CREATE e DROP di colonne temporanee

#### di [Sergio Govoni](http://mvp.microsoft.com/en-us/mvp/Sergio%20Govoni-4029181) - Microsoft MVP

Blog: <http://www.ugiss.org/sgovoni/>

Twitter: [@segovoni](https://twitter.com/segovoni)

* 1. 

*Giugno, 2013*

Introduzione

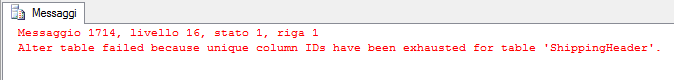
La vista di sistema [sys.columns](http://msdn.microsoft.com/en-us/library/ms176106.aspx), come citano i books on-line, restituisce una riga per ogni colonna contenuta negli oggetti di un database che possono contenere colonne come, ad esempio, viste o tabelle. Gli oggetti database in grado di contenere colonne sono elencati di seguito:

* Funzioni assembly con valori di tabella
* Funzioni in-line con valori di tabella
* Tabelle interne
* Tabelle di sistema
* Funzioni con valori di tabella
* Tabelle utente
* Viste

In particolare la colonna column\_id della vista sys.columns espone l’identificativo univoco (ID) assegnato ad ogni colonna presente all’interno di un oggetto, che da questo momento in poi ipotizziamo, per semplicità, essere una tabella.

Scenario

Un po’ di tempo fa, ho avuto l’occasione di occuparmi del problema che sta alla base del Messaggio di Errore 1714 illustrato nella figura seguente (su una istanza SQL Server 2005).

* 1. 
  2. Figura 1 – Messaggio di Errore 1714 (SQL Server 2005)

L’errore è stato riscontrato in una stored procedure che gestisce la generazione dei documenti di trasporto. L’ipotetico cliente che ha segnalato questo errore, emette ogni giorno migliaia di DdT; la segnalazione è pervenuta proprio durante la generazione di un nuovo documento. Per la memorizzazione delle testate dei documenti di trasporto, si utilizza la tabella dbo.ShippingHeader.

Messaggio di Errore 1714 (SQL Server 2005)

Il seguente frammento di codice T-SQL implementa la creazione (semplificata) della tabella dbo.ShippingHeader nel database di sistema tempdb. Per completezza vengono create anche le tabelle dbo.Product (anagrafica prodotti) e dbo.Customer (anagrafica clienti).

* 1. USE [tempdb];
  2. GO
  3. -- dbo.Product
  4. CREATE TABLE dbo.Product
  5. (
  6. ProductID VARCHAR(25) NOT NULL
  7. ,SafetyStockLevel SMALLINT NOT NULL
  8. ,Size VARCHAR(5) NULL
  9. ,ModifiedDate DATETIME NOT NULL DEFAULT GETDATE()
  10. ,Status BIT NOT NULL DEFAULT(1)
  11. ,CONSTRAINT PK\_Product PRIMARY KEY(ProductID)
  12. );
  13. GO
  14. -- dbo.Customer
  15. CREATE TABLE dbo.Customer
  16. (
  17. CustomerID INT IDENTITY(1, 1) NOT NULL PRIMARY KEY
  18. ,CustomerName VARCHAR(40) NOT NULL
  19. );
  20. GO
  21. -- dbo.ShippingHeader
  22. CREATE TABLE dbo.ShippingHeader
  23. (
  24. ShippingID INTEGER IDENTITY(1, 1) NOT NULL
  25. ,ProductID VARCHAR(25) NOT NULL
  26. FOREIGN KEY (ProductID)
  27. REFERENCES dbo.Product(ProductID)
  28. ,ShipDate DATETIME DEFAULT GETDATE() NOT NULL
  29. ,ShipNumber VARCHAR(20)
  30. ,CustomerID INT DEFAULT(1) NOT NULL
  31. FOREIGN KEY (CustomerID)
  32. REFERENCES dbo.Customer(CustomerID)
  33. ,ShipName VARCHAR(20) DEFAULT('Name')
  34. ,ShipAddress VARCHAR(40) DEFAULT('Address')
  35. ,ShipCity VARCHAR(20) DEFAULT('City')
  36. ,ShipPostalCode VARCHAR(20) DEFAULT('Postal code')
  37. ,ShipCountry VARCHAR(20) DEFAULT('Country')
  38. ,DeliveryDate DATETIME DEFAULT GETDATE()
  39. PRIMARY KEY(ShippingID)
  40. );
  41. GO

La tabella dbo.ShippingHeader è utilizzata per memorizzare i documenti di trasporto emessi dall’azienda. Su questa tabella, **per ogni nuovo documento**, viene **creata e distrutta la colonna temporanea** TestField, utilizzata per salvare alcune informazioni durante la generazione del DdT.

La stored procedure di generazione DdT eseguiva un frammento di codice T-SQL simile a quello riportato di seguito, dove nei tratti commentati c’era la logica di generazione del documento.

* 1. BEGIN
  2. -- ...
  3. -- Creazione colonna temporanea
  4. ALTER TABLE dbo.ShippingHeader ADD TestField INTEGER;
  5. -- ...
  6. -- ...
  7. -- ...
  8. -- Memorizzazione di un valore temporaneo
  9. UPDATE
  10. dbo.ShippingHeader
  11. SET
  12. TestField = @value
  13. WHERE
  14. (<condizione>)
  15. -- ...
  16. -- ...
  17. -- ...
  18. -- Utilizzo del valore memorizzato
  19. SELECT
  20. TestField
  21. FROM
  22. dbo.ShippingHeader
  23. WHERE
  24. (<condizione>)
  25. -- ...
  26. -- ...
  27. -- ...
  28. -- Eliminazione colonna temporanea
  29. ALTER TABLE dbo.ShippingHeader DROP COLUMN TestField
  30. -- ...
  31. -- ...
  32. -- ...
  33. END;
  34. GO

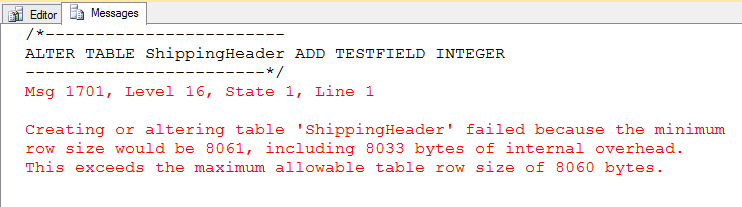
Dopo alcuni mesi di lavoro della stored procedure (in produzione), è stato raggiunto il limite massimo degli identificativi univoci (ID) assegnabili in fase di creazione di una nuova colonna.

**Non** era quindi più possibile **aggiungere colonne alla tabella** dbo.ShippingHeader. SQL Server restituiva il messaggio di errore indicato in precedenza perché il contatore colid della tabella di sistema sys.syscolpars **non poteva essere incrementato** in quanto raggiunto il limite massimo di valori rappresentabili.

La colonna colid, nella versione 2005 di SQL Server, è di tipo smallint e con questo tipo di dato si possono rappresentare positivamente 2^15 - 1 elementi, ossia 32.767… dopo qualche mese, il cliente aveva inserito più di 32.767 documenti di trasporto!

Messaggio di Errore 1701 (SQL Server 2012)

In SQL Server 2012, il tipo di dato della colonna column\_id della vista sys.columns è Integer, può quindi rappresentare positivamente 2^31 - 1 elementi, molti di più di quelli che potevano essere rappresentati in precedenza, ma la nostra stored procedure prima di raggiungere il limite massimo restituisce il messaggio di errore illustrato nella figura seguente.

* 1. 
  2. Figura 2 – Messaggio di Errore 1701 (SQL Server 2012)

Dopo aver modificato la stored procedure che implementa la logica di generazione dei documenti di trasporto, per non eseguire l’ADD e il DROP di una colonna temporanea ad ogni esecuzione, ci siamo subito posti il problema di come poter applicare futuri aggiornamenti (aggiunta di nuove colonne) alla tabella dbo.ShippingHeader.

Soluzione

La soluzione adottata consiste nel ricreare la tabella dbo.ShippingHeader. Ricreando la tabella, il contatore colid (column\_id nella vista sys.columns) verrà resettato. L’assegnazione dei prossimi identificativi univoci (ID) ripartirà dal valore successivo a quello estratto nella colonna MAX\_ColID nel seguente comando T-SQL:

* 1. USE [tempdb];
  2. GO
  3. SELECT
  4. MAX(c.column\_id) AS MAX\_ColID
  5. FROM
  6. sys.columns AS c
  7. WHERE
  8. (c.object\_id = object\_id('dbo.ShippingHeader'));
  9. GO

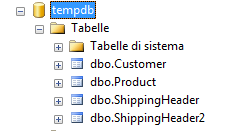
Per ricreare la tabella dbo.ShippingHeader possiamo eseguire, nell’ordine, le seguenti attività:

* Duplicazione della tabella (e copia dei dati)
* Eliminazione delle integrità referenziali definite su dbo.ShippingHeader
* Eliminazione della tabella dbo.ShippingHeader
* Rinomina (in dbo.ShippingHeader) della tabella precedentemente copiata
* Applicazione delle integrità referenziali

Duplicazione della tabella e copia dei dati

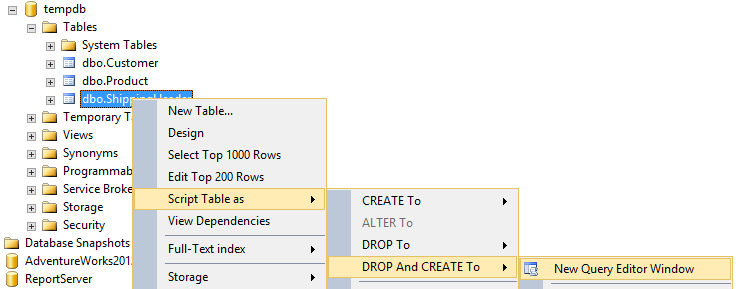
Per duplicare (copiando i dati) la tabella dbo.ShippingHeader abbiamo utilizzato il Tool di [Importazione/Esportazione guidata di SQL Server](http://msdn.microsoft.com/it-it/library/ms188032(v=sql.105).aspx).

Completata la procedura, il database conterrà una nuova tabella che in questo esempio abbiamo chiamato dbo.ShippingHeader2 come illustra la figura seguente.

* 1. 
  2. Figura 3 – Tabella dbo.ShippingHeader2 (copia dalle tabella dbo.ShippingHeader)

Eliminazione delle integrità referenziali definite su dbo.ShippingHeader

Per ottenere, in modo semplice e veloce, i comandi di eliminazione delle integrità referenziali definite sulla tabella dbo.ShippingHeader abbiamo utilizzato le funzioni di scripting di SQL Server, in particolare abbiamo utilizzato la funzione di generazione del codice per le istruzioni DROP e CREATE, come illustrato in figura 4.

* 1. 
  2. Figura 4 – Funzioni di scripting di SQL Server

Abbiamo quindi eseguito la cancellazione dei vincoli definiti sulla la tabella dbo.ShippingHeader, il seguente frammento di codice T-SQL riporta gli statement eseguiti:

* 1. USE [tempdb]
  2. GO
  3. ALTER TABLE [dbo].[ShippingHeader] DROP CONSTRAINT [FK\_\_ShippingH\_\_Produ\_\_1A14E395]
  4. GO
  5. ALTER TABLE [dbo].[ShippingHeader] DROP CONSTRAINT [FK\_\_ShippingH\_\_Custo\_\_1CF15040]
  6. GO
  7. ALTER TABLE [dbo].[ShippingHeader] DROP CONSTRAINT [DF\_\_ShippingH\_\_Deliv\_\_22AA2996]
  8. GO
  9. ALTER TABLE [dbo].[ShippingHeader] DROP CONSTRAINT [DF\_\_ShippingH\_\_ShipC\_\_21B6055D]
  10. GO
  11. ALTER TABLE [dbo].[ShippingHeader] DROP CONSTRAINT [DF\_\_ShippingH\_\_ShipP\_\_20C1E124]
  12. GO
  13. ALTER TABLE [dbo].[ShippingHeader] DROP CONSTRAINT [DF\_\_ShippingH\_\_ShipC\_\_1FCDBCEB]
  14. GO
  15. ALTER TABLE [dbo].[ShippingHeader] DROP CONSTRAINT [DF\_\_ShippingH\_\_ShipA\_\_1ED998B2]
  16. GO
  17. ALTER TABLE [dbo].[ShippingHeader] DROP CONSTRAINT [DF\_\_ShippingH\_\_ShipN\_\_1DE57479]
  18. GO
  19. ALTER TABLE [dbo].[ShippingHeader] DROP CONSTRAINT [DF\_\_ShippingH\_\_Custo\_\_1BFD2C07]
  20. GO
  21. ALTER TABLE [dbo].[ShippingHeader] DROP CONSTRAINT [DF\_\_ShippingH\_\_ShipD\_\_1B0907CE]
  22. GO

Eliminazione della tabella dbo.ShippingHeader

Dopo aver eliminato i vincoli (integrità referenziali e di dominio), è stato possibile eliminare la tabella dbo.ShippingHeader.

* 1. USE [tempdb]
  2. GO
  3. DROP TABLE [dbo].[ShippingHeader]
  4. GO

Rinomina (in dbo.ShippingHeader) della tabella precedentemente copiata

Per rinominare la tabella dbo.ShippingHeader2 in dbo.ShippingHeader, abbiamo utilizzato la stored procedure di sistema [sp\_rename](http://msdn.microsoft.com/en-us/library/ms188351.aspx) che consente di modificare il nome di un oggetto creato dall’utente, nel database corrente. Il seguente comando T-SQL illustra l’utilizzo di sp\_rename:

* 1. USE [tempdb];
  2. GO
  3. EXEC sp\_rename
  4. @objname = 'dbo.ShippingHeader2'
  5. ,@newname = 'ShippingHeader';
  6. GO

Applicazione delle integrità referenziali

L’ultimo step consiste nel ripristinare la PRIMARY KEY e i vincoli della tabella dbo.ShippingHeader. I comandi di creazione della PRIMARY KEY e delle integrità sulla tabella dbo.ShippingHeader rigenerata sono stati precedentemente generati (nello step 2), vengono ora utilizzati come illustrato di seguito:

* 1. USE [tempdb]
  2. GO
  3. ALTER TABLE [dbo].[ShippingHeader] ADD PRIMARY KEY([ShippingID])
  4. GO
  5. ALTER TABLE [dbo].[ShippingHeader] ADD DEFAULT (getdate()) FOR [ShipDate]
  6. GO
  7. ALTER TABLE [dbo].[ShippingHeader] ADD DEFAULT ((1)) FOR [CustomerID]
  8. GO
  9. ALTER TABLE [dbo].[ShippingHeader] ADD DEFAULT ('Name') FOR [ShipName]
  10. GO
  11. ALTER TABLE [dbo].[ShippingHeader] ADD DEFAULT ('Address') FOR [ShipAddress]
  12. GO
  13. ALTER TABLE [dbo].[ShippingHeader] ADD DEFAULT ('City') FOR [ShipCity]
  14. GO
  15. ALTER TABLE [dbo].[ShippingHeader] ADD DEFAULT ('Postal code') FOR [ShipPostalCode]
  16. GO
  17. ALTER TABLE [dbo].[ShippingHeader] ADD DEFAULT ('Country') FOR [ShipCountry]
  18. GO
  19. ALTER TABLE [dbo].[ShippingHeader] ADD DEFAULT (getdate()) FOR [DeliveryDate]
  20. GO
  21. ALTER TABLE [dbo].[ShippingHeader] WITH CHECK ADD FOREIGN KEY([CustomerID])
  22. REFERENCES [dbo].[Customer] ([CustomerID])
  23. GO
  24. ALTER TABLE [dbo].[ShippingHeader] WITH CHECK ADD FOREIGN KEY([ProductID])
  25. REFERENCES [dbo].[Product] ([ProductID])
  26. GO

Conclusioni

L’utilizzo **ciclico** di una **colonna temporanea creata** e **distrutta** su una tabella, ad esempio, per ogni esecuzione di una stored procedure o in generale per ogni esecuzione di una determinata elaborazione **non è una buona pratica di programmazione**.

Lo scenario descritto rappresenta una semplificazione di un caso reale, non abbiamo trattato la rigenerazione di eventuali indici definiti nella tabella dbo.ShippingHeader.

Prima di eseguire la procedura descritta in un ambiente di produzione è necessario averla provata in modo completo su un backup del DB in ambiente di test.

#### di [Sergio Govoni](http://mvp.microsoft.com/en-us/mvp/Sergio%20Govoni-4029181) - Microsoft MVP

Blog: <http://www.ugiss.org/sgovoni/>

Twitter: [@segovoni](https://twitter.com/segovoni)