatewaya), ktoré idú do primárnych archívov. Výpisy majú tvar
OldValues delay XXX sec for ID \$YYY, source object \$ZZZ
Pomocou tejto kategórie (zapnutie, vypnutie) je možné zisťovať i do archívu nechodia staré hodnoty, ktoré následne vyvolávajú prepoty závislých archívnych objektov.
Pozn: po vypnutí ladiacej kategórie sa zobrazia najdlhšie dosiahnuté oneskorenia.
- Ladiaca kategória [DBG.ARCHIV.OLDVAL.COMMIT](#) súvisí s kategóriou [DBG.ARCHIV.MONITOR.OV](#). Zapína výpisy pri aplikovaných "commitoch" (OLDVALUE COMMIT), ktoré nasledujú po skončení zápisu starých hodnôt (ítaných z komunikácie, z gatewaya), ktoré idú do primárnych archívov. Výpisy majú tvar
Old value commit TransactId XXX <time_from, time_to>
Pomocou tejto kategórie je možné zisťovať, za aké obdobie do archívu chodia staré hodnoty, ktoré následne vyvolávajú prepoty závislých archívnych objektov (tiež za toto obdobie).
- Výkon archívu na platforme Linux (najmä pre vypočítané a štatistické archívy) je možné výrazne zvýšiť nasadením optimalizovanej unixODBC knižnice [libodbc.so.2.0.0](#). Navyše odporúčame nasadenie patchov archívu z októbra 2021 a novších, ktoré majú implementované recyklovanie ODBC kurzorov, ktoré rieši pomalosť knižnice unixODBC spôsobenú použitím dátových štruktúr (lineárne zoznamy) nevhodných pre aplikácie s niekoľkými desiatkami tisíc predkompilovaných kurzorov.
- Na všetkých platformách (operaných systémoch) odporúčame použiť pre potreby archivácie databázu PostgreSQL a venovať pozornosť správnejmu nastaveniu jej [parametrov](#).

- Zrýchlenie archívu pri ítaní (a teda aj pri prepotoch štatistických a výpočtaných archívov) je možné dosiahnuť zapnutím [izochrónnej cache](#).
- Zrýchlenie archívu pri zápise je možné dosiahnuť zvýšením potu zapisovacích taskov - parameter [WriteThreadsCount](#). Táto úprava vyžaduje rozšírenie licencie o [High Performance Archiv](#).
- Zrýchlenie archívu pri ítaní je možné dosiahnuť zvýšením potu ítacích taskov - parameter [ReadThreadsCount](#). Táto úprava vyžaduje rozšírenie licencie o [High Performance Archiv](#).
- Odporúčame zapnutie [asových rezov](#) aspo pre štruktúrované archívy (`DataTableSlices=2`). asové rezy zrýchlia a zjednodušia údržbu dátových tabuliek a indexov najmä pre prípad štruktúrovaných archívov.
- Parameter [ReportLongRecalc](#) môže byť použitý na detekciu prepotov intervalov za dlhšie obdobie ako je špecifikovaná hodnota (v sekundách). Takéto prepoty môžu byť dôsledkom vkladania dát, ktoré sú príliš staré (alebo existencie dát, ktoré majú as v budúcnosti).
- Tell príkaz [FREEZE](#) umožňuje zastaviť spracovanie dát v archíve na definovaný počet sekúnd a následne nahromadené požiadavky spracovať. Takto je možné uskutočniť výkonnostný test archívu (ak sa jedná o D2000 systém s redundantnými archívmi, tak ideálne na pasívnom archíve, aby pozastavenie archivácie nemalo vplyv na funkciu).

Archív - debugovanie

Táto sekcia obsahuje tipy na ladenie jednotlivých archívnych objektov.

- Tell príkaz [SHOW_DYN_INFO](#) je určený na jednorazové zobrazenie informácií o konkrétnom archívnom objekte.
- Tell príkaz [DI ON](#) je určený na aktiváciu ladiacich informácií o konkrétnom archívnom objekte. Budú sa zobrazovať informácie o vkladaní hodnôt, v prípade výpočtaných a štatistických archívov o prepotoch hodnôt a prípadne intervalov. Aktivácia ladiacej kategórie `DBG.ARCHIV.DATA` pre takéto archívne objekty zapne zobrazovanie výpočtaných hodnôt ako aj hodnôt zdrojových objektov idúcich do výpotu.
- Ladiaca kategória `DBG.ARCHIV.GET_VALUE_FOR_TIME` slúži na zapnutie výpisov pri ítaní hodnoty pre konkrétny as z databázy.
- Ladiaca kategória `DBG.ARCHIV.READ` zapína informácie o externých požiadavkách na ítanie z archívu (napr. z procesu [D2000 HI](#) alebo ako dôsledok akcií [GETARCH*](#)). Je možné tak sledovať vplyv ítania (napr. zo skriptu) na výkon D2000 Archívu.
- Ladiaca kategória `DBG.ARCHIV.READ.TREZOR` zapína informácie o požiadavkách na ítanie z trezorovej databázy. Keže trezorových databáz môže byť pripojených veľa, ítanie z trezorov môže výrazne prispieť k záťaži D2000 Archívu, SQL databázy a I/O subsystému.
- Ladiaca kategória `DBG.ARCHIV.READ.EMPTY_EXTERNAL` zapína informácie o požiadavkách na ítanie z archívu (napr. z procesu D2000 HI alebo ako dôsledok akcií [GETARCH*](#)), ktoré nevrátili žiadne dáta (tj. týkali sa prázdnych archívov). Tieto požiadavky v dôsledku optimalizácií v archíve nespôsobia ítanie z archívnej databázy.
- Ladiaca kategória `DBG.ARCHIV.READ.EMPTY_INTERNAL` zapína informácie o požiadavkách na ítanie generovaných priamo archívom (pri výpotoch výpočtaných/štatistických archívov), ktoré nevrátili žiadne dáta (tj. týkali sa prázdnych archívov). Tieto požiadavky v dôsledku optimalizácií v archíve nespôsobia ítanie z archívnej databázy.

Archív - debugovanie pre vývojárov

Nasledujúce ladiace kategórie sú určené pre vývojárov D2000.

- Ladiaca kategória `DBG.ARCHIV.CACHE` zapína ladiace informácie týkajúce sa práce s archívou cache (vkladanie, mazanie).
- Ladiaca kategória `DBG.ARCHIV.CACHE.HIT` zapína ladiace informácie týkajúce sa neúspešného ítania z archívnej cache.
- Ladiaca kategória `DBG.ARCHIV.COMPRESS.TREZOR` zapína ladiace informácie týkajúce sa [kompresie dát trezorov](#).
- Ladiaca kategória `DBG.ARCHIV.DISK_SPACE` zapína ladiace informácie týkajúce sa zisovania obsadeného miesta v archívnej databáze.
- Ladiaca kategória `DBG.ARCHIV.SLICES` zapína ladiace informácie týkajúce sa vytvárania [asových rezov](#).
- Ladiaca kategória `DBG.ARCHIV.TRANSTREE` zapína ladiace informácie týkajúce sa stromu transakcií. Vzhľadom k existencii viacerých zapisovacích taskov (vi parameter [WriteThreadsCount](#)) je nutné synchronizovať prepoty výpočtaných a štatistických archívnych objektov, aby prebiehali v správnom poradí. Na toto slúži strom transakcií.
- Ladiaca kategória `DBG.ARCHIV.WARNTRANS` zapína ladiace informácie týkajúce sa neukončených transakcií v strome transakcií. V správne fungujúcom archíve sú všetky transakcie ukončované korektné.
- Ladiace kategórie `DBG.ARCHIV.WRITECACHE` a `DBG.ARCHIV.WRITECACHE.DETAILS` zapínajú ladiace informácie týkajúce sa "write cache". Toto je krátkodobá cache určená na uchovávanie hodnôt, ktoré boli zapísané do databázy, ale neboli ešte vykonané commit (a teda nie sú ešte viditeľné inými taskami).
- Ladiaca kategória `DBG.ARCHIV.WARN_ZERO` zapína ladiace informácie týkajúce sa porovnávania hodnôt, pokiaľ sú takmer identické (vi parameter [AlmostZero](#)).