CAPITOLO 4

Configurazione e ottimizzazione dei Server SQL

- Accesso ai dati di configurazione di SQL Server **112**
- Tecniche di gestione delle opzioni di configurazione di SQL Server **114**
- Configurazione di SQL Server con le stored procedure **123**

Microsoft SQL Server 2012 è stato progettato per adattarsi automaticamente al carico di lavoro e ottimizzare le impostazioni di configurazione. Per esempio, SQL Server può aumentare o ridurre l'utilizzo della memoria in base alla richiesta generale di memoria di sistema. SQL Server gestisce inoltre l'efficienza della memoria, specialmente quando si tratta di query e connessioni utente; ma la memoria è solo una delle molte aree in cui la configurazione viene modificata automaticamente.

Benché la funzionalità di ottimizzazione automatica di SQL Server funzioni bene, a volte è comunque necessario configurare le impostazioni di SQL Server manualmente.

Per esempio, se sta girando un database di grandi dimensioni con vincoli particolari e il database non funziona come dovrebbe, si potrebbe voler personalizzare la configurazione. Potrebbe inoltre essere necessario modificare le impostazioni di configurazione per account SQL Server, autenticazione e controllo.

Gli strumenti chiave che si utilizzano per configurare e ottimizzare SQL Server sono:

- **Query del catalogo di sistema** Offre un modo diretto per stabilire le caratteristiche di configurazione del database e le relative impostazioni.
- **Stored procedure** Consentono di visualizzare e gestire le impostazioni di configurazione attraverso le stored procedure, per esempio sp_configure.
- **SQL Server Management Studio** Offre un'interfaccia di facile utilizzo per aggiornare il database e le impostazioni del Registro di sistema.
- **SQLServr.exe** Avvia SQL Server dalla riga di comando. Può essere utilizzato per impostare i parametri di configurazione all'avvio

In questo capitolo sono descritte le strutture disponibili per configurare e ottimizzare SQL Server. Per prima cosa si analizza il catalogo di sistema di SQL Server 2012, poi si procede con una discussione sulle query del catalogo e le stored procedure. Qui viene fornito il background essenziale per comprendere come configurare e ottimizzare SQL Server 2012.

Accesso ai dati di configurazione di SQL Server

SQL Server 2012 utilizza un approccio basato sugli oggetti per rappresentare server, database e tutte le altre caratteristiche di configurazione e i contenuti dei dati. Il fulcro di questa struttura è il catalogo di sistema, che descrive gli oggetti in un'istanza particolare di SQL Server, oltre ai relativi attributi. Per esempio, gli attributi di un database descrivono:

- Il numero e i nomi di tabelle e viste
- Il numero e i nomi delle colonne di una tabella o vista
- Il tipo, la scala e la precisione dei dati della colonna
- I trigger e i vincoli definiti in una tabella
- Le chiavi e gli indici definiti in una tabella
- Le statistiche utilizzate dall'ottimizzatore query per generare i piani delle query.

Nelle query è possibile accedere a queste e ad altre informazioni del catalogo di sistema utilizzando

- **Viste del catalogo** Consente l'accesso ai metadati archiviati in un database, che include gli attributi del database e i relativi valori. Le viste del catalogo possono essere utilizzate per accedere a tutti i metadati disponibili per l'utente, tranne quelli di replica, backup, piani di manutenzione del database e SQL Server Agent.
- **Viste di compatibilità** Consentono l'accesso a molte delle tabelle di sistema di versioni precedenti di SQL Server. Queste viste servono solo per la compatibilità con le versioni precedenti, in quanto mostrano gli stessi metadati disponibili in SQL Server 2000 Non mostrano i metadati delle nuove funzionalità per esempio il partizionamento e il mirroring del database.
- **Viste dello schema delle informazioni** Consente l'accesso a un sottoinsieme di metadati archiviati in un database, che include gli attributi del database e i relativi valori. Tali viste sono basate su definizioni di viste del catalogo secondo lo standard SQL-92 e non contengono metadati specifici su SQL Server 2012. Le applicazioni che utilizzano queste viste possono essere trasferite tra sistemi eterogenei di database compatibili con SQL-92.
- **Funzioni ODBC del catalogo** Offre un'interfaccia utilizzabile dai driver Open Database Connectivity (ODBC) per restituire set di risultati contenenti informazioni del catalogo di sistema. I set di risultati presentano le informazioni del catalogo in un modo indipendente dalla struttura delle tabelle di catalogo sottostanti.

- **Set di righe dello schema OLE DB** Fornisce un'interfaccia IDBSchemaRowset che i provider OLE DB possono utilizzare per accedere alle informazioni del catalogo di sistema. Il set di righe presenta le informazioni del catalogo in modo indipendente rispetto alla struttura delle tabelle di catalogo sottostanti.
- **Stored procedure e funzioni di sistema** Forniscono stored procedure e funzioni Transact-SQL (T-SQL) che restituiscono informazioni del catalogo.

Questo principalmente perché le viste presentano i metadati in un formato indipendente rispetto a qualsiasi implementazione di tabella di catalogo, quindi le viste non vengono influenzate dalle modifiche apportate nelle tabelle di catalogo sottostanti. Quando si desidera configurare o gestire un server, in genere si utlizzano le stored procedure per semplificare l'esecuzione delle attività necessarie. Le stored procedure forniscono la funzionalità per visualizzare e gestire con facilità la configurazione di SQL Server e i relativi database.

Le viste del catalogo contengono informazioni utilizzate da Motore di database di SQL Server 2012. Forniscono l'interfaccia più generica per i metadati del catalogo e rappresentano il modo più diretto per accedere a tali informazioni e per utilizzarle. Tutti i metadati a disposizione degli utenti nel catalogo di sitema vengono mostrati attraverso le viste del catalogo. Tali viste non contengono informazioni relative a replica, backup, piani di manutenzione del database o SQL Server Agent.

Come tutte le strutture nei database SQL Server 2012, le viste del catalogo seguono una gerarchia basata sugli oggetti in cui gli oggetti di livello inferiore ereditano gli attributi di quelli di livello superiore. Alcune viste del catalogo ereditano le righe da altre viste. Per esempio la vista del catalogo relativa alle tabelle eredita tutte le colonne della vista relativa agli oggetti. Quindi, oltre alle colonne specifiche della vista relativa alle tabelle, questa vista include tutte le colonne della vista relativa agli oggetti.

Le stored procedure di sistema possono essere utilizzate per visualizzare i dettagli della configurazione di SQL Server e per eseguire attività di amministrazione generiche. SQL Server 2012 dispone di due categorie principali di stored procedure di sistema:

- Quelle destinate agli amministratori
- Quelle utilizzate per implementare funzionalità per le API.

Naturalmente, tratteremo delle stored procedure di sistema destinate agli amministratori. Le stored procedure di sistema sono scritte utilizzando Transact-SQL (T-SQL). La maggior parte restituisce un valore pari a 0 per indicare la riuscita e un valore diverso da zero per indicare un errore. Per esempio, sp_dboption è una stored procedure per la gestione delle opzioni di configurazione dei database SQL Server (tranne i database *master* e *tempdb*). Quando si usa sp_dboption per impostare un valore di configurazione del database, se viene restituito un codice pari a 0 significa che l'opzione è stata impostata come previsto. Se viene restituito un codice pari a 1 significa che si è verificato un errore e l'opzione non è stata impostata come previsto.

Nell'esempio seguente il database Personnel viene tenuto non in linea, se non sono presenti utenti:

T-SQL

```
USE master;
GO
EXEC sp_dboption "Personnel", "offline", "TRUE";
GO
```

PowerShell

```
Invoke-Sqlcmd -Query "USE master; EXEC sp_dboption 'Personnel',
'offline', 'TRUE';" -ServerInstance "CorpServer17\DataServices"
```

Se la stored procedure restituisce 0, il database è stato messo correttamente non in linea. Se viene restituito un valore pari a 1 significa che si è verificato un problema, quindi il database è ancora in linea. Per ulteriori informazioni sull'utilizzo delle stored procedure, consultare il paragrafo "Configurazione di SQL Server con le stored procedure" più avanti nel capitolo. Notare che è possibile modificare alcune opzioni di sp_configure solo quando show advanced options è impostato su 1, come negli esempi seguenti:

T-SQL

```
exec sp_configure "show advanced options", 1
```

PowerShell

```
Invoke-Sqlcmd -Query "exec sp_configure 'show advanced options', 1"
-ServerInstance "DbServer18\OrderSystem"
```

NOTA **Quando si usa Invoke-Sqlcmd, bisogna specificare l'istanza del Motore di database tramite il parametro –ServerInstance nel formato –ServerInstance "*NomeServer*" per l'istanza predefinita ovvero –ServerInstance "*NomeServer\NomeIstanza*" per un'istanza personalizzata. Per maggiori informazioni sull'utilizzo di Invoke-Sqlcmd, fare riferimento al Capitolo 1, "Gestione dei Server SQL".**

Tecniche di gestione delle opzioni di configurazione di SQL Server

Le opzioni di configurazione possono essere considerate un insieme di regole che definiscono il modo in cui viene configurato e utilizzato SQL Server. Le singole istanze server possono avere diverse configurazioni, così come i database che supportano, le connessioni fatte dalle applicazioni e ogni istruzione o programma batch eseguito.

Impostazione delle opzioni di configurazione

Le opzioni di configurazione possono essere impostate per

- **Un'istanza server specifica** Le opzioni server vengono definite anche opzioni a livello di istanza e vengono impostate tramite la stored procedure sp_configure.

- **Un database specifico** Le opzioni di database vengono definite anche opzioni a livello di database e vengono impostate eseguendo l'istruzione ALTER DATABASE. Il livello di compatibilità del database può essere impostato eseguendo la stored procedure sp_dbcmptlevel.
- **Una connessione specifica** Le opzioni di connessione vengono impostate tramite Microsoft OLE DB Provider per SQL Server o le proprietà driver di SQL Server ODBC e tramite le opzioni ANSI SET quando la connessione viene stabilita.
- **Un'istruzione specifica o un batch di istruzioni** Le opzioni a livello di batch vengono specificate tramite le istruzioni SET. Le opzioni a livello di istruzione vengono specificate in singole istruzioni Transact-SQL e hanno effetto sulle rimanenti istruzioni all'interno del batch.

Ognuna di queste aree di configurazione può essere considerata un livello nella gerarchia di configurazione di SQL Server. Quando un'opzione è supportata in più di un livello, l'impostazione applicabile viene stabilita in base al seguente ordine di precedenza:

1. Opzione server
2. Opzione database
3. Opzione di connessione (ANSI SET) o batch (SET)
4. Opzione per istruzione specifica (HINT)

NOTA **La stored procedure sp_configure fornisce l'opzione *user options*, che consente di modificare i valori predefiniti di molte opzioni SET. Benché *user options* sembri un'opzione di istanza, è un'opzione SET. Nelle versioni precedenti di SQL Server, le opzioni a livello di batch venivano chiamate *opzioni a livello di connessione*. Quando si disabilita Multiple Active Result Sets (MARS), le opzioni a livello di batch vengono considerate anche opzioni a livello di connessione.**

ALTER DATABASE si utilizza per modificare le impostazioni di un database, sp_configure per modificare le impostazioni a livello di server, mentre l'istruzione SET per modificare le impostazioni che influiscono solo sulla sessione attuale. In caso di conflitti tra le opzioni di configurazione, le opzioni applicate in un momento successivo hanno la precedenza su quelle già impostate. Per esempio, le opzioni di connessione hanno la precedenza su quelle del database o del server.

Utilizzo delle opzioni SET

In genere, le opzioni SET vengono configurate dagli utenti all'interno di un batch o uno script e vengono applicate fino a quando non vengono reimpostate o fino a quando non termina la sessione utente con il server. Le opzioni SET possono inoltre essere configurate all'interno di una stored procedure o di un trigger. In tal caso, le opzioni SET vengono applicate fino a quando non vengono reimpostate all'interno della stored procedure o del trigger, oppure fino a quando il controllo restituisce il codice che ha richiamato la stored procedure o il trigger.

Le opzioni SET vengono applicate in fase di parsing (analisi) o di esecuzione. Le opzioni della fase di analisi sono QUOTED_IDENTIFIER, PARSEONLY, OFFSETS e FIPS_FLAGGER. Tutte le altre opzioni SET appartengono alla fase di esecuzione. Le opzioni della fase di analisi vengono applicate durante l'analisi man mano che vengono incontrate. Quelle della fase di esecuzione vengono applicate durante l'esecuzione del codice in cui sono specificate.

Le istruzioni batch vengono analizzate per intero prima dell'esecuzione. Ciò significa che le istruzioni del flusso di controllo non influiscono sulle impostazioni della fase di analisi. Per contro, sia il flusso di controllo che l'esecuzione influiscono sull'impostazione delle opzioni della fase di esecuzione. Le opzioni di questa fase vengono impostate solo se il controllo viene modificato in una sezione del batch che contiene le opzioni della fase di esecuzione e le relative istruzioni vengono eseguite senza errori. Se l'esecuzione non riesce prima dell'impostazione di un'opzione della fase di esecuzione o durante l'elaborazione dell'istruzione che imposta l'opzione, tale opzione non viene impostata.

Quando un utente si connette a un database, alcune opzioni potrebbero essere impostate su ON automaticamente. Tali opzioni possono essere impostate tramite le opzioni utente, le opzioni server o le proprietà di connessione ODBC e OLE DB. Se l'utente modifica le opzioni SET all'interno di un batch SQL dinamico o uno script, tali modifiche vengono applicate solo per la durata di tale batch o script.

NOTA **Le connessioni attivate tramite MARS hanno un elenco dei valori predefiniti delle opzioni SET. Quando in tale connessione viene eseguito un batch o uno script, i valori predefiniti delle opzioni SET vengono copiati nell'ambiente della richiesta attuale. Tali valori restano validi a meno che vengano reimpostati all'interno della connessione. Quando il batch o lo script termina, l'ambiente di esecuzione viene ricopiato in quello predefinito della sessione. In questo modo si assicura che nell'esecuzione simultanea di più batch all'interno della stessa connessione venga eseguito un ambiente isolato di opzioni SET. Tuttavia, dal momento che l'ambiente di esecuzione viene ricopiato in quello predefinito della sessione quando viene completata l'esecuzione del batch o dello script, l'ambiente predefinito attuale della connessione dipende dall'ultimo batch o script che completa l'esecuzione.**

Nella tabella 4-1 vengono elencate le opzioni SET di batch/connessione disponibili, oltre alle corrispondenti opzioni database e server supportate in SQL Server 2012 e alle impostazioni predefinite (dove possibile). Le voci contrassegnate da un asterisco fornire delle informazioni di diagnostica aggiuntive (e non sono usate per la configurazione). L'istruzione SET ANSI_DEFAULTS viene fornita come collegamento per l'impostazione delle opzioni standard SQL-92 ai valori predefiniti. Le opzioni ripristinate quando viene utilizzata questa istruzione sono le seguenti: SET ANSI_NULLS, SET CURSOR_CLOSE_ON_COMMIT, SET ANSI_NULL_DFLT_ON, SET IMPLICIT_TRANSACTIONS, SET ANSI_PADDING, SET QUOTED_IDENTIFIER e SET ANSI_WARNINGS.

NOTA **Nelle prossime versioni di SQL Server, SET OFFSETS non sarà più disponibile. Inoltre, ANSI_NULLS, ANSI_PADDING e CONCAT_NULLS_YIELDS_NULL saranno sempre attivati e non sarà possibile disattivarli. ANSI_NULLS e ANSI_PADDING devono essere attivati quando si creano o si modificano gli indici nelle colonne calcolate o le viste indicizzate. Il driver ODBC di SQL Server Native Client e il provider OLE DB di SQL Server Native Client per SQL Server impostano automaticamente ANSI_NULLS e ANSI_PADDING su attivo alla connessione. Per le connessioni dalle applicazioni DB-Library, tuttavia, l'impostazione predefinita per SET ANSI_NULLS e SET ANSI_PADDING is OFF è disattivata.**

TABELLA 4-1 Opzioni SET

OPZIONE SET	OPZIONE DATABASE	OPZIONE SERVER	IMPOSTAZIONE PREDEFINITA
ANSI_DEFAULTS	Nessuno	Nessuno	N/D
ANSI_NULL_DFLT_OFF ANSI_NULL_DFLT_ON	ANSI_NULL_ DEFAULT	user options predefinita	OFF
ANSI_NULLS	ANSI_NULLS	user options predefinita	OFF
ANSI_PADDING	ANSI_PADDING	user options predefinita	ON
ANSI_WARNINGS	ANSI_WARNINGS	user options predefinita	OFF
ARITHABORT	ARITHABORT	user options predefinita	OFF
ARITHIGNORE	Nessuno	user options predefinita	OFF
CONCAT_NULL_ YIELDS_NULL	CONCAT_NULL_ YIELDS_NULL	Nessuno	OFF
CONTEXT_INFO	Nessuno	Nessuno	OFF
CURSOR_CLOSE_ ON_COMMIT	CURSOR_CLOSE_ ON_COMMIT	user options predefinita	OFF
DATEFIRST	Nessuno	Nessuno	7
DATEFORMAT	Nessuno	Nessuno	mdy
DEADLOCK_PRIORITY	Nessuno	Nessuno	NORMAL
FIPS_FLAGGER	Nessuno	Nessuno	OFF
FMTONLY	Nessuno	Nessuno	OFF
FORCEPLAN	Nessuno	Nessuno	OFF
IDENTITY_INSERT	Nessuno	Nessuno	OFF
IMPLICIT_TRANSACTIONS	Nessuno	user options predefinita	OFF
LANGUAGE	Nessuno	Nessuno	us_english
LOCK_TIMEOUT	Nessuno	Nessuno	Nessun limite
NOCOUNT	Nessuno	user options predefinita	OFF

OPZIONE SET	OPZIONE DATABASE	OPZIONE SERVER	IMPOSTAZIONE PREDEFINITA
NOEXEC	Nessuno	Nessuno	OFF
NUMERIC_ ROUNDABORT	NUMERIC_ ROUNDABORT	Nessuno	OFF
OFFSETS	Nessuno	Nessuno	OFF
PARSEONLY	Nessuno	Nessuno	OFF
QUERY_GOVERNOR_ COST_LIMIT	Nessuno	query governor cost limit	OFF
QUOTED_IDENTIFIER	quoted identifier	user options predefinita	OFF
REMOTE_PROC_ TRANSACTIONS	Nessuno	Nessuno	OFF
ROWCOUNT	Nessuno	Nessuno	OFF
SHOWPLAN_ALL*	Nessuno	Nessuno	OFF
SHOWPLAN_TEXT*	Nessuno	Nessuno	OFF
SHOWPLAN_XML*	Nessuno	Nessuno	OFF
STATISTICS IO*	Nessuno	Nessuno	OFF
STATISTICS PROFILE*	Nessuno	Nessuno	OFF
STATISTICS TIME*	Nessuno	Nessuno	OFF
STATISTICS XML*	Nessuno	Nessuno	OFF
TEXTSIZE	Nessuno	Nessuno	OFF
TRANSACTION ISOLATION LEVEL	Nessuno	Nessuno	N/D
XACT_ABORT	Nessuno	Nessuno	OFF

Utilizzo delle opzioni server

Le opzioni del server possono essere impostate tramite le finestre di dialogo delle proprietà di SQL Server Management Studio o la stored procedure sp_configure. La differenza tra i due metodi sta nelle opzioni disponibili per l'impostazione. In SQL Server Management Studio sono disponibili solo le opzioni di configurazione del server utilizzate più comunemente, mentre con sp_configure si può accedere a tutte le opzioni di configurazione. Nella tabella 4-2 vengono elencate le opzioni del server disponibili, oltre alle corrispondenti opzioni SET e di database supportate in SQL Server 2012 e alle impostazioni predefinite (dove possibile).

NOTA **L'opzione allow updates è obsoleta. Se utilizzata, potrebbe generare un errore.**

TABELLA 4-2 Opzioni server

OPZIONE SERVER	OPZIONE SET	OPZIONE DATABASE	IMPOSTAZIONE PREDEFINITA
allow updates	Nessuno	Nessuno	0
backup compression default	Nessuno	Nessuno	0
clr enabled	Nessuno	Nessuno	0
cross db ownership chaining	Nessuno	Nessuno	0
default language	Nessuno	Nessuno	0
filestream access level	Nessuno	Nessuno	0
max text repl size	Nessuno	Nessuno	65536
nested triggers	Nessuno	Nessuno	1
remote access	Nessuno	Nessuno	1
remote admin connections	Nessuno	Nessuno	0
remote login timeout	Nessuno	Nessuno	20
remote proc trans	Nessuno	Nessuno	0
remote query timeout	Nessuno	Nessuno	600
server trigger recursion	Nessuno	Nessuno	1
show advanced options	Nessuno	Nessuno	0
user options	ANSI_NULL_DFLT_ON ANSI_NULL_DFLT_OFF	ANSI_NULL_ DEFAULT	OFF

OPZIONE SERVER	OPZIONE SET	OPZIONE DATABASE	IMPOSTAZIONE PREDEFINITA
	ANSI_NULLS	ANSI_NULLS	OFF
	ANSI_PADDING	ANSI_PADDING	ON
	ANSI_WARNINGS	ANSI_ WARNINGS	OFF
	ARITHABORT	ARITHABORT	OFF
	ARITHIGNORE	Nessuno	OFF
	CURSOR_CLOSE_ON_ COMMIT	CURSOR_ CLOSE_ON_ COMMIT	OFF
	DISABLE_DEF_CNST_CHK	Nessuno	OFF
	IMPLICIT_TRANSACTIONS	Nessuno	OFF
	NOCOUNT	Nessuno	OFF
	QUOTED_ IDENTIFIER	QUOTED_ IDENTIFIER	OFF

Utilizzo delle opzioni di database

Le opzioni di database vengono impostate eseguendo l'istruzione ALTER DATABASE. Nelle nuove installazioni SQL Server, le impostazioni nei database *model* e *master* sono le stesse. Quando vengono creati nuovi database, le opzioni predefinite vengono prese dal database *model*. Ogni volta che viene modificata un'opzione di database, Motore di database compila nuovamente la cache. Nella tabella 4-3 vengono elencate le opzioni di database standard, oltre alle corrispondenti opzioni SET e del server supportate in SQL Server 2012 e alle impostazioni predefinite (dove possibile).

NOTA **Microsoft consiglia di impostare ANSI_NULLS e ANSI_PADDING sempre su ON, al fine di evitare problemi con le prossime versioni di SQL Server. AUTO_UPDATE_STATISTICS_ASYNC non produce alcun effetto, a meno che AUTO_UPDATE_STATISTICS sia impostata su ON.**

TABELLA 4-3 Opzioni database

OPZIONE DATABASE	OPZIONE SET	OPZIONE SERVER	IMPOSTAZIONE PREDEFINITA
ANSI_NULL_DEFAULT	ANSI_NULL_DFLT_ON ANSI_NULL_DFLT_OFF	user options predefinita	OFF
ANSI_NULLS	ANSI_NULLS	user options predefinita	OFF

OPZIONE DATABASE	OPZIONE SET	OPZIONE SERVER	IMPOSTAZIONE PREDEFINITA
ANSI_PADDING	ANSI_PADDING	user options predefinita	ON
ANSI_WARNINGS	ANSI_WARNINGS	user options predefinita	OFF
AUTO_CLOSE	Nessuno	Nessuno	OFF
AUTO_CREATE_STATISTICS	Nessuno	Nessuno	ON
AUTO_SHRINK	Nessuno	Nessuno	OFF
AUTO_UPDATE_STATISTICS	Nessuno	Nessuno	ON
AUTO_UPDATE_STATISTICS_ ASYNC	Nessuno	Nessuno	OFF
CONCAT_NULL_YIELDS_NULL	CONCAT_NULL_YIELDS_ NULL	Nessuno	OFF
CURSOR_CLOSE_ON_COMMIT	CURSOR_CLOSE_ON_ COMMIT	user options predefinita	OFF
CURSOR_DEFAULT	Nessuno	Nessuno	GLOBAL
MERGE PUBLISH	Nessuno	Nessuno	FALSE
PUBLISHED	Nessuno	Nessuno	FALSE
QUOTED_IDENTIFIER	QUOTED_IDENTIFIER	user options predefinita	ON
READ_ONLY	Nessuno	Nessuno	FALSE
RECOVERY BULK_LOGGED	Nessuno	Nessuno	FALSE
RECOVERY SIMPLE	Nessuno	Nessuno	TRUE
RECURSIVE_TRIGGERS	Nessuno	Nessuno	FALSE
RESTRICTED_USER	Nessuno	Nessuno	FALSE
SINGLE_USER	Nessuno	Nessuno	FALSE
SUBSCRIBED	Nessuno	Nessuno	TRUE
TORN_PAGE_DETECTION	Nessuno	Nessuno	TRUE

Gestione della compatibilità del database

Per impostazione predefinita, quando in SQL Server 2012 viene creato un nuovo database, il livello di compatibilità predefinito è 110 (a meno che il database *model* abbia un livello di compatibilità inferiore). Quando un database viene aggiornato a SQL Server 2012, quelli precedenti a SQL Server 2005 vengono aggiornati al livello di compatibilità 90. Tutti gli altri database mantengono il livello esistente:

- 90 per il livello di compatibilità con SQL Server 2005
- 100 per il livello di compatibilità con SQL Server 2008
- 110 per il livello di compatibilità con SQL Server 2012

Benché il livello di compatibilità del database *master* non possa essere modificato, è possibile modificare quello del database *model*. Tale flessibilità consente la creazione di nuovi database con un livello di compatibilità non predefinito. Per modificare il livello di compatibilità utilizzare l'istruzione ALTER DATABASE.

Tale istruzione consente di impostare il livello di compatibilità da utilizzare per uno specifico database. L'istruzione ALTER DATABASE consente di impostare alcuni comportamenti di database in modo che siano compatibili con una specifica versione precedente di SQL Server. Nell'esempio seguente il livello di compatibilità del database Personnel viene modificato per SQL Server 2008:

T-SQL

```
ALTER DATABASE Personnel
SET COMPATIBILITY_LEVEL = 100;
GO
```

PowerShell

```
Invoke-Sqlcmd -Query "ALTER DATABASE Personnel;
SET COMPATIBILITY_LEVEL = 100;" -ServerInstance "DbServer17\Cwhouse"
```

In caso di possibili conflitti tra le impostazioni di compatibilità (e altre impostazioni), è importante sapere quale contesto di database è in uso. In genere, il contesto di database attuale corrisponde al database definito tramite l'istruzione USE se si trova in un batch/script, oppure al database contenente la stored procedure se si trova in una stored procedure applicata a tale istruzione. Quando una stored procedure viene eseguita da un batch o da un'altra stored procedure, l'esecuzione avviene secondo le impostazioni di opzione del database in cui è salvata. Per esempio, quando una stored procedure del database Support richiama una stored procedura del database Personnel, la procedura di Support viene eseguita in base all'impostazione del livello di compatibilità del database Support, mentre la procedura di Personnel viene eseguita in base al livello di compatibilità del database Personnel.

Configurazione di SQL Server con le stored procedure

Utilizzando la finestra di dialogo Proprietà server è possibile configurare molte aree di SQL Server, come illustrato nel Capitolo 1 Come si è appreso in questo capitolo, è possibile configurare SQL Server anche tramite le stored procedure, per esempio sp_configure. Tali stored procedure e altre query vengono eseguite in SQL Server Management Studio. SQL Server Management Studio dispone di uno strumento client incorporato che invia comandi a un'istanza SQL Server, che a sua volta analizza, compila ed esegue i comandi.

Nelle sezioni seguenti viene illustrato come configurare SQL Server utilizzando SQL Server Management Studio e le stored procedure. Negli altri capitoli sono disponibili ulteriori informazioni sull'utilizzo di SQL Server Management Studio.

Utilizzo di SQL Server Management Studio per le query

Per avviare SQL Server Management Studio e accedere allo strumento client delle query incorporato, completare la seguente procedura:

1. Fare clic su Start, Tutti i programmi, Microsoft SQL Server, SQL Server Management Studio. Oppure fare clic su Start, nella casella Esegui digitare **ssms**, quindi premere Invio.
2. Nella finestra di dialogo Connetti al server, illustrato nella figura 4-1, utilizzare l'elenco Tipo server per selezionare il componente al quale ci si desidera connettere, come Motore di database.

FIGURA 4-1 La finestra di dialogo Connetti al server.

3. Nel campo Nome server, digitare il nome del server su cui è in esecuzione SQL Server, per esempio CorpSvr04.

NOTA **È possibile connettersi a qualsisasi server visibile a patto che l'accesso remoto sia abilitato e il nome del server venga risolto. Selezionare l'opzione <Cerca...> nella casella a discesa Nome server, per eseguire una ricerca dei server disponibili. È anche possibile registrare il server con il quale si desidera lavorare. Fare riferimento al paragrafo "Gestione dei server" nel Capitolo 1 per maggiori dettagli.**

4. Utilizzare la casella a discesa Autenticazione per scegliere il tipo di autenticazione, per esempio Autenticazione di Windows o Autenticazione di SQL Server (in base ai tipi di autenticazione consentiti al momento dell'installazione del server). Fornire un nome utente Windows oppure l'ID di accesso e password SQL Server in base alle necessità.
   - **Autenticazione di Windows** Utilizza l'account di dominio e password correnti per stabilire la connessione database. Questa opzione funziona solo se viene attivata l'autenticazione Windows e si posseggono i privilegi appropriati.
   - **Autenticazione SQL Server** Consente di specificare un ID di accesso SQL Server e una password.
5. Fare clic su Connetti. Viene stabilita la connessione all'istanza predefinita (a meno di aver configurato precedentemente un'altra impostazione predefinita). Per modificare l'istanza a cui ci si desidera connettere, fare clic sul pulsante Opzioni prima di fare clic su Connetti, selezionare la scheda Proprietà connessione, quindi utilizzare l'elenco a discesa Connetti al database per selezionare il database desiderato.
6. In SQL Server Management Studio è possibile connettersi al database selezionato precedentemente facendo clic sulla barra degli strumenti Nuova query e saltando i passaggi restanti. Per connettersi a un database diverso, fare clic su File, Nuovo, quindi selezionare il tipo di query, per esempio Query del Motore di database.
7. Nella finestra di dialogo Connetti al motore di database, specificare il nome del server, oppure selezionare Sfoglia nell'elenco a discesa per cercare tutti i computer in cui viene eseguito SQL Server all'interno di un'intera foresta Active Directory oppure alle differenti istanze in esecuzione sul servizio SQL Browser.
8. Specificare la tecnica di autenticazione da utilizzare. Fare clic su Connetti. Come prima, viene stabilita la connessione al database predefinito (a meno di aver configurato precedentemente un'altra impostazione predefinita). Per modificare il database a cui ci si connette, fare clic su Opzioni, selezionare la scheda Proprietà di connessione, quindi utilizzare l'elenco a discesa Connetti al database per selezionare il database al quale si desidera connettersi.

Se si sta utilizzando un database attivo in SQL Server Management Studio ed è già stata effettuata l'autenticazione della connessione, è possibile connettersi automaticamente all'istanza server del database attualmente selezionata e utilizzare le informazioni di autenticazione attuali per l'accesso. Per fare tutto ciò, fare clic con il pulsante destro del mouse sulla vista Esplora oggetti di SQL Server Management Studio, quindi selezionare Nuova query.

Esecuzione di query e modifica delle impostazioni

La finestra di query in SQL Server Management Studio normalmente è divisa in due riquadri (vedere la figura 4-2). Il riquadro superiore consente l'inserimento delle query, mentre in quello inferiore vengono visualizzati i risultati.

Se nella parte inferiore della finestra non viene visualizzato un altro riquadro, non preoccuparsi. Viene visualizzato automaticamente quando viene digitata la query. È anche possibile impostare il riquadro in modo che si apra automaticamente selezionando Mostra riquadro risultati dal menu Finestra.

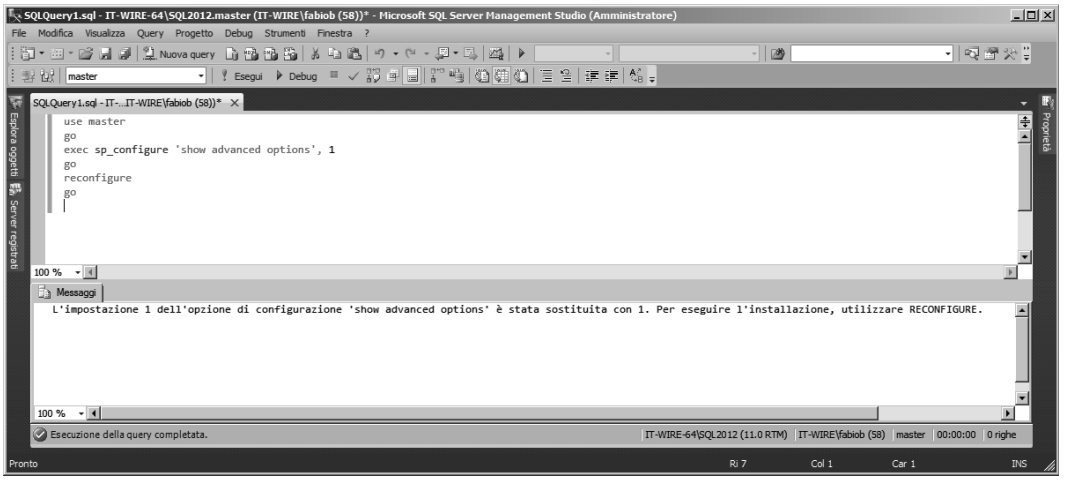

FIGURA 4-2 Esecuzione di query in SQL Server Management Studio.

Come è noto, è possibile utilizzare sp_configure per visualizzare e modificare le impostazioni di configurazione di SQL Server. Sono disponibili due tipi di impostazioni di configurazione: dinamici e non dinamici. In questo caso, un'impostazione dinamica è quella che si può modificare senza dover arrestare e riavviare SQL Server. Per eseguire sp_configure o altri tipi di query, digitare un comando nel riquadro superiore, quindi fare clic sul pulsante Esegui della barra degli strumenti (il punto esclamativo di colore rosso). È inoltre possibile eseguire i comandi utilizzando le sequenze di tasti illustrate di seguito:

- F5
- Ctrl+E
- Alt+X

NOTA **Per impostazione predefinita, tutti gli utenti dispongono delle autorizzazioni per eseguire sp_configure, così da poter visualizzare le impostazioni. Tuttavia, solo gli utenti con l'autorizzazione a livello server Alter Settings possono utilizzare sp_configure per modificare le opzioni di configurazione. Per impostazione predefinita, solo i membri dei ruoli server fissi sysadmin e serveradmin posseggono tale autorizzazione. Con sp_configure, solo gli utenti con l'autorizzazione a livello server Alter Settings possono eseguire il comando RECONFIGURE o RECONFIGURE WITH OVERRIDE.**

Ogni volta che si utilizza sp_configure per modificare le impostazioni, le modifiche di fatto non vengono applicate fino a quando non si esegue il comando RECONFIGURE. Per modificare alcune impostazioni molto rischiose a livello di sicurezza è necessario utilizzare il commando RECONFIGURE WITH OVERRIDE. WITH OVERRIDE disabilita il controllo dei valori di configurazione. Sebben WITH OVERRIDE possa essere utilizzato con qualunque opzione di configurazione, è specificamente richiesto per impostare l recovery interval, min server memory e max server memory. WITH OVERRIDE è richiesto anche quando si imposta una opzione a un valore che non è generalmente consigliato.

Le impostazioni sp_configure sono divise in due categorie: standard e avanzate I comandi standard possono essere eseguiti in qualunque momento, mentre quelli avanzati possono essere eseguiti solo quando Show advanced options è impostato su 1.

Quando questa impostazione è attivata, è possibile modificare sia le impostazioni standard sia quelle avanzate. Per modificare le impostazioni avanzate, completare la seguente procedura:

1. In SQL Server Management Studio, digitare quanto segue:

   ```
   exec sp_configure 'show advanced options', 1
   go
   reconfigure
   go
   ```

   SUGGERIMENTO **È possibile disattivare le opzioni avanzate in seguito, modificando il valore in 0.**

2. Eseguire i comandi premendo Ctrl+E.
3. Cancellare quanto presente nella finestra delle query.
4. Digitare un comando sp_configure per ogni opzione che si desidera modificare.
5. Digitare **reconfigure** (o **reconfigure with override**).
6. Digitare **go**.
7. Eseguire i comandi premendo Ctrl+E.
8. Se sono state modificate impostazioni non dinamiche, arrestare e riavviare il server. Consultare la tabella 4-4 e 4-5 per ulteriori dettagli.

Verifica e impostazione dei parametri di configurazione

Nella tabella 4-4 viene fornito un riepilogo dei parametri di configurazione standard. I parametri sono elencati in ordine alfabetico, con riportati i valori minimi, massimi e predefiniti.

Diversi valori sono indicati in secondi nelle colonne Valore minimo e Valore massimo. Nella colonna dei parametri dinamici viene indicato se l'impostazione è dinamica.

Quando in questa colonna è presente una "N", significa che è necessario arrestare e riavviare il server per applicare le modifiche.

Va ricordato che per abilitare database indipendenti nell'istanza è necessario abilitare l'opzione contained database authentication.

TABELLA 4-4 Riepilogo rapido dei parametri di configurazione standard

OPZIONE DI CONFIGURAZIONE	VALORE MINIMO	VALORE MASSIMO	VALORE PREDEFINITO	DINAMICA: YES/NO
allow updates	0	1	0	Y
backup compression default	0	1	0	Y
clr enabled	0	1	0	Y (a eccezione dei server WOW64)
contained database authentication	0	1	0	Y
cross db ownership chaining	0	1	0	Y
default language	0	9999	0	Y
filestream access level	0	2	0	Y
max text repl size	-1	2147483647	65536	Y
nested triggers	0	1	1	Y
remote access	0	1	1	N
remote admin connections	0	1	0	Y
remote login timeout	0 (s)	2147483647 (s)	20	Y
remote proc trans	0	1	0	Y
remote query timeout	0 (s)	2147483647 (s)	600	Y
server trigger recursion	0	1	1	Y
show advanced options	0	1	0	Y
user options	0	32767	0	Y

Nella tabella 4-5 viene fornito un riepilogo dei parametri di configurazione avanzati. Per visualizzare o modificare tali parametri è necessario impostare il parametro Show advanced options su 1. Le opzioni che si configurano automaticamente hanno un asterisco (*) dopo il nome. Diversi valori sono indicati in secondi, byte e altre unità di misura come si denota nelle colonne Valore minimo e Valore massimo.

Con max worker threads, 1.024 è il valore massimo consigliato per i sistemi operativi a 32 bit. Il valore predefinito zero (0) calcola il valore effettivo con la seguente formula: 256 + (numero di processori – 4) * 8).

NOTA **Non è possibile modificare alcune opzioni avanzate, possono essere solo visualizzate. Notare anche che open objects è obsoleto e disponibile solo per la compatibilità con le versioni precedenti. In SQL Server, il numero di oggetti database aperti è gestito dinamicamente e limitato solo dalla memoria disponibile.**

TABELLA 4-5 Riepilogo rapido dei parametri di configurazione avanzati

OPZIONE DI CONFIGURAZIONE	VALORE MINIMO	VALORE MASSIMO	VALORE PREDEFINITO	DINAMICA: YES/NO
ad hoc distributed queries	0	1	0	Y
access check cache bucket count	0	16384	0	Y
access check cache quota	0	2147483647	0	Y
affinity I/O mask	-2147483648	2147483647	0	N
affinity64 I/O mask	-2147483648	2147483647	0	N
affinity mask	-2147483648	2147483647	0	Y
affinity64 mask	-2147483648	2147483647	0	Y
Agent XPs	0	1	0; 1	Y
awe enabled	0	1	0	N
blocked process threshold	0 (s)	86400 (s)	0	Y
c2 audit mode	0	1	0	N
common criteria compliance enabled	0	1	0	N
cost threshold for parallelism	0	32767	5	Y
cursor threshold	-1	2147483647	-1	Y
Database Mail XPs	0	1	0	Y
default full-text language	0	2147483647	1033	Y
default trace enabled	0	1	1	Y
disallow results from triggers	0	1	0	Y
EKM provider enabled	0	1	0	N

OPZIONE DI CONFIGURAZIONE	VALORE MINIMO	VALORE MASSIMO	VALORE PREDEFINITO	DINAMICA: YES/NO
fill factor	0 (%)	100 (%)	0	N
ft crawl bandwidth (max)	0	32767	100	Y
ft crawl bandwidth (min)	0	32767	0	Y
ft notify bandwidth (max)	0	32767	100	Y
ft notify bandwidth (min)	0	32767	0	Y
index create memory*	704 (kb)	2147483647 (kb)	0	Y
in-doubt xact resolution	0	2	0	Y
lightweight pooling	0	1	0	N
locks*	5000	2147483647	0	N
max degree of parallelism	0	64; 1024	0	Y
max full-text crawl range	0	256	4	Y
max server memory*	16 (mb)	2147483647 (mb)	2147483647	N
max text repl size	–1 (b)	2147483647 (b)	65536	Y
max worker threads	128	32767	0	N
media retention	0	365	0	N
min memory per query	512 (kb)	2147483647 (kb)	1024	Y
min server memory*	0 (mb)	2147483647 (mb)	0	Y
network packet size	512 (b)	32767 (b)	4096	Y
Ole Automation Procedures	0	1	0	Y
open objects	0	2147483647	0	N
optimize for ad hoc workloads	0	1	0	Y
ph_timeout	1 (s)	3600 (s)	60	Y
precompute rank	0	1	0	Y

OPZIONE DI CONFIGURAZIONE	VALORE MINIMO	VALORE MASSIMO	VALORE PREDEFINITO	DINAMICA: YES/NO
priority boost	0	1	0	N
query governor cost limit	0	2147483647	0	Y
query wait	-1	2147483647 (s)	-1	Y
recovery interval*	0 (min)	32767 (min)	0	Y
Replication XPs	0	1	0	Y
scan for startup procs	0	1	0	N
set working set size	0	1	0	N
SMO and DMO XPs	0	1	1	Y
SQL Mail XP	0	1	0	Y
transform noise words	0	1	0	Y
two digit year cutoff	1753	9999	2049	Y
user connections*	0	32767	0	N
xp_cmdshell	0	1	0	Y

È possibile visualizzare le impostazioni attuali di tutte le opzioni di configurazione eseguendo la query seguente:

T-SQL

```
exec sp_configure
go
```

PowerShell

```
Invoke-Sqlcmd -Query "exec sp_configure"
-ServerInstance "DataServer91\CorpServices"
```

NOTA **Per poter visualizzare le opzioni avanzate Show Advanced Options deve essere impostato su 1.**

Per visualizzare l'impostazione attuale di un'opzione di configurazione, eseguire la query seguente:

T-SQL

```
exec sp_configure 'optionName'
go
```

PowerShell

```
Invoke-Sqlcmd -Query "exec sp_configure 'optionName'"
```

```
-ServerInstance "DataServer91\CorpServices"
```

dove *optionName* è il nome dell'opzione che si desidera visualizzare, per esempio:

T-SQL

```
exec sp_configure 'max worker threads'
go
```

PowerShell

```
Invoke-Sqlcmd -Query "exec sp_configure 'max worker threads'"
-ServerInstance "DataServer91\CorpServices"
```

Per modificare il valore di un'impostazione, eseguire la query seguente:

T-SQL

```
exec sp_configure 'optionName', newValue
go
reconfigure with override
go
```

PowerShell

```
Invoke-Sqlcmd -Query "exec sp_configure 'optionName', newValue;
reconfigure with override" -ServerInstance "DataServer91\CorpServices"
```

dove *optionName* è il nome dell'opzione che si desidera visualizzare, mentre *newValue* è il nuovo valore per questa opzione, per esempio:

T-SQL

```
exec sp_configure 'locks', 10000
go
reconfigure
go
```

PowerShell

```
Invoke-Sqlcmd -Query "exec sp_configure 'locks', 10000;
reconfigure " -ServerInstance "DataServer91\CorpServices"
```

NOTA **Non bisogna sempre utilizzare WITH OVERRIDE. A parte che con recovery intervals, minimum server memory e maximum server memory, questo valore è necessario solo quando si eseguono aggiornamenti ad hoc e si imposta un'opzione a un valore generalmente non consigliato. Considerare che alcune modifiche alle impostazioni vengono applicate solo quando l'istanza di SQL Server viene riavviata.**

Modifica delle impostazioni con ALTER DATABASE

Per SQL Server 2012 e versioni successive, l'istruzione ALTER DATABASE sostituisce la stored procedure sp_dboption nella modifica delle impostazioni di database. Sebbene sp_dboption può ancora essere utilizzata, sarà rimossa nelle prossime versioni di SQL Server. Per modificare tali impostazioni è necessario essere membro di un ruolo a cui è concessa l'autorizzazione ALTER sul database o a cui tale autorizzazione è assegnata in modo esplicito.

Quando si esegue un'istruzione ALTER DATABASE, si verifica un checkpoint nel database per cui l'opzione è stata modificata, con la conseguente applicazione immediata della modifica. Nella tabella 4-6 viene offerta una panoramica delle opzioni di database che possono essere impostate con ALTER DATABASE.

NOTA **Con i database indipendenti è possibile impostare le opzioni che diversamente dovrebbero essere impostate a livello del server. DEFAULT_FULLTEXT_LANGUAGE imposta il valore della lingua predefinita per le colonne full-text o indicizzate. DEFAULT_LANGUAGE imposta il valore della lingua predefinita per i nuovi login creati. NESTED_TRIGGERS determina se i trigger AFTER hanno un effetto a cascata e attivano altri trigger. TRANSFORM_NOISE_WORDS può essere utilizzato per eliminare messaggi di errore relativi a parole non significative che causano errori nelle operazioni Booleane durante una query full-text. TWO_DIGIT_YEAR_CUTOFF specifica un intero da 1753 a 9999 che rappresenta l'anno dal quale si iniziano a interpretare gli anni rappresentati da due cifre come anni a quattro cifre.**

TABELLA 4-6 Riepilogo rapido delle opzioni di database

OPZIONE	QUANDO TRUE O IMPOSTATA SUL VALORE...	VALORI ACCETTATI
ALLOW_SNAPSHOT_ ISOLATION	Abilita l'isolamento dello SNAPSHOT della transazione, in cui le istruzioni vedono uno snapshot di dati come esistente all'inizio della transazione. Al posto del blocco viene utilizzato il controllo delle versioni delle righe.	ON | OFF
ANSI_NULL_DEFAULT	CREATE TABLE utilizza regole SQL-92 per stabilire se una colonna consente valori null.	ON | OFF
ANSI_NULLS	Tutti i confronti con un valore null restituiscono UNKNOWN (quando OFF, i valori non-UNICODE devono restituire TRUE se entrambi i valori sono NULL).	ON | OFF
ANSI_PADDING	Nei valori carattere vengono inseriti degli spazi finali, mentre nei valori binari vengono inseriti degli zero finali per l'adattamento alla lunghezza della colonna.	ON | OFF
ANSI_WARNINGS	Errori o avvisi vengono inviati quando si verificano condizioni quali la divisione per zero.	ON | OFF

OPZIONE	QUANDO TRUE O IMPOSTATA SUL VALORE...	VALORI ACCETTATI
ARITHABORT	La query o il batch viene interrotto per un overflow o un errore di divisione per zero. Se l'errore si verifica in una transazione, tale transazione viene annullata. (quando questa opzione è OFF viene visualizzato un messaggio di avviso, ma l'esecuzione continua se non si verifica alcun errore).	ON | OFF
AUTO_CLEANUP	Le informazioni sul rilevamento delle modifiche vengono eliminate automaticamente dopo il periodo di conservazione.	ON | OFF
AUTO_CREATE_ STATISTICS	Vengono generate automaticamente tutte le statistiche mancanti necessarie per l'ottimizzazione della query.	ON | OFF
AUTO_UPDATE_ STATISTICS	Vengono generate automaticamente tutte le statistiche non aggiornate necessarie per l'ottimizzazione della query.	ON | OFF
AUTOCLOSE	Dopo l'ultima disconnessione dell'utente, il database viene chiuso normalmente e vengono rilasciate le relative risorse.	ON | OFF
AUTOSHRINK	I database vengono compattati periodicamente in modo automatico.	ON | OFF
CHANGE_RETENTION	Quando il rilevamento delle modifiche è impostato su ON, imposta il periodo di conservazione delle informazioni sul rilevamento delle modifiche.	*RetPeriod* {DAYS | HOURS | MINUTES}
CHANGE_TRACKING	Attiva o disattiva il rilevamento delle modifiche.	ON | OFF
COMPATIBILITY_LEVEL	Imposta il livello di compatibilità del database.	80 | 90 | 100
CONCAT_NULL_ YIELDS_NULL	Quando uno degli operandi di un'operazione di concatenazione è NULL, il risultato è NULL.	ON | OFF
CONTAINMENT	I login sono autenticati dal database, ovvero mappati a login presenti nel database master.	NONE | PARTIAL

OPZIONE	QUANDO TRUE O IMPOSTATA SUL VALORE...	VALORI ACCETTATI
CURSOR_CLOSE_ON_COMMIT	Tutti i cursori aperti quando si esegue il commit o il rollback di una transazione vengono chiusi (quando questa opzione è impostata su è OFF, i cursori restano aperti quando viene eseguito il commit di una transazione. Il rollback di una transazione chiude tutti i cursori tranne quelli definiti come INSENSITIVE o STATIC).	ON | OFF
CURSOR_DEFAULT	Per impostazione predefinita i cursori vengono definiti LOCAL.	LOCAL | GLOBAL
DATE_CORRELATION_OPTIMIZATION	SQL Server mantiene le statistiche di correlazione tra le tabelle di un database collegate da un vincolo FOREIGN KEY e con colonne di tipo datetime.	ON | OFF
DB_CHAINING	Il database può essere l'origine o la destinazione di una catena di proprietà tra database. L'istanza di SQL Server riconoscerà questa impostazione solo quando l'opzione server cross db ownership chaining è impostata su 0 (OFF). In caso contrario, tutti i database utente possono partecipare alle catene di proprietà tra database, indipendentemente dal valore di questa opzione.	ON | OFF
EMERGENCY	Contrassegna il database come read-only, disattiva la registrazione e consente l'accesso solo ai membri del ruolo sysadmin.	EMERGENCY
ENCRYPTION	La crittografia trasparente dei dati viene attivata per il database.	ON | OFF
MULTI_USER	Più utenti possono accedere al database.	MULTI_USER
NUMERIC_ROUNDABORT	Viene generato un errore quando in un'espressione si verifica una perdita di precisione Quando questa opzione è impostata su OFF, le perdite di precisione non generano messaggi di errore e il risultato viene arrotondato alla precisione della colonna o della variabile in cui viene salvato il risultato.	ON | OFF

OPZIONE	QUANDO TRUE O IMPOSTATA SUL VALORE...	VALORI ACCETTATI
OFFLINE	Il database viene contrassegnato come non in linea (altrimenti il database è disponibile per l'utilizzo).	OFFLINE
ONLINE	Il database viene contrassegnato come in linea e disponibile per l'utilizzo.	ONLINE
PAGE_VERIFY	SQL Server è in grado di rilevare le pagine del database danneggiate.	CHECKSUM | TORN_PAGE_ DETECTION | NONE
PARAMETERIZATION	SQL Server esegue la parametrizzazione di tutte le query del database.	SIMPLE | FORCED
QUOTED_IDENTIFIER	Gli identificatori possono essere delimitati tra virgolette doppie.	ON | OFF
READ_COMMITTED_ SNAPSHOT	Al posto del blocco viene utilizzato il controllo delle versioni delle righe; tutte le istruzioni vedono uno snapshot di dati come esistente all'inizio della transazione indipendentemente dal fatto che le transazioni utilizzano l'isolamento dello SNAPSHOT.	ON | OFF
READ_ONLY	Il database è impostato su read-only (ma può essere eliminato utilizzando l'istruzione DROP DATABASE). Il database non può essere in uso quando viene impostata questa opzione (tranne per *master*).	READ_ONLY
RECOVERY	Comporta la reimpostazione del modello di recupero. SIMPLE consente un checkpoint per tagliare la parte inattiva del log.	FULL | BULK_ LOGGED | SIMPLE
RECURSIVE_TRIGGERS	Consente l'attivazione ricorsiva di trigger (quando è OFF, impedisce la ricorsione diretta ma non quella indiretta. Per disattivare la ricorsione indiretta, impostare l'opzione del server nested triggers su 0 utilizzando sp_configure).	ON | OFF
RESTRICTED_USER	Il database può essere utilizzato solo dal proprietario.	RESTRICTED_ USER
SINGLE_USER	Al database può accedere un utente per volta.	SINGLE_USER

OPZIONE	QUANDO TRUE O IMPOSTATA SUL VALORE...	VALORI ACCETTATI
TORN_PAGE_ DETECTION	Consente il rilevamento di pagine incomplete (questa opzione viene sostituita da PAGE_VERIFY).	ON | OFF
TRUSTWORTHY	I moduli di database che utilizzano un contesto di rappresentazione possono accedere alle risorse esterne al database.	ON | OFF

La maggior parte delle opzioni elencate nella Tabella 4-6, accetta un valore ON o OFF utilizzato per impostare lo stato dell'opzione. Per esempio, è possibile attivare la crittografia trasparente dei dati sul database CustomerSupport utilizzando il seguente comando:

T-SQL

```
USE master;
GO
ALTER DATABASE CustomerSupport
SET ENCRYPTION ON;
GO
```

PowerShell

```
Invoke-Sqlcmd -Query "USE master; ALTER DATABASE CustomerSupport
SET ENCRYPTION ON;" -ServerInstance "DataServer91\CorpServices"
```

Alcune opzioni impostano in modo esplicito uno stato specifico. Per esempio, se al database *CustomerSupport* non fossero connessi degli utenti, il database potrebbe essere impostato su read-only utilizzando il seguente comando:

T-SQL

```
USE master;
GO
ALTER DATABASE CustomerSupport
SET READ_ONLY;
GO
```

PowerShell

```
Invoke-Sqlcmd -Query "USE master; ALTER DATABASE CustomerSupport
SET READ_ONLY;" -ServerInstance "DataServer91\CorpServices"
```