

MySQL Oracle SQL Server

Backup, Ripristino, Manutenzione e
Automazione

MySql - Backup

È possibile effettuare il backup dei dati in un database MySQL in 4 modi distinti:

1. Effettuando una copia fisica della struttura del database
2. Utilizzando i comandi via shell messi a disposizione da MySQL (es. mysqldump)
3. Utilizzando tool grafici come MySQL Administrator oppure utility (es. mysqlhotcopy)
4. Replicazione

Copia Fisica dei File:

MySQL salva tutta la struttura del database nella cartella "data" quindi copiando l'intera directory è possibile effettuare un salvataggio dello stato corrente del sistema. Ricordarsi di bloccare le tabelle e successivamente sbloccarle per evitare che nel frattempo i dati vengano modificati creando inconsistenze nel backup.

Il percorso di salvataggio può essere impostato nel file `/etc/my.cnf`.

MySql - Backup

Mysqldump:

Uno strumento estremamente utile messo a disposizione da MySQL è “mysqldump” tramite il quale è possibile effettuare una copia del database creando un file (chiamato dump) contenente tutte le query sql che permettono di ricreare la struttura e i dati del database selezionato.

MySQLdump può creare inconsistenze pertanto è preferibile bloccare il database prima di effettuare il dump dello stesso, tali problematiche sull'inconsistenza dei dati possono essere risolte utilizzando i LOG incrementali di MySQL (più avanti nella manutenzione).

Esempio di utilizzo:

`mysqldump --all-databases > destinazione` (destinazione è il percorso e il nome del file dove verrà scritto il backup, `--all-databases` specifica che verranno salvati tutti i database presenti sul server)

`mysqldump nome_db > destinazione` (nome_db è il nome del database da salvare)

MySQL - Backup

Mysqhotcopy:

È uno script in Perl il cui compito è copiare i dati del db specificato in una directory diversa nella stessa macchina. Effettua il LOCK sulle tabelle ed chiama il comando "cp" per effettuare la copia fisica.

Esempio:

mysqhotcopy db_name [/path/to/new_directory] ATTENZIONE viene utilizzato per salvare solo tabelle di tipo MyISAM (quindi non le transazionali INNODB).

MySql Administrator:

Tool grafico messo a disposizione da MySql che permette, tramite la relativa interfaccia, la creazione di progetti di backup. In ogni progetto è possibile specificare le tabelle da salvare, i dati, e la struttura del database (viste, stored procedure etc..), inoltre permette la temporizzazione dello stesso nel tempo automatizzando i salvataggi.

L'Administrator utilizza mysqldump per effettuare il salvataggio dei dati, quindi il file che viene generato sarà un insieme di query sql per ricreare l'intero database.

Replicazione:

MySql permette la replicazione del server. Viene fatta una copia speculare del database basandosi sull'utilizzo del bin-log (vedremo in manutenzione).

Per la sua realizzazione viene decisa una macchina master e una (o più) slave. La macchina master ha il compito di inviare i bin-log alle slave ricreando di volta in volta la struttura del DB quindi senza perdita di consistenza sui dati. La replicazione è una procedura asincrona questo significa che il master può inviare i dati quando vuole.

Nelle impostazioni della procedura è possibile specificare quali database e tabelle replicare sui server slave remoti.

Mysql - Ripristino

Il ripristino del database può avvenire per due cause:

1. Ripristino di una versione vecchia del database (necessaria copia di backup)
3. Crash del Sistema Operativo

Ripristino da copia di Backup

Si può fare utilizzando la linea di comando utilizzando “mysql” oppure tramite l'interfaccia grafica Mysql Administrator.

Per importare un precedente backup si utilizza il comando

```
mysql < file_di_backup (file di backup è il percorso e il nome del file contenente i dati)
```

se invece si è salvato solo un database si possono importare la sua struttura e i suoi dati nel seguente modo

```
mysql nome_db < file_di_backup
```

se si utilizza MySQL administrator è presente un pannello chiamato “Restore” dove è possibile specificare il file da ripristinare e quali parti dello stesso.

Crash del Sistema Operativo:

Per le tabelle transazionali (INNODB) al riavvio del server vengono controllati i bin log e se vi sono transazioni pendenti queste non vengono completate e viene fatto il rollback di tutta la transazione riportando il sistema in uno stato di consistenza.

Nel caso di tabelle MyISAM è presente un'utility chiamata myisamchk che effettua un controllo sull'integrità della struttura.

MySql - Ripristino

Point-in-Time Recovery

Se il server è stato eseguito in modalità `--log-bin` è possibile ripristinare i dati in esso contenuti partendo dai log.

L'utilizzo di `point-in-time` permette il ripristino da un certo istante temporale di un certo log tramite l'utility `mysqlbinlog`.

Esempio:

```
mysqlbinlog --stop-date="2005-04-20 9:59:59" \ /var/log/mysql/bin.001 | mysql -u root -p
```

```
mysqlbinlog --start-date="2005-04-20 9:55:00" \ --stop-date="2005-04-20 10:05:00" / /var/log/mysql/bin.001
```

specificando il parametro `stop-date` e `start-date` si indicano l'inizio e la fine del ripristino.

MySQL - Manutenzione

Comandi SQL principali:

Check Table

Optimize Table

Check table viene utilizzata in caso di crash e di problemi sulla struttura (anche fisica) della tabella.

Esempio:

```
myisamchk -r -q tbl_nome (q r indicano quick repair per effettuare un controllo rapido)
```

Optimize table

viene utilizzato nel caso in cui vengano eliminate grandi quantità di dati o eseguite parecchie modifiche alla struttura. In questi casi le righe eliminate vengono semplicemente scollegate dalla tabella e lo spazio continua a risultare occupato anche se gli indici vengono riutilizzati per altri record. Per riutilizzare quello spazio viene chiamato il comando optimize table.

```
OPTIMIZE [LOCAL | NO_WRITE_TO_BINLOG] TABLE tbl_name [, tbl_name]
```

MySql - Manutenzione

Log:

MySQL prevede 4 tipi di log.

1. Error Log
2. Query Log
3. Binary Log
4. Slow Query Log

Error Log:

Mysql aggiorna l'error log quando viene avviato , quando viene chiuso, se viene riavviato e, in caso di crash, viene salvato lo stack trace per poter , eventualmente, individuare la causa dell'errore.

Query Log:

Contiene tutte le query e le connessioni che vengono eseguite sul server

Binary Log:

Log estremamente importante nel caso di ripristino e backup. In questo log vengono memorizzate tutte le query che modificano la struttura o i dati del database. Viene utilizzato nelle tabelle INNODB per le mantenere la congruenza dei dati e nella replicazione del server. Per abilitarli è necessario far partire mysql con l'opzione --bin -log.

MySQL aggiunge alla fine del nome del file del log un numero progressivo a 6 cifre. Tale numero viene incrementato se il file supera la dimensione massima consentita oppure in caso di riavvio del server.

Slow Query Log:

Nelle variabili di configurazione di mysql è possibile specificare il tempo massimo di esecuzione di una query, nel caso in cui questa superi il tempo massimo indicato viene salvata nel file di log per le query "lente".

MySql - Manutenzione

MySql Administrator

Sempre in questo tool vengono messi a disposizione dei comodi strumenti per il controllo del sistema.

Nella scheda Health e Options si possono controllare lo stato di “salute” del nostro server, i valori delle variabili di ambiente (si possono modificare tramite il file `/etc/my.cnf`) e alcuni grafici indicativi del traffico generato, della memoria utilizzata e una serie di statistiche di utilizzo.

MySQL - Automazione

Per quel che riguarda l'automazione MySQL mette a disposizione due strumenti:

1. Triggers

I trigger sono un insieme di query che vengono eseguite quando accade un determinato evento legato ad una tabella.

La sintassi per la creazione di un trigger è la seguente:

```
CREATE  
[DEFINER = { user | CURRENT_USER }]  
TRIGGER trigger_name trigger_time trigger_event  
ON tbl_name FOR EACH ROW trigger_stmt
```

Il trigger rimane associato a `tbl_name`, (non è possibile associare un trigger ad una tabella temporanea o ad una vista).

`Trigger_time` può essere "BEFORE" o "AFTER" e indicano in che momento dell'azione verrà eseguito.

`Trigger_event` indica il tipo di evento collegato al trigger e può essere "INSERT", "LOAD DATA", "REPLACE", "UPDATE", "DELETE".

MySql - Automazione

2. Event Scheduler

È possibile creare degli eventi e automatizzarli in funzione del tempo. L'EVENT SCHEDULER non è altro che un database con un insieme di query SQL che vengono eseguite in un determinato istante.

ATTENZIONE! Non è possibile abilitare o disabilitare l'Event Scheduler mentre il server è attivo.

Tale strumento è disponibile dalla versione 5.1.16 e prevede 3 comandi:

CREATE EVENT (creazione di un evento)

ALTER EVENT (modifica di un evento)

DROP EVENT (eliminazione di un evento)

Oracle - Backup

Oracle prevede 4 modalità di backup:

1. Backup logici che utilizzano export (e l'utility complementare Import vedi ripristino)
2. Backup logici con Data Pump export e import
3. Backup fisici (online e offline)
4. Backup fisici incrementali tramite RMA

Data Pump export e import

Ideali per operazioni di manutenzione (notevoli vantaggi).

CAPITOLO 22

Export

Export esegue un dump del server o dei database indicati.

Il file scritto da export conterrà tutti i comandi necessari a ricreare completamente tutti gli oggetti e dati scelti.

La funzione export prevede 4 livelli di funzionamento:

- Full
Tutto il DB viene esportato (dizionario e utenti compresi)
- Tablespace
Tutti gli oggetti della tablespace selezionata
- User
Tutti gli oggetti di un singolo utente (proprietario)
- Table
Esporta una tabella specifica (con o senza dati).

Oracle - Backup

La export può essere eseguita in modo interattivo tramite oracle enterprise manager oppure con i file dei comandi.

Si possono specificare dei parametri che, se sono in conflitto, possono far fallire il comando.
Es: full = Y e owner = HR (esportazione “full” e esportazione “user”) sono incoerenti in quanto nell'esportazione full è già compresa l'esportazione “user”

Esempio: `exp system/manager file=expdat.dmp compress=Y owner=(HR,THUMPER)`
L'esportazione viene eseguita dall'user **system** che ha password **manager** nel file **expdat.dmp**, **comprimendo** tutti i dati relativi agli utenti **HR** e **THUMPER**

ESPORTAZIONI COERENTI:

Export esporta una tabella per volta (quindi in momenti differenti) cosicché queste potrebbero variare nel frattempo creando inconsistenze nel DB.

Si dovrebbero programmare le esportazioni quando nessuno sta modificando il DB (startup restrict, ad esempio, assicura che solo i DBA siano connessi al momento dell'esportazione) però non sempre è possibile.

Il parametro consistent (solo per esportazioni complete) cerca di creare una versione coerente a partire dall'inizio dell'esportazione. Se non riesce la creazione di questa versione compariranno degli errori “snapshot too old”.

Per evitare questi errori bisogna creare segmenti di rollback molto grandi o una tablespace di undo di grandi dimensioni

Rendere inattivo il database limitando le connessioni ai soli utenti con privilegi SYSDBA.

Vengono inoltre creati due file (parfile). Uno conterrà i nomi della maggior parte delle tabelle, usato per la prima esportazione con consistent = N. Il secondo file contiene i nomi delle tabelle per cui si deve mantenere la consistenza (consistent = Y). Riducendo in questo modo le dimensioni dei segmenti di rollback

Oracle - Backup

ESPORTAZIONE DELLE TABLESPACES:

Vengono esportati tutti gli oggetti della tablespace con il comando

Exp utente/pswd tablespaces=nome_t

BACKUP FISICO OFFLINE

Viene effettuato il Backup di tutti i file del database, il comando viene eseguito dopo che il db è stato **chiuso**. Questi file offrono un'immagine completa del db.

Vengono copiati:

File di dati

Control file

Redo log in linea

Essendo chiuso fornisce un'immagine completa del db così com'era nel momento in cui è stato chiuso.

Idealmente, tutti i file di dati si trovano in directory allo stesso livello su ciascun dispositivo.

Questo sistema è molto affidabile propria per la natura del tipo di backup però implica un blocco nell'accesso al database.

Oracle - Backup

BACKUP FISICO ONLINE

Possono essere utilizzati quando il server è eseguito in modalità ARCHIVELOG, questa modalità mantiene su disco tutti i redo log file.

Se i file di log dovessero riempirsi LGWR li sovrascrive in quel momento ARC0-ARC9 archivia i file di log prima che questi vengano sovrascritti.

La tablespace su cui si effettua il backup viene impostata nello stato di backup (l'utility RMAN lo fa in automatico).

Ora il database può essere recuperato completamente e il rollforward può essere effettuato in qualsiasi momento tramite i file di redo log.

Viene fatto il backup dei seguenti file:

- File di dati
- Redo log file archiviati
- Control file

In questo modo il database rimane sempre aperto e quindi accessibile dagli utenti.

I database non eseguiti in modalità archive log potranno essere recuperati solo nel momento in cui avviene il backup fisico.

E' consigliabile effettuare il backup nelle ore di minore attività a causa dell'elevato uso di risorse del sistema.

Oracle - Ripristino

RECOVERY MANAGER

RMAN è una utility fornita da Oracle per la gestione dei recuperi.

Tiene traccia dei backup tramite un recovery catalog.

È composto da quattro componenti:

- RMAN
- Database di destinazione
- Database col catalogo dei recuperi
- Software gestione supporti

Vengono eseguiti backup fisici incrementali dei file di dati.

Ad ogni livello di backup vengono archiviati solo i blocchi che sono stati utilizzati dall'ultimo backup cumulativo o completo.

Il vantaggio più grande sta nel backup dei blocchi e non delle tablespaces.

Sql Server 2005 - Backup

In Sql Server l'operazione di backup ha un effetto trascurabile sull'esecuzione di transazioni, per cui è possibile eseguire operazioni di backup durante il normale funzionamento del database, cosa ben diversa negli altri sistemi di database. Durante l'esecuzione di un backup, in Microsoft SQL Server i dati vengono copiati direttamente dai file di database nelle periferiche di backup (dischi o nastro). I dati non vengono modificati e le transazioni in esecuzione durante il backup non vengono mai posticipate.

Sono previste 3 modelli di backup:

- Il modello di recupero con registrazione minima
Nell'ambito di questo modello tutte le parti inattive del log vengono automaticamente liberate per poter essere riutilizzate, ovvero troncate, dopo i checkpoint automatici. La gestione del backup in un database in cui si utilizza questo tipo di modello è semplice. Il database, tuttavia, può essere ripristinato solo fino al momento dell'ultimo backup dei dati. Tutti gli aggiornamenti successivi all'ultimo backup e precedenti all'errore vengono persi.
- Il modello di recupero con registrazione completa
Il modello di recupero con registrazione completa è più complesso, poiché comporta l'esecuzione del backup e del ripristino del log delle transazioni (backup del log). Il vantaggio di questo modello di recupero è la possibilità di ripristinare un database rispetto a qualsiasi data e ora all'interno di un backup del log (recupero temporizzato). Presupponendo che sia possibile eseguire il backup del log attivo in seguito a un'emergenza, è possibile ripristinare il database fino al punto di errore, senza alcuna perdita di dati.
- Il modello di recupero con registrazione minima delle transazioni di massa
Può essere utilizzato occasionalmente a complemento del modello di recupero con registrazione completa. Questo tipo di recupero consente di migliorare le prestazioni durante alcune operazioni di massa su larga scala, ma supporta solo il ripristino di backup completi. Il recupero temporizzato non è supportato. Per questo e per altri motivi, è consigliabile utilizzare questo modello solo occasionalmente nel corso di operazioni di massa di notevole entità, ad esempio le importazioni di massa di grandi quantità di dati. In tutti gli altri casi è consigliabile utilizzare il modello di recupero con registrazione completa.

SQL Server 2005 - Backup

Backup Completi

È possibile eseguire il backup completo del database, comprensivo di parte del log delle transazioni, che ne consente il recupero. I backup completi del database rappresentano il database nel momento in cui il backup è stato completato. Il log delle transazioni incluso nel backup consente di riportare il database alle condizioni in cui si trovava nel momento in cui il backup è stato completato.

Quando si utilizza il modello di recupero con registrazione minima, dopo ogni backup, un'eventuale emergenza comporta il rischio di perdita dei dati. Il rischio di perdita dei dati cresce ad ogni aggiornamento fino al successivo backup dei dati, quando torna a zero e inizia un nuovo ciclo di esposizione al rischio.

Quando si utilizza il modello di recupero con registrazione completa, è necessario eseguire backup del log delle transazioni.

Per creare backup completo del database, eseguire dal linea di comando l'istruzione `BACKUP DATABASE` specificando:

- * Il nome del database di cui eseguire il backup.
- * La periferica di backup in cui archiviare il backup completo del database.

SQL Server 2005 - Backup

Backup Differenziali

Un backup differenziale del database registra solo i dati modificati dopo l'ultimo backup completo del database. I backup differenziali del database hanno dimensioni ridotte e sono più veloci rispetto ai backup completi del database. Semplificano pertanto i backup frequenti e riducono il rischio di perdita dei dati. Un backup differenziale del database si basa sul precedente backup completo del database.

Per creare il backup differenziale del database, eseguire da linea di comando l'istruzione `BACKUP DATABASE` specificando:

- * Il nome del database di cui eseguire il backup.
- * La periferica di backup in cui archiviare il backup completo del database.
- * La clausola `DIFFERENTIAL`, per specificare che viene eseguito il backup solo delle parti del database modificate dopo la creazione dell'ultimo backup completo.

In alternativa è possibile utilizzare il tool fornito da Microsoft seguendo la procedura che trovate a questo link:

<http://msdn2.microsoft.com/it-it/library/ms188248.aspx>

SQL Server 2005 - Manutenzione

I piani di manutenzione consentono di creare un flusso di lavoro delle attività di manutenzione necessarie per assicurare prestazioni ottimali del database, eseguire regolarmente il backup del database in caso di errori del sistema e verificare le inconsistenze. Sebbene sia possibile utilizzare Creazione guidata piano di manutenzione per creare le attività di manutenzione principali, la creazione manuale dei piani risulta molto più efficiente. In SQL Server 2005 Database Engine (Motore di database), i piani di manutenzione creano un processo che esegue automaticamente le attività di manutenzione a intervalli pianificati.

<http://msdn2.microsoft.com/it-it/library/ms189036.aspx>

SQL Server 2005 - Automazione

Microsoft mette a disposizione un tool chiamato Notification Services che permette l'invio di informazioni a utenti, il grande vantaggio è che vengono supportate svariate tipologie di dispositivi per la ricezione.

Esempi di scenari possono essere visionabili al seguente link:

<http://msdn2.microsoft.com/it-it/library/ms171255.aspx>

Bibliografia & Linkografia

MySQL:

- www.mysql.com

Oracle:

- www.oracle.com

SQL Server:

- MSDN Microsoft

<http://msdn2.microsoft.com/it-it/library/ms130214.aspx>