Parsování BTEQ logů

Historie dokumentu

|  |  |  |  |
| --- | --- | --- | --- |
| Verze | Datum | Autor | Poznámka |
| 1.0 | 27.01.2014 | David Šťastný |  |

Obsah dokumentu

[1. Úvod 1](#_Toc378677615)

[2. Požadavky 1](#_Toc378677616)

[2.1 Omezení rozsahu 2](#_Toc378677617)

[3. Řešení 2](#_Toc378677618)

[3.1 Technologie 2](#_Toc378677619)

[3.2 Omezení 3](#_Toc378677620)

[3.3 Spuštění 3](#_Toc378677621)

[3.4 Výstup 3](#_Toc378677622)

[3.5 Integrace 4](#_Toc378677623)

# Úvod

Tento dokument analyzuje možnosti parsování BTEQ logů pro řešení problematiky identifikace změn v tabulkách.

# Požadavky

BIS OS, BIS DI a BID potřebují snadno identifikovat změny v tabulkách v rámci BTEQ skriptů spouštěných pomocí Informatica workflow. Při spuštění BTEQ skriptu je vyroben soubor hlášení Teradata <InformaticaServer>/WorkflowLogs/<workflow>/sbteq\_<název skriptu>.log, obsahující podrobný záznam činnosti. Pro identifikaci změn v tabulkách jsou zajímavé řádky logu ve tvaru:

* \*\*\* Insert completed. 1234 rows added.
* \*\*\* Update completed. 1234 rows updated.
* \*\*\* Delete completed. 1234 rows deleted.
* \*\*\* Total time was 12 seconds

Tyto řádky obsahují informace, které poslouží jako podklad pro sledování odchylek v očekávaném počtu změn v tabulkách.

## Omezení rozsahu

POC má tyto omezení rozsahu zpracování:

* Zpracovány jsou pouze „insert into“ statementy. Ostatní typy dotazů (update a delete) budou doplněny v implementační fázi.
* Statementy, které zhavarují (tzn. mají v logu „Failure“), se nezpracovávají. Obecně se předpokládá, že do zpracování půjdou pouze ty logy, které neobsahují „Failure“
* Implicitní transakce jsou vynechány
* Explicitní transakce jsou omezeny pouze na první „insert“

# Řešení

## Technologie

Pro implementaci řešení tohoto PoC byly reálně zvažovány dvě technologie:

* Java
* PowerShell skript

Možnost nativní Windows aplikace (např. C++) byla zavržena s ohledem na uzavřenost pro další vývoj a vysoké náklady na vytvoření.

### Java

Technologie Java byla nakonec zavržena ve prospěch PowerShellu s ohledem na tyto skutečnosti:

* Java není standardní součástí serverů, kde běží Informatica
* Pro spuštění zpracování BTEQ logů se musí pokaždé znovu nastartovat proces virtuálního stroje Java, což prodlužuje dobu zpracování a zbytečně vytěžuje CPU
* Řešení v Java se musí při každé změně překompilovat do byte-kódu a dostatečně zabalit pro snadné spuštění

### PowerShell

Technologie PowerShell byla upřednostněna před Javou s ohledem na tyto skutečnosti:

* Prostředí PowerShell 2.0 je standardní součástí serverů, kde běží Informatica (Windows Server 2008 R2), není tedy třeba dalších zásahů do softwarového vybavení
* Prostředí PowerShell je implementováno jako nativní služba Windows, a tedy v době spuštění skriptu pro zpracování BTEQ logů je již zavedeno do paměti. Ke spuštění zpracování tedy dochází téměř okamžitě
* PowerShellový skript, který řeší zpracování BTEQ logů, je ve své podstatě textový soubor, který je prostředím PowerShell interpretován. Toto umožňuje velmi snadno upravit zdrojový kód v případě nalezení chyby nebo pro implementaci rozšíření

## Omezení

Omezení řešení odpovídají omezením řešení popsaných v kapitole 2.1. Navíc v rámci PoC vykazuje skript další omezení:

* Logika skriptu není zcela optimalizována, v implementační fázi je možné logiku optimalizovat a tím snížit dobu zpracování.
* Skript neošetřuje výjimečné stavy v případě chyb. Výchozím chováním v případě výskytu chyby je okamžité ukončení skriptu.

## Spuštění

Skript parseBteqLog.ps1 se spouští standardním způsobem v prostředí PowerShellu. Skript vyžaduje dva vstupní parametry. Pokud některý z parametrů chybí, skript okamžitě skončí s výpisem informace o použití.

PS> .\parseBteqLog.ps1 CPDWD\_333\\*.log CPDWD\_333.csv

První parametr určuje soubor nebo soubory BTEQ logů. Druhý parametr určuje název výstupního souboru.

### Výkon

Současná PoC verze skriptu zpracuje 150 log souborů za 2 a půl minuty na běžném kancelářském notebooku.

## Výstup

Výstupem PowerShellového skriptu je CSV soubor, jehož název je nutno zadat jako druhý parametr při spuštění skriptu.

### Formát

Formát výstupního souboru je standardní CSV, které neobsahuje hlavičku – tj. první řádek již obsahuje data – a hodnoty jsou odděleny středníkem.

Sloupce jsou následující:

* Date – datum (yyyy-mm-dd), kdy byl spuštěn skript. Toto datum se načítá přímo z logu
* ScriptName – název BTEQ skriptu, ke kterému je zprácováván log. Název se načítá přímo z logu, nikoliv z názvu souboru logu
* LineNumber – číslo řádky, na které začíná sledovaný „insert into“ statement
* StatementIndex – pořadí statementu v rámci skriptu
* DatabaseName – jméno databáze upravované cílové tabulky
* TableName – jméno upravované cílové tabulky použité ve statementu
* RowsInserted – počet přidaných řádků pro „insert“ statement
* RowsUpdated – počet aktualizovaných řádků pro „update“ statement – *zatím neimplementováno*
* RowsDeleted – počet smazaných řádků pro „delete“ statement – *zatím neimplementováno*
* TimeInSeconds – doba trvání statementu v sekundách

## Integrace

Skript pro parsování BTEQ logů je snadno integrovatelný do command tasku v Informatica workflow. Tato integrace již není v rámci tohoto POC.