

SQL Server sistemsko okruženje

U ovom poglavlju opisano je nekoliko različitih osobina SQL Servera za sistemsko okruženje. Pre svega je dat detaljan opis SQL Server elemenata skladišta na disku, sistemskih baza podataka i pogodnosti. Posle toga su objašnjeni pojmovi Unicode i podrška nacionalnim jezicima. Na kraju je o SQL Server sistemu diskutovano i ilustrovana je arhitektura.

Skladište na disku

Arhitektura memorije SQL Servera sadrži dve jedinice za skladištenje objekata baze podataka:

- **stranicu**
- **proširenje.**

Glavna jedinica za memorisanje podataka u SQL Serveru je stranica. Veličina stranice u SQL Serveru 7 je 8k, što se razlikuje od prethodne verzije, kod koje je veličina iznosila 2K. Svaka stranica ima zaglavlje od 96 bajtova, koje se koristi za smeštaj sistemskih informacija. Odmah iza zaglavlja slede nizovi podataka. U SQL Serveru niz ne može bude dugačak dve, ili više stranica. Zato je maksimalna veličina pojedinačnog niza (izuzimajući TEXT, NTEXT i slikovne podatke) 8.060 bajtova.

SQL Server poseduje šest različitih tipova stranica:

- **stranice podataka**
- **stranice indeksa**
- **text/slika stranice**
- **stranice globalne alokacione mape**
- **stranice slobodnog prostora**
- **stranice indeksne alokacione mape.**

Kada kreirate tabelu, ili index SQL Server odvaja fiksni prostor za smeštaj podataka koji pripadaju tabeli, ili indeksu. Kada se taj prostor popuni SQL Server mora da odvoji prostor za dodatno memorisanje. Fizički deo memorije koji se odvaja za tabelu (indeks) naziva se proširenje. Jedno proširenje obuhvata osam kontinuelnih stranica, ili 64k. SQL Server ima dve vrste proširenja:

- **uniformno**
- **mešano.**

Uniformno proširenje pripada samo pojedinačnoj tabeli, ili indeksu, dok mešano deli do osam tabela, ili indeksa. Nova tabela, ili indeks se uvek prvo dodeljuju stranicama iz mešanog proširenja. Nakon toga, ako je veličina tabele (indeksa) veća od osam stranica, prebacuje se u uniformno proširenje.

Sistemske baze podataka

Tokom instalacije SQL Servera generišu se četiri sistemske baze podataka:

- **master**
- **model**
- **tempdb**
- **msdb.**

Master baza podataka je najvažnija u SQL Server sistemu. Ona obuhvata sve sistemske tabele koje su neophodne za rad sa sistemom baza podataka. Na primer, **master** baza podataka sadrži informaciju o svim ostalim bazama kojima manipuliše SQL Server, sistemskim vezama sa klijentima, i korisničkim autorizacijama.

Model baza podataka se koristi kao šablon kada se kreira korisnički definisana baza podataka. Ona sadrži podskup svih sistemskih tabela **master** baze, koje su potrebne svakoj od korisnički definisanih baza. Administrator sistema može da promeni osobine **modela** baze podataka, zbog prilagođavanja potrebama svom sistemu.

Baza podataka **tempdb** obezbeđuje memorijski prostor za privremene tabele i ostale privremene potrebne objekte. Na primer, SQL Server smešta međurezultate izračunavanja svakog kompleksnog izraza u **tempdb** bazu podataka. Bazu **Tempdb** koriste sve baze koje pripadaju celom sistemu. Njihov sadržaj se briše svaki put kada se završava korisnički proces, ili se sistem stopira.

Informacije koje dozvoljavaju SQL Server agentu da funkcioniše smeštene su u **msdb** bazi podataka. To znači da se **msdb** baza koristi za čuvanje upozorenja, zadataka i za operatore snimanja (za detaljni opis **msdb** baze o SQL Server agentu pogledajte Poglavlje 22).

NAPOMENA

Postoji druga sistemska baza podataka, nazvana distribuciona baza podataka, koja se instalira na distribucionom serveru u slučaju kada se podaci repliciraju (za više obaveštenja o distribucionom serveru i povraćaju podataka videti Poglavlje 25).

Pogodnosti

Pogodnosti su komponente SQL Servera koje omogućavaju različite osobine, kao što su verodostojnost podataka, definicija podataka i statistički podržavajuće funkcije. Sve SQL Server pogodnosti poseduju dva glavna svojstva:

- **Pozivaju se korišćenjem komandi** operativnog sistema.
- **Svaka pogodnost ima više opcionih parametara.**

U ovom poglavlju opisane su sledeće pogodnosti:

- **bcp**
- **isql**
- **osql**
- **console.**

Pogodnost bcp

Pogodnost **bcp** (BulkCoPy) kopira sadržaje tabele, ili pogleda u dokument operativnog sistema. Takođe, može da se koristi za učitavanje podataka iz dokumenta operativnog sistema u tabelu. Dalje, mogu da se specificiraju različiti formati za učitavanje, ili iščitavanje dokumenata. Ova pogodnost je opisana detaljno u Poglavlju 21, zato što su učitavanje i čitanje podataka, obično, zadaci administratora sistema.

Pogodnost isql

Pogodnost **isql** je jedan Transact-SQL interfejs za SQL Server. Koristi DB-biblioteku za komunikaciju sa serverom. Generalna forma ove pogodnosti je

```
Isql {option [parameter] }...
```

Option je specifična opcija pogodnosti, dok parameter određuje vrednost definisane opcije. Pogodnost **isql** ima dosta opcija. Najvažnija je opisana u tabeli 15.1.

PRIMER 15.1

```
Isql -SNTB11900-U sa -ims0510.sql -O ms0520.rpt
```

U primeru 15.1 administrator SQL Server sistema, nazvanog NTB11900, izvršava grupu koja je smeštena u dokument **ms0510.sql** i memoriše rezultate u izlazni dokument **ms0510.rpt**. Sistem zahteva lozinku od administratora sistema, jer je opcija **-P** izostavljena.

Pogodnost osql

Pogodnost **osql** je drugi Transact-SQL interfejs az SQL Server, sličan isqlu. Jedina razlika je što **osql** pogodnost koristi ODBC za komunikaciju sa serverom. **Osql** pogodnost opcije su iste kao i one za **isql** pogodnost.

Pogodnost console

Pogodnost **console** prikazuje poruke za proces za backup i smešta u bazu podataka. Ona mora da se aktivira pre izvršenja DUMP, ili LOAD iskaza (Za definiciju DUMP i LOAD iskaza pogledati Poglavlje 20). Administrator sistema i dump operatori mogu da koriste ovu pogodnost. Sintaksa je

```
Console [/S server_name] [/P pipe_name]
```

server_name određuje server računar u mreži (ova opcija se zahteva kada se startovanje izvršenja console komande vrši preko udaljenog računara). pipe_name je protok koji se koristi za startovanje servera (console je pogodnost koju koriste Named protoci).

Tabela 15.1: Korisne opcije isql pogodnosti

Opcija	Opis
-S server_name	Specificira se ime SQL Servera sa kojim se ostvaruje veza. Ako se ova opcija izostavi, uspostavlja se veza sa serverom baze podataka postavljenim pomoću promenljive okruženja isqlserver . Ako ova promenljiva okruženja nije postavljena, SQL Server će pokušati da se veže za lokalnu mašinu.
-U login_id	Specificira SQL Server login. Login se mora kreirati korišćenjem sistemske procedure sp_addlogin . Ako je ova opcija izostavljena, koristi se vrednost promenljive okruženja isqluser .
-P password	Specificira lozinku koja odgovara SQL server loginu. Ako se ova opcija ne navede, koristi se vrednost promenljive okruženja isqlpassword . Ako promenljiva nije postavljena, pogodnost će zatražiti lozinku.
-c command_end	Specificira se komandni terminator (podrazumevana vrednost je go). Ova opcija može da se koristi za postavljanje komandnog terminatora u “;”, što je podrazumevani terminator za skoro sve ostale sisteme baza podataka.
-i input_file	Specificira ime dokumenta, koji sadrži grupu, ili memorisanu proceduru. Dokument mora da sadrži najmanje jedan komandni terminator. Može da se koristi znak <, umesto -i.
-o output_file	Specificira ime dokumenta, koji prihvata rezultate od uslužnog programa. Znak > može da se koristi umesto -o.

Unicode

Pošto se SQL Server koristi širom sveta, on treba da procesira specifične podatke bilo kog jezika (ili države). To znači da korisnici van SAD mogu da koriste znakove sopstvenog sistema pisanja. Oni očekuju da SQL Server podržava ne samo osnovne poslove, kao što su izdanja na maternjem jeziku, već i da radi na istom nivou. Ceo proces podrške nacionalnim jezicima se naziva *internacionalizacija* softvera.

Internacionalizacija SQL Servera utiče na sledeće komponente sistema:

- **selekciju skupa karaktera**
- **selekciju kolacione sekvencije**
- **instalaciju nacionalno-specifičnih poruka o greškama**
- **specifikacije datuma i novčanih formata.**

Kodiranje karaktera

Mešavina standarda diktira kako kodirati karaktere. Neki standardi su sedmobitni i nazivaju se jednobajtni, a ostali su 8-bitni i zovu se multibajtni skupovi znakova. Prethodne verzije SQL Servera koriste Code Page Model (Model kodnih stranica); SQL Server 7 koristi Unicode.

Model kodnih stanica

Microsoftovi operativni sistemi su ranije, mahom, podržavali engleski i nekoliko evropskih jezika. Sve kodne stranice su tada bile zasnovane na 256 znakova (svaki je predstavljan jednobajtnom numeričkom vrednošću). Bili su dostupni znakovi interpunkcije, dvadeset i šest slova engleskog alfabeta (i velika i mala), grčka slova, znakovi za crtanje linija i veznici (veznik je kombinacija dva, ili više znakova, koja se koristi za predstavljanje pojedinačnog tipografičkog znaka). Svaka kodna stranica se sastoji iz dva dela. ASCII znakovi sa dužinom koda od 0 do 127 i prošireni znakovi sa dužinom koda od 128 do bilo kog broja (slovo se naziva High-ASCII dužina koda). High-ASCII dužina koda uključuje istaknute znakove i različite simbole u skupu znakova sa Zapada i ideografe u dalekoistočnoj kodnoj stranici (ideograf je znak kineskog porekla, koji označava reč, ili slog i uopšteno se koristi u više azijskih jezika). Svaka država pridružuje svoju kodnu stranicu sa svojim domaćim znakovima u proširenom kodnom obliku.

Kodiranje znakova korišćenjem Unicodea

Unicode je 16-bitni metod za kodiranje, pozajmljen od Microsoftovog Windowsa NT. Ovo kodiranje znakova je razvio, ustanovio i promovisao Unicode Consortium. Unicode sadrži skoro sve današnje znakove koji se koriste u radu sa računarima. Ovo uključuje većinu sve tskih pisama, izdavačkih znakova i matematičkih i tehničkih simbola. Oko 35.000 kodnih tačaka je već iskorišćeno za obeležavanje znakova. Kao dodatak modernim jezicima, Unicode pokriva i klasični grčki i hebrejski jezik. Za korisnički definisane simbole na raspolaganju stoje 6.500 kodnih tačaka. Za razliku od prethodnih kodnih stranica, koje nisu bile projektovane vodeći računa o ekspanziji, Unicode je uniformni metod kodiranja znakova.

To znači da je sa njim moguće procesirati znakove različitih sistema pisanja u jednom dokumentu. Zbog toga, on eliminiše potrebu za kodiranjem nepristupačnih mnogostrukih kodnih stranica.

Unicode standard direktno adresira samo kodirani i interpretirani tekst, ali ne i sve ostale akcije preduzete nad tekstem. To znači da Unicode standard ne opisuje kako se identifikovane kodne vrednosti tumače na ekranu, ili papiru (softverska, ili hardverska mašina za prevođenje u računar su odgovorne za pojavljivanje znakova na ekranu, ili papiru).

Unicode tipovi podataka i funkcije

SQL Server podržava tri Unicode tipa podataka:

- **NCHAR**
- **NVARCHAR**
- **NTEXT**

NCHAR tip podataka skladišti Unicode karaktere fiksirane dužine. Valja ga koristiti za definisanje alfanumeričkih kolona, gde su skoro sve vrednosti podataka približno iste dužine. NVARCHAR tip podataka skladišti Unicode karaktere različite dužine. Treba ga koristiti za definisanje alfanumeričkih kolona u kojima dužina podataka znatno varira. NTEXT tip podataka skladišti velike karakter podatke (sa više od 4.000 karaktera).

Postoje i dve funkcije povezane sa Unicode:

- **UNICODE** ('char_expresion')
- **NCHAR** (int_expresion)

Funkcija Unicode vraća celobrojnu vrednost za prvi karakter Unicode niza karaktera **char_expresion**. NCHAR vraća Unicode karakter za celobrojnu vrednost **int_expresion**.

Primer 15.2 vraća Unicode karakter L zajedno sa njegovom celobrojnomo vrednošću.

PRIMER 15.2

```
DECLARE @var1 NVCHAR (40)
SET @var1 = N'Latin character'
SELECT UNICODE (@var1)'Integer value', NCHAR(UNICODE(@var1))'Unicode
character'
```

Rezultat je:

Integer value

73

Unicode character

L

Arhitektura SQL Servera

SQL Server 7 je visokoskalabilan sistem baza podataka. To znači da se ista mašina baze podataka može instalirati i koristiti na isti način na različitim računarima, od mobilnih laptopova, koji rade pod Windows 95/98, do SMP računara pod Windows NT Server enterprise Edition. Niski memorijski zahtevi SQL Servera 7 dozvoljavaju da se on koristi na laptopovima i pod Windows 95 operativnim sistemom. Sa druge strane, upotrebom SQL Server sistema sa SMP računarima koriste se prednosti njihove višenitne arhitekture.

Pre objašnjenja arhitekture SQL Servera pregledajmo postojeće multiprocesorske hardverske arhitekture.

Multiprocesorske hardverske arhitekture

Postoje tri pristupa arhitekturi multiprocesor računara:

- **simetrični multiprocesor (SMP)**
- **klasteri**
- **masivno paralelni procesori (MPP).**

Simetrično multiprocesiranje određuje sistem u kome veći broj procesora deli glavnu memoriju i disk (ovakvi sistemi se, takođe, nazivaju sistemi sa deljivim diskom, zato što oni dele jedan operativni sistem koji upravlja sa jednim, ili sa više diskova). SMP sistemi su, obično, povezani tako da su omogućene visoke brzine prenosa za sve procesore. Prednosti SMP sistema su:

- **jednostavna administracija**
- **malo reprogramiranje DBMS.**

SMP sistem je jednostavan za administraciju, zato što izgleda kao da jedan računar upravlja operativnim sistemom. Sa druge strane, ako su SMP sistemi skalirani na više CPU-ova, postaju komplikovaniji za administraciju.

Uopšteno govoreći, DBMS ne zahtevaju reprogramiranje da bi se koristili sa SMP sistemom, ako se različiti procesi jednostavno dodeljuju različitim CPU-ovima. Međutim, DBMS je reprogramiran da bi podržao multinitnu arhitekturu, gde su procesi podeljeni na niti, što pomaže DBMS-u da iskoristi prednosti SMP sistema (definiciju niti data je u sledećem odeljku).

Mana SMP sistema je ograničena skalabilnost. To znači da je broj procesora koji mogu da se međusobno sinhronizuju ograničen veličinom glavne memorije.

Klaster, obično, određuje grupu SMP računara koja je povezana u mrežu. Njegove prednosti su bolja skalabilnost i pouzdanost (klaster može da toleriše propuste jednog računara, zato što ostali računari u klasteru mogu da preuzmu procesiranje).

MPP sistemi nude najveću skalabilnost. Oni, kao i klasteri, specificiraju većinu računara povezanih pomoću mreže. Ovi sistemi se, takođe, zovu nepodeljeni sistemi, zato što svaki procesor ima svoju glavnu memoriju, sopstveni operativni sistem i sopstveni DBMS. Iako MPP sistemi nude najveću skalabilnost, oni su nepogodni za administraciju.

NAPOMENA

U vreme pisanja ovog teksta SQL Server je podržavao SMP sisteme i NT 2 Machine klastere. Uprkos tome, SQL Server sa SMP hardverom odgovara bazama podataka skoro svih veličina, zato što se SMP implementacije stalno poboljšavaju. To znači da broj procesora i snaga svakog od njih rastu. Takođe raste brzina memorije, a poboljšava se upravljanje busom. Zbog svih ovih faktora, SQL Server može da upravlja najvećom bazom podataka koja postoji.

Višenitna arhitektura SQL Servera

SQL Server podržava SMP sisteme, koristeći dva specijalna arhitekturna poboljšanja:

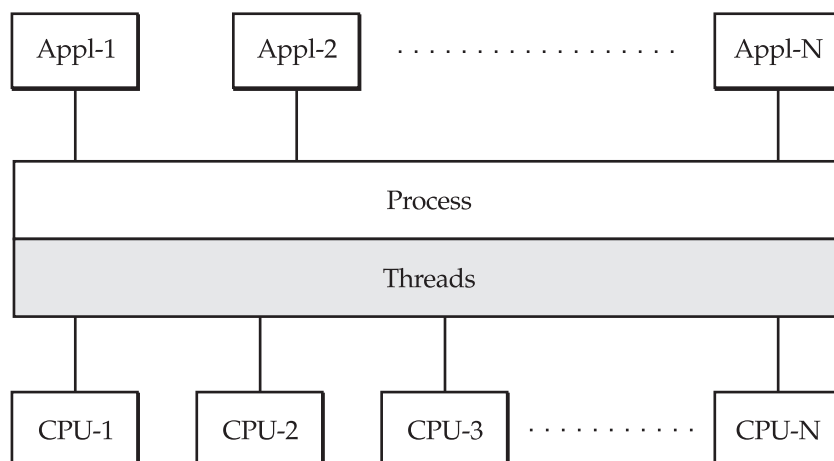
- **N:1 arhitektura**
- **domaće višenitno procesiranje**

N:1 arhitektura definiše arhitekturu DBMS, gde se N korisničkih aplikacija izvršava korišćenjem pojedinačnog procesa (vidite sliku 15.1). Kao dodatak N:1 arhitekturi, postoji i 1:1 arhitektura, gde svaka korisnička aplikacija startuje jedan proces. Nepogodnost drugog modela je u tome da kad broj korisnika raste performanse sistema opadaju, zato što neke tačke u glavnoj memoriji ne mogu da upravljaju svim postojećim procesima (N:1 arhitektura se, takođe, zove jednoprocena arhitektura).

SQL Server podržava SMP sisteme korišćenjem niti pre nego procesa. Niti su delovi procesa koji se zasebno izvršavaju. Zato se one, ponekad, nazivaju laki procesi. Sa njima je moguće da DBMS radi efikasno na više aplikacija baza podataka korišćenjem samo nekoliko procesora. Prebacivanje sa jedne niti na nit drugog procesora može da se izvrši mnogo brže nego prebacivanje sa jednog na drugi proces.

NAPOMENA

Domaće višenitno procesiranje znači da SQL Server podržava multiprocesiranje korišćenjem Windows NT niti. Korišćenje niti operativnog sistema, raspoređivanje zadataka u SQL Serveru je preventivan proces, a to dozvoljava balansirani dinamički unos kroz više CPU-ova.

**Slika 15.1**

Višenitna jednoprocena arhitektura SQL Servera

SMP računari se, zajedno sa SQL Server multinitnom arhitekturom, koriste za paralelno izvršavanje različitih zadataka baze podataka. Sledeći zadaci mogu da se paralelizuju korišćenjem SQL Servera 7:

- unos podataka
- backup i oporavak
- brzo izvršenje upita.

SQL Server omogućava paralelni unos podataka korišćenjem bcp pogodnosti. Tabela u koju se podaci unose ne sme da poseduje ni jedan indeks, a operacija unosa ne sme da bude logovana (više informacija o bcp pogodnosti naći ćete u Poglavlju 20).

NAPOMENA

Samo aplikacije koje koriste API-e bazirane na ODBC, ili SQLOLEDB mogu izvesti paralelan unos podataka u jednu tabelu.

SQL Server može da izvrši backup baza podataka, ili transakcionih logova na više uređaja (traka, ili disk) korišćenjem paralelnih backupova. U ovom slučaju više niti jednog proširenja čitaju stranice baze podataka istovremeno.

SQL Server omogućava paralelne upite, da bi poboljšao njihovo izvršavanje. Nezavisni delovi SELECT iskaza mogu biti izvršeni korišćenjem više domaćih niti na SMP računaru. Svaki upit koji je planiran za paralelno izvršenje sadrži operator izmene u svom planu izvršenja (operator izmene je operator u planu izvršenja upita koji omogućava upravljanje procesom, redistribucijom podataka i kontrolom toka). Za ovakav upit SQL Server generiše paralelni plan izvršenja upita. Paralelni upiti značajno poboljšavaju performanse SELECT iskaza, koji obrađuje velike količine podataka.

Zaključak

SQL Server je napredni relacioni DBMS koji je superioran u odnosu na druge DBMS-ove u dve oblasti:

- u **internalizaciji softvera baze podataka**
- u **arhitekturi sistema baze podataka**

SQL Server je jedan od prvih sistema baze podataka koji koristi Unicode kao budući standard za internalizaciju softvera. Korišćenjem Unicodea moguće je obraditi karaktere različitih sistema pisanja u jednom dokumentu, eliminišući potrebu za kodom za višekodne stranice. SQL Server poseduje tehnologiju za upotrebu multiprocesor sistema, kao što su SMP računari. Ova tehnologija je bazirana na jednoprocesnoj arhitekturi i domaćem multinitnom procesiranju.

Ovim poglavljem završen je drugi deo knjige. Sledeći deo posvećen je zadacima systemske administracije SQL Servera. U Poglavlju 16 obradićemo opšte zadatke administratora sistema i administrativnih alata SQL Servera.