

CÁC CHUYÊN GIA GIỌNG NÓI<sup>®</sup> TRONG SQL MÁY CHỦ

# Xây dựng Một Dữ liệu Kho

Với Ví dụ TRONG SQL Máy chủ

Vincent Rainardi

apress<sup>®</sup>



# Data Extraction

**Bây** giờ chúng ta đã tạo các bảng trong NDS và DDS, đã đến lúc điền thông tin vào chúng. Trước chúng tôi Có thể cư trú họ, mặc dù, chúng tôi nhu cầu ĐẾN lấy lại cái dữ liệu từ cái nguồn hệ thống.

Cái này là Gì dữ liệu chiết xuất là tất cả Về.

Đầu tiên Ôm bàn luận cái tổng quan nguyên tắc của dữ liệu chiết xuất, khác biệt các loại của nguồn hệ thống và một số kỹ thuật trích xuất dữ liệu. Sau đó tôi sẽ giới thiệu công cụ mà chúng ta sẽ sử dụng, SQL Máy chủ Tích hợp Dịch vụ. Sau đó giới thiệu cái dụng cụ, Ôm sau đó trình diễn Làm sao ĐẾN sử dụng Nó để trích xuất cái dữ liệu từ cái nguồn hệ thống TRONG của chúng tôi trường hợp học: Ngọc bích, Tháp Web, Và Sao Mộc.

## Giới thiệu ĐẾN ETL

**ETL là viết tắt của Extract, Biến đổi và Tải.** Đây là quá trình truy xuất và chuyển đổi dữ liệu từ hệ thống nguồn và đưa vào kho dữ liệu. Quá trình này đã diễn ra trong nhiều thập kỷ và đã phát triển và cải thiện rất nhiều kể từ khi ra đời.

Ở đó là một số nền tảng nguyên tắc ĐẾN hiểu khi trích xuất dữ liệu từ Một nguồn hệ thống nhằm mục đích lấp đầy một kho dữ liệu. Đầu tiên, khối lượng dữ liệu được truy xuất là lớn, có thể là hàng trăm megabyte hoặc hàng chục gigabyte. Một hệ thống OLTP được thiết kế sao cho dữ liệu là đã lấy lại TRONG nhỏ bé miếng, không TRONG lớn số lượng giống cái này, Vì thế Bạn có ĐẾN là cần thận không làm chậm hệ thống nguồn quá nhiều. Chúng tôi muốn việc trích xuất diễn ra nhanh nhất có thể, chẳng hạn như năm phút nếu như chúng tôi Có thể, không ba giờ (cái này phụ thuộc TRÊN cái chiết xuất phương pháp, cái mà Tôi sẽ che sau đó TRONG cái này chương). Chúng tôi Mà còn muốn Nó ĐẾN là BẢNG bé nhỏ BẢNG khả thi, như là BẢNG 10MB mỗi ngày nếu chúng ta Có thể, không 1GB mỗi ngày. TRONG phép cộng, chúng tôi muốn Nó ĐẾN là BẢNG không thường xuyên BẢNG khả thi, như là BẢNG một lần một ngày nếu như chúng tôi Có thể, không mọi năm phút. Chúng tôi muốn cái thay đổi TRONG cái nguồn hệ thống ĐẾN là BẢNG tối thiểu BẢNG khả thi, như là BẢNG KHÔNG thay đổi Tại tất cả nếu như chúng tôi Có thể, không tạo ra kích hoạt vì chụp ảnh dữ liệu thay đổi trong từng bảng.

Đoạn trước nói về một nguyên tắc cơ bản trong việc trích xuất dữ liệu. Nếu có là điều duy nhất bạn sẽ rút ra được từ chương này, tôi hy vọng đó là: khi trích xuất dữ liệu từ Một nguồn hệ thống, Bạn có ĐẾN là cần thận không ĐẾN quấy rầy cái nguồn hệ thống cũng vậy nhiều.

Sau khi trích xuất dữ liệu, chúng tôi muốn đưa dữ liệu vào kho dữ liệu càng sớm càng tốt, lý tưởng nhất là ngay lập tức, mà không cần chạm vào đĩa (tức là không lưu trữ tạm thời). trong cơ sở dữ liệu hoặc trong các tập tin). Chúng ta cần áp dụng một số phép biến đổi cho dữ liệu từ hệ thống nguồn để phù hợp với định dạng và cấu trúc của dữ liệu trong NDS và DDS.

Đôi khi phép biến đổi dữ liệu chỉ là định dạng và chuẩn hóa, chuyển đổi sang một số hoặc định dạng ngày nhất định, cắt bớt khoảng trắng theo sau hoặc số không đứng đầu và tuân thủ theo các chuẩn. Khác hẳn cái sự sửa đổi là Một tra cứu, như là BẢNG dịch thuật khách hàng trạng thái 2 để Hoạt động hoặc dịch danh mục sản phẩm “Nhạc Pop” thành 54. Một chuyển đổi khác là



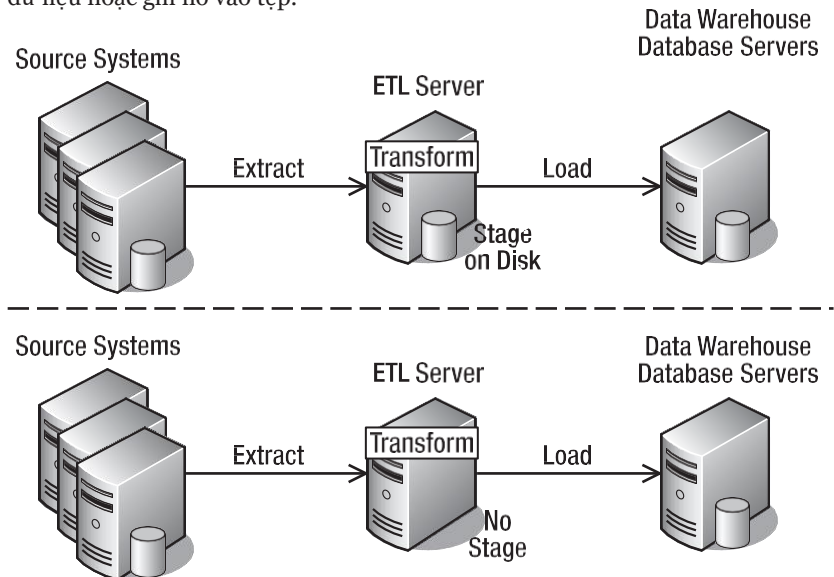
thường xuyên đã sử dụng TRONG dữ liệu kho bãi là tổng hợp, cái mà có nghĩa tóm tắt dữ liệu ở mức cao hơn mức độ.

Chúng tôi Mà còn muốn cái dữ liệu chúng tôi đặt vào trong cái kho ĐẾN là lau dọn Và của Tốt chất lượng. Ví dụ, chúng tôi không muốn số điện thoại không hợp lệ, địa chỉ email không có ký tự @ trong Nó, Một sản phẩm mã số cái đó làm không hiện hữu, Một DVD với Một dung tích của 54GB, Một Địa chỉ với Một thành phố của Amsterdam Nhưng Một tình trạng của California, hoặc Một đơn vị giá của \$0 vì Một cụ thể âm thanh sách. Đối với điều này mục đích, chúng tôi nhu cầu ĐẾN LÂM nhiều kiểm tra trước đặt cái dữ liệu vào trong cái kho.

Hai nguyên tắc quan trọng khác là *rò rỉ* và *khả năng phục hồi*. Rò rỉ xảy ra khi quá trình ETL nghĩ rằng nó đã tải xuống toàn bộ dữ liệu từ hệ thống nguồn nhưng trên thực tế đã bỏ sót một số bản ghi. Một quy trình ETL tốt không nên có bất kỳ rò rỉ nào. Khả năng phục hồi có nghĩa là quy trình ETL phải mạnh mẽ để trong trường hợp xảy ra lỗi, nó có thể phục hồi mà không bị mất hoặc hư hỏng dữ liệu. Tôi sẽ thảo luận tất cả những điều này trong chương này.

## ETL Các cách tiếp cận Và Ngành kiến trúc

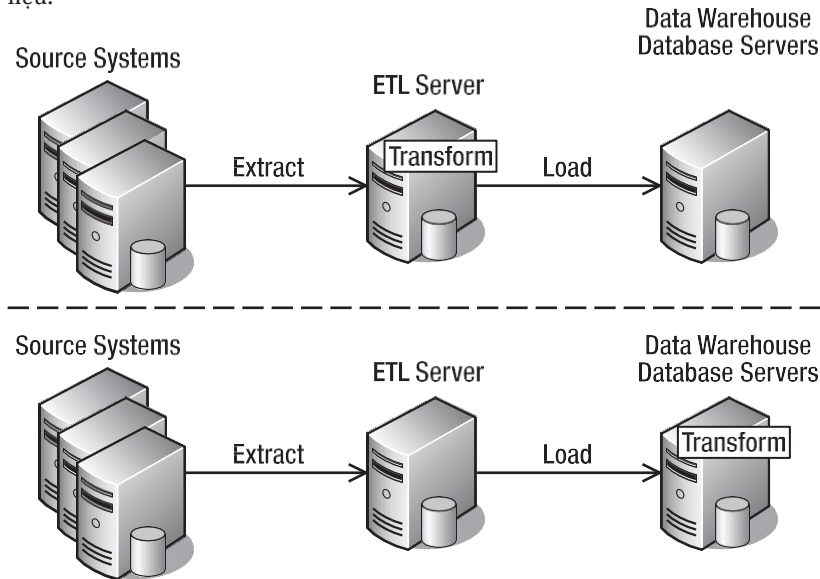
Ở đó là một số **cách tiếp cận của thực hiện ETL**. Một truyền thống tiếp cận là ĐẾN sự lôi kéo dữ liệu từ các hệ thống nguồn, đưa nó vào vùng dàn dựng, sau đó chuyển đổi và tải nó vào cái kho, BẢNG mỗi cái đứng đầu sơ đồ của Nhân vật 7-1. Ngoài ra, thay vì của đặt dữ liệu TRONG Một dàn dựng khu vực, Thỉnh thoảng cái ETL máy chủ làm cái sự biến đổi TRONG ký ức và sau đó cập nhật kho dữ liệu trực tiếp (không có giai đoạn dàn dựng), như được hiển thị trong sơ đồ dưới cùng của Hình 7-1. Khu vực dàn dựng là một cơ sở dữ liệu vật lý hoặc các tập tin. Đưa dữ liệu vào khu vực dàn dựng khu vực có nghĩa là chèn nó vào cơ sở dữ liệu hoặc ghi nó vào tệp.



**Nhân vật 7-1.** ĐẾN sân khấu TRÊN đĩa hoặc LÂM sự biến đổi TRONG ký ức

Biến đổi cái dữ liệu TRONG ký ức là nhanh hơn hơn đặt Nó TRÊN đĩa Đầu tiên. Nếu như cái dữ liệu đủ nhỏ, bạn có thể chuyển đổi trong bộ nhớ, nhưng nếu dữ liệu lớn, bạn cần phải đưa nó vào đĩa Đầu tiên. Việc bạn có thể đưa dữ liệu vào bộ nhớ hay không phụ thuộc vào dung lượng bộ nhớ của ETL máy chủ có.

Giải pháp thay thế cho hai phương pháp ETL được thể hiện trong Hình 7-1 được gọi là Trích xuất, Tải và Chuyển đổi (ELT), như thể hiện ở nửa dưới của Hình 7-2. Trong phương pháp ELT, chúng tôi lấy dữ liệu từ các hệ thống nguồn, tải dữ liệu vào kho dữ liệu, sau đó áp dụng phép chuyển đổi qua đang cập nhật cái dữ liệu TRONG cái kho. TRONG cái Tiếng Anh giao tiếp tiếp cận, Thiết yếu chúng tôi sao chép dữ liệu hệ thống nguồn (OLTP) vào kho dữ liệu và chuyển đổi nó ở đó. Mọi người thường áp dụng phương pháp ETL nếu họ có máy chủ ETL mạnh và phần mềm mạnh mẽ, giàu tất cả các loại quy trình chuyển đổi và chất lượng dữ liệu.



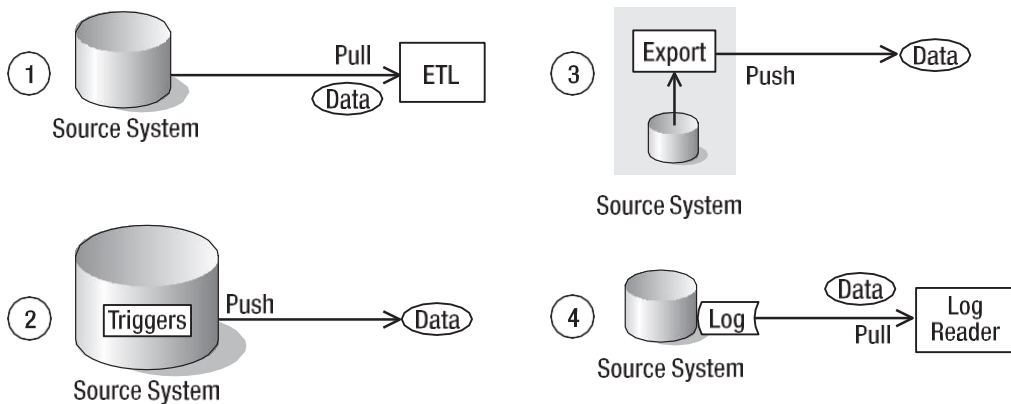
**Hình 7-2.** ETL Và Tiếng Anh: sự lựa chọn của Ở đâu ĐẾN trình diễn cái sự biến đổi

Mọi người thường áp dụng phương pháp ELT nếu họ có hệ thống cơ sở dữ liệu kho dữ liệu mạnh, thường là hệ thống cơ sở dữ liệu MPP. Xử lý song song hàng loạt (MPP) là một nhóm máy chủ (gọi là nút) và mỗi nút có bộ nhớ, bộ xử lý và đĩa riêng. Ví dụ về hệ thống cơ sở dữ liệu MPP là Teradata, Netezza và Neoview. Khi bạn nhóm hai hoặc nhiều Máy chủ SQL, bạn sẽ có tính khả dụng cao và mỗi nút có bộ nhớ và bộ xử lý riêng, nhưng bạn vẫn chia sẻ các đĩa. Trong hệ thống cơ sở dữ liệu MPP, mỗi nút có bộ nhớ, bộ xử lý và đĩa riêng. Nó được gọi là kiến trúc *không chia sẻ gì*. Hệ thống cơ sở dữ liệu MPP mạnh hơn các hệ thống có đĩa dùng chung vì dữ liệu tải diễn ra song song trên nhiều nút có đĩa riêng.

Các chủ yếu lợi thế của MPP cơ sở dữ liệu hệ thống là cái đó cái hiệu suất tăng là tuyến tính. Nếu Bạn đặt mười SQL Máy chủ TRONG MỘT hoạt động-hoạt động cụm, cái hiệu suất tăng lên, Nhưng không 10 Máy chủ SQL đơn. Trong hệ thống cơ sở dữ liệu MPP, nếu bạn đưa vào mười nút, hiệu suất gần như mười lần cái hiệu suất của Một đơn nút.

Xét về việc *ai* di chuyển dữ liệu ra khỏi hệ thống nguồn, chúng ta có thể phân loại các phương pháp ETL thành bốn cách tiếp cận (xem Hình 7-3):

- Một ETL quá trình kéo cái dữ liệu ngoài qua truy vấn cái nguồn hệ thống cơ sở dữ liệu thường xuyên. Điều này là cái hầu hết chung tiếp cận. Các ETL kết nối ĐẾN cái nguồn hệ thống cơ sở dữ liệu, truy vấn dữ liệu và đưa dữ liệu ra.
- Các kích hoạt trong cơ sở dữ liệu hệ thống nguồn đẩy dữ liệu thay đổi ra ngoài. Một kích hoạt cơ sở dữ liệu là Một bộ sưu tập của SQL các tuyên bố cái đó thực hiện mọi thời gian ở đó là Một chèn, cập nhật, hoặc xóa TRÊN Một bàn. Qua sử dụng kích hoạt, chúng tôi Có thể cửa hàng cái đã thay đổi hàng TRONG khác bàn.
- Một quy trình theo lịch trình trong hệ thống nguồn sẽ xuất dữ liệu thường xuyên. Điều này tương tự như cách tiếp cận đầu tiên, nhưng chương trình truy vấn cơ sở dữ liệu không phải là chương trình ETL bên ngoài. Thay vào đó, nó là chương trình xuất khẩu *nội bộ* chạy trên máy chủ hệ thống nguồn.
- Trình đọc nhật ký đọc các tệp nhật ký cơ sở dữ liệu để xác định các thay đổi dữ liệu. Tệp nhật ký cơ sở dữ liệu chứa bản ghi các giao dịch được thực hiện với cơ sở dữ liệu đó. Trình đọc nhật ký là chương trình hiểu định dạng dữ liệu trong tệp nhật ký. Nó đọc các tệp nhật ký, lấy dữ liệu ra và lưu trữ dữ liệu ở nơi khác.



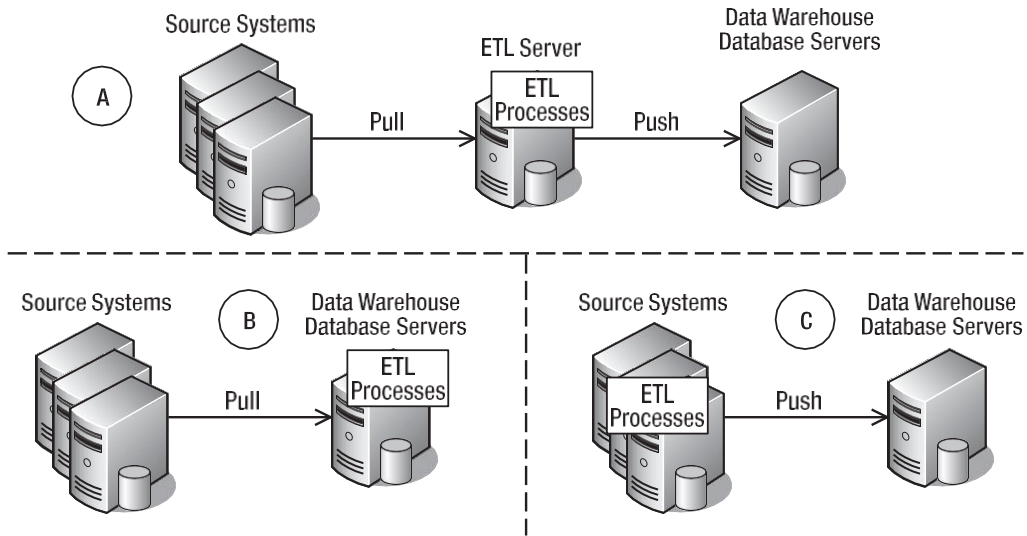
**Nhân vật 7-3.** Bốn ETL cách tiếp cận dựa trên TRÊN Ai di chuyển cái dữ liệu ngoài của cái nguồn hệ thống

Xét về *vị trí* thực hiện các quy trình di chuyển dữ liệu ra ngoài, chúng ta có thể phân loại ETL thành ba phương pháp (xem Hình 7-4):

- Thực hiện các quy trình ETL trong một máy chủ ETL riêng biệt nằm giữa hệ thống nguồn và máy chủ kho dữ liệu. Cách tiếp cận này cung cấp hiệu suất cao nhất. ETL chạy trên máy chủ riêng của nó, do đó nó không sử dụng tài nguyên của kho dữ liệu máy chủ hoặc máy chủ hệ thống nguồn. Nó đắt hơn hai lựa chọn tiếp theo vì bạn phải mua thêm giấy phép máy chủ và phần mềm.
- Thực hiện cái ETL các quá trình TRONG cái dữ liệu kho máy chủ. Cái này tiếp cận Có thể là đã sử dụng nếu chúng ta có dự phòng dung tích TRONG cái dữ liệu kho máy chủ hoặc nếu như chúng tôi có Một thời gian chỗ khi kho dữ liệu không được sử dụng (ví dụ như vào ban đêm). Nó rẻ hơn phương pháp đầu tiên vì chúng tôi không cần cung cấp thêm máy chủ.



- Thực hiện các quy trình ETL trong máy chủ lưu trữ hệ thống nguồn. Phương pháp này được triển khai khi chúng ta cần kho dữ liệu thời gian thực. Nói cách khác, thời điểm dữ liệu trong hệ thống nguồn thay đổi, sự thay đổi được truyền đến kho dữ liệu. Điều này có thể đạt được bằng cách sử dụng các kích hoạt cơ sở dữ liệu trong hệ thống nguồn. Trong thực tế, cách tiếp cận này là đã thực hiện TRONG sự liên kết với hoặc của cái trước phương pháp tiếp cận. Nghĩa là, phương pháp này chỉ được sử dụng cho một vài bảng và các bảng còn lại sẽ được điền bằng hai phương pháp đầu tiên.



**Nhân vật 7-4.** Lựa chọn của Ở đâu ĐẾN đặt ETL các quá trình

## Tổng quan Những cân nhắc

Các hệ thống chúng tôi là trích xuất từ có thể không là Một cơ sở dữ liệu. Nó có thể là Một tài liệu hệ thống, Một hàng đợi, dịch vụ hoặc email. Nếu dữ liệu nằm trong cơ sở dữ liệu, thông thường chúng ta sẽ truy xuất dữ liệu bằng ADO.NET, ODBC, OLEDB, JDBC hoặc kết nối cơ sở dữ liệu độc quyền. Ngày nay, hầu hết các cơ sở dữ liệu đều là quan hệ, nhưng đôi khi chúng ta bắt gặp các cơ sở dữ liệu phân cấp, chẳng hạn như Adabas và ISM hoặc tuần tự tài liệu kho, như là BẢNG ĐẠI HỌC ISAM. ĐẾN trích xuất cái dữ liệu từ họ, chúng tôi nhu cầu ĐẾN có bên phải cơ sở dữ liệu tài xế hoặc viết Một dữ liệu xuất khẩu kịch bản sử dụng cái cơ sở dữ liệu cụ thể ngôn ngữ.

Nếu như cái dữ liệu là TRONG Một tài liệu hoặc tập tin, Nó có thể là có cấu trúc, bán cấu trúc, hoặc không có cấu trúc.

Một có cấu trúc tài liệu là giống cái này:

NHÂN DẠNG	Ngày	Cửa hàng	Sản phẩm	Số lượng
2893	2/1/08	32	A160	150
2897	2/4/08	29	B120	240

là một tệp có vị trí cố định, nghĩa là các trường nằm ở các vị trí cột nhất định. Hoặc có thể như thế này:



*ID|Ngày| Cửa hàng| Sản phẩm| Số lượng*

*2893| 2/1/08| 32| A160| 150*

*2897| 2/4/08| 29| B120| 240*

cái mà là Một được phân định tài liệu. TRONG cái này trường hợp, Nó là đường ống được phân định.

Tập cấu trúc chứa dữ liệu dạng bảng (bảng), nghĩa là dữ liệu về cơ bản ở định dạng cột và hàng. Tập bản cấu trúc chứa dữ liệu dạng bảng và dữ liệu không phải dạng bảng. Một tập bản cấu trúc trông giống như tập XML này:

```
<trật tự Mã số: 2893">
  <ngày>2/1/08</ngày>
  <cửa hàng>32</cửa hàng>
  <sản phẩm>A160</sản phẩm>
  <số lượng>150</số lượng>
</Đặt hàng>
```

```
<trật tự Mã số: 2897">
  <ngày>2/4/08</ngày>
  <cửa hàng>29</cửa hàng>
  <sản phẩm>B120</sản phẩm>
  <số lượng>240</số lượng>
</Đặt hàng>
```

```
<khách hàng Mã số: 83
  <tên>John Smith</tên>
  <email>jsmith@aol.com</email>
</Đặt hàng>
```

Hai phần đầu tiên (dữ liệu đơn hàng) ở dạng bảng, nhưng phần thứ ba (dữ liệu khách hàng) không ở dạng bảng.

Dữ liệu phi cấu trúc như sau: “Vào ngày 2/1/2008, chúng tôi nhận được đơn hàng có mã số 2893 từ cửa hàng 32, yêu cầu 150 đơn vị sản phẩm A160. Ba ngày sau, cửa hàng 29 yêu cầu 240 đơn vị sản phẩm B120.” Đây là nội dung điển hình của email. Chúng tôi trích xuất thông tin trong dữ liệu phi cấu trúc bằng cách sử dụng khai thác văn bản.

Thỉnh thoảng, cái dữ liệu chiết xuất Có thể là xong chỉ một trong lúc chắc chắn lần, như là BẢNG sau đó một đợt hoặc không thể thực hiện vào một số thời điểm nhất định, chẳng hạn như vào thời điểm sao lưu hoặc trong quá trình xử lý cuối tháng. Thỉnh thoảng, nếu như chúng tôi là may mắn, chúng tôi Có thể truy cập cái nguồn dữ liệu Tại bất kì thời gian, như là như khi nào ở đó là Một sơ trung chỉ đọc máy chủ cụ thể cung cấp vì quảng cáo học truy vấn Và Bảng ETL 7-1 danh sách cái tiềm năng vấn đề TRONG dữ liệu chiết xuất, cái các kịch bản, Và cái chung kỹ thuật trong những tình huống đó.

**Bàn 7-1. Tiềm năng Vấn đề TRONG Dữ liệu Chiết xuất**

Tiềm năng	Tình huống vấn đề	Chung Kỹ thuật ĐẾN khác phục
Dữ liệu chỉ có sẵn tại gian với một số lần.	Lịch trình lô dài chạy qua đêm theo sau là một OLTP DBA lớn sao lưu cơ sở dữ liệu. Trích xuất trích liệu nhỏ hơn trong ngày sẽ được chia thành nhiều phần. chậm cái Giao dịch trực tuyến (OLTP)	Đàm phán một khoảng thời . Phá vỡ xuất vào trong dữ liệu nhỏ hơn trong ngày sẽ được chia thành nhiều phần. chậm cái Giao dịch trực tuyến (OLTP)
Cơ sở dữ liệu OLTP không phải là ngày đó. có sẵn vào những ngày nhất tháng	Vào ngày đầu tiên của mỗi tháng, Nói người sử dụng cái lý do vì cái tron ngày.	Bỏ qua việc trích xuất vào OLTP chạy một đợt cuối giới hạn.
OLTP DBA quan tâm dữ liệu ĐẾN đưa cho “đọc- ngoài hỏi cái OLTP nội bộ.	Nhóm dự án bao gồm ETL của chúng tôi có thể “làm hỏng” chỉ là” truy cập. Hỏi cái OLTP hệ thống của họ . Quản trị viên cơ sở dữ liệu ĐẾN xuất khẩu dữ liệu vì Bạn.	Hỏi cái Quản trị viên cơ sở các nhà tư vấn bên chỉ cần học
Có một mối lo ngại rằng ETL sẽ cấp/quá tải OLTP dụng Một ít cơ sở dữ liệu. cho lớn	Không có “thời gian yên tĩnh” vì OLTP là toàn cầu cơ sở dữ liệu.	Đọc từ sự căng thẳng thứ OLTP cơ sở dữ liệu. Sử trích xuất thường xuyên
OLTP DBA lo ngại dữ liệu họ có thể ter- yêu cầu máy chủ trong trường hợp khẩn cấp. chạy lại	Có một tình huống khẩn cấp rằng họ không thể ngăn chặn ETL pro- được chỉ định Bạn Tại bất kì thời khởi động lại.	bảng, như là BẢNG chỉ một Chủ Nhật. Kế cái Quản trị viên cơ sở tion TRONG OLTP cái đó gian (nói với các cesses chúng như thế nào) nhưng sau đó (thủ công gọi).
Họ không thể bắt đầu ETL các quá trình.	Cái ETL ĐẾN chạy lại tràn ngập.	Có là Một OLTP chương trình sao lưu tự động sau đó.

Một của cái hầu hết quan trọng đồ đặc Tại cái này sân khấu của cái dự án (các bắt đầu của ETL) là để có được ĐẾN cái dữ liệu, cái đó là, ĐẾN là có thể ĐẾN kết nối ĐẾN cái nguồn dữ liệu. Thỉnh thoảng Nó mất thời gian ĐẾN ở trong Một chức vụ Ở đâu chúng tôi Có thể truy vấn cái nguồn hệ thống (nếu như Nó là Một cơ sở dữ liệu) hoặc đọc cái tập tin (nếu như Nó một hệ thống tập tin). Đôi khi máy chủ cơ sở dữ liệu nguồn được đặt ở một lục địa khác và ở đó là KHÔNG kết nối ĐẾN cái đó máy chủ. Tôi đã đã làm việc TRÊN một số dự án Ở đâu chúng tôi có sắp xếp vì cái mạng kết nối ĐẾN là đã mở, cái đó là, ĐẾN cho phép chắc chắn Giao thức TCP giao thông ĐẾN chảy qua cái tường lửa Vì thế cái đó chúng tôi có thể lấy ĐẾN cái dữ liệu. Bàn 7-2 danh sách cái chung lý do vì không thể tiếp cận được dữ liệu nguồn cũng như các giải pháp chung của họ.

**Bàn 7-2. Lý do vì Không Hiện tại Có thể ĐẾN Lấy cái Nguồn Dữ liệu**

Lý do chung	Giải pháp
Ở đó là KHÔNG kết nối ĐẾN cái mục tiêu cơ sở dữ liệu, có lẽ bởi vì của KHÔNG lộ trình hoặc cái giao thông chỉ, cái đích đến bị chặn TRÊN cái tường lửa.	Hỏi cái mạng kỹ sư ĐẾN mở cái tường Cung cấp cái nguồn Địa chỉ IP Địa chỉ, Và cái cảng con số ĐẾN mở. Sử dụng cái Đã đăng ký Địa chỉ IP Địa chỉ nếu như yêu cầu.
Cơ sở dữ liệu đích yêu cầu trình điều khiển cụ thể. nó trên Chúng tôi không thể sử dụng ODBC, ADO.NET hoặc OLEDB.	Đạt được hoặc mua trình điều khiển và cài đặt máy chủ ETL.

Các mục tiêu máy chủ tên là không được công nhận. Hãy thử Nó với Địa chỉ IP Đầu tiên. Nếu như cái này tác phẩm, hỏi cái mạng

đội ĐẾN thêm vào cái tên TRÊN cái Tên miền.  
Một số cơ sở dữ liệu hệ thống như là BẢNG  
Dữ liệu Teradata thêm vào Một hậu tố đến  
tên.

*Tiếp tục*

**Bàn 7-2. Tiếp tục**

Lý	do chung Giải pháp
Chúng ta có thể truy cập vào cơ sở dữ liệu nhưng không thể truy vấn viên cơ sở dữ liệu ĐẾN tạo nên hai tài khoản: a vì chúng tôi không có quyền truy cập vào các bảng hoặc chế độ xem.	Hỏi cái OLTP Quản trị tài khoản cá nhân để phát triển và a tài khoản chức năng cho sản xuất. Luôn yêu cầu quyền truy cập chỉ đọc.
Phiên bản cơ sở dữ liệu của OLTP mới hơn điều khiển. trình điều khiển trên máy chủ ETL.	Upgrade cái trình
RDBMS yêu cầu một ngôn ngữ hoặc công cụ cụ thể, như Progress 4GL. hoặc yêu cầu	Hoặc lấy phiên bản ODBC của trình điều khiển (chấp nhận các truy vấn SQL thông thường)  Quản trị viên cơ sở dữ liệu ĐẾN viết Một kịch bản ĐẾN xuất khẩu cái dữ liệu.

Trong ba phần tiếp theo, tôi sẽ thảo luận về cách trích xuất dữ liệu từ cơ sở dữ liệu, hệ thống tệp và các loại nguồn khác.

## Trích xuất Quan hệ Cơ sở dữ liệu

Sau khi kết nối với dữ liệu nguồn, chúng ta có thể trích xuất dữ liệu. Khi trích xuất dữ liệu từ cơ sở dữ liệu quan hệ bao gồm các bảng, chúng ta có thể sử dụng một trong bốn phương pháp chính:

- Trộn bàn mọi thời gian
- Tăng dần trích xuất
- Đã sửa phạm vi
- Đẩy tiếp cận

### Trộn Bàn Mọi Thời gian

Chúng tôi sử dụng cái trộn bàn mọi thời gian khi cái bàn là bé nhỏ, như là BẢNG với Một bàn với ba số nguyên hoặc varchar(10) cột gồm có của Một một vài hàng. MỘT hơn chung lý do là bởi vì có KHÔNG dấu thời gian hoặc danh tính cột cái đó chúng tôi Có thể sử dụng vì biết cái mà hàng đã từng được cập nhật kể từ cái cuối cùng trích xuất. Vì ví dụ, Một tổng quan giải mã bàn giống Bàn 7-3 có thể là được trích xuất bằng phương pháp “toàn bộ bảng mỗi lần”.

**Bàn 7-3. MỘT Tổng quan Giải mã Bàn**

Bàn	Mã số	Sự miêu tả
PMT	1	Trực tiếp ghi nợ
PMT	2	Hàng tháng hóa đơn
PMT	3	Hàng năm TRONG nâng cao
STS	Máy chủ	Tích cực
STS	SU	Cấm

182

CHAPTER 7 • DATA EXTRACTION

STS	TỰ NHIÊN	Sự cân bằng nổi bật
ĐĂNG KÝ	S	Đã đăng ký
ĐĂNG KÝ	Bạn	Đã hủy đăng ký
...		

---

Cái này mã-giải mã bàn là đã sử dụng khắp cái hệ thống. Hơn là hơn tạo ra Một bảng riêng biệt vì cái sự chi trả mã số, khách hàng trạng thái, đăng ký trạng thái, Và Vì thế TRÊN, một số hệ thống kinh doanh sử dụng Một chung bàn, cái đó là, Một đơn giải mã bàn phục vụ cái trọn hệ thống. Tất nhiên rồi, cái này là ít hơn linh hoạt, Nhưng Tôi đã đã xem cái này đã thực hiện TRONG Một nguồn hệ thống. Bởi vì đã có KHÔNG dấu thời gian, KHÔNG giao dịch ngày (của nó không Một giao dịch bàn), Và KHÔNG danh tính cột hoặc, ở đó đã từng là KHÔNG đường chúng tôi có thể có thể tải về cái dữ liệu tăng dần, bởi vì chúng tôi không thể nhận dạng cái mà hàng đã từng mới, đã cập nhật, hoặc đã xóa. May mắn thay, cái bàn đã từng là chỉ một khoảng 15.000 hàng Và lấy đi Về Một phút ĐẾN trích xuất ĐẾN cái sân khấu.

TRONG Cửa tôi kinh nghiệm, Một bàn cái đó chứa 1.000 ĐẾN 5.000 hàng (nói 10 ĐẾN 30 cột, hoặc lên đến 500 byte TRONG chiều rộng) TRÊN Một 100Mbps Mạng LAN thông thường mất một ĐẾN năm giây ĐẾN tải xuống từ DB/2 trên AS/400 bằng trình điều khiển iSeries OLE DB hoặc từ Informix trên Unix bằng trình điều khiển OLE DB gốc của Informix vào bảng phân đoạn không có chỉ mục trong SQL Server 2005 hoặc 2000 trên Một Cửa sổ 2000 hoặc 2003 máy chủ, được cấu hình với Đốt kích 5 vì Gổ MDF Và Đốt kích 1 vì LDF sử dụng đĩa 15.000 RPM, chạy trên Dell 2850 với bộ nhớ 8MB, được thực hiện bằng SSIS hoặc DTS, không có chuyển đổi hoặc tập lệnh (chỉ là ánh xạ cột sang cột đơn thuần với dữ liệu con- phiên bản). Tôi không thể cung cấp cho bạn công thức tính thời gian cần thiết trong môi trường của bạn vì các cấu hình phần cứng và phần mềm khác nhau có kết quả khác nhau, nhưng tôi có thể hướng dẫn bạn: bạn có thể ước tính thời gian thực hiện trong môi trường sản xuất của mình bằng cách đo thời gian chạy chương trình ETL trên môi trường phát triển hoặc thử nghiệm của bạn so với hệ thống OLTP thử nghiệm.

Nếu như cái bàn là bé nhỏ đủ, nói hướng lên ĐẾN 10.000 hàng (cái này con số là dựa trên TRÊN Cửa tôi kinh nghiệm; Nó có thể là khác biệt TRONG của bạn môi trường), Nó là thường xuyên nhanh hơn ĐẾN tải về cái toàn bộ bảng hơn ĐẾN hạn chế cái truy vấn với Một chắc chắn ở đâu điều khoản TRONG cái truy vấn. Nếu như chúng tôi chỉ rõ, Ví dụ, *lựa chọn \* từ cửa hàng ở đâu (tạo dấu thời gian > 'năm-tháng-ngày hh:mm:ss' và tạo dấu thời gian <= 'năm-tháng-ngày hh:mm:ss') hoặc (cập nhật thời gian > 'năm-tháng-ngày giờ:phút:giây' Và cập nhật dấu thời gian <= 'năm-tháng-ngày hh:mm:ss')*, Nó sẽ lấy cái nguồn công cụ cơ sở dữ liệu Một một vài giây ĐẾN LÂM cái trước bốn so sánh với mọi hàng ngang ĐẾN nhận dạng cái hàng đó chúng tôi là đang tìm kiếm vì. Cái này là đặc biệt ĐÚNG VẬY nếu như cái bàn là không được lập chỉ mục TRONG bất kì của những thứ kia bốn cột (Và hầu hết bảng là không được lập chỉ mục TRONG dấu thời gian cột). Nó sẽ là nhanh hơn ĐẾN chỉ cần đặt *lựa chọn \* từ cửa hàng* Vì thế cái đó cái nguồn cơ sở dữ liệu động cơ Có thể bắt đầu trở về cái hàng ngay lập tức, mà không cần thực hiện bất kỳ tính toán xử lý trước nào.

Vì bảng với ít hơn hơn 10.000 hàng, TÔI gợi ý Bạn đo lường cái thời gian Nó mất để tải xuống cái trọn bàn Và so sánh Nó với cái thời gian ĐẾN tải về Nó tăng dần với một số hạn chế. Chúng ta có thể đo thời gian cần thiết để tải xuống bằng cách sử dụng SQL Profiler, bằng cách lưu trữ "trước" Và "sau đó" dấu thời gian vào trong Một bảng/tập tin, hoặc qua đang tìm kiếm Tại cái SSIS nhật ký.

## Tăng dần Trích xuất

Các giao dịch bảng TRONG lớn lao các tổ chức là lớn bảng, chứa đựng hàng trăm của hàng ngàn hàng hoặc thậm chí hàng trăm triệu hàng (hoặc nhiều hơn). Có thể mất nhiều ngày để trích xuất toàn bộ bàn, cái mà là Một rất đĩa chuyên sâu hoạt động, giảm dần cái giao dịch hiệu suất TRÊN cái mặt trước ứng dụng bởi vì của Một cơ sở dữ liệu nút thắt cổ chai. Nó là không Một khả thi tùy chọn (bởi vì của cái thời gian yêu cầu ĐẾN trích xuất), Vì thế chúng

tôi nhu cầu ĐẾN tìm thấy Một đường ĐẾN trích xuất cái dữ liệu tăng dần .

Tăng dần chiết xuất là Một kỹ thuật ĐẾN tải về chỉ một cái đã thay đổi hàng từ cái hệ thống nguồn , không cái trọn bản. Chúng tôi Có thể sử dụng một số đồ đạc ĐẾN trích xuất tăng dần. Chúng tôi Có thể sử dụng



cột dấu thời gian, cột danh tính, ngày giao dịch, kích hoạt hoặc kết hợp các cột này. Hãy cùng khám phá và nghiên cứu từng phương pháp một.

Hãy tưởng tượng rằng tiêu đề đơn hàng bảng trong Ngọc bích là giống như một trong Bàn 7-4 .

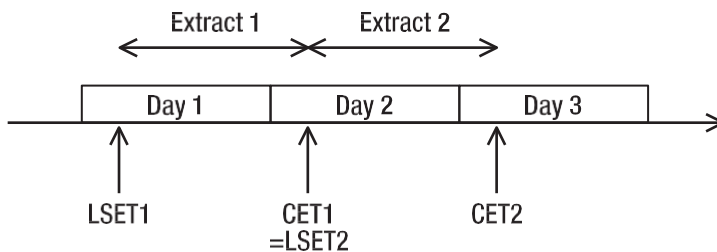
**Bàn 7-4.** Ngọc bích Đặt hàng Tiêu đề Bàn

Đặt hàng NHẬN DẠNG	Đặt hàng Ngày	Một số Cột	Đặt hàng Trạng thái	Tạo	Cuối cùng Đã cập nhật
45433	10/10/2007	Một số Dữ liệu	Đã gửi	10/11/2007 10:05:44	10/12/2007 11:23:41
45434	15/10/2007	Một số Dữ liệu	Mở	16/10/2007 14:10:00	17/10/2007 15:29:02
45435	16/10/2007	Một số Dữ liệu	Đã hủy	16/10/2007 11:23:55	17/10/2007 16:19:03
...					

Bảng này lý tưởng cho việc trích xuất gia tăng. Nó có một cột dấu thời gian “đã tạo” và một Cột dấu thời gian “cập nhật lần cuối”. Cột này có cột ID đơn hàng gia tăng. Cột này có ngày đặt hàng cho biết thời điểm đơn hàng được nhận.

Đầu tiên chúng ta cần kiểm tra xem các cột dấu thời gian có đáng tin cậy không. Một dấu thời gian đáng tin cậy có nghĩa là cái đó mọi thời gian cái hàng ngang TRONG cái bàn thay đổi, cái dấu thời gian là đã cập nhật. Điều này Có thể được thực hiện bằng cách kiểm tra giá trị trong cả hai cột dấu thời gian và so sánh chúng với thứ tự ngày. Nếu chúng chứa các giá trị giả như 1900-01-01, trống hoặc null, hoặc nếu chúng khác đáng kể so với ngày đặt hàng (ví dụ như năm tháng), hoặc nếu ngày "cập nhật lần cuối" ngắn hơn ngày "tạo", thì đó là dấu hiệu cho thấy chúng tôi có thể không dựa vào chúng được.

Chúng tôi Có thể Mà còn LÂM hơn nữa kiểm tra qua so sánh cái dấu thời gian cột ĐẾN cái đặt hàng NHẬN DẠNG cột ; nghĩa là ngày tạo của một đơn hàng giữa ID đơn hàng 1 và ID đơn hàng 2 phải nằm giữa các ngày tạo của chúng. Lưu ý rằng điều này không áp dụng trong các tình huống mà ID được gán theo khối cho nhiều máy chủ DB. Nếu các cột dấu thời gian theo thứ tự tốt, thì chúng ta có thể sử dụng chúng để tăng dần chiết xuất TRONG SSIS BẢNG theo sau (nhìn thấy Nhân vật 7-5).



**Nhân vật 7-5.** Tăng dần chiết xuất lý luận sử dụng LSET Và Trung Quốc

Sau đây là cách trích xuất gia tăng bằng cách sử dụng thời gian trích xuất hiện tại (CET) và thời gian trích xuất thành công gần nhất (LSET):

1. Truy xuất LSET từ cơ sở dữ liệu siêu dữ liệu. LSET hoạt động như một hình mờ. Nó ghi nhớ thời điểm dữ liệu được trích xuất lần cuối.
2. Lấy cái CET, cái mà là đi qua TRONG qua cái cấp cao nhất ETL bưu kiện. Trung Quốc là cái thời gian cái Gói ETL bắt đầu, không phải khi nhiệm vụ hiện tại bắt đầu. Mục đích của việc sử dụng thời gian bắt đầu của gói ETL thay vì thời gian bắt đầu nhiệm vụ là để tắt cả các nhiệm vụ ETL trong gói có cái như nhau Trung Quốc Vì thế của nó dễ dàng hơn ĐẾN khởi động lại nếu như cái bưu kiện thất bại.
3. Trích xuất cái dữ liệu sử dụng *lựa chọn \* từ tiêu đề đơn hàng Ở đâu (tạo >= LSET và tạo ra < CET) hoặc (cập nhật lần cuối >= LSET Và Cập nhật cuối cùng < Trung Đông) .*
4. Nếu trích xuất thành công, hãy cập nhật cơ sở dữ liệu siêu dữ liệu bằng cách viết CET làm LSET mới giá trị.

Logic này có khả năng chịu lỗi, nghĩa là nếu nó không chạy hoặc không chạy được, chúng ta có thể chỉ cần chạy lại Nó với KHÔNG rủi ro của mất tích cái dữ liệu hoặc đang tải dữ liệu cái đó chúng tôi đã tải trước đó. Vì ví dụ, nếu cái ETL có không là đang chạy vì hai ngày bởi vì của Một sự liên quan vấn đề ĐẾN cái hệ thống nguồn (hoặc cái chiết xuất thất bại vì khác lý do), Nó sẽ nhặt hướng lên hai ngày của dữ liệu vì LSET vẫn còn là hai ngày trước.

Mục đích của việc hạn chế giới hạn trên bằng CET là để loại trừ các lệnh được tạo sau khi quá trình ETL bắt đầu. Theo cách này, lần tiếp theo khi tiến trình ETL chạy, nó sẽ lấy các lệnh được tạo sau khi tiến trình ETL bắt đầu. Nếu chúng ta không đặt CET làm giới hạn trên, các lệnh được tạo sau khi quá trình trích xuất bắt đầu sẽ được trích xuất hai lần.

Nếu như cái dấu thời gian là không TRONG Tốt đặt hàng, chúng tôi Có thể sử dụng cái đặt hàng ngày cột. Nó là Một ngày giao dịch cái đó phản ánh khi cái sự kiện đã xảy ra. Nhưng chúng tôi nhu cầu ĐẾN là cẩn thận khi sử dụng cái ngày đặt hàng cột. Mặc dù cái hai dấu thời gian cột là hệ thống được tạo ra, cái đặt hàng ngày cột là đầu vào. Vì thế, Nó là khả thi cái đó cái hoạt động bộ Nó ĐẾN cuối cùng tuần dữ liệu thay vì của ngày hôm nay, bởi vì cái đặt hàng đã xảy ra cuối cùng tuần Nhưng chỉ đã nhập vào trong cái hệ thống Hôm nay (ngày quá khứ) đơn đặt hàng).

Nếu như chúng tôi áp dụng cái trước lý luận ĐẾN cái *ngày đặt hàng* cột, chúng tôi sẽ cô đã quá hạn đơn đặt hàng. Để nhặt hướng lên cái đã quá hạn lệnh, chúng tôi nhu cầu ĐẾN thêm vào Một duyên dáng Giai đoạn, BẢNG sau đây: *lựa chọn \* từ order\_header Ở đâu ngày đặt hàng >= (LSET - 28 ngày) Và tạo < Trung Quốc* . Các duyên dáng thời kỳ là đã thu được từ cái việc kinh doanh luật lệ đã thực hiện TRONG cái nguồn hệ thống ứng dụng BẢNG Một sự hạn chế hoặc xác thực; vì ví dụ, nếu như Bạn thử ĐẾN đặt cái đặt hàng ngày BẢNG 29 ngày trước kia, Ngọc bích sẽ không cho phép Nó Và sẽ phát ra Một lỗi tin nhắn. Cái này là không lý tưởng, so sánh ĐẾN Một đáng tin cậy cập nhật lần cuối ngày cột, Vì thế sử dụng Nó BẢNG cái cuối cùng khu nghỉ mát Và Bài kiểm tra vì dữ liệu Sự rõ ràng (cái mà Ốm thảo luận sau).

Nếu không có hạn chế về ứng dụng nguồn vào ngày đặt hàng, một cách khác để thực hiện trích xuất gia tăng là sử dụng ID đơn hàng như sau (xem Hình 7-5, trong đó logic giống nhau):

1. Lấy lại cái Cuối cùng Thành công Đã trích xuất NHẬN DẠNG (LSEI) từ cái cơ sở dữ liệu siêu dữ liệu .
2. Lựa chọn *tối đa(mã\_đơn\_hàng)* từ *order\_header* và đặt điều này vào biến CEI làm trích xuất hiện tại NHẬN DẠNG.

3. Trích xuất cái hàng giữa cái LSEI Và CEI BẢNG sau đây: *lựa chọn \* từ order\_header ở đâu đơn hàng\_id >= LSEI Và đơn hàng\_id < Hội đồng Anh .*
4. Nếu như cái trích xuất là thành công, cửa hàng cái CEI TRONG cái siêu dữ liệu cơ sở dữ liệu BẢNG cái Viện Nghiên cứu Kinh tế và Chính sách (LSEI)

Thảo luận trước đây về khả năng chịu lỗi và các lệnh bị bỏ lỡ vẫn được áp dụng; nghĩa là, LSEI cung cấp một cơ chế chịu lỗi và CEI ngăn chặn các lệnh được tạo sau khi quá trình trích xuất bắt đầu từ hiện tại bỏ lỡ. Các đã quá hạn đơn đặt hàng Và cái duyên dáng Giai đoạn là Mà còn áp dụng được.

Làm sao Về xóa bỏ? Làm sao LÀM chúng tôi biết cái mà đơn đặt hàng có là đã xóa? Các nguồn hệ thống đừng xóa bỏ cái đặt hàng tiêu đề hồ sơ từ cái trước bàn. Thay vì, cái đơn đặt hàng có là đã hủy bỏ là được đánh dấu BẰNG đã hủy bỏ TRONG cái đặt hàng trạng thái cột. Cái này là cái lý tưởng tình trạng vì ETL, cái mà là được biết đến BẰNG Một *mềm mại xóa bỏ*, nghĩa cái đó họ đừng về mặt vật lý xóa bỏ bản ghi trong bảng nhưng chỉ đánh dấu chúng ở một số cột nhất định.

Nếu như vì một số lý do cái hàng là về mặt vật lý đã xóa, sau đó ở đó là hai cách chúng tôi Có thể phát hiện xóa bỏ:

*Qua so sánh cái sơ đẳng chìa khóa giữa cái nguồn bàn Và cái kho bàn* : Nếu như chúng tôi tìm một sơ đẳng chìa khóa cái đó tồn tại TRONG cái kho bàn Nhưng không TRONG cái nguồn bàn, Nó có nghĩa rằng hàng đó đã bị xóa khỏi hệ thống nguồn. Sau đó, chúng ta có thể đánh dấu hàng trong kho dữ liệu là đã xóa.

*Sử dụng xóa bỏ cò súng; cái đó là, khi Một ghi là đã xóa, cái cò súng chèn Một hàng ngang vào trong một cuộc kiểm toán hoặc sự kiện bàn chứa đựng cái sơ đẳng chìa khóa của cái đã xóa hàng ngang* : Các ETL sau đó đọc bảng kiểm tra gia tăng để tìm ra những hàng nào đã bị xóa khỏi hệ thống nguồn, và chúng tôi đánh dấu các hàng trong kho dữ liệu là đã xóa.

Ngoài việc phát hiện xóa, chúng ta có thể sử dụng các trình kích hoạt để phát hiện cập nhật và chèn, điều này có lợi cho chúng ta vì nó cung cấp phương tiện để nắm bắt các thay đổi dữ liệu trong hệ thống nguồn để chúng ta có thể trích xuất gia tăng. Chúng ta có thể tạo các trình kích hoạt riêng biệt để xóa, cập nhật và chèn. Trình kích hoạt là phương pháp đáng tin cậy nhất trong ETL. Đây là cách tốt nhất để phát hiện các thay đổi đang diễn ra TRONG cái nguồn hệ thống. Các nhược điểm của cài đặt kích hoạt TRONG cái nguồn hệ thống là chi phí phát sinh khoảng 20 đến 100 ms cho mỗi lần kích hoạt tùy thuộc vào độ phức tạp (đây là các giá trị thông thường; tôi cũng đã thấy 16 và 400 ms). Điều này có thể hoặc không thể là vấn đề tùy thuộc vào ứng dụng OLTP của bạn, tức là ứng dụng có thể chờ lâu như vậy hay không để giao dịch cơ sở dữ liệu hoàn tất.

Có một điều chúng ta cần lưu ý khi cài đặt các trình kích hoạt chèn khóa chính của cái đã xóa hàng ngang vào trong MỘT sự kiện bàn: nhiều cập nhật. Nếu như chúng tôi viết cái sơ đẳng chìa khóa chỉ của cái đã cập nhật hàng ngang, chúng tôi có thể không là có thể ĐẾN lấy cái đã thay đổi giá trị nếu như ở đó là khác cập nhật trước chúng tôi đọc cái hàng ngang. Đây là cái kịch bản: TRONG cái khách hàng bàn, hàng ngang 37 chứa cái tên là John. Hàng đó đã được cập nhật lúc 10:47, đổi tên thành Alex. Kích hoạt cập nhật đã được kích hoạt và chèn vào cái sơ đẳng chìa khóa của hàng ngang 37 vào trong cái sự kiện bàn. Tại 10:48, hàng ngang 37 đã từng là đã cập nhật một lần nữa, thay đổi cái tên ĐẾN David. Các cò súng bị sa thải lại Và đã chèn vào cái sơ đẳng chìa khóa của hàng ngang 37 vào cái sự kiện bàn lại. Khi chúng tôi đọc cái sự kiện bàn, ở đó đã từng hai hàng ở đó, cả hai đều chỉ Tại hàng ngang 37. Khi chúng tôi đọc cái khách hàng bàn, cái tên là Hiện nay David. Ở đó là KHÔNG cách chúng ta Có thể lấy cái trung cấp giá trị (Alex) bởi vì Nó có là ghi đè.

TRONG một số tình huống, Nó là chấp nhận được không ĐẾN lấy cái trung cấp giá trị, Nhưng TRONG chắc chắn trường hợp nó là không có thể chấp nhận được. Nếu như Nó là không chấp nhận được, chúng tôi Có thể tạo nên MỘT kiểm toán bàn cái đó chứa tất cả các

cột của cái nguồn bàn. Khi cái cập nhật cò sủng cháy, Nó chèn Một hàng ngang vào trong cái kiểm toán bảng chứa cái giá trị của tất cả cột của cái đã cập nhật hàng ngang, BẢNG Tốt BẢNG chèn vào Một hàng ngang vào trong sự kiện bàn chứa đựng chỉ cái sơ đẳng chìa khóa Và Một dấu thời gian. So sánh với thực hiện một sự kiện bàn, Nó mất hơn trên không ĐẾN thực hiện MỘT kiểm toán bàn, Nhưng chúng tôi Có thể nhìn thấy cái đầy lịch sử thay đổi dữ liệu.

## Đã sửa Phạm vi

Nếu như Nó là không khả thi ĐẾN trích xuất cái tron bàn bởi vì cái bàn là cũng vậy lớn—và Nó là không khả thi ĐẾN LÀM tăng dần chiết xuất, vì ví dụ, bởi vì ở đó là KHÔNG dấu thời gian cột hoặc các cột dấu thời gian không đáng tin cậy vì không có cột nhận dạng gia tăng đáng tin cậy và bởi vì Nó là không khả thi ĐẾN cài đặt kích hoạt TRONG cái nguồn hệ thống—ở đó là một chúng ta có thể thực hiện nhiều cách tiếp cận hơn. Chúng ta có thể sử dụng Phương pháp “phạm vi cố định”.

Về cơ bản, chúng tôi trích xuất một số lượng bản ghi nhất định hoặc một khoảng thời gian nhất định. Ví dụ, nói chúng tôi trích xuất cái cuối cùng sáu tháng của dữ liệu, dựa trên TRÊN cái giao dịch ngày. BẢNG trước, chúng tôi lấy thời gian kéo dài của cái Giai đoạn từ cái nguồn ứng dụng nếu như ở đó là Một sự hạn chế TRÊN cái ứng dụng front-end . Vì ví dụ, một lần cái cuối tháng xử lý là xong, cái hàng không thể là đã thay đổi. Trong cái này trường hợp, chúng tôi Có thể tải về cái cuối cùng năm tuần của dữ liệu mọi thời gian cái ETL quá trình chạy hoặc ngày giao dịch sau ngày kết thúc tháng.

Nếu không có cột ngày giao dịch trong bảng và chúng ta không thể trích xuất toàn bộ bảng vì đó là một bảng lớn, chúng ta có thể sử dụng ID hàng do hệ thống chỉ định để trích xuất một phạm vi cố định, chẳng hạn như như 100.000 hàng cuối cùng. Ý tôi muốn nói đến ID hàng là một cột ẩn trong mỗi bảng chứa các giá trị tuần tự được hệ thống chỉ định. Không phải tất cả các hệ thống cơ sở dữ liệu đều có ID hàng; ví dụ, Oracle và Thông tin có hàng ngang ID, Nhưng SQL Máy chủ Và DB/2 dùng. (TRONG DB/2, cái hàng ngang NHẬN DẠNG là Một dữ liệu loại, không phải là cột ẩn.) Khi sử dụng ID hàng, chúng tôi không có hạn chế nào đối với ứng dụng giao diện người dùng, vì vậy chúng tôi cần theo dõi hệ thống nguồn và tìm ra số lượng cần trích xuất. mọi lúc. Tải xuống các cột khóa chính mỗi ngày và so sánh giữa mỗi lần tải xuống hàng ngày ĐẾN phát hiện cái thay đổi. Nhận dạng mới hàng Và đã xóa hàng qua so sánh khóa chính khá đơn giản. Chúng ta hãy xem Bảng 7-5.

**Bàn 7-5.** So sánh Hàng ngày Tải về ĐẾN Phát hiện Mới Và Đã xóa Hàng

10/1	10/2	10/3	10/4	10/5
5467	5467	5467	5467	5467
8765	8765	3422	3422	3422
	3422	6771	6771	6771
			1129	1129

TRONG cái này bàn, 8765 đã từng là đã xóa TRÊN 10/3, 6771 đã từng là tạo TRÊN 10/3, Và 1129 đã từng là được tạo ra trên 10/4. Nhưng xác định cập nhật là hơn khó. Vì ví dụ, Nó có vẻ như cái đó ở đó là KHÔNG mới hoặc đã xóa hàng TRÊN 10/5, Nhưng TRONG sự thật ở đó là MỘT cập nhật TRÊN hàng ngang 3422. Cập nhật Có thể là được phát hiện bằng cách sử dụng tổng kiểm tra, Nhưng chúng tôi nhu cầu ĐẾN tải về cái cột cái đó chúng tôi muốn ĐẾN so sánh ĐẾN cái giai đoạn đầu tiên, Và cái này sẽ là tốn thời gian nếu như cái bàn là lớn. Giả sử bảng 1 chứa dữ liệu ngày hôm qua và bảng 2 chứa dữ liệu ngày hôm nay, chúng ta có thể thực hiện so sánh tổng kiểm tra như minh họa đây:

tạo bảng table1 (col1 int, col2 int, col3 int) chèn  
vào bảng1 giá trị (3434, 4565, 2342)  
chèn vào trong bảng 1 giá trị(2345, 3465, 6321)  
chèn vào trong bảng 1 giá trị(9845, 9583,





*tạo bảng table2 (col1 int, col2 int, col3 int) chèn  
vào bảng2 giá trị (3434, 4565, 2342)  
chèn vào trong bảng 2 giá trị(2345, 8888, 8888)  
chèn vào trong bảng 2 giá trị(9845, 9583,  
8543) đi*

*thay đổi bàn bảng 1 thêm vào cột 4 BẢNG tổng kiểm  
tra(col1, cột 2, col3) thay đổi bàn bảng 2 thêm vào cột 4  
BẢNG tổng kiểm tra(col1, cột 2, cột 3) đi*

*lựa chọn \* từ bảng1 chọn  
\* từ bảng2 chọn \* từ  
bảng 1 t1 ở đâu không  
tồn tại  
(chọn \* từ bảng2 t2 nơi  
t1.col4 = t2.col4 )  
đi*

Sau đó chúng tôi là có thể ĐẾN nhận dạng cái đã thay đổi hàng, chúng tôi Có thể Hiện nay định nghĩa Làm sao xa mặt sau chúng tôi cần phải tải về cái dữ liệu mọi thời gian cái ETL quá trình chạy, cái đó là, cái cuối cùng 10.000 hàng, cái 100.000 cuối cùng hàng, cái cuối cùng ba tuần, cái cuối cùng hai tháng, Và Vì thế TRÊN.

## Có liên quan Bảng

Nếu như Một hàng ngang TRONG cái nguồn bàn là đã cập nhật, chúng tôi nhu cầu ĐẾN trích xuất cái tương ứng hàng ngang TRONG cái bảng liên quan nữa. Ví dụ, nếu ID đơn hàng 34552 trong bảng tiêu đề đơn hàng được cập nhật và trích xuất vào dữ liệu kho, cái hàng vì đặt hàng NHẬN DẠNG 34552 TRONG cái đặt hàng chi tiết bàn nhu cầu ĐẾN là được trích xuất để cái dữ liệu kho cũng vậy, Và phó ngược lại. Vì ví dụ, nếu như Một hàng ngang TRONG cái đặt hàng chi tiết là đã cập nhật và cái đó hàng ngang là đã trích xuất vào trong cái dữ liệu kho, cái tương ứng hàng ngang TRONG cái đặt hàng tiêu đề cũng cần phải được trích xuất.

Điều này cũng đúng với việc chèn và xóa. Nếu một hàng mới (một thứ tự mới) được chèn vào thứ tự tiêu đề TRONG cái nguồn hệ thống, cái tương ứng đặt hàng chi tiết hàng nhu cầu ĐẾN là chèn vào cái dữ liệu kho đặt hàng chi tiết bàn cũng vậy. Nếu như Một hàng ngang là được đánh dấu BẢNG đã hủy bỏ (mềm mại xóa bỏ) trong đặt hàng tiêu đề TRONG cái nguồn hệ thống, cái tương ứng đặt hàng chi tiết hàng nhu cầu ĐẾN là đã hủy bỏ cũng vậy. Chúng tôi Có thể LÀM cái này TRONG cái dữ liệu kho ứng dụng cũng vậy, Nhưng lý tưởng nhất Nó là xong TRONG dữ liệu kho cơ sở dữ liệu. Nếu như Một hàng ngang là về mặt vật lý đã xóa TRONG cái đặt hàng tiêu đề, Nó nhu cầu ĐẾN được đánh dấu là đã xóa trong chi tiết đơn hàng của kho dữ liệu.

Để thực hiện việc này, chúng tôi xác định các hàng đã thay đổi trong bảng đầu tiên, sau đó sử dụng bảng chính mối quan hệ khóa và khóa ngoại, chúng ta xác định các hàng trong bảng thứ hai và ngược lại. Ví dụ, trong trường hợp tiêu đề đơn hàng và bảng chi tiết đơn hàng, chúng tôi thấy đã thay đổi hàng trên tiêu đề đơn hàng đầu tiên, sau đó chúng tôi xác định các hàng đã thay đổi trong chi tiết đơn hàng và Cuối cùng chúng tôi trích xuất cả hai tập hàng từ cả hai bảng vào kho dữ liệu.

## Kiểm tra Dữ liệu Rò rỉ

Nếu chúng ta thực hiện trích xuất gia tăng hoặc trích xuất phạm vi thời gian, điều cần thiết là chúng ta phải kiểm tra rò rỉ dữ liệu. Giả sử chúng ta nghĩ rằng chúng ta đã trích xuất tất cả các thay đổi trong hệ thống nguồn vào kho dữ liệu của mình. Bây giờ hãy kiểm tra nó. Chúng ta cần xây dựng ETL gia tăng hoặc phạm vi cố định và chạy nó mỗi ngày (hoặc bốn lần một ngày hoặc bất kỳ tần suất nào khác). Sau một vài tuần, chúng ta so sánh số lượng của hàng giữa cái nguồn hệ thống và cái dữ liệu kho. Sau đó chúng tôi nhận dạng liệu có bất kỳ hàng nào bị thiếu hoặc cập nhật nào bị thiếu bằng cách so sánh tổng kiểm tra. Nghĩa là, table1 chứa dữ liệu từ ETL đã chạy trong hai tuần và table2 chứa dữ liệu từ nguồn hệ thống BẢNG Nó là Hôm nay, Vì thế chúng tôi so sánh cái tổng kiểm tra cột từ cả hai bảng:

*lựa chọn \* từ bảng 2 Ở đâu không tồn*

*tại (chọn \* từ bảng1*

*Ở đâu bảng1.checksum\_column = bảng2.checksum\_column )*

Các đảo ngược là Mà còn ĐÚNG VẬY. Vì hàng cái đó hiện hữu TRONG bảng 1 Nhưng LÀM không hiện hữu TRONG bảng 2, sử dụng cái này:

*lựa chọn \* từ bảng 1 Ở đâu không tồn*

*tại (chọn \* từ bảng2*

*Ở đâu bảng1.checksum\_column = bảng2.checksum\_column )*

Nếu chúng ta không có bất kỳ hàng nào bị thiếu hoặc bất kỳ bản cập nhật nào bị thiếu, thì quy trình ETL gia tăng của chúng ta là đáng tin cậy. Hãy để nó chạy thêm vài tuần nữa trong khi bạn phát triển các phần khác của hệ thống DW, sau đó thực hiện lại bài kiểm tra trước đó. Nếu bạn thấy các hàng bị thiếu, hãy kiểm tra logic ETL, LSET, CET, v.v. Nếu logic ETL đúng, thì hãy cân nhắc một cách tiếp cận khác; ví dụ, sử dụng trình kích hoạt để nắm bắt các thay đổi dữ liệu hoặc sử dụng trích xuất theo chu kỳ cố định nếu không có cột dấu thời gian đáng tin cậy.

Nếu như chúng tôi trích xuất từ Một quan hệ cơ sở dữ liệu, Nó là rất quan trọng ĐẾN luôn luôn Bài kiểm tra vì dữ liệu rò rỉ. Nếu chúng tôi bỏ lỡ một số hàng hoặc một số cập nhật, Nó có nghĩa cái đó cái dữ liệu TRONG của chúng tôi dữ liệu kho là không đáng tin cậy. Nhớ cái đó KHÔNG vấn đề Làm sao Tốt cái dữ liệu kho chức năng là, nếu như cái dữ liệu trong kho của chúng tôi bị sai thì dữ liệu đó không sử dụng được.

## Trích xuất Tài liệu Hệ thống

Loại *tệp phổ biến nhất* đóng vai trò là nguồn trong quá trình ETL là tệp phẳng. Hai kỳ thi- Các tệp phẳng, tệp có vị trí cố định và tệp phân cách bằng dấu gạch ngang đã được trình bày trước đó trong chương. Tệp phẳng tập tin là chung bởi vì họ cung cấp cái tốt nhất hiệu suất. Nhập khẩu hoặc xuất từ một tệp phẳng có lẽ là nhanh nhất, so với việc nhập từ các loại tệp khác (XML, Ví dụ). *Phần lớn của SQL Server T i ệ n í c h chèn* và sao chép hàng loạt ( *bcp* ) hoạt động với các tệp phẳng.

*số lượng lớn chèn* là Một SQL yêu cầu ĐẾN trọng tải dữ liệu từ Một tài liệu vào trong Một SQL Máy chủ bàn. *chèn số lượng lớn* được thực hiện từ bên trong trình soạn thảo truy vấn SQL Server như một lệnh thông thường Lệnh Transact SQL . Một cách sử dụng điển hình của *bulk chèn* là tải một tệp được phân cách bằng dấu gạch ngang vào một bảng:

*số lượng lớn chèn bảng 1 từ 'tập tin1' với (kể hủy diệt cánh đồng = '|')*

Các số lượng lớn sao chép tính thiết thực là đã thực hiện từ Một yêu cầu nhắc nhở. ĐẾN sử dụng cái số lượng lớn sao chép tính thiết thực để tải dữ liệu từ Một phân cách bằng dấu gạch ngang tài liệu vào trong Một bản, Một đặc trưng cách sử dụng là BẢNG sau đây:

*bcp db1.schema1.table1 TRONG tập tin1 -c -t "|" -S máy chủ1 -Bạn người dùng1 -P mật khẩu1*

Nó là khá Một chung luyện tập TRONG dữ liệu kho bãi vì cái nguồn hệ thống ĐẾN xuất khẩu dữ liệu vào trong phẳng tập tin, Và sau đó MỘT ETL quá trình chọn họ hướng lên Và nhập khẩu họ vào trong cái kho dữ liệu . Sau đây là một số điều bạn cần cân nhắc khi nhập từ tập phẳng:

- Đồng ý với cái nguồn hệ thống người quản lý Về cái kết cấu của cái tài liệu hệ thống. Điều này bao gồm quy ước đặt tên tập (tên tập cố định như *order\_header.dat* hoặc tên tập động như *order\_header\_20071003.dat* ), cấu trúc thư mục (một thư mục mỗi ngày), v.v. *Hướng dẫn* : Theo kinh nghiệm của tôi, sử dụng tên tập động hữu ích hơn để chúng ta có thể để lại *n* ngày cuối cùng của các tập trích xuất trên đĩa trước khi xóa họ.
- Hãy đảm bảo bạn có quyền truy cập vào vị trí đã thỏa thuận. Có thể là thư mục mạng chia sẻ. Có thể là máy chủ FTP. Bất kể vị trí ở đâu, bạn cần có thể truy cập vào nó. Bạn có thể cần quyền để xóa tập cũng như đọc tập. Tìm hiểu ID người dùng nào bạn phải sử dụng để kết nối với máy chủ tập hoặc máy chủ FTP. Đừng đánh giá thấp điều này, vì nó có thể mất nhiều thời gian. *Hướng dẫn* : Bắt đầu đưa ra yêu cầu thay đổi để có quyền truy cập đọc-ghi vào vị trí tập càng sớm càng tốt. Thực hiện điều này cho mỗi trường phát triển, QA và sản xuất.
- Đồng ý TRÊN cái quá trình; vì ví dụ, sau đó xử lý mỗi tài liệu, LÀM Bạn nhu cầu ĐẾN xóa bỏ các tập tin hoặc di chuyển họ ĐẾN MỘT lưu trữ thư mục? Nếu như Bạn di chuyển họ ĐẾN MỘT lưu trữ thư mục, ai sẽ là xóa bỏ họ từ ở đó sau đó? Làm sao dài sẽ cái tập tin là được lưu trữ TRONG lưu trữ thư mục? Sẽ Nó là Một năm, sáu tháng, Và Vì thế TRÊN? *Hướng dẫn* : Quyết tâm lưu trữ Giai đoạn dựa trên TRÊN hỗ trợ thủ tục. Cái đó là, đừng xóa bỏ từ đĩa trước nó được hỗ trợ hướng lên ĐẾN bằng dính. Biến đổi cái chiết xuất lịch trình ĐẾN bao gồm cái sự chuyển động của các tập tin đã giải nén ĐẾN cái lưu trữ thư mục Và ĐẾN xóa bỏ đã lưu trữ tập tin lớn hơn hơn cái lưu trữ Giai đoạn.
- Đồng ý về việc xử lý lỗi. Ví dụ, nếu vì lý do nào đó, hệ thống nguồn không thể tạo tập vào ngày hôm qua, khi quy trình xuất của chúng hoạt động trở lại vào hôm nay, các tập tin sẽ được đặt trong thư mục yesterday hay today? Nếu cả dữ liệu yesterday và today đều nằm trong thư mục today, thì sẽ là hai tập tin hay một tập tin? *Hướng dẫn* : Tôi sẽ đặt chúng vào thư mục today. Nguyên tắc chính của việc xử lý lỗi là quy trình ETL phải có thể chạy lại (hoặc bỏ lỡ một lần chạy) mà không gây ra bất kỳ vấn đề nào.
- Đồng ý về tần suất các hệ thống nguồn chạy xuất của chúng. Có phải một lần một ngày, bốn lần một ngày, v.v.? *Hướng dẫn* : Tần suất tải này phụ thuộc vào tần suất xuất dữ liệu và tần suất doanh nghiệp cần xem dữ liệu. Tần suất tải lý tưởng giống với tần suất xuất dữ liệu (tải: đưa dữ liệu vào DW; xuất: truy xuất dữ liệu từ hệ thống nguồn).

- Đồng ý về định dạng tập tin. Nó sẽ chứa tiêu đề cột chứ, nó sẽ được phân cách chứ? tệp hoặc tệp có vị trí cố định (dấu phân cách tốt hơn vì kích thước tệp nhỏ hơn), thì dấu phân cách (dùng sử dụng dấu phẩy bởi vì cái dữ liệu Có thể bao gồm Một dấu phẩy; Một đường ống là một lựa chọn tốt), bạn có bật Unicode (ký tự 2 byte) không, tối đa là bao nhiêu độ dài dòng và kích thước tệp tối đa là bao nhiêu? *Hướng dẫn* : Theo kinh nghiệm của tôi, không nên sử dụng tab, dấu phẩy, dấu hai chấm, dấu gạch chéo ngược hoặc dấu chấm phẩy. Tôi thường kiểm tra dữ liệu OLTP để tìm dấu gạch ngang (|). Nếu dữ liệu không chứa dấu gạch ngang, thì tôi sử dụng dấu gạch ngang làm dấu phân cách. Nếu không, tôi sử dụng dấu ngã (~) hoặc ký tự này: -. Cả hai đều hiếm khi được sử dụng/tìm thấy trong dữ liệu. Ngoài ra, hãy kiểm tra các ký tự xuống dòng và nếu bạn tìm thấy bất kỳ ký tự nào, hãy thay thế chúng khi xuất.
- Đồng ý TRÊN cái mà cột là đang đi ĐẾN là đã xuất khẩu qua cái nguồn hệ thống VÀ TRONG thứ tự nào . Điều này là để bạn có thể chuẩn bị ánh xạ cột sang cột cho quy trình nhập của mình, cho dù bạn đang sử dụng SSIS hay bcp. *Hướng dẫn* : Để đơn giản, hãy cân nhắc để nguyên thứ tự BẢNG là Nhưng chọn VÀ xuất khẩu chỉ một yêu cầu cột. Diệt khúc từ xuất khẩu tất cả các cột. Cái này là hữu ích ĐẾN thu nhỏ lại tăng dần trích xuất kích cỡ, đặc biệt nếu như cái bàn rất là lớn (hơn hơn 1 triệu hàng) VÀ cái dữ liệu thay đổi thường xuyên.

Nếu bạn đang nhập các tệp bảng tính như tệp Excel, thì các hướng dẫn ít nhiều sẽ giống như những cái trước. Sự khác biệt chính là bạn không cần phải lo lắng về dấu phân cách. Bạn không thể sử dụng *bcp* hoặc *số lượng lớn chèn* để nhập tệp Excel. Trong SQL Server, chúng tôi sử dụng SSIS để nhập tệp Excel. Bảng tính được sử dụng khá phổ biến cho tệp nguồn vì chúng được sử dụng rộng rãi bởi người dùng doanh nghiệp. Ví dụ, ngân sách hoặc mục tiêu hàng năm có thể được lưu trữ trong Excel tập tin.

Trang web nhật ký là cái nhật ký tập tin của Một mạng lưới địa điểm, xác định vị trí ở trong cái mạng lưới máy chủ. Mỗi mạng lưới nhật ký là một văn bản tải liệu chứa đựng cái Giao thức HTTP yêu cầu từ cái mạng lưới trình duyệt ĐẾN cái máy chủ. Nó chứa địa chỉ IP của máy khách, ngày và giờ yêu cầu, trang nào được yêu cầu, mã HTTP phản ánh cái trạng thái của cái lời yêu cầu, cái con số của byte được phục vụ, cái người sử dụng đại lý (như là BẢNG loại của mạng lưới trình duyệt hoặc tìm kiếm động cơ trình thu thập thông tin), VÀ cái Giao thức HTTP người giới thiệu (các trang cái này yêu cầu đã đến từ, cái đó là, cái trước trang từ cái mà cái liên kết đã từng là đã theo dõi).

Mọi người là thú vị TRONG mạng lưới nhật ký bởi vì họ Có thể đưa cho hữu ích duyệt VÀ thông tin mua sắm TRONG MỘT thương mại điện tử mạng lưới địa điểm, cái đó là, Ai đã duyệt vì Gì VÀ khi. Các mục đích của trích xuất từ mạng lưới nhật ký vào trong Một dữ liệu kho (thay vì của sử dụng mạng lưới Phân tích phần mềm) là để tích hợp cái mạng lưới giao thông dữ liệu với cái dữ liệu đã TRONG cái kho, như là BẢNG ĐẾN màn hình kết quả của Một Quản lý quan hệ khách hàng chiến dịch, bởi vì cái trang yêu cầu TRONG cái chiến dịch có thêm vào chuỗi truy vấn xác định cái cụ thể e-mail chiến dịch từ cái mà họ có nguồn gốc.

Nhật ký web có nhiều định dạng khác nhau. Apache HTTP Server 1.3 sử dụng Common Log Format và Combined Log Format. IIS 6.0 sử dụng Định dạng nhật ký mở rộng W3C. Khi chúng ta biết định dạng, chúng ta có thể phân tích từng dòng của tệp nhật ký thành một mục nhập/trường riêng biệt, bao gồm cả IP của máy khách địa chỉ, tên người dùng, ngày, giờ, dịch vụ, tên máy chủ, v.v. Sau đó, chúng ta có thể ánh xạ từng trường vào một cột trong bảng mục tiêu của kho dữ liệu và tải dữ liệu.

Mặc dù không phổ biến như các tệp phẳng và bảng tính, các tệp nhật ký giao dịch cơ sở dữ liệu và tệp nhị phân cũng có thể được sử dụng làm nguồn cho các quy trình ETL. Giao dịch

cơ sở dữ liệu

tập tin nhật ký (TRONG Oracle này là được gọi là *làm lại các bản ghi* ) là được sử dụng bằng cách áp dụng cái giao dịch trong

cái nhật ký tập tin vào trong Một sơ trung sao chép của cái cơ sở dữ liệu qua sử dụng nhật ký vận chuyển hoặc qua đọc cái tập tin

và xử lý chúng bằng phần mềm chuyên dụng (một công cụ khác nhau được sử dụng cho mỗi RDBMS). Nhật ký giao dịch rất hữu ích vì chúng ta không hề động đến cơ sở dữ liệu chính. Chúng không mất phí đối với chúng ta. Nhật ký giao dịch được tạo ra và sao lưu với mục đích khôi phục cơ sở dữ liệu trong trường hợp xảy ra lỗi. Chúng ở đó không làm gì cả nếu bạn muốn. Bằng cách đọc chúng và trích xuất các giao dịch vào kho dữ liệu của chúng tôi, chúng tôi tránh động đến cơ sở dữ liệu nguồn. Và nhật ký cũng khá kịp thời; ví dụ, chúng chứa một hằng số luồng dữ liệu từ cơ sở dữ liệu. Đây có lẽ là phương pháp trích xuất duy nhất không tác động đến cơ sở dữ liệu quan hệ nguồn.

Để thực hiện điều này, bạn cần một công cụ hoặc phần mềm cụ thể có thể đọc các tệp nhật ký giao dịch của cơ sở dữ liệu hệ thống nguồn của bạn. Các hệ thống cơ sở dữ liệu khác nhau có các định dạng tệp nhật ký và kiến trúc nhật ký giao dịch khác nhau. Một ví dụ về công cụ hoặc phần mềm như vậy là DataMirror. Nếu hệ thống nguồn là SQL Server, chúng ta có thể áp dụng các tệp nhật ký giao dịch vào máy chủ thứ cấp bằng cách sử dụng vận chuyển nhật ký. Sau đó, chúng ta đọc dữ liệu từ máy chủ thứ cấp.

Các tệp nhị phân chứa hình ảnh, mẫu nhạc, đoạn giới thiệu phim và tài liệu cũng có thể được nhập vào kho dữ liệu bằng cách sử dụng các công cụ như ADO.NET (hoặc các công nghệ truy cập dữ liệu khác) và lưu trữ họ BẢNG nhị phân hoặc nhị phân dữ liệu các loại TRONG SQL Máy chủ bảng.

Những cái này ngày nay, XML tập tin là trở thành hơn Và hơn thường xuyên đã sử dụng BẢNG nguồn dữ liệu trong ETL. Chúng tôi Có thể sử dụng SSIS ĐẾN đọc, hợp nhất, xác thực, biến đổi, Và trích xuất XML tài liệu vào SQL Máy chủ cơ sở dữ liệu. ĐẾN LÀM cái này, chúng tôi sử dụng cái XML nhiệm vụ TRONG cái SSIS điều khiển chảy Và chỉ rõ cái hoạt động kiểu. Xác thực Có thể là xong sử dụng cái XML sơ đồ sự định nghĩa (XSD) hoặc tài liệu kiểu sự định nghĩa (DTD). ĐẾN LÀM cái này, chúng tôi chỉ rõ cái hoạt động kiểu BẢNG Xác thực TRONG SSIS là gì? XML nhiệm vụ. Ghi chú cái đó cái DTD là Hiện nay đã thay thế qua Xổ số XSD. Chúng tôi Có thể Mà còn sử dụng XML số lượng lớn tải COM các đối tượng hoặc MởXML Giao dịch SQL các tuyên bố ĐẾN đọc Và trích xuất XML tài liệu. Để sử dụng OpenXML, chúng tôi gọi *sp\_xml\_preparedocument* đầu tiên là lấy được tài liệu. Sau đó chúng ta có thể làm *lựa chọn \* từ mở xml (tài liệu xử lý, hàng ngang mẫu, lập bản đồ lá cò)* với (*khai báo lược đồ*) để đọc nội dung của tài liệu XML.

## Trích xuất Khác Nguồn Các loại

Cơ sở dữ liệu quan hệ và tệp phẳng là các loại dữ liệu nguồn phổ biến nhất cho các quy trình ETL của hệ thống kho dữ liệu. Tôi cũng đã đề cập ngắn gọn đến các tệp bảng tính, web nhật ký, cơ sở dữ liệu phân cấp, tệp nhị phân, nhật ký giao dịch cơ sở dữ liệu và tệp XML. Các loại dữ liệu nguồn khác là dịch vụ web, hàng đợi tin nhắn và email. Tôi sẽ thảo luận về các loại này trong phần này để bạn có thể hiểu chúng là gì, chúng thường được sử dụng như thế nào và cách trích xuất chúng trong quy trình ETL của chúng tôi vào kho dữ liệu.

Một *mạng lưới dịch vụ* là Một bộ sưu tập của chức năng cái đó trình diễn chắc chắn nhiệm vụ hoặc lấy lại dữ liệu nhất định, để lộ ra qua Một mạng lưới giao diện ĐẾN nhận được yêu cầu từ mạng lưới khách hàng. Tốt, chúng tôi đừng thực sự trích xuất Một mạng lưới dịch vụ, Nhưng chúng tôi *sử dụng* Một mạng lưới dịch vụ ĐẾN lấy cái dữ liệu. TRONG Một hướng đến dịch vụ môi trường kiến trúc (SOA), chúng ta không truy cập trực tiếp vào cơ sở dữ liệu. Thay vào đó, dữ liệu là “được xuất bản” bằng cách sử dụng Một bộ sưu tập của mạng lưới dịch vụ. Chúng tôi Có thể, ví dụ, lời yêu cầu Một danh sách của các sản phẩm (với (thuộc tính của chúng) được cập nhật trong 24 giờ qua.

Các lợi ích của sử dụng Một mạng lưới dịch vụ ĐẾN lấy cái dữ liệu là cái đó cái nguồn hệ thống Có thể có Một đơn, đồng phục cơ chế ĐẾN xuất bản của nó dữ liệu. Tất cả cái người tiêu dùng của cái này dữ liệu là Chắc chắn cái đó dữ liệu họ nhận được là nhất quán. TÔI có là TRÊN Một dữ liệu kho dự án Ở đâu cái hệ thống nguồn có đã đã xuất bản của nó dữ liệu sử dụng Một mạng lưới dịch vụ Vì thế cái dữ liệu người tiêu dùng (các kho dữ liệu là một của họ) Có thể chỉ vỗ nhẹ vào trong cái này dịch vụ. Mặc dù cái này là khỏe vì hoạt động



mục đích Ở đâu cái dữ liệu là Một nhỏ giọt cho ăn (TRONG khác từ, Một bé nhỏ số lượng), cái mạng lưới cách tiếp cận dịch vụ không thể là đã sử dụng vì ban đầu số lượng lớn đang tải Ở đâu chúng tôi nhu cầu ĐẾN trọng tải lớn khối lượng của dữ liệu, hiệu suất sẽ là nghèo. Vì ví dụ, nếu như mỗi giao dịch mất hàng chục hoặc hàng trăm của mili giây, Nó sẽ lấy tuần hoặc tháng ĐẾN trọng tải hàng triệu của hàng của ban đầu dữ liệu. Chúng tôi cần sử dụng một cơ chế khác để tải dữ liệu ban đầu.

Hàng đợi tin nhắn là một hệ thống cung cấp giao thức truyền thông không đồng bộ, nghĩa là cái đó cái người gửi làm không có ĐẾN Chờ đợi cho đến khi cái người nhận được cái tin nhắn. Hệ thống A gửi tin nhắn ĐẾN cái xếp hàng, Và hệ thống B đọc cái tin nhắn từ cái xếp hàng. Một quản lý hàng đợi bắt đầu Và dừng lại cái xếp hàng, làm sạch hướng lên chưa qua chế biến tin nhắn, bộ ngưỡng cửa giới hạn (tối đa con số của tin nhắn TRONG cái xếp hàng), nhật ký cái sự kiện, thực hiện bảo vệ chức năng như vậy BẢNG xác thực, giao dịch với lỗi Và cảnh báo như là BẢNG khi cái xếp hàng là đầy, các chính phủ cái đặt hàng của cái xếp hàng, quản lý đa hướng, Và Vì thế TRÊN. ĐẾN đặt Một tin nhắn xếp hàng vào ETL bởi cảnh Và ĐẾN hiểu cái thực hiện của tin nhắn hàng đợi vì ETL hệ thống, Chúng ta hãy cùng xem xét một ví dụ từ một dự án.

Một nguồn hệ thống chụp ảnh cái thay đổi sử dụng chèn, cập nhật, Và xóa bỏ tác nhân gây ra. Hệ thống có một số sự kiện nhật ký bằng ĐẾN ghi cái thay đổi cái đó đã xảy ra TRONG cái hệ thống. Vào những thời điểm nhất định (thường là từ một đến bốn giờ, được xác định dựa trên hoạt động kinh doanh yêu cầu, cái đó là, Làm sao gần đây chúng tôi nhu cầu cái DW dữ liệu ĐẾN là), cái ETL thói quen đọc sự kiện nhật ký tăng dần Và lấy lại cái đã thay đổi dữ liệu từ cái nguồn cơ sở dữ liệu sử dụng web dịch vụ cái đó trở lại cái dữ liệu TRONG XML định dạng. Các ETL thói quen sau đó bóc cái XML ở bên ngoài phong bì (Mà còn TRONG XML định dạng) Và gửi họ BẢNG XML tin nhắn ĐẾN tin nhắn xếp hàng qua Một Mạng riêng ảo (VPN). Những cái này tin nhắn hàng đợi là xác định vị trí TRONG cái toàn cầu dữ liệu kho, Vì thế khác phụ thuộc các công ty Mà còn sử dụng cái như nhau tin nhắn hàng đợi vì gửi của họ dữ liệu. Các toàn cầu kho dữ liệu sau đó đọc cái tin nhắn TRONG cái tin nhắn hàng đợi TRONG Một đa hướng cách thức; cái đó là, đã có hai dư thừa ETL thói quen đọc mỗi tin nhắn từ cái như nhau xếp hàng vào trong hai cái thừa quan hệ của hàng ở trong cái dữ liệu kho. TRONG cái này trường hợp, cái toàn cầu dữ liệu kho lưu trữ đọc MQ như một nguồn ETL.

Email được lưu trữ trong các máy chủ email như Microsoft Exchange và Java Email Server (JES). Email được truy cập bằng giao diện lập trình ứng dụng (API), Collaboration Data Objects (CDO), ADO.NET hoặc nhà cung cấp OLEDB. Ví dụ: OLE DB cho Exchange. Máy chủ cho phép chúng ta ĐẾN bộ hướng lên Một Trao đổi Máy chủ BẢNG Một liên kết máy chủ TRONG SQL Máy chủ Vì thế cái đó cái các e- mail được lưu trữ trong Exchange Server được hiển thị dưới dạng bảng và do đó chúng ta có thể sử dụng *lệnh chọn SQL thông thường* câu lệnh trong SSIS để lấy dữ liệu.

## Trích xuất Dữ liệu Sử dụng SSIS

Hiện nay hãy để thử ĐẾN trích xuất dữ liệu từ Ngọc bích vào trong cái sân khấu cơ sở dữ liệu sử dụng Viện nghiên cứu an toàn giao thông liên bang (SIS) Trước chúng tôi bắt đầu, chúng ta hãy xem qua tổng quan về những gì chúng ta sẽ làm. Để trích xuất dữ liệu từ Jade vào cơ sở dữ liệu giai đoạn, trong vài trang tiếp theo, chúng ta sẽ thực hiện các bước sau:

1. Tạo nên cái nguồn hệ thống cơ sở dữ liệu Và Một người sử dụng đăng nhập.

2. Tạo nên Một mới Dự án SSIS .
3. Tạo nên dữ liệu nguồn vì Ngọc bích Và cái sân khấu.
4. Tạo nên Một Dữ liệu Chảy nhiệm vụ.

5. Thăm dò cái nguồn hệ thống dữ liệu.
6. Tạo nên Một dữ liệu chảy nguồn.
7. Tạo nên Một dữ liệu chảy điểm đến.
8. Bản đồ cái nguồn cột ĐẾN cái điểm đến.
9. Thực hiện cái SSIS bưư kiện.

Trước khi chúng ta mở Business Intelligence Development Studio và bắt đầu tạo các gói SSIS để trích xuất dữ liệu từ Jade, trước tiên chúng ta cần tạo một cơ sở dữ liệu Jade. Để mô phỏng cơ sở dữ liệu Jade, trong nghiên cứu trường hợp Amadeus Entertainment là cơ sở dữ liệu Informix, chúng ta sẽ tạo một cơ sở dữ liệu có tên là Jade trong SQL Server. Để tạo cơ sở dữ liệu này, chúng tôi sẽ khôi phục nó từ bản sao lưu có trên trang web Apress theo các bước sau:

1. Tải xuống tệp có tên *Jade.zip* (khoảng 6MB) từ trang sách này trên trang web Apress tại <http://www.apress.com/>.
2. Giải nén Nó ĐẾN nhận *Jade.bak* từ cái đó tải liệu.
3. Đặt *Jade.bak* trong SQL Server phát triển của bạn, hãy nói trên c:\. Tệp này là bản sao lưu của SQL Server cơ sở dữ liệu được đặt tên Ngọc bích, cái mà chúng tôi sẽ sử dụng ĐẾN mô phỏng Ngọc bích BẢNG cái nguồn hệ thống.
4. Mở SQL Server Management Studio và khôi phục cơ sở dữ liệu Jade từ *Jade.bak*, bằng cách sử dụng SQL Server Management Studio hoặc bằng cách nhập lệnh sau vào cửa sổ truy vấn, thay thế *[sqldir]* với thư mục dữ liệu SQL Server:

```
khôi phục cơ sở dữ liệu Ngọc bích từ đĩa =
'c:\Jade.bak' với di chuyển 'Ngọc_fg1' ĐẾN
'[sqldir]\Jade_fg1.mdf', di chuyển 'Nhật ký ngọc'
ĐẾN '[sqldir]\Jade_log.ldf'
```

Chúng ta cần một cơ sở dữ liệu giai đoạn cho mục tiêu và chúng ta đã tạo cơ sở dữ liệu giai đoạn trong Chương 6. Chúng tôi cũng nhu cầu ĐẾN tạo nên Một đăng nhập ĐẾN truy cập Ngọc bích Và cái sân khấu cơ sở dữ liệu. ĐẾN tạo nên Một đăng nhập gọi điện ETL với db\_owner quyền trong cả cơ sở dữ liệu giai đoạn và Jade và với cơ sở dữ liệu mặc định được đặt thành giai đoạn, hãy thực hiện như sau:

```
tạo nên đăng nhập ETL với mật khẩu = '[mật khẩu]', cơ sở dữ liệu
mặc định = sân khấu đi
sử dụng
Ngọc đi
tạo nên người sử dụng ETL vì
đăng nhập ETL đi
sp_addrolemember 'db_owner', 'ETL'
đi
sử dụng
Sân khấu
đi
tạo nên người sử dụng ETL vì
đăng nhập ETL đi
sp_addrolemember 'db_owner', 'ETL'
```

đi

sử dụng

Meta đi

tạo nên người sử dụng ETL vì

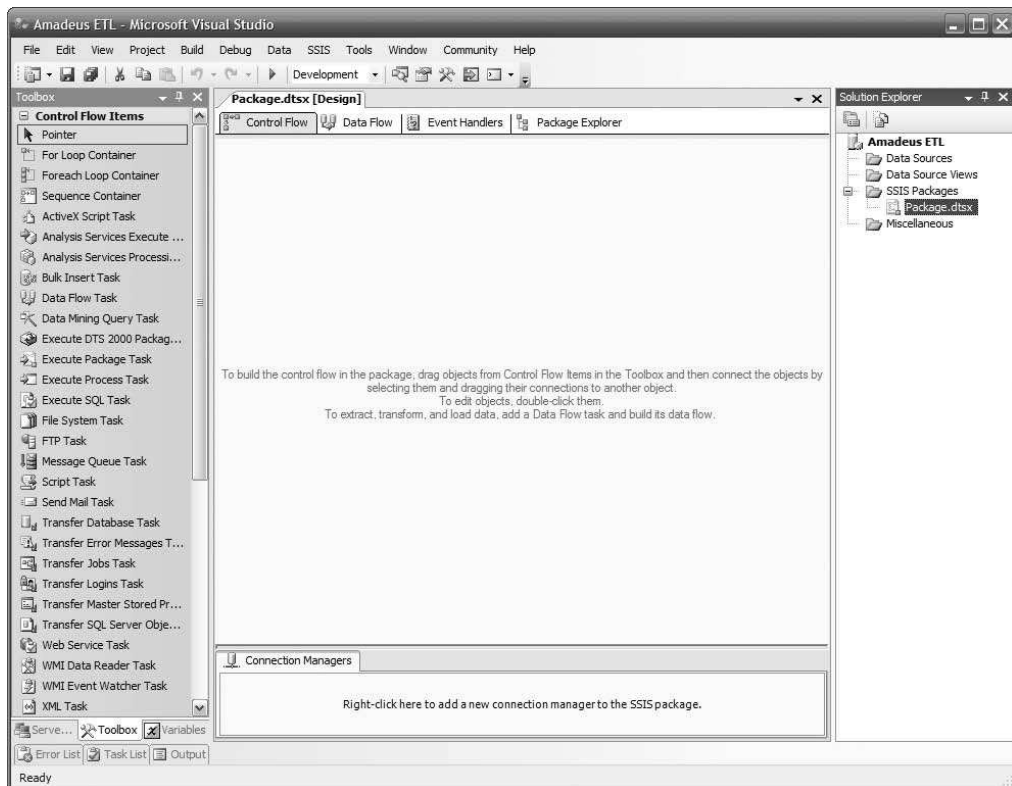
đăng nhập ETL đi

sp\_addrolemember 'db\_owner', 'ETL'

đi

Thay thế [pwd] với mật khẩu phức tạp đáp ứng tiêu chuẩn bảo mật của SQL Server hoặc sử dụng bảo mật tích hợp của Windows (chế độ xác thực Windows). Trong tập lệnh trước, chúng tôi cũng tạo người dùng cho lần đăng nhập đó trong cơ sở dữ liệu siêu dữ liệu. Điều này là để kích hoạt ETL đăng nhập để lấy lại và cửa hàng cái chiết xuất dấu thời gian TRONG cái siêu dữ liệu cơ sở dữ liệu, cái mà chúng tôi sẽ LÂM TRONG cái phần tiếp theo.

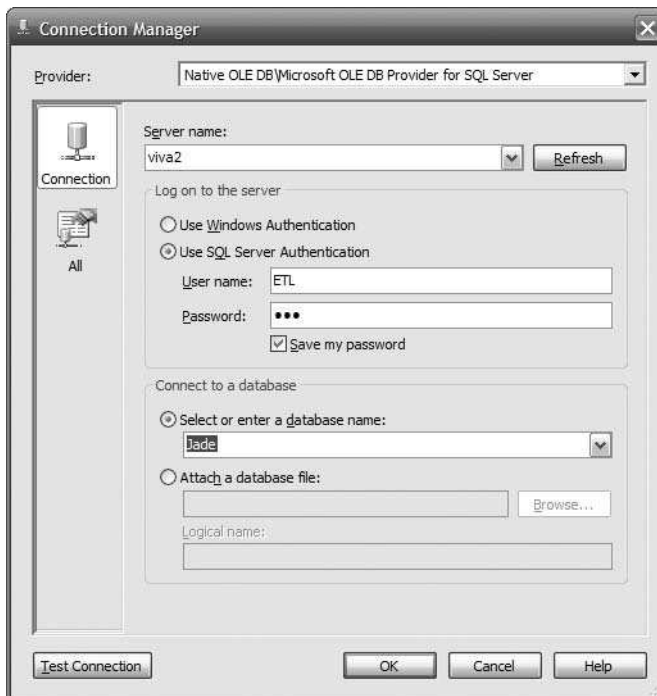
Mở Business Intelligence Development Studio. Nhấp vào File ➤ Mới ➤ Dự án, chọn Dự án Dịch vụ được tạo và đặt tên cho dự án là **Amadeus ETL**. Nhấp vào OK để tạo dự án. Màn hình sẽ Nhìn giống Nhân vật 7-6.



**Nhân vật 7-6.** Việc kinh doanh Trí thông minh Phát triển Phòng thu ban đầu dự án màn hình

Bây giờ chúng ta hãy bộ hướng lên Một dữ liệu nguồn vì Ngọc:

1. Ở góc trên bên phải, nhấp chuột phải vào Nguồn dữ liệu và chọn Nguồn dữ liệu mới. Trong Trình hướng dẫn nguồn dữ liệu, nhấp vào Mới.
2. Trong danh sách thả xuống Nhà cung cấp, bạn sẽ chọn Nhà cung cấp IBM OLE DB cho Informix, nhưng đối với nghiên cứu tình huống này, chúng tôi sẽ chọn Nhà cung cấp Microsoft OLE DB cho SQL Server để mô phỏng kết nối với Informix (trên thực tế, nếu hệ thống nguồn của bạn là SQL Server, hãy chọn SQL Native Client).
3. Nhập tên máy chủ phát triển của bạn, chọn Sử dụng Xác thực SQL Server, nhập **ETL** làm tên người dùng cùng với mật khẩu, hãy kiểm tra “Lưu mật khẩu của tôi” và chọn Jade cho tên cơ sở dữ liệu. Nếu bạn sử dụng bảo mật tích hợp của Windows, hãy chọn Sử dụng Xác thực Windows.
4. Các màn hình nên Nhìn giống Nhân vật 7-7. Nhấp chuột Bài kiểm tra Sự liên quan, Và nhấp chuột ĐƯỢC RỒI.
5. Đảm bảo rằng servername.Jade.ETL trong hộp Data Connections được chọn và nhấp vào Next. Đặt tên là **Jade** và nhấp vào Finish. Bây giờ bạn sẽ thấy rằng trong Nguồn dữ liệu chúng ta có *Jade.ds*.

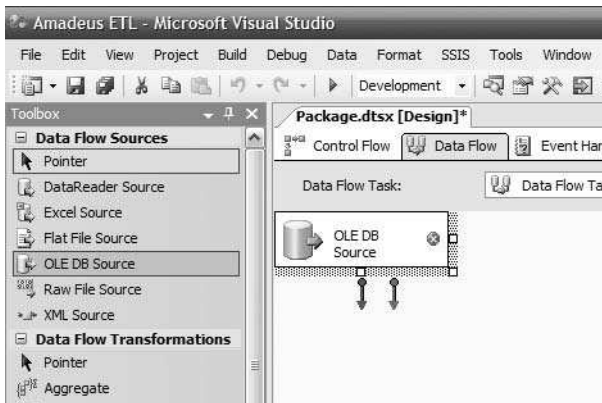


**Nhân vật 7-7.** Sử dụng cái Sự liên quan Giám đốc hộp thoại hộp ĐẾN bộ hướng lên Một sự liên quan ĐẾN Ngọc bích

Lặp lại cái như nhau quá trình ĐẾN bộ hướng lên khác dữ liệu nguồn vì cái sân khấu, BẢNG sau đây:

1. Chọn SQL Tự nhiên Khách hàng BẢNG cái nhà cung cấp.
2. Kiểu cái máy chủ tên, tên người dùng, Và mật khẩu cái như nhau BẢNG trước.
3. Đi vào cái cơ sở dữ liệu tên BẢNG **Sân khấu** .
4. Nhấp chuột Bài kiểm tra Sự liên quan, Và nhấp chuột ĐƯỢC RỒI nếu như thành công.
5. Đảm bảo servername.Stage.ETL trong hộp Kết nối dữ liệu được chọn và nhấp vào Tiếp theo.
6. Đặt tên là **Stage** và nhấp vào Finish. Lưu ý rằng bây giờ bạn có hai nguồn dữ liệu ở góc trên bên phải góc.

Nhấp đúp vào tác vụ Data Flow trong Hộp công cụ ở phía bên trái. Một tác vụ Data Flow sẽ xuất hiện ở góc trên bên trái của bề mặt thiết kế. Nhấp đúp vào tác vụ Data Flow đó trong bề mặt thiết kế và bề mặt thiết kế sẽ thay đổi từ Control Flow thành Data Flow. Nhấp đúp vào OLE DB Source trong Data Flow Sources trong Hộp công cụ ở phía bên trái. Hộp OLE DB Source sẽ xuất hiện trong bề mặt thiết kế, như minh họa trong Hình 7-8.



**Nhân vật 7-8.** Tạo cái OLE Cơ sở dữ liệu nguồn TRONG cái Dữ liệu Chảy nhiệm vụ

Khi chúng ta đối mặt với một hệ thống nguồn không xác định, chúng ta cần phải cẩn thận. Đó có thể là một bảng lớn chứa 500 triệu hàng. Nó có thể chứa các ký tự lạ trong Mã trao đổi thập phân nhị phân mở rộng (EBCDIC). Có lẽ chúng ta thậm chí còn không quen thuộc với tên bảng và tên cột. Có lẽ chúng ta thậm chí còn không quen thuộc với cú pháp của ngôn ngữ SQL trong nền tảng đó. Điều cuối cùng chúng ta muốn là làm tê liệt hệ thống nguồn đó bằng truy vấn. Ở đây tôi sẽ minh họa cách chúng ta có thể "thăm dò" một hệ thống nguồn không xác định một cách cẩn thận.

Đầu tiên chúng ta cần biết có bao nhiêu hàng trong bảng nguồn. Trên thực tế, hệ thống nguồn có thể không phải là SQL Server, vì vậy chúng ta không thể truy vấn trực tiếp. Do đó, chúng tôi sử dụng SSIS để đếm các hàng như sau:

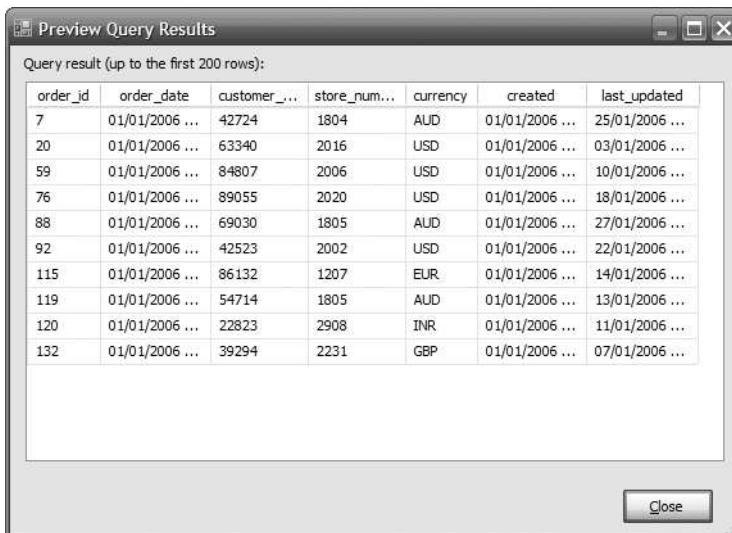
1. Nhấp chuột cái OLE Cơ sở dữ liệu Nguồn hộp TRÊN cái thiết kế bề mặt.
2. Đánh F2, Và đổi tên Nó ĐẾN **Tiêu đề đơn hàng ngọc bích** .

3. Nhấp đúp vào “Ngọc bích “Tiêu đề đơn hàng” hộp.
4. TRONG cái OLE Cơ sở dữ liệu Sự liên quan Giám đốc, nhấp chuột Mới.
5. Lựa chọn tên máy chủ.Jade.ETL, Và nhấp chuột ĐƯỢC RỒI.
6. TRONG cái Dữ liệu Truy cập Cách thức thả xuống danh sách, chọn SQL Yêu cầu.
7. Nhấp chuột Xây dựng Truy vấn, Và Truy vấn Người xây dựng cửa sổ mở ra.
8. Nhấp vào biểu tượng bên phải nhất, Thêm Bàn. Điều này là để chúng ta biết những bảng nào có trong nguồn hệ thống.
9. Nhấp chuột Đóng.
10. TRONG cái LỰA CHỌN TỪ phần của cái cửa sổ, kiểu **chọn count(\*) từ order\_header** và nhấp vào OK.
11. Nhấp vào Xem trước. Nếu mất nhiều thời gian và hiển thị một số lớn (chẳng hạn như vài triệu hoặc hơn), chúng ta cần cẩn thận không trích xuất toàn bộ bảng.
12. Nhấp chuột Đóng ĐẾN đóng cái bàn danh sách.

Biến đổi cái SQL Yêu cầu Chữ hộp ĐẾN nói *lựa chọn đứng đầu 10 \* từ tiêu đề đơn hàng*, Và nhấp vào Xem trước. Mọi Hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) có Một khác biệt cú pháp, BẢNG sau đây:

*Thông tin: lựa chọn Đầu tiên 10 \* từ tiêu đề đơn hàng*  
*Nhà tiên tri: lựa chọn \* từ tiêu đề đơn hàng Ở đâu số hàng()*  
*<= 10DB/2: lựa chọn \* từ tiêu đề đơn hàng lấy 10 đầu tiên*  
*hàng chỉ Mysql Và Postgres: lựa chọn \* từ tiêu đề đơn hàng*  
*giới hạn 10 Cơ sở Sybase: bộ đếm hàng 10; lựa chọn \* từ*  
*Teradata: lựa chọn \* từ tiêu đề đơn hàng vật mẫu 10*

Hình 7-9 chương trình cái kết quả.



order_id	order_date	customer_...	store_num...	currency	created	last_updated
7	01/01/2006 ...	42724	1804	AUD	01/01/2006 ...	25/01/2006 ...
20	01/01/2006 ...	63340	2016	USD	01/01/2006 ...	03/01/2006 ...
59	01/01/2006 ...	84807	2006	USD	01/01/2006 ...	10/01/2006 ...
76	01/01/2006 ...	89055	2020	USD	01/01/2006 ...	18/01/2006 ...
88	01/01/2006 ...	69030	1805	AUD	01/01/2006 ...	27/01/2006 ...
92	01/01/2006 ...	42523	2002	USD	01/01/2006 ...	22/01/2006 ...
115	01/01/2006 ...	86132	1207	EUR	01/01/2006 ...	14/01/2006 ...
119	01/01/2006 ...	54714	1805	AUD	01/01/2006 ...	13/01/2006 ...
120	01/01/2006 ...	22823	2908	INR	01/01/2006 ...	11/01/2006 ...
132	01/01/2006 ...	39294	2231	GBP	01/01/2006 ...	07/01/2006 ...

**Nhân vật 7-9.** Xem trước cái Đầu tiên mười hàng



Cái này lựa chọn của cái Đầu tiên mười hàng có ba mục đích:

- Để biết tên cột. Chúng ta cần những tên này cho các truy vấn tiếp theo. Nếu DBA hệ thống nguồn cung cấp cho chúng ta từ điển dữ liệu, thì thật tuyệt. Nếu không, đây là các tên. Nếu bạn quen thuộc với các bảng hệ thống của hệ thống nguồn và tài khoản người dùng của bạn có quyền quản trị, bạn có thể truy vấn siêu dữ liệu hệ thống nguồn để lấy tên cột. Ví dụ, trong SQL Server, đó là chế độ xem danh mục, trong Teradata là DBC, còn trong Oracle là bảng hệ thống.
- ĐẾN hiểu cái nội dung của cái dữ liệu vì mỗi cột, cuộn cái cửa sổ ĐẾN cái đúng. Trong cụ thể, Nhìn vì cột chứa đựng không chuẩn mực nhân vật (thường xuyên đã hiển thị dưới dạng các hình chữ nhật trông nhỏ) và định dạng ngày tháng.
- Nếu như *đếm(\*)* là lớn, cái này “mười hàng” truy vấn cho phép chúng ta ĐẾN hiểu liệu cái hệ thống nguồn có Tốt phản ứng thời gian hoặc là rất chậm. Nếu như cái bàn là to lớn Nhưng Nó là phần vùng và được lập chỉ mục Tốt, chúng tôi nên trông chờ cái Đầu tiên “mười hàng” truy vấn ĐẾN trở lại TRONG ít hơn hơn hai giây. Con gấu TRONG tâm trí cái đó cái phản ứng thời gian là Mà còn ảnh hưởng qua khác các yếu tố như là như hệ thống trọng tải, phần cứng Và phần mềm cấu hình, điều tiết, Và Vì thế TRÊN.

Sau đó lựa chọn cái đã thay đổi dữ liệu qua đánh máy cái này TRONG cái SQL Yêu cầu Chữ cửa sổ:

```
lựa chọn * từ tiêu đề đơn hàng
Ở đâu (tạo > '2007-12-01 03:00:00'
Và tạo <= '2007-12-02 03:00:00')
hoặc (cập nhật lần cuối > '2007-12-01 03:00:00'
Và cập nhật lần cuối <= '2007-12-02 03:00:00')
```

Thay đổi ngày thành ngày hôm qua và ngày hôm nay, rồi nhấp vào Xem trước. Đừng lo lắng về các ngày được mã hóa cứng; sau trong chương này, bạn sẽ học cách làm cho điều này trở nên năng động bằng cách truy xuất và lưu trữ vào cơ sở dữ liệu siêu dữ liệu. Đừng lo lắng về rủi ro rằng nó sẽ trả về 400 triệu hàng TRONG 4 giờ; SSIS sẽ trở lại chỉ một cái Đầu tiên 200 hàng bởi vì SSIS giới hạn đầu ra khi xem trước dữ liệu. Cuộn xuống và cuộn sang phải để kiểm tra dữ liệu. Nếu mọi thứ có vẻ ổn, hãy thay đổi chế độ truy cập dữ liệu thành Bảng hoặc Xem, chọn *order\_header* làm tên bảng, nhấp vào Xem trước, nhấp vào Đóng, sau đó nhấp vào OK.

Cuộn xuống trong Hộp công cụ. Trong Data Flow Destination, nhấp đúp vào SQL Server Destination. Hộp SQL Server Destination sẽ xuất hiện trên bề mặt thiết kế. Nhấp vào hộp đó, nhấn F2 và đổi tên thành **Stage order header**. Thay đổi kích thước hộp nếu cần. Nhấp vào “Jade order tiêu đề” hộp, Và sự lôi kéo cái màu xanh lá mũi tên ĐẾN cái “Sân khấu đặt hàng tiêu đề” hộp. Nhấp đúp chuột cái “Sân khấu” hộp “tiêu đề đơn hàng”. Trong Danh sách thả xuống “OLE DB connection manager”, nhấp vào New. Trong hộp Data Connection, chọn servername.Stage.ETL và nhấp vào OK.

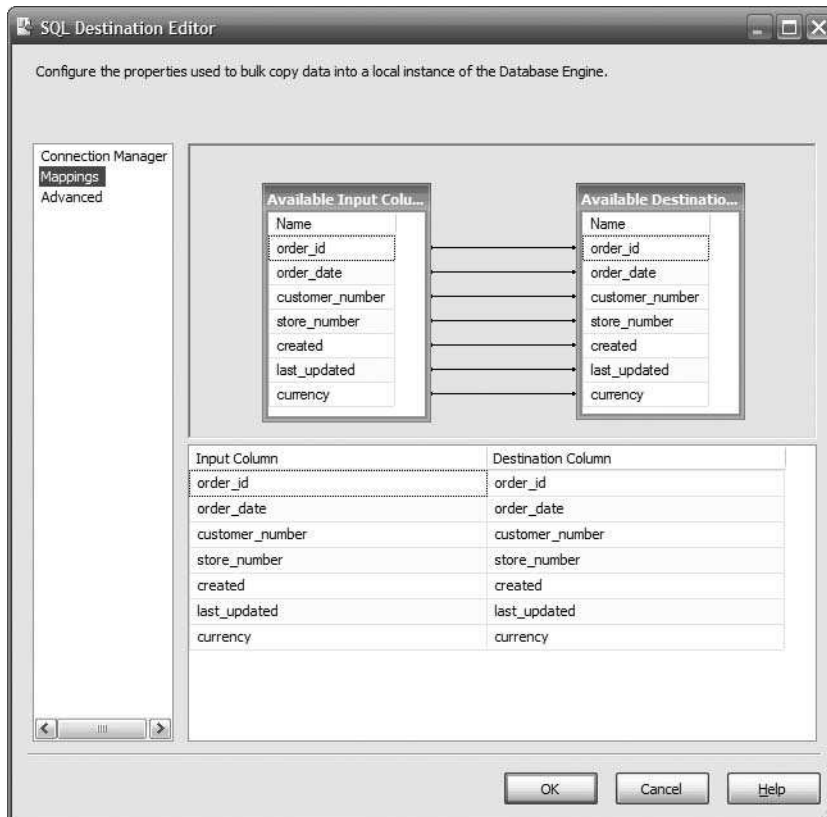
Bạn có thể tạo bảng đích trong giai đoạn này theo cách thủ công, dựa trên kiểu dữ liệu cột hệ thống nguồn. Bạn cũng có thể tạo nó ngay lập tức bằng cách sử dụng nút Mới ở bên phải bảng hoặc Xem danh sách thả xuống. Tôi thích tạo thủ công để có thể chắc chắn về các kiểu dữ liệu, tên bảng, tên cột, ràng buộc và vị trí vật lý/nhóm tệp như sau (tất nhiên bạn có thể sử dụng SQL Server Management Studio thay vì nhập câu lệnh SQL):

```

tạo nên bàn order_header
( order_id      int
, order_date    ngày giờ
, số_khách_hàng int
, số_cửa_hàng   int
, ngày giờ đã tạo
, ngày giờ cập nhật cuối cùng
) trên stage_fg2
đi

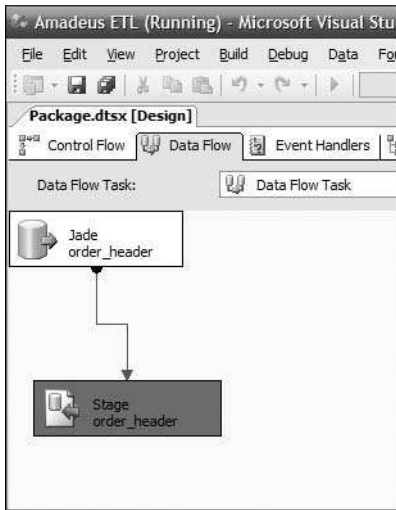
```

Nhấp vào OK trong cửa sổ Tạo bảng để tạo bảng *order\_header* trong cơ sở dữ liệu giai đoạn. Trong bảng hoặc chế độ xem, chọn *order\_header* mà chúng ta vừa tạo. Nhấp vào Xem trước và nó sẽ hiển thị một bảng trống. Nhấp vào Đóng, sau đó nhấp vào Ảnh xạ ở phía bên trái. Hình 7-10 hiển thị kết quả.



**Nhân vật 7-10.** Bản đồ nguồn cột ĐẾN cái sân khấu bàn cột

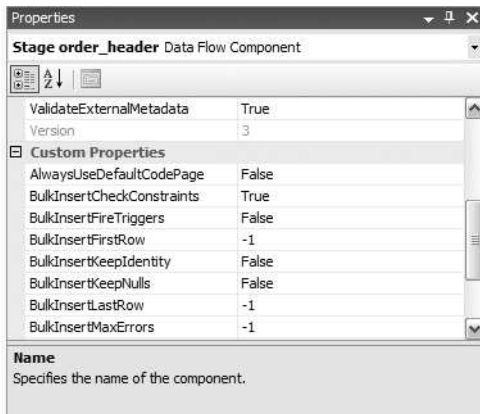
Nếu tên cột khác nhau, bạn cần ánh xạ chúng theo cách thủ công. Nếu tên cột giống nhau, chúng sẽ được ánh xạ tự động. Nhấp vào OK để đóng cửa sổ này. Nhấn F5 hoặc biểu tượng hình tam giác màu xanh lá cây trên thanh công cụ để chạy gói. Nó sẽ không thành công với hộp “Stage order\_header” được đánh dấu màu đỏ, như thể hiện trong Hình 7-11.



**Nhân vật 7-11.** Đang chạy cái SSIS bưu kiện


Nhấp vào tab Tiến trình. Bạn sẽ thấy nó nói rằng “Bạn không có quyền sử dụng tải trọng lớn tuyên bố.” Nhấn Shift+F5 hoặc cái màu xanh da trời hình chữ nhật ĐẾN dừng lại đang chạy cái bưu kiện. SSIS cho thấy Thực hiện Kết quả tab, hiển thị cái như nhau lỗi tin nhắn. Cái này là Một chung lỗi. Nguyên nhân là do người dùng *ETL* không có phần chèn lớn.

Nhấp vào tab Luồng dữ liệu, sau đó nhấp vào Hộp “Stage order\_header”. Cuộn xuống các thuộc tính ở phía bên phải, như thể hiện trong Hình 7-12.



**Nhân vật 7-12.** Số lượng lớn chèn của cái

Diểm đến SQL Server sử dụng lệnh chèn hàng loạt để tải dữ liệu vào các bảng SQL. Do đó, thông tin đăng nhập mà chúng tôi sử dụng cần có vai trò máy chủ bulkadmin. Bạn có thể sử dụng SQL Server Management Studio hoặc câu lệnh SQL để chỉ định vai trò này. Câu lệnh SQL là *sp\_addsrvrolemember 'ETL', 'quản trị viên hàng loạt'*.

Chạy lại cái bưu kiện. Hiện nay cả hai hộp TRONG cái Dữ liệu Chảy tab nên là màu xanh lá, nghĩa SSIS đó có thành công nhập khẩu những thứ kia hàng từ cái nguồn bàn ĐẾN cái sân khấu bàn. Nhấp chuột Dừng lại, hoặc nhấn Shift+F5. Nhấp chuột cái màu xanh da trời đĩa cái nút (hoặc lựa chọn Tài liệu  Cứu) ĐẾN cứu của bạn SSIS gói. Đi đến SQL Server Management Studio và truy vấn *order\_header* bảng trong cơ sở dữ liệu giai đoạn để xác minh rằng bạn có các hàng trong giai đoạn đó.

## Ghi nhớ cái Cuối cùng Chiết xuất Dấu thời gian

Ghi nhớ cái cuối cùng chiết xuất dấu thời gian có nghĩa cái đó chúng tôi ghi nhớ cái cuối cùng dấu thời gian của dữ liệu chúng tôi trích xuất để lần trích xuất tiếp theo chúng tôi có thể bắt đầu từ điểm đó. Chúng tôi vừa trích xuất thành công các bản ghi tiêu đề đơn hàng từ Jade vào giai đoạn. Nhưng chúng tôi đã sử dụng các giá trị cố định cho cái ngày phạm vi. Hiện nay hãy để cửa hàng cái cuối cùng chiết xuất dấu thời gian TRONG cái siêu dữ liệu cơ sở dữ liệu để có thể mọi thời gian cái ETL quá trình chạy Nó sẽ trích xuất khác biệt hồ sơ. Các ETL quá trình sẽ trích xuất chỉ một những thứ kia hồ sơ cái đó đã từng đã thêm sau đó cái cuối cùng trích xuất.

Trước chúng tôi bắt đầu, hãy để đi bởi vì MỘT Tổng quan của Gì chúng tôi là đang đi ĐẾN LÀM. TRONG cái Kế tiếp vài trang, chúng tôi sẽ sử dụng dấu thời gian trích xuất cuối cùng để kiểm soát lần trích xuất tiếp theo. Điều này cho phép chúng tôi trích xuất cái nguồn bàn tăng dần. ĐẾN LÀM cái đó, TRONG cái Kế tiếp một vài trang chúng tôi sẽ là đang làm sau đây các bước:

1. Tạo nên Một bàn TRONG cái siêu dữ liệu cơ sở dữ liệu ĐẾN cửa hàng cái dấu thời gian.
2. Tạo nên Một dữ liệu nguồn vì cái siêu dữ liệu cơ sở dữ liệu.
3. Biến đổi cái dữ liệu chảy BẢNG sau đây:
  - a. Lưu trữ hiện hành thời gian.
  - b. Lấy cái cuối cùng chiết xuất dấu thời gian từ cái bàn.
  - c. Thêm vào dấu thời gian ĐẾN cái chiết xuất truy vấn BẢNG tham số.
  - d. Cập nhật cái dấu thời gian TRONG cái bàn.
4. Thông thoáng cái mục tiêu bàn TRONG cái sân khấu cơ sở dữ liệu.
5. Bộ cái ban đầu dấu thời gian TRONG cái siêu dữ liệu bàn.
6. Thực thi gói để điền vào bảng mục tiêu. Trước

tiên, hãy tạo bảng *data\_flow* như sau:

*sử dụng  
meta dữ*

*nếu như tồn tại*

*( lựa chọn \* từ sys.tables ở*

*đâu tên = 'dòng dữ liệu' )*

*làm rơi bàn data\_flow*

*đi*

```

tạo nên bàn luồng dữ liệu
( id          int          không vô giá trị danh tính(1,1)
, tên        varchar(20) không vô giá trị
, Ngày giờ LSET
, Ngày giờ CET
, ràng buộc pk_data_flow
  chính chìa khóa nhóm lại
  (nhận dạng)
)
đi

```

```

tạo nên chỉ số data_flow_name
trên data_flow(tên)
đi

```

```

tuyên bố @LSET ngày giờ, @CET ngày giờ
thiết lập @LSET = '2007-12-01 03:00:00'
bộ @CET = '2007-12-02 03:00:00'
chèn vào trong luồng dữ liệu (tên, trạng thái, LSET,
  Giá trị CET) ('tiêu đề đơn hàng', 0, @LSET, @CET)
chèn vào trong luồng dữ liệu (tên, trạng thái, LSET,
  Giá trị CET) ('chi tiết đơn hàng', 0, @LSET, @CET)
chèn vào trong luồng dữ liệu (tên, trạng thái, LSET,
  Giá trị CET) ('khách hàng', 0, @LSET, @CET)
chèn vào trong luồng dữ liệu (tên, trạng thái, LSET,
  Giá trị CET) ('sản phẩm', 0, @LSET, @CET)
đi

```

---

• **Lưu ý** LSET đứng vì Cuối cùng Thành công Chiết xuất Dấu thời gian, Và Trung Quốc có nghĩa Hiện hành Dấu thời gian trích xuất .

---

Xác minh cái đó chúng tôi có cái đó *tiêu đề đơn hàng* hàng ngang TRONG cái *luồng dữ liệu* bàn: *lựa chọn* \* từ *data\_flow* . Bạn nên nhìn thấy bốn hàng, BẢNG đã hiển thị TRONG Bàn 7-6.

**Bàn 7-6.** luồng dữ liệu Bàn

nhậ n dạn g	tên	LSET	Trung Quốc
1	<i>tiêu đề đơn hàng</i>	2007-12-01 03:00:00.000	2007-12-02 03:00:00.000
2	<i>chi tiết đơn hàng</i>	2007-12-01 03:00:00.000	2007-12-02 03:00:00.000
3	<i>khách hàng</i>	2007-12-01 03:00:00.000	2007-12-02 03:00:00.000

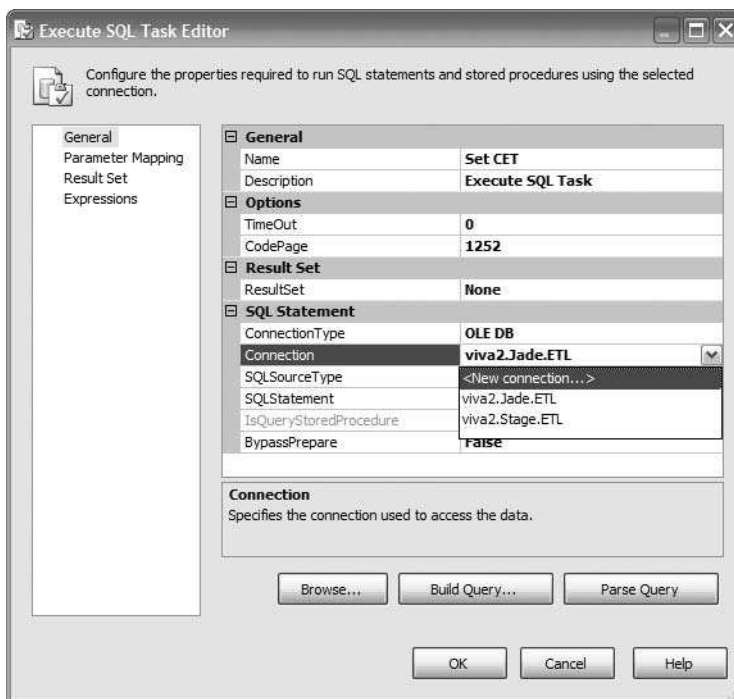
4	<i>sản phẩm</i>	2007-12-01 03:00:00.000	2007-12-02 03:00:00.000
---	-----------------	-------------------------	-------------------------

---

Được rồi, bây giờ chúng ta hãy sửa đổi tác vụ Luồng dữ liệu *order\_header* trong gói *Amadeus ETL* của chúng ta để rằng trước tiên nó đặt CET thành thời gian hiện tại và sau đó đọc LSET. Nếu luồng dữ liệu được thực thi thành công, chúng tôi cập nhật LSET và trạng thái cho luồng dữ liệu đó. Nếu không thành công, chúng tôi đặt trạng thái thành không thành công và chúng tôi không cập nhật LSET. Theo cách này, lần tiếp theo luồng dữ liệu được chạy, nó sẽ chọn các bản ghi từ cùng một LSET.

Vì vậy, hãy tạo một nguồn dữ liệu mới có tên là Meta kết nối với cơ sở dữ liệu siêu dữ liệu bằng cách sử dụng người dùng *ETL*. Quy trình này giống như lần trước. Đây là lý do tại sao trong phần trước, khi chúng tôi tạo cái Ngọc bích cơ sở dữ liệu, chúng tôi Mà còn được giao cái người sử dụng *ETL BẢNG chủ sở hữu db* TRONG cơ sở dữ liệu siêu dữ liệu. Bây giờ chúng ta sẽ đặt CET theo thời gian hiện tại cho tất cả các hàng.

1. Trong Hộp công cụ ở bên trái, nhấp đúp vào tác vụ Thực thi SQL. Sẽ có một hộp mới trên bề mặt thiết kế có nhãn Thực thi SQL Nhiệm vụ. Nhấp chuột phải và đổi tên hộp này thành **Đặt CET**.
2. Nhấp đúp vào hộp này. Trong phần Câu lệnh SQL, nhấp vào danh sách thả xuống Kết nối và chọn <Kết nối mới...>, như thể hiện trong Hình 7-13.



**Nhân vật 7-13.** Cấu hình cái sự liên quan vì cái Thực hiện SQL Nhiệm vụ

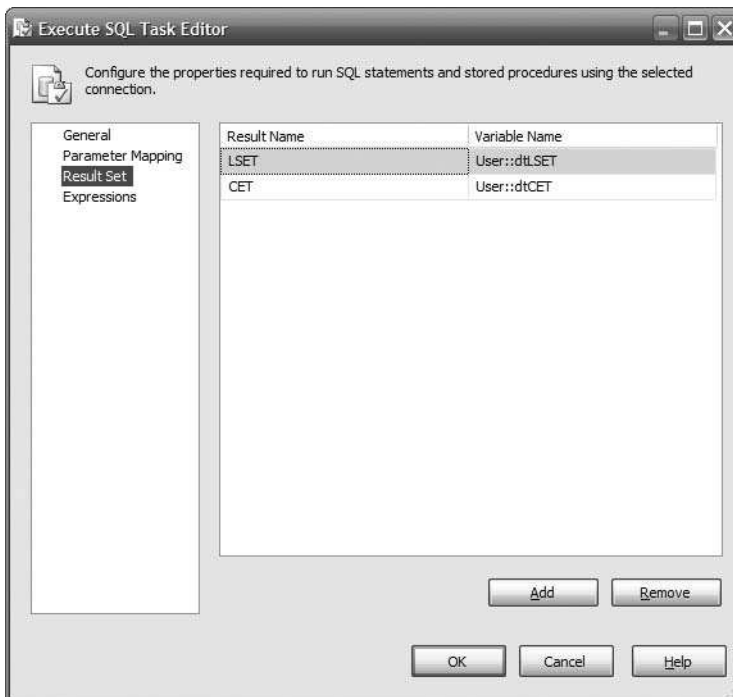
3. Chọn cái tên máy chủ.Meta.ETL dữ liệu sự liên quan, Và nhấp chuột ĐƯỢC RỒI.
4. TRONG cái SQL Tuyên bố cánh đồng, kiểu *cập nhật luồng dữ liệu bộ Trung Quốc* = lấy ngày() Ở đâu tên = 'order\_header'. Nhấp vào OK để đóng hộp thoại Execute SQL Task Editor.

Ở các bước từ 5 đến 7, chúng ta sẽ lấy thời gian trích xuất thành công gần đây nhất và

thời gian trích xuất hiện tại từ bảng siêu dữ liệu luồng dữ liệu.

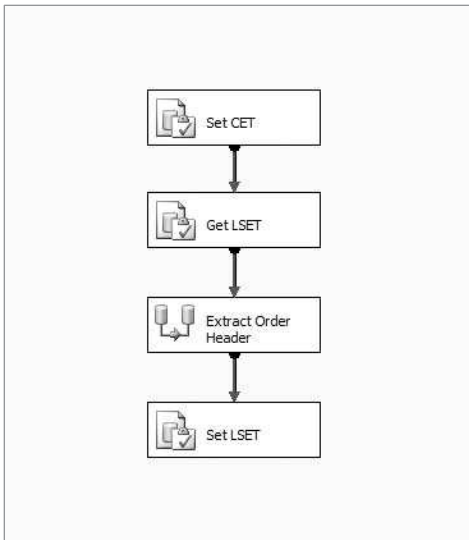


5. Trong Hộp công cụ, nhấp đúp vào tác vụ Thực thi SQL một lần nữa để tạo tác vụ mới trên bề mặt thiết kế. Đổi tên hộp này thành **Nhận LSET**.
6. Nhấp đúp vào hộp để chỉnh sửa. Đặt kết nối thành servername.Meta.ETL. Đặt Result-Set thành Single Row.
7. TRONG cái SQL Tuyên bố cánh đồng, kiểu **lựa chọn LSET, Trung Quốc từ luồng dữ liệu Ở đầu tên = 'order\_header'**. Nhấp vào Phân tích truy vấn để xác minh.  
Ở các bước từ 8 đến 12, chúng ta sẽ lưu trữ kết quả truy vấn vào các biến. Chúng sẽ được sử dụng sau này để giới hạn truy vấn tiêu đề thứ tự.
8. TRONG cái bên trái cột, nhấp chuột Kết quả Bộ. Nhấp chuột Thêm vào. Thay đổi Tên Kết quả Mới ĐẾN **LỚP 1**.
9. Nhấp chuột cái Biến đổi Tên tế bào, mở rộng cái thả xuống danh sách, Và chọn <Mới biến...>. Đặt vùng chứa thành Gói, nghĩa là phạm vi của biến là gói SSIS, không chỉ là tác vụ Lấy LSET, nghĩa là chúng ta có thể sử dụng biến trong các tác vụ khác.
10. Thay đổi cái tên ĐẾN **dtLSET**. Rời khỏi Không gian tên bộ ĐẾN Người dùng.
11. Thay đổi Giá trị Nhập vào **DateTime**. Đặt giá trị là **2007-10-01 00:00:00** (hoặc bất kỳ ngày nào bạn thích) và nhấp vào OK.
12. Nhấp vào Thêm một lần nữa để thêm một biến khác. Đặt Tên kết quả là **CET** và Tên biến là một biến mới có tên là **dtCET** thuộc kiểu DateTime. Kết quả sẽ trông giống như Hình 7-14.



**Nhân vật 7-14.** Lưu trữ cái kết quả bộ TRONG biến số

13. Nhấp chuột ĐƯỢC RỒI ĐẾN đóng cái Thực hiện SQL Nhiệm vụ Biên tập viên hộp thoại hộp. TRONG các bước 14 ĐẾN 16 chúng tôi sẽ kết nối cái nhiệm vụ TRÊN cái thiết kế bề mặt Và ngăn nắp hướng lên cái cách trình bày.
14. Kết nối mũi tên thành công màu xanh lá cây từ hộp Set CET với hộp Get LSET. Đổi tên tác vụ Data Flow thành **Extract Order Header** và kết nối mũi tên màu xanh lá cây từ hộp Get LSET với hộp này.
15. TRONG cái Hộp công cụ, nhấp đúp cái Thực hiện SQL nhiệm vụ ĐẾN tạo nên Một mới hộp, Và đổi tên Nó ĐẾN **Đặt LSET** . Kết nối cái màu xanh lá mũi tên từ Trích xuất Đặt hàng Tiêu đề hộp ĐẾN cái này Bộ LSET hộp.
16. Chúng ta hãy ngăn nắp hướng lên cái cách trình bày Một nhỏ bé chút. TRONG cái thực đơn thanh, lựa chọn Định dạng ☉ Tự động Cách trình bày ☉ Biểu đồ. Các kết quả trông giống Nhân vật 7-15 .



**Nhân vật 7-15.** Sắp xếp cái cách trình bày sau đó kết nối cái nhiệm vụ hộp

Ở các bước từ 17 đến 20, chúng ta sẽ sửa đổi để trích xuất tác vụ tiêu đề đơn hàng nhằm bao gồm dấu thời gian làm tham số để giới hạn truy vấn.

17. Nhấp đúp vào hộp Extract Order Header và trên bề mặt thiết kế Data Flow, nhấp đúp vào hộp Jade Order Header để chỉnh sửa.
18. Cập nhật văn bản lệnh SQL bằng cách thay đổi tất cả các ngày cố định thành dấu chấm hỏi, như sau:

*lựa chọn \* từ tiêu đề đơn hàng*

*Ở đâu (tạo > ? Và tạo <= ?)*

*hoặc (cập nhật lần cuối > ? Và cập nhật lần cuối <= ?)*

19. Nhấp vào nút Tham số và đặt tham số 0 thành User::dtLSET, tham số 1 thành User:dtCET, tham số 2 ĐẾN Người dùng::dtLSET, Và tham số 3 ĐẾN Người





**Nhân vật 7-16.** Làm cái ngày phạm vi TRONG cái Dữ liệu Chảy nhiệm vụ năng động

20. Nhấp vào OK và nhấp vào OK lần nữa để đóng cửa sổ và quay lại bề mặt thiết kế luồng dữ liệu.

Ở các bước từ 21 đến 22, chúng ta sẽ sửa đổi tác vụ Set LSET để cập nhật dấu thời gian được lưu trữ trong bảng *data\_flow*.

21. Trên bề mặt thiết kế, nhấp vào tab Control Flow. Nhấp đúp vào hộp Set LSET và đặt kết nối tới *servername.Meta.ETL*.
22. Bộ cái SQL tuyên bố ĐẾN *cập nhật luồng dữ liệu bộ LSET = Trung Quốc Ở đâu tên = 'order\_header'* và nhấp vào OK. Lưu gói bằng cách nhấn Ctrl+S.

Ở các bước từ 23 đến 25, chúng ta sẽ chuẩn bị thực thi gói bằng cách làm trống bảng mục tiêu và đặt giá trị dấu thời gian.

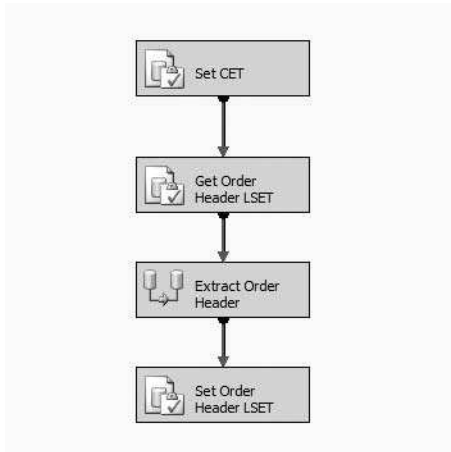
23. Trước khi chúng ta chạy gói, hãy xóa tất cả các hàng khỏi *order\_header* bảng trong giai đoạn, như sau: *cắt ngắn bàn giai đoạn.dbo.order\_header*.
24. Đặt cột CET cho *order\_header* trên luồng dữ liệu bảng trong cơ sở dữ liệu siêu dữ liệu đến ngày hôm qua và cột LSET đến ngày hôm kia, như sau: *cập nhật meta.dbo.data\_flow bộ LSET = lấy ngày()-2, Trung Quốc = lấy ngày()-1 Ở đâu tên = 'tiêu đề đơn hàng'*.
25. Xác minh cái hàng TRONG Của Jade *tiêu đề đơn hàng* bàn cái đó chúng tôi là đang đi ĐẾN trích xuất qua sử dụng cái tiếp theo truy vấn:

*lựa chọn \* từ ngọc bích.dbo.order\_header*  
*Ở đâu (tạo giữa lấy ngày()-1 Và lấy ngày()) hoặc*  
*(cập nhật lần cuối giữa lấy ngày()-1 Và lấy ngày())*

Ghi chú Làm sao nhiều hàng cái này truy vấn trả về.

TRONG các bước 26 ĐẾN 29, chúng tôi sẽ thực hiện cái buur kiện Và kiểm tra cái thực hiện kết quả.

26. Quay lại Business Intelligence Development Studio và nhấn F5 (hoặc nhấp vào hình tam giác màu xanh lá cây cái nút TRONG cái thanh công cụ) ĐỂ chạy cái buur kiện. Tất cả bốn hộp trở nên màu vàng và sau đó chuyển sang màu xanh lá cây từng cái một, như thể hiện trong Hình 7-17.

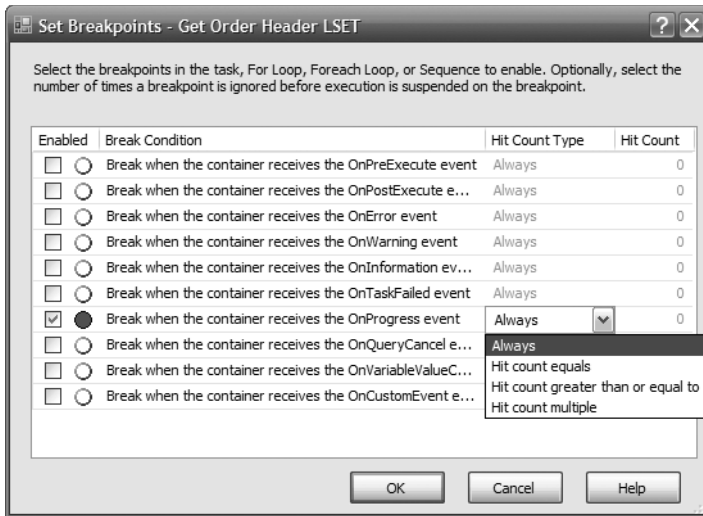


**Nhân vật 7-17.** Đang chạy SSIS buur kiện cái đó ghi nhớ cái cuối cùng chiết xuất thời gian

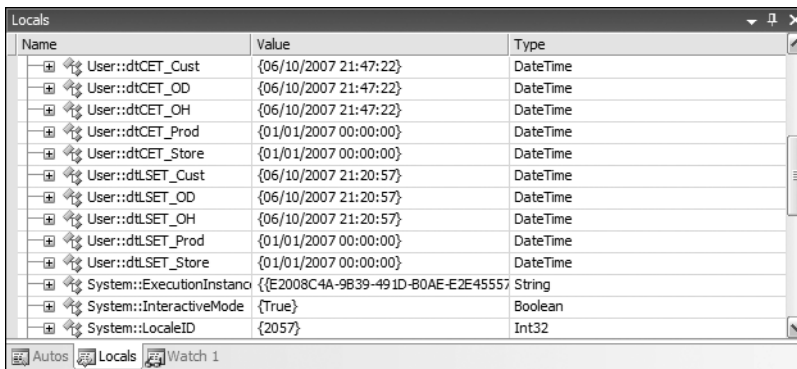
27. Nếu một trong số chúng chuyển sang màu đỏ, đừng hoảng sợ. Nhấp vào tab cuối cùng (Tiến trình) và xem thông báo lỗi. Bạn có thể gỡ lỗi bằng cách nhấp chuột phải vào các hộp trên luồng điều khiển và chọn Chính sửa điểm dừng. Hộp thoại Set Breakpoints sẽ bật lên, như thể hiện trong Hình 7-18, và bạn có thể chọn thời điểm bạn muốn breakpoint xảy ra. Bạn cũng có thể chỉ định breakpoint luôn xảy ra hay chỉ xảy ra ở một số lần nhất định, tức là số lần breakpoint bị bỏ qua trước khi thực thi dừng lại.

Bạn Có thể Mà còn nhìn thấy cái giá trị của cái biến đổi qua nhấp chuột cái Đồng hồ 1 hoặc Người dân địa phương các tab TRONG góc dưới bên trái góc của cái màn hình. Nhân vật 7-19 chương trình cái đó chúng tôi Có thể quan sát cái giá trị của các biến cục bộ khi thực thi bị tạm dừng tại điểm dừng. Ví dụ, giá trị của người sử dụng biến đổi *dtCET\_OH* là 06/10/2007 21:47:22 Và cái dữ liệu kiểu là Ngày giờ.

28. Nhấn Shift+F5 để dừng gói đang chạy. Nhấp vào tab Execution Results và xác minh rằng tất cả các tác vụ đã được thực hiện thành công.
29. Mở SQL Server Management Studio và kiểm tra xem các cột LSET và CET cho *order\_header* trong luồng dữ liệu bảng trong cơ sở dữ liệu siêu dữ liệu được cập nhật chính xác theo thời gian hiện tại. Ngoài ra, hãy kiểm tra xem *order\_header* trong cơ sở dữ liệu giai đoạn có chứa đúng số hàng từ bảng *order\_header* của *Jade* như đã lưu ý trước đó không.



**Nhân vật 7-18.** Cài đặt hướng lên Một điểm dừng



**Nhân vật 7-19.** Các Người dân địa phương của số hiển thị cái biến đổi giá trị Và dữ liệu kiểu

30. Hiện tại, tên của gói SSIS trong Solution Explorer là *Package.dtsx* . Đổi tên thành *order\_header.dtsx* bằng cách nhấp chuột phải vào nó. Khi được hỏi "Bạn có muốn đổi tên đối tượng gói không", hãy nhấp vào Có. Theo mặc định, SSIS chỉ đổi tên tệp gói, nhưng bên trong tên đối tượng không bị thay đổi (bạn có thể thấy tên đối tượng bằng cách nhấp vào Quan điểm thực đơn Và lựa chọn Cửa sổ Thuộc tính). Qua trả lời Đúng, chúng tôi đổi tên đối tượng gói cũng như tệp gói.
31. Nhấp vào menu File và chọn Save All để lưu toàn bộ giải pháp. Nhấp vào menu File và chọn Close Project để đóng gói.

Với điều đó, chúng ta đã hoàn thành việc thực hiện các bước cơ bản để trích xuất dữ liệu gia tăng từ một quan hệ. cơ sở dữ liệu. Trước chúng tôi đóng cái này chương, hãy để LÀM Một dữ liệu trích xuất từ Một tài liệu bởi vì trong các dự án kho dữ liệu thực tế, điều này xảy ra rất nhiều.

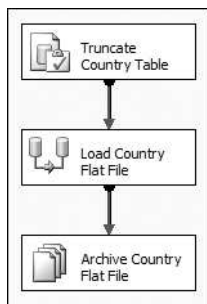
## Trích xuất từ Tập tin

Bây giờ chúng ta đã trích xuất dữ liệu từ cơ sở dữ liệu, hãy thử trích xuất dữ liệu từ các tệp. Trong vài trang, chúng tôi sẽ trích xuất dữ liệu từ Một phẳng tài liệu. Vì cái mục đích của cái này bài tập, chúng tôi có Quốc gia ISO dữ liệu cung cấp ĐẾN chúng ta TRONG phân cách bằng dấu gạch ngang tập tin. Chúng tôi sẽ nhập khẩu cái này tài liệu ĐẾN cái sân khấu cơ sở dữ liệu. Các tài liệu sẽ là cung cấp một lần Một tuần, từ Một bên ngoài dữ liệu nhà cung cấp.

Ở các trang sau, chúng ta sẽ tạo một gói SSIS để nhập tệp phẳng vào cơ sở dữ liệu giai đoạn. Sau đó nhập khẩu thành công ĐẾN cái sân khấu, chúng tôi sẽ lưu trữ cái tài liệu qua đi chuyển Nó đến một người khác thư mục. Chúng tôi sẽ hoàn thành cái này qua đang làm cái tiếp theo các bước:

1. Tải về cái phẳng tài liệu chứa đựng cái Tiêu chuẩn ISO quốc gia dữ liệu.
2. Mở cái phẳng tài liệu ĐẾN nhìn thấy cái nội dung.
3. Tạo nên cái mục tiêu bàn TRONG cái sân khấu cơ sở dữ liệu.
4. Tạo nên Một mới Gói SSIS .
5. Tạo nên Một Thực hiện SQL nhiệm vụ ĐẾN cắt ngắn cái mục tiêu bàn.
6. Tạo nên Một Dữ liệu Chảy nhiệm vụ ĐẾN trọng tải cái phẳng tài liệu ĐẾN cái sân khấu cơ sở dữ liệu.
7. Tạo nên Một Tài liệu Hệ thống nhiệm vụ ĐẾN lưu trữ cái tài liệu.
8. Thực hiện cái bưu kiện.

Gói SSIS mà chúng tôi đang xây dựng trông giống như Hình 7-20. Nó cắt bớt bảng mục tiêu, trích xuất dữ liệu từ tệp phẳng quốc gia, sau đó lưu trữ tệp.



**Nhân vật 7-20.** SSIS bưu kiện ĐẾN trích xuất dữ liệu từ tài liệu

Được rồi, chúng ta hãy bắt đầu. Trước tiên hãy tải tệp phẳng từ trang web của cuốn sách tại <http://www.apress.com/> . Nó là xác định vị trí TRONG cái thư mục /dữ liệu/bên ngoài . Các tài liệu tên là *country.dat* (8KB). Đặt tệp này vào ổ đĩa cục bộ của bạn, ví dụ: *c:\* .

Chúng ta hãy mở cái *quốc gia.dat* tài liệu sử dụng Sổ tay ĐẾN nhìn thấy cái nội dung. Nó trông giống cái này:

*Mã* | *Quảng*  
*cáo quốc*  
*gia* | *Andorra*  
*ae* | *Hoa Kỳ Tiếng Ả Rập*  
*Emirates af* | *Afghanistan*  
*ag* | *Antigua Và Barbuda*  
*Ai* | *Anguilla*  
*al* | *Albania*

Gồm có hai cột và được phân cách bằng dấu gạch ngang. Cột đầu tiên là mã quốc gia ISO gồm hai ký tự và cột thứ hai là tên quốc gia. Dòng đầu tiên chứa tiêu đề cột: *Mã* cho cột đầu tiên và *Quốc gia* cho cột thứ hai.



Bây giờ chúng ta đã biết nội dung và định dạng của tệp, hãy tạo gói SSIS để tải cái này tải liệu vào trong cái sân khấu. Nhưng trước cái đó, chúng tôi nhu cầu ĐẾN tạo nên cái sân khấu bàn Đầu tiên. Nếu như chúng tôi nhìn vào cái dữ liệu tải liệu, cái dài nhất quốc gia tên là 34 nhân vật (Thánh Tập sách (Sao Tập) và Principe), vì vậy đối với bài tập này, chúng ta sẽ đặt chiều rộng của cột tên quốc gia là 50 ký tự. Chúng ta hãy mở Sự quản lý Phòng thu Và tạo nên cái mục tiêu bàn gọi điện *quốc gia* TRÊN cái cơ sở dữ liệu giai đoạn như sau:

```

tạo nên bàn quốc gia
( mã quốc gia      char(2)
, country_name     varchar(50)
)

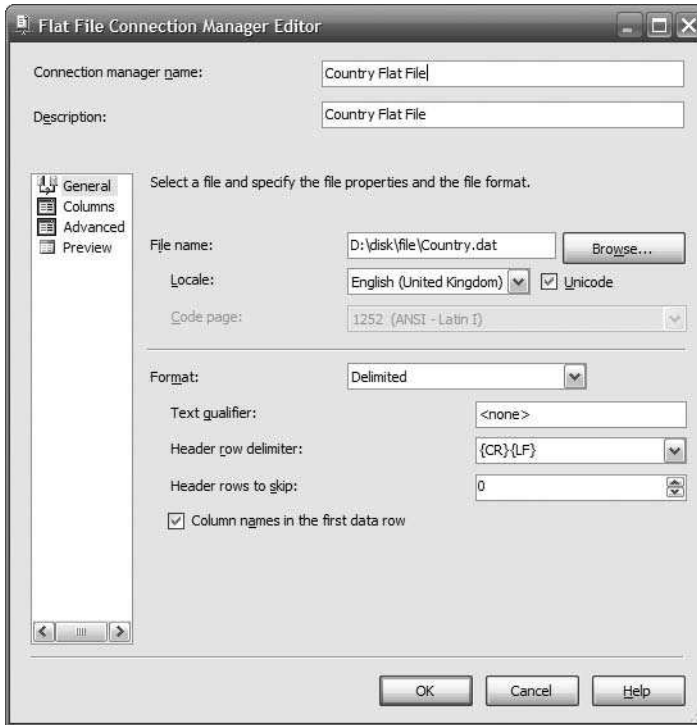
```

Hiện nay chúng tôi Có thể tạo nên cái SSIS buu kiện:

1. Mở Business Intelligence Development Studio và mở dự án Amadeus ETL (chọn Tài liệu  Mở  Dự án, điều hướng ĐẾN *Amadeus ETL.sln* , Và nhấp đúp Nó).
2. TRONG Giải pháp Nhà thám hiểm, nhấp chuột phải SSIS Gói hàng, Và chọn Mới SSIS Bưu kiện. MỘT bề mặt thiết kế trống mở ra.
3. ĐẾN là có thể ĐẾN chạy cái buu kiện nhiều lần, chúng tôi nhu cầu ĐẾN cắt ngắn cái mục tiêu bàn trước khi chúng ta tải dữ liệu. Nhấp vào tab Kiểm soát luồng và nhấp đúp vào tác vụ Thực thi SQL trong Hộp công cụ.
4. Đổi tên hộp **Truncate Country Bảng** và nhấp đúp vào hộp để chỉnh sửa. Đặt kết nối tới servername.Stage.ETL.
5. Đặt câu lệnh SQL để *cắt bớt bàn quốc gia* và nhấp vào OK. Nhấp vào OK một lần nữa để đóng hộp thoại Execute SQL Task Editor.
6. Bây giờ chúng ta hãy tạo một tác vụ Luồng dữ liệu để tải tệp phẳng vào cơ sở dữ liệu giai đoạn. nhấp vào Luồng dữ liệu Nhiệm vụ trong Hộp công cụ và đổi tên hộp mới thành **Load Country Flat File** . Kết nối mũi tên màu xanh lá cây từ Truncate Country Table đến Load Country Flat File.
7. Nhấp đúp vào Tài tệp phẳng quốc gia để chỉnh sửa. Hãy thêm tệp phẳng quốc gia làm nguồn. Nhấp đúp vào Nguồn tệp phẳng trong Hộp công cụ và đổi tên hộp mới thành **Tệp phẳng quốc gia** .
8. Nhấp đúp chuột cái Quốc gia Phẳng Tài liệu hộp, Và nhấp chuột cái Mới cái nút.

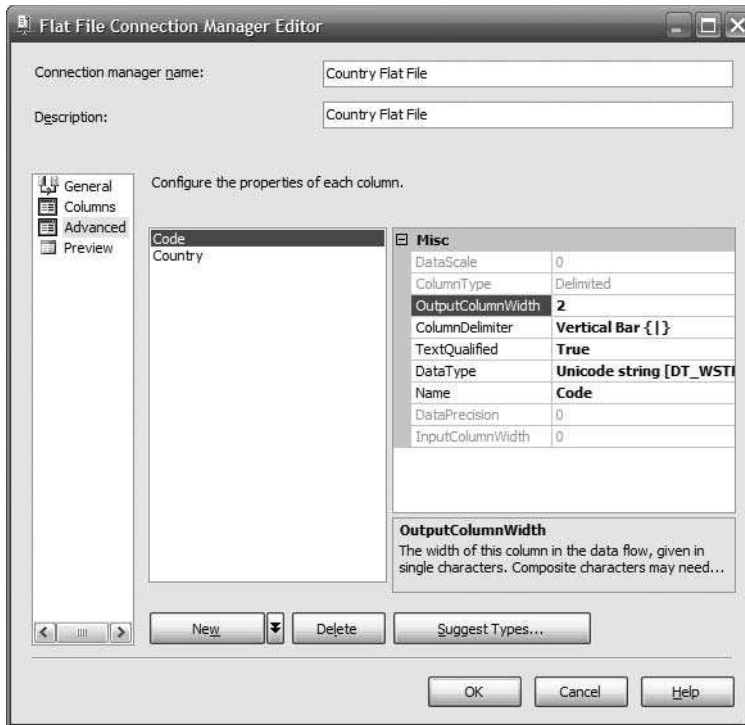


9. Hộp thoại Trình biên tập Trình quản lý kết nối tệp phẳng mở ra, như thể hiện trong Hình 7-21. Đặt cái tên Và Sự miêu tả BẢNG Quốc gia Phẳng Tài liệu, nhấp chuột Duyệt, Và chọn đất *nước.dat* tài liệu Bạn đã tải về trước đó. Bộ cái khác của cái BẢNG đã hiển thị trên Hình 7-21.



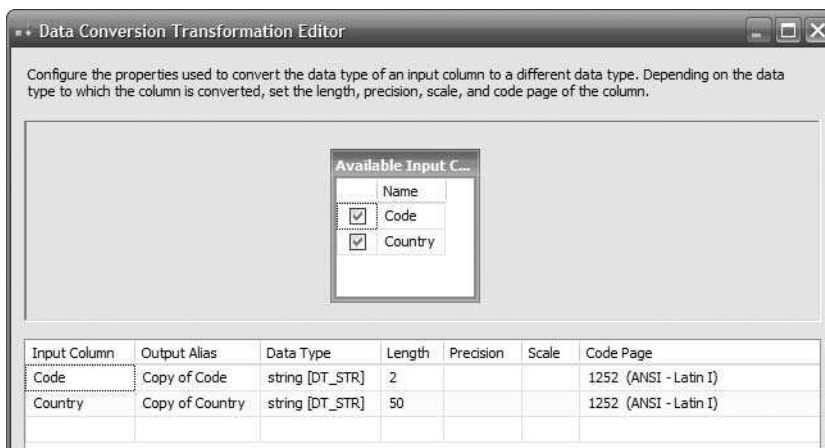
**Nhân vật 7-21.** Cấu hình cái phẳng tài liệu sự liên quan của cái

10. TRONG cái khung cửa sổ TRÊN cái bên trái, nhấp chuột Cột, Và chỉ rõ cái cột dấu phân cách BẢNG Thăng đứng Thanh { | }. Rời khỏi cái hàng ngang dấu phân cách BẢNG {CR}{LF}.
11. Vì mã quốc gia rộng hai ký tự nên chúng ta cần thiết lập cột đầu ra chiều rộng tương ứng. Trong trang bên trái, nhấp vào Advanced và thay đổi Output-ColumnWidth cho cột Code từ 50 thành 2, như thể hiện trong Hình 7-22. Để chiều rộng cột Country là 50 vì cột này có 50 ký tự, như chúng ta đã thảo luận trước đó.
12. Nhấp chuột ĐƯỢC RỒI ĐẾN đóng cái Phẳng Tài liệu Sự liên quan Giám đốc Biên tập viên Và trở lại ĐẾN cái Phẳng Cửa sổ File Source Editor. Nhấp vào Preview để xem dữ liệu mẫu, sau đó nhấp vào Close. Nhấp vào OK để quay lại bề mặt thiết kế.
13. Bây giờ chúng ta hãy tạo một phép chuyển đổi Dữ liệu để chuyển đổi đầu ra tệp phẳng, là Unicode, để phù hợp với cột trên bảng mục tiêu, đó là chuỗi ký tự ANSI. Nhấp đúp cái Dữ liệu Chuyển đổi sự biến đổi TRONG Hộp công cụ, Và kết nối mũi tên màu xanh lá cây từ Tệp phẳng quốc gia sang Chuyển đổi dữ liệu.



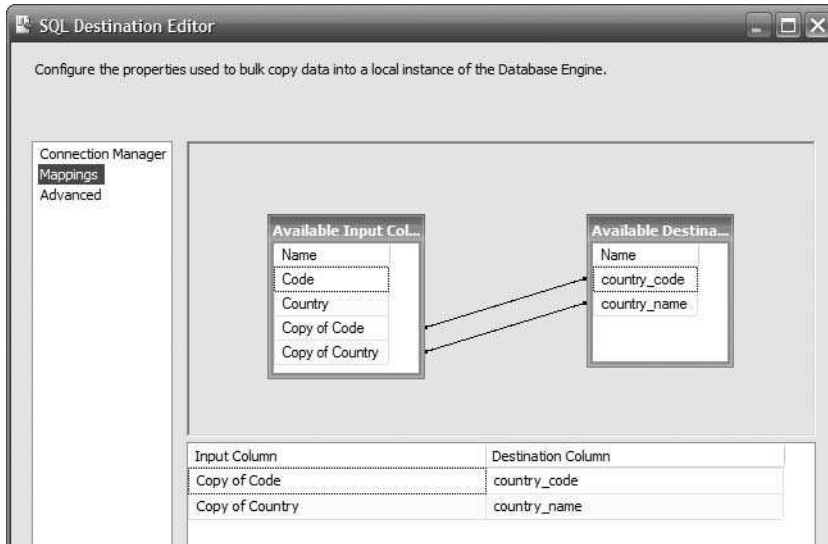
**Nhân vật 7-22.** Cấu hình cái của cái của mỗi cột

14. Nhấp đúp vào Chuyển đổi dữ liệu để chỉnh sửa và kiểm tra cả cột Mã và Quốc gia trong Cột đầu vào khả dụng. Trong Dữ liệu Nhập cột, chọn chuỗi [DT\_STR] vì cả hai cái Mã số Và Quốc gia cột, BẢNG đã hiển thị TRONG Nhân vật 7-23.



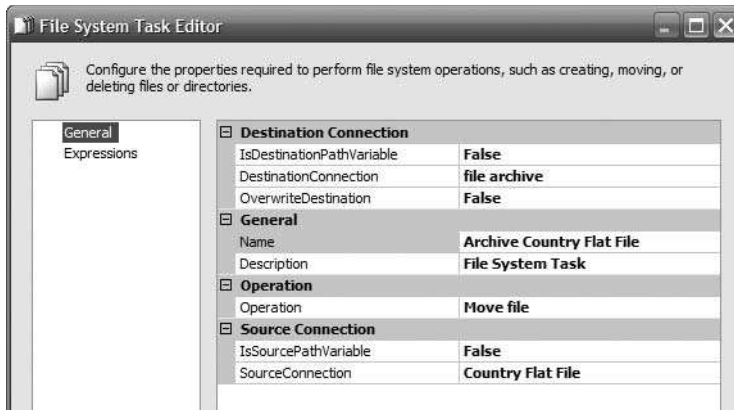
**Nhân vật 7-23.** Dữ liệu Chuyển đổi Sự biến đổi Biên tập viên

15. Trong Hộp công cụ, nhấp đúp vào SQL Server Destination và đổi tên hộp mới thành **Country Stage** . Kết nối mũi tên màu xanh lá cây trên hộp Data Conversion với hộp Country Stage. Nhấp đúp vào hộp Country Stage và đặt trình quản lý kết nối OLE DB thành servername.Stage.ETL.
16. Đặt “Sử dụng bảng hoặc chế độ xem” thành *quốc gia* mà chúng ta đã tạo trước đó. Trong bảng điều khiển bên trái, nhấp vào Mappings. Nhấp và kéo cột Copy of Code bên trái đến *mã quốc gia* cột bên phải. Nhấp và kéo Sao chép cột Quốc gia bên trái vào cột *country\_name* bên phải, như thể hiện trong Hình 7-24.

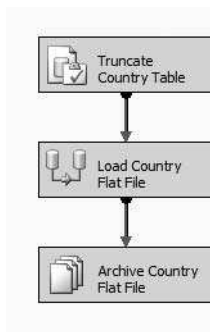


**Nhân vật 7-24.** Bản đồ cái nguồn cột ĐẾN cái mục tiêu cột

17. Nhấp vào OK để quay lại bề mặt thiết kế. Nhấp vào tab Control Flow ở trên cùng của bề mặt thiết kế. Chúng ta cần thêm một nhiệm vụ nữa ở đây, đó là lưu trữ tệp phẳng quốc gia bằng cách di chuyển nó đến một thư mục khác.
18. Nhấp đúp chuột cái Tài liệu Hệ thống nhiệm vụ TRONG cái Hộp công cụ, Và đổi tên cái mới tạo hộp  
**Lưu trữ Tệp phẳng quốc gia** .
19. Nhấp đúp chuột cái Lưu trữ Quốc gia Phẳng Tài liệu hộp ĐẾN biên tập Nó. Bộ Nguồn Sự liên quan để chỉ ra ĐẾN cái quốc gia phẳng tài liệu. Bộ Điểm đến Sự liên quan ĐẾN điểm ĐẾN MỘT trống thư mục. Đặt Hoạt động ĐẾN Di chuyển Tài liệu, BẢNG đã hiển thị TRÊN Nhân vật 7-25.
20. Nhấp vào OK để quay lại bề mặt thiết kế. Hãy sắp xếp lại bố cục một chút. Nhấp vào menu Edit và chọn Select All. Nhấp vào menu Format, chọn Auto Layout và chọn Diagram. Ba hộp bây giờ đã được căn chỉnh và cách đều nhau.
21. Nhấp vào menu File và chọn Save All để lưu toàn bộ giải pháp. Nhấn F5 để chạy gói. Bây giờ nó sẽ chạy OK. Mỗi hộp sẽ chuyển sang màu vàng rồi chuyển sang màu xanh lá cây, như thể hiện trong Hình 7-26.



**Nhân vật 7-25.** Cấu hình cái Tài liệu Hệ thống nhiệm vụ ĐẾN lưu trữ cái tài liệu



**Nhân vật 7-26.** Nhập khẩu dữ liệu quốc gia từ phẳng tập tin vào cái sân khấu

Thậm chí mặc dù chúng tôi là không trích xuất tăng dần đây, chúng tôi vẫn có ĐẾN thêm vào cái bit siêu dữ liệu như chúng ta đã làm trước đó khi trích xuất tiêu đề đơn hàng. Điều này có nghĩa là chúng ta cần lưu trữ dấu thời gian chạy thành công cuối cùng trong *data\_flow* bảng. Ghi lại kết quả của từng luồng dữ liệu ETL là quan trọng Vì thế cái đó khi cái ETL thất bại Và chúng tôi muốn ĐẾN khởi động lại từ sự thất bại, chúng tôi biết ở đâu ĐẾN khởi động lại từ bởi vì chúng tôi biết cái mà nhiệm vụ đã từng thành công Và cái mà nhiệm vụ thất bại để chạy. Chúng tôi thực hiện điều này bằng cách lưu trữ kết quả thực thi trong cùng một *data\_flow* bảng siêu dữ liệu như được hiển thị trong Nhân vật 7-27. (TÔI sẽ là đang thảo luận siêu dữ liệu TRONG Chương 10.)

key	name	status	LSET	CET
1	stg_order_header	2	2007-10-06 21:20:57.730	2007-10-06 21:47:22.750
2	stg_order_detail	2	2007-10-06 21:20:57.730	2007-10-06 21:47:22.750
3	stg_customer	2	2007-10-06 21:20:57.730	2007-10-06 21:47:22.750
4	stg_product	2	2007-10-06 21:20:57.730	2007-10-06 21:47:22.750
5	stg_product_status	1	2007-04-22 13:13:58.560	2007-04-22 13:13:58.560
6	stg_product_category	1	2007-04-22 13:13:58.560	2007-04-22 13:13:58.560
7	stg_product_type	1	2007-04-22 13:13:58.560	2007-04-22 13:13:58.560
27	stg_country	1	2007-10-07 16:43:19.733	2007-10-07 16:43:19.733

**Nhân vật 7-27.** Các luồng dữ liệu siêu dữ liệu bàn của hàng thực hiện trạng thái, cái LSET, Và cái Trung Quốc

## Bản tóm tắt

Chương này bắt đầu với các phương pháp tiếp cận và kiến trúc cho các hệ thống ETL, tức là ETL, ELT, sử dụng máy chủ ETL chuyên dụng hay không, đẩy so với kéo, lựa chọn máy chủ nào để đặt các quy trình ETL, v.v. Sau đó, chương này sẽ đi qua các loại hệ thống nguồn khác nhau, chẳng hạn như cơ sở dữ liệu quan hệ, bảng tính, tệp phẳng, tệp XML, nhật ký web, tệp nhật ký cơ sở dữ liệu, hình ảnh, dịch vụ web, hàng đợi tin nhắn và email. Chương này cũng thảo luận về cách trích xuất gia tăng toàn bộ bảng mỗi lần bằng cách sử dụng các phạm vi cố định và cách phát hiện rò rỉ dữ liệu.

Sau đó, tôi chỉ cách trích xuất dữ liệu tiêu đề đơn hàng từ Jade vào giai đoạn bằng SSIS. Chúng tôi đưa ra một số quy trình để ghi nhớ bản trích xuất cuối cùng để chúng tôi có thể trích xuất gia tăng. Chúng tôi cũng trích xuất tệp phẳng quốc gia vào cơ sở dữ liệu giai đoạn bằng cách sử dụng SSIS.

Cái này chương đã đưa cho Bạn Một Tốt sự thành lập của dữ liệu chiết xuất nguyên tắc, BẢNG Tốt BẢNG Một hiểu biết rõ ràng của Làm sao ĐẾN LÀM Nó sử dụng Viện nghiên cứu an toàn giao thông liên bang (SIS) TRONG cái Kế tiếp chương, chúng tôi sẽ mang đến cái dữ liệu cái đó chúng tôi đã nhập vào giai đoạn NDS và DDS.