

ITA-102 Hurtownie Danych

Marcin Gorawski, Michał Jan Gorawski

Moduł 1

Wersja 1.0

Proces ekstrakcji danych (ETL) I

Spis treści

Proces ekstrakcji danych (ETL) I	1
Informacje o module	3
Przygotowanie teoretyczne	4
Przykładowy problem	4
Podstawy teoretyczne.....	4
Wybrane zagadnienia procesu ekstrakcji.....	6
Selekcja danych.....	6
Czyszczenie danych i standaryzacja	6
Transformacja	7
Integracja	7
Ładowanie tablic wymiarów	8
Zarządzanie wymiarami zmiennymi.....	8
Ładowanie tablic faktów	9
Ładowanie danych historycznych	9
Agregacja.....	10
Odtwarzanie – powtórne ładowanie	10
Zarządzanie ekstrakcją.....	11
Przetwarzanie równoległe	11
Przykładowe rozwiązanie.....	12
Porady praktyczne	12
Uwagi dla studenta	12
Dodatkowe źródła informacji.....	13
Laboratorium podstawowe	14
Problem 1 (czas realizacji 30 min).....	14

Problem 2 (czas realizacji 15min) - kontynuacja problemu 1	19
Laboratorium rozszerzone	21
Zadanie 1 (czas realizacji 30 min).....	21
Zadanie 2 (czas realizacji 60 min).....	21

Informacje o module

Opis modułu

W module tym znajdziesz informacje dotyczące zagadnień związanych z procesem ekstrakcji danych. Poznasz podstawową wiedzę na temat ekstrakcji danych. Zobaczysz, czym jest oraz jak zaprojektować proces ETL (Ekstrakcja, Transformacja i Ładowanie danych) w środowisku SQL Server 2008.

Cel modułu

Celem modułu jest przedstawienie możliwości użycia pakietu Integration Services, jednego z komponentów SQL Server 2008, przy projektowaniu i implementowaniu procesu ETL o przeciętnej złożoności. Zadaniach ETL tego modułu uwzględniają proces czyszczenia.

Uzyskane kompetencje

Po zrealizowaniu modułu będziesz:

- wiedział, czym jest oraz jak projektować proces ETL,
- potrafił zbudować mało złożony proces ETL w SQL Server 2008,
- rozumiał mechanikę tworzenia procesów ekstrakcji danych w SQL Server 2008.

Wymagania wstępne

Przed przystąpieniem do pracy z tym modulem powinieneś:

- dobrze orientować się w zagadnieniach baz danych,
- znać zasady pracy w środowisku Visual Studio.

Mapa zależności modułu

Przy rozpoczęciu pracy z modulem 1 nie jest wymagana znajomość innych modułów.

Przygotowanie teoretyczne

Przykładowy problem

Jesteś głównym specjalistą bazodanowym w dużej firmie. Jak dotąd zarząd firmy nie zdecydował się na budowę hurtowni danych, lecz coraz częściej słyszysz rozmowy o korzyściach, jakie niesie ze sobą wprowadzenie tej technologii do firmy. W końcu postanowiłeś sam poszerzyć swoją wiedzę i zaczerpnąć informacji na temat hurtowni danych, oraz poznać różnicę pomiędzy hurtownią danych a bazą danych. Dotąd traktowałeś bazę danych głównie, jako repozytorium danych. Dowiedziałeś się, że hurtownia danych to o wiele więcej. Oprócz przechowywania wysokiej jakości danych, hurtownia udostępnia cały szereg analiz jakie można wykonać korzystając z przechowywanych danych, oraz przygotować na podstawie tych analiz raporty. Od razu zauważyłeś jedną ważną informację jaka podawana była we wszystkich źródłach – podstawą implementacji hurtowni danych jest umiejętne zaprojektowanie i wdrożenie procesu ETL (ang. Extraction, Transformation, Load) czyli procesu Ekstrakcji, Transformacji i Ładowania danych.

Podstawy teoretyczne

Zespół czynności związanych z przenoszeniem danych z systemów źródłowych do hurtowni danych zwany jest potocznie ekstrakcją danych. W procesie ekstrakcji możesz wyróżnić trzy odrębne etapy:

- pobranie danych z systemu źródłowego (*ang. Extraction*),
- przekształcenie danych do pożądanej postaci (*ang. Transformation*),
- załadowanie danych do hurtowni (*ang. Load*).

Stąd też często programy ekstrakcji określane są mianem systemów ETL.



Rys. 1. Kroki budowania hurtowni danych: proces ETL

Na rysunku 1 możesz zaobserwować wielowarstwową architekturę hurtowni danych z zaznaczonym procesem ETL. Warstwę ETL tworzą aplikacje procesu ETL, aplikacje zarządzające (harmonogramowanie, mapowanie, logowanie, monitorowanie, odświeżanie, odtwarzanie), obszar pośredni (*ang. Data Stage Area*) oraz repozytorium metadanych. Aplikacje procesu ETL tworzą zbiór kolejno następujących po sobie zadań ekstrakcji tj.:

- Seleksja i pobranie danych (ekstrakcja wymiarów i faktów).
- Transformacja wymiarów.
- Nadanie nowych kodów wartości atrybutów w tablicach wymiarów.
- Kontrola jakości i czyszczenie danych.

- Transformacja faktów.
- Konwersja indeksów na kody wartości atrybutów.
- Eliminacja błędnych rekordów.
- Standaryzacja danych.
- Agregacja danych.
- Ładowanie tablic wymiarów, faktów, agregacji:
 - ładowanie tablic wymiarów (przeniesienie danych do wybranej statycznej tablicy wymiarów i zmiennej tablicy wymiarów oraz przeniesienie danych do pozostałych tablic wymiarów);
 - ładowanie tablic faktów (algorytmy załadowania danych historycznych i ładowania przyrostowego;
 - ładowanie agregatów (algorytmy agregacji danych);
- analizę logów bazy danych (automatyzacja procesu ładowania (głównie przyrostowego)).

W obszarze pośrednim dane są przygotowywane, przetwarzane, a następnie przenoszone do hurtowni danych. Jest to miejsce, gdzie dokonywana jest selekcja i pobranie danych, a następnie ich transformacja, standaryzacja i czyszczenie. Obszar pośredni może być zrealizowany jako: relacyjna baza danych, zbiór plików lub może być rozumiany jako pamięć operacyjna komputera, na którym wykonywana jest ekstrakcja. To, który z wariantów zostanie wybrany zależy od wielu czynników np. ilości danych oraz ich pochodzenia. Gdy danych jest niewiele lub znajdują się na tym samym komputerze co hurtownia danych, można wybrać metodę wykorzystującą pamięć operacyjną komputera. Obszar pośredni nigdy nie stanowi bazy danych, do której kierowane są zapytania użytkowników.

Gdy dane zostały już wstępnie przetworzone i zgromadzone w obszarze pośrednim, następuje ich integracja zgodnie z formatem wymagany w hurtowni danych oraz załadowanie do systemu docelowego. Z zagadnieniem procesu ETL wiąże się również problem przyrostowego, okresowego ładowania danych do hurtowni, czyli odświeżania danych oraz ładowania uzupełniającego danych, w których wcześniej wykryto błędy.

Aby rozpocząć przenoszenie danych muszą być wcześniej spełnione określone warunki:

- rozpoznanie struktury systemów transakcyjnych,
- wstępna selekcja danych - specyfikacja danych źródłowych do hurtowni,
- rozpoznanie wymagań stawianych hurtowni danych i określenie transformacji wykonywanych na danych źródłowych,
- zdefiniowanie architektury i modelu danych,
- implementacja baz hurtowni danych.

Dopiero po spełnieniu tych wymagań możesz rozpocząć analizę problemu ekstrakcji danych.

Ekstrakcja jest procesem silnie obciążającym systemy komputerowe, stąd celowe jest stosowne do potrzeb, przygotowanie zasobów przeznaczonych na rzecz tworzenia hurtowni danych. Warto również ustalić środowisko pracy programistów, terminologię i nazewnictwo. Wszystkie te elementy powinny być zdefiniowane przed przystąpieniem do właściwych prac projektowych i realizacyjnych. Ze względu na wielość i różnorodność zadań silny nacisk należy położyć na szczegółowe dokumentowanie tego procesu.

Generalnie projektowanie procesu ekstrakcji zamyka się w następujących zadaniach:

- Selekcja (i pobranie danych z heterogenicznych źródeł zewnętrznych).
- Czyszczenie i standaryzacja (wykrywanie błędów i ich poprawa).
- Transformacja (konwersja danych do formatu obowiązującego w HD).
- Integracja (budowa map transformacji).

- Ładowanie (sortowanie, sumowanie, konsolidowanie, wyliczanie widoków, sprawdzanie integralności, budowanie indeksów oraz partycji).
- Odświeżanie (uaktualnianie danych w hurtowni na podstawie danych źródłowych).
- Odtwarzanie (ładowanie uzupełniające HD po jej „upadku”).
- Zarządzanie ekstrakcją.
- Ekstrakcja równoległa.

Poniżej omówimy niektóre z nich.

Wybrane zagadnienia procesu ekstrakcji

Selekcja danych

Podstawowym zadaniem jest odszukanie i identyfikacja danych przydatnych w tworzonej hurtowni. Wiele danych transakcyjnych nie jest potrzebnych w hurtowni. Dopiero na bazie reguł biznesowych możesz ocenić przydatność danych, znajdujących się w archiwach. Zadanie to jest żmudne i wymaga dużej cierpliwości. Podczas analizy danych historycznych warto być zbierał informacje statystyczne o systemach transakcyjnych, z których będą czerpane informacje. Dane te przydadzą Ci się do oszacowania czasu ładowania i lepszego rozplanowania zadań.

Czyszczenie danych i standaryzacja

W całym procesie ekstrakcji musisz zwrócić szczególną uwagę na kontrolę poprawności danych. Przed jej rozpoczęciem musisz określić kryteria, wg których będziesz oceniać poprawność załadowanych danych. Definiując te kryteria musisz uwzględnić:

- zgodność danych pomiędzy systemami źródłowymi a danymi w hurtowni,
- kompletność danych (odpowiada na pytanie: czy wszystkie dane w systemach źródłowych mają swoje odpowiedniki w hurtowni danych),
- wewnętrzna spójność danych w hurtowni,
- unikalność, która gwarantuje niepowtarzalność danych w hurtowni,
- częstotliwość aktualizacji danych w hurtowni (powinna być wystarczająca na potrzeby analizy danych i wspomagania podejmowania decyzji).
- W danych pochodzących z systemów źródłowych występuje wiele usterek, które wymagają korekty. Typowe usterki to:
 - niekonsekwentne używanie kodów i znaczników,
 - niektóre identyfikatory mogą mieć różną interpretację,
 - dane mogą być zapisane niepoprawnie albo ten sam obiekt może być dwukrotnie wprowadzony do bazy z różnymi nazwami (duplikaty, rekordy sprzeczne).
 - Niektóre błędy można usunąć przed rozpoczęciem ekstrakcji, przez:
 - identyfikację i wybór najbardziej wiarygodnych źródeł danych,
 - sprawdzenie, jaka część danych źródłowych jest niepoprawna,
 - współpracę z projektantami systemu transakcyjnego przy usuwaniu wad w danych,
 - współpracę z użytkownikami systemów transakcyjnych w zakresie uzupełniania danych.

Pozostałe usterki musisz usuwać w obszarze pośrednim, przed załadowaniem danych do HD.

Ogólnie, czyszczenie danych dotyczy kilku faz tj.:

- Analizy danych - celem wykrycia wszystkich rodzajów błędów i niekonsekwencji, które są do usunięcia; programy analizujące powinny korzystać z metadanych;
- Definiowania strumienia transformacji i reguł mapowania - w zależności od liczby źródeł, stopnia ich heterogeniczności i stopnia „zabrudzenia” danych, musi być wykonana duża ilość transformacji danych i akcji czyszczenia. Schemat translacji używany jest do mapowania źródeł we wspólny model danych HD w postaci relacyjnej. Pierwsze kroki czyszczenia danych

mogą poprawić jakość przygotowywanych danych w pojedynczym źródle danych do ich integracji. Dalsze kroki zajmują się schematami/danymi integracji i usuwaniem problemów wystąpień wielu źródeł np. duplikatów. Dla HD, kontrola i strumień danych dla tych transformacji oraz kroki czyszczenia powinny być określone w strumieniu prac, który definiuje proces ETL;

- Weryfikacji - poprawność i efektywność strumienia transformacji i definicji transformacji powinny być testowane i oceniane np. na próbkach lub kopii danych źródłowych, aby poprawić definicje jeśli będzie to konieczne. Wiele iteracji kroków analizy, projektowania i weryfikacji może być potrzebne, ponieważ np. tylko kilka błędów pozostało widocznych po zastosowaniu kilku transformacji;
- Poprawiania danych - powrotny strumień czyszczonych danych - po usunięciu błędów, oczyszczone dane powinny zastąpić „brudne” dane również w oryginalnych źródłach w celu uniknięcia potrzeby ponownej pracy czyszczenia dla przyszłych ekstrakcji danych.

Fazy czyszczenia powinny być określone przez deklaracyjny język zapytań i mapowania, aby umożliwić automatyczną generację kodu transformacji.

Standaryzacja danych zapewnia konwersję wartości atrybutów w spójny i jednolity format, co ułatwia dopasowanie i integrację danych. Np. wpisy daty i czasu powinny być zebrane w specyficznym formacie; nazwy i inne ciągi danych powinny być przekonwertowane albo na duże albo na małe znaki itd. Dane tekstowe mogą być zebrane i zunifikowane przez wykonanie tematykacji, usuwanie prefiksów, sufiksów i słów przystankowych. Co więcej, skróty i schematy kodowania powinny być spójnie rozwiązywane przez konsultacje ze słownikami synonimów lub zastosowanie predefiniowanych reguł konwersji.

Transformacja

Proces transformacji wymaga oczywiście dużych ilości metadanych, takich jak: schematy, poziomy wystąpień charakterystyk danych, mapowania transformacji, definicji przepływu danych, informacje o dziedzicznych obiektach transformowanych i ich zmianach, itd. Dla spójności, elastyczności i łatwości ponownego użycia, te informacje powinny być utrzymywane w repozytorium metadanych (rys.1).

Przykładami transformacji danych z obszaru pośredniego do postaci wymaganej w HD mogą być operacje:

- nadania unikalnych indeksów,
- konwersji typów danych (np. z EBCDIC na ASCII),
- dopracowania szczegółów (np. kapitalizacja pierwszych liter w imionach i nazwiskach).

O ile ostatnie dwa przykłady wymagają jedynie klasycznych mechanizmów konwersji, to nadanie nowych indeksów ma znaczenie szczególne. W systemach transakcyjnych indeksy mają różnorodną postać, często nieodpowiednią dla przetwarzania w HD. Zdarza się, że dany wymiar pochodzi z wielu źródeł. Rzadko przy tym poszczególne wartości atrybutów wymiarowych mają unikalne indeksy. Konieczne jest wówczas stworzenie nowych indeksów, unikalnych w ramach hurtowni. Dlatego opracowuje się specjalną mapę konwersji: z indeksów transakcyjnych na indeksy hurtowni. Każda tabela wymiarów, dla której tworzone są indeksy powinna posiadać taką mapę.

Integracja

Pierwszym Twoim zadaniem jest stworzenie schematów ekstrakcji. Twój wstępny schemat powinien zawierać bardzo ogólne założenia: z jakich modułów, jakie dane (jakościowo i ilościowo) oraz dokąd są przenoszone. Kolejne uściślenia poszczególnych ścieżek pozwalają na otrzymanie szczegółowej mapy przekształceń danych transakcyjnych na dane hurtowni (mapa transformacji). Taka strategia tworzenia schematów ekstrakcji i transformacji metodą zstępującą chroni Cię przed przypadkową utratą pewnych połączeń. Dopiero wówczas możliwe będzie zdefiniowanie

szczegółowych algorytmów transformacji danych. Staną się one podstawą praktycznej implementacji.

Jeśli będziesz zajmował się problemami wielu źródeł musisz zintegrować schematów, wykonując takie kroki jak dzielenie, scalanie, składanie, rozkładanie atrybutów i tablic. Na przykład, zadanie eliminacji duplikatów jest typowo wykonywane po większości transformacji i kroków czyszczenia pojedynczych źródeł danych. Eliminacja jest wykonywana na co najmniej dwóch czyszczonych źródłach w tym samym czasie lub na pojedynczym już zintegrowanym zbiorze danych. Eliminacja duplikatów wymaga identyfikacji podobnych rekordów dotyczących tego samego obiektu, a następnie ich scalanie w jeden rekord zawierający wszystkie istotne atrybuty (bez redundancji). Na tym etapie pracy warto abyś dokonał wyboru narzędzi programistycznych. Im wcześniej takiego wyboru dokonamy, tym bardziej nasz schemat będzie dostosowany do narzędzi, których będziemy używać. Pozwoli to ograniczyć konieczność ponownego projektowania niektórych transformacji.

Ładowanie tablic wymiarów

Ładowanie tablic wymiarów jest drugim etapem procesu ekstrakcji danych. Na początek powinieneś wybrać najprostszą statyczną tablicę wymiaru. Statyczną tzn. niezmienną, w której zestaw atrybutów jest stały i niezmienny. Powodem, dla którego dobieramy najprostszy przypadek jest ograniczenie ewentualnych trudności do problemów związanych z komunikacją, bezpieczeństwem i transferem danych. Istotna jest tutaj kwestia wyboru metody tworzenia i transmisji wyodrębnionych danych. Stosuje się zasadniczo dwie metody: przez plik i przez strumień.

Kiedy dane zostaną poprawnie przygotowane, możesz przystąpić do ładowania tablic wymiarów. Dzięki przygotowaniu danych w obszarze pośrednim proces ładowania jest stosunkowo prosty, jednak warto stosować pewne stałe reguły postępowania:

- Musisz unikać posługiwania się klasycznymi poleceniami wstawiania rekordów do bazy jako nieefektywnymi z powodu transakcyjnych cech relacyjnych baz danych. Większość RDBMS'ów realizuje transakcje przez jednoczesne zapisywanie zmian w rejestrach transakcyjnych, co czyni wszelkie operacje bardzo czasochłonnymi.
- Powinieneś używać programów do masowego wprowadzania danych (*ang. loader*).
- Wyłącz OLTP – nawet, jeśli ładowanie się nie powiedzie, to możliwe jest powtórzenie ładowania – pliki źródłowe znajdują się w obszarze pośrednim.
- Posortuj pliki źródłowe – operacja ta znacznie przyspiesza tworzenie indeksów, natomiast powinieneś unikać przekształcania danych podczas ładowania – nawet wykorzystanie dedykowanych programów wspomagających ładowanie może być nieefektywne, bowiem rekordy są przetwarzane i ładowane pojedynczo.
- Usuń indeksy przed ładowaniem i utwórz ponownie po zakończeniu ładowania – to zalecenie jest istotne, gdy ładowana jest duża porcja danych. Jako granicę przyjmuje się poziom objętości indeksów rzędu 10-15 procent bazy, po przekroczeniu którego warto indeksy usunąć i po ładowaniu odtworzyć.
- Poprawnie zdefiniuj parametry bazy danych – większość obecnych baz danych automatycznie alokuje miejsce pod tworzenie nowych zapisów w bazach HD.
- Określ odpowiednie wartości wskaźników wypełnienia i przygotować odpowiednią ilość wolnego miejsca.

Gdy ładowanie najprostszej tablicy wymiarów zakończy się powodzeniem, można przystąpić do opracowania algorytmu ekstrakcji jednej wybranej zmiennej tablicy wymiarów.

Zarządzanie wymiarami zmiennymi

Wymiary zmienne charakteryzują się zmianami opisów, dotyczących tych samych wartości atrybutów, które znajdują się już w bazach hurtowni. Możesz wyróżnić trzy rodzaje obsługi sytuacji ze zmieniającymi się wymiarami:

- Typ 1: Nadpisanie – „stara” wartość atrybutu znajdująca się w tablicy wymiarowej jest nadpisywana przez wartość „nową”. Ten model postępowania powoduje utratę poprzedniej wartości atrybutu. Przykładem takiego atrybutu jest zmodyfikowana nazwa tego samego produktu.
- Typ 2: Utworzenie nowego rekordu w tablicy wymiarów – nowa wartość wymiaru otrzymuje nowy indeks w tablicy hurtowni, wraz z nowym opisem.
- Typ 3: Umieszczenie nowego opisu w dodatkowych polach rekordu tablicy wymiarów – w tym przypadku poprzedni i nowy opis atrybutu egzystuje w różnych polach tego samego rekordu tablicy wymiarów.

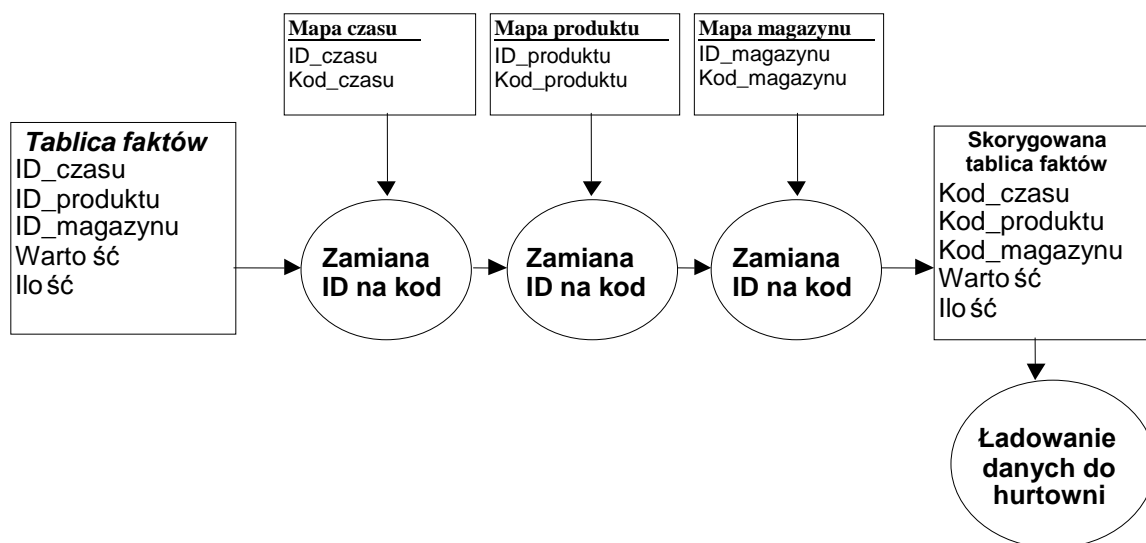
Zarządzając tablicami wymiarów możesz posłużyć się każdą z powyższych technik.

Wszystkie powyższe techniki zakładają posiadanie wiedzy o tym, że poszczególne wymiary zostały zmienione. Najczęściej stosowaną, zapewniającą najlepsze efekty jest technika typu 2.

Po przeprowadzeniu ładowania jednej z wybranych tablic wymiarów: statycznej i zmiennej musisz przeprowadzić ekstrakcję pozostałych wymiarów. Jest to realizowane w oparciu o doświadczenia uzyskane w dotychczasowych działaniach. Niektóre tablice wymiarów są znacznych rozmiarów i ich ekstrakcja charakteryzuje się cechami wspólnymi z ekstrakcją dużych tablic faktów.

Ładowanie tablic faktów

W proces ekstrakcji musisz uwzględnić podział na ładowanie pierwsze i przyrostowe. Ładowanie pierwsze zakłada przeniesienie do nowo tworzonej hurtowni danych historycznych i aktualnych. Ekstrakcja danych historycznych to proces ładowania i transformacji dużych zbiorów informacji. Ładowanie przyrostowe (zwane też odświeżaniem) to przeniesienie do istniejącej hurtowni danych aktualnych, na bieżąco zbieranych w systemach transakcyjnych.



Rys. 2.. Proces przekształcania tablic faktów.

Ładowanie danych historycznych

Pierwsze ładowanie oznacza zebranie danych historycznych, czasami pochodzących z wielu źródeł (dodatkowym utrudnieniem może być fakt przechowywania tych danych na nośnikach archiwizacyjnych np. taśmach). Proces przetwarzania tablic faktów ilustruje rys.2.

Tablice faktów muszą odnosić się do indeksów hurtowni, zawartych w tablicach wymiarów. Podczas przetwarzania tablic wymiarów zostaną utworzone nowe kody wartości atrybutów każdego z wymiarów. Indeksy w źródłowych tablicach systemów transakcyjnych należy odpowiednio przekształcić na nowe kody.

Cały proces ekstrakcji należy tak kontrolować, aby w tablicy faktów nie znalazły się wartości kodów, których nie ma w tablicy wymiarów. Sprawdzenia tego należy dokonywać programowo, a nie z wykorzystaniem wbudowanych mechanizmów relacyjnej bazy danych, jakimi są więzy integralności. Postępowanie takie jest wskazane, ponieważ ładowanie każdego rekordu pociąga za sobą jednocześnie sprawdzenie więzów integralności, co z kolei wydłuża czas działania.

Agregacja

Końcowym etapem ładowania danych do hurtowni jest agregacja danych. Rekordy w tablicach agregacji tworzone są na podstawie już wygenerowanych tablic faktów. Problem pojawia się, gdy ilość danych do agregacji jest bardzo duża. Nie możesz wówczas zagregować wszystkich danych, a jedynie te, które zostały załadowane podczas ostatniego ładowania. Masz tu do czynienia z agregacją przyrostową. Problemu nie możesz łatwo rozwiązać, gdy dane z ostatniego ładowania obejmują nie tylko informacje z ostatniego okresu. Gdy danych przyrostowych jest dużo i sięgają zapisów historycznych, należy przeprowadzić pełną agregację. Przy wyborze struktur agregujących jako projektant musisz kierować się następującymi przesłankami:

- tworzenie agregatów powinno wspomagać wykonanie raportów,
- tworzenie agregatów powinno wspomagać odpowiedzi na pytania użytkowników.

Innym zagadnieniem jest dobór narzędzi, z użyciem których agregacja będzie realizowana oraz utrzymywanie agregatów. Jeżeli wykorzystasz tutaj mechanizm bazy danych, pociąga za sobą uruchomienie mechanizmów transakcyjnych: logowania, tworzenia rejestrów transakcyjnych itp. Jednak realizacja struktur agregujących w oparciu o perspektywy zmaterializowane (Oracle, Informix Red Brick) pozwala na automatyczną (bez udziału operatora) aktualizację struktur bezpośrednio po wypełnieniu struktur bazowych.

Odtwarzanie – powtórne ładowanie

Jeśli w trakcie ładowania danych wystąpi błąd, to jednym z możliwych sposobów postępowania jest cofnięcie całej operacji i powtórzenie jej od nowa. Lepszym rozwiązaniem jest jednak kontynuowanie ładowania od miejsca, w którym zostało przerwane. W tym celu musisz rozpoznać, do jakiego miejsca ładowanie było poprawne i nie wymaga powtarzania. Niektóre algorytmy wznowienia ładowania wykorzystują śledzenie poprawności ładowania i wymagają dodatkowego narzutu na operacje: ładowania, specyficznych transformacji danych lub wysokich kosztów odtwarzania danych.

Akcja wznowienia procesu ekstrakcji polegająca na dokończeniu przerwane go procesu ładowania danych nazywa jest odtwarzaniem.

Jeżeli rozważysz tylko błędy na poziomie systemowym (np. błąd RDBMS, błąd oprogramowania, brak miejsca na dysku) i przyjmiesz, że jeśli w jednym z procesów wystąpił błąd, ładowanie danych zostanie wstrzymane, to jedynymi danymi stanowiącymi podstawę wznowienia procesu ładowania są dane załadowane do hurtowni oraz dane źródłowe.

Stosowane są koncepcje postępowania - metody odtwarzania tj.:

- Metoda podziału danych wejściowych (ang. *batching*) - dane wejściowe dzielone są na bloki danych (ang. *batch*) przetwarzane sekwencyjnie, blok po bloku. W razie przerwania procesu ekstrakcji odtwarzanie jest wznowiane od uszkodzonego bloku, pozostałe poprawnie załadowane bloki nie wymagają powtórzonego przetwarzania.
- Metoda kopii migawkowych i punktów powrotu - opisuje sposób tworzenia i wykorzystania okresowych kopii migawkowych (ang. *snapshots*) przetwarzanych danych oraz punktów powrotu (ang. *savepoints*). W razie przerwania ekstrakcji, każda transformacja danych wznowia swoje przetwarzanie od ostatniego punktu powrotu.

- Metoda podziału procesu ekstrakcji (ang. staging) - polega na dekompozycji procesu ekstrakcji na powiązane logicznie grupy podprocesów. Dane wyjściowe danej grupy są zapisywane i jednocześnie podawane na wejście kolejnej grupy podprocesów. Odtwarzanie polega na restarcie grupy podprocesów obejmujących wcześniej wykonane kopie danych wejściowych tej grupy.
- Metoda powtarzania procesu ekstrakcji od początku (ang. redo) – polega na wznowieniu od samego początku każdego przerwanej procesu ekstrakcji, natomiast proces ładujący dane do hurtowni filtruje przetworzone już krotki tak, aby do bazy hurtowni nie trafiały dane już tam zapisane przed wystąpieniem przerwania.
- Metoda odtwarzania Design-Resume (DR) - opisuje zmodyfikowaną metodę odtwarzania typu Redo. Algorytm DR wykorzystuje dane już przetworzone i załadowane do hurtowni danych przez przerwany proces ekstrakcji eliminując potrzebę powtórnego przetwarzania wszystkich danych wejściowych. Dane te są wykorzystywane w czasie filtrowania danych wejściowych i danych przetwarzanych przez wznowiony proces ekstrakcji. Takie podejście zasadniczo skraca czas odtwarzania.

Odrębnym zagadnieniem jest obsługa wyjątków, czyli sytuacji, w których wykryto błąd danych źródłowych. Zwykle procedura ekstrakcji, która rozpoznała, że rekord danych przygotowany do załadowania do hurtowni nie spełnia kryteriów poprawności, zapisuje ten rekord w innym zbiorze i sygnalizuje administratorowi procesu ekstrakcji wystąpienie niepoprawnej sekwencji danych wejściowych.

Zarządzanie ekstrakcją

Proces ekstrakcji danych nie jest zadaniem jednorazowym. Dlatego bardzo ważne jest zorganizowanie zarządzania ekstrakcją przyrostową (okresowe odświeżanie danych), jak i uzupełniającą (ładowanie danych, które nie zostały umieszczone w hurtowni z powodu wykrycia błędów ładowania). Z praktyki wiadomo, że automatyczna kontrola procesu ekstrakcji powinna obejmować:

- zdefiniowanie zadań do wykonania i określenie zależności między nimi,
- zaplanowanie wykonania zadań,
- monitorowanie i rejestrowanie wyników wykonania procesów elementarnych,
- obsługę wyjątków i błędów,
- szeregowanie procesów – kończący się proces powinien zainicjować następny,
- informowanie o zakończeniu procesów i zadań.

Dla przeprowadzenia automatycznej ekstrakcji wykorzystuje się najczęściej wbudowane w system mechanizmy automatycznego uruchamiania zadań. Zastosowanie specjalizowanych narzędzi ekstrakcji, które mają wbudowane mechanizmy kontroli wykonania zadań, znacznie upraszcza zarządzanie i automatyzację procesu ekstrakcji.

Przetwarzanie równoległe

Równoległe przetwarzanie stwarza ogromne możliwości zwiększenia wydajności procesu ETL. Zadania równoległe wykorzystują różne zasoby: procesor, pamięć, sieć, porty wejścia/wyjścia. Dlatego istotne jest wyodrębnienie niezależnych zadań, które wykorzystują różne zasoby. Jeśli jakaś operacja musi być poprzedzona wykonaniem innej, należy uwzględnić ten porządek wykonania – samo zrównoleglenie zadań nie zawsze powoduje przyspieszenia wykonania ekstrakcji.

Zagadnienie ekstrakcji równoległej możesz rozważać w następujących aspektach [4]:

- Wykorzystanie wbudowanych mechanizmów przetwarzania równoległego danych większości relacyjnych baz danych - szczególnie indeksowanie danych może być realizowane równoległe: każdy procesor działa na swojej niezależnej części bazy;

- Zastosowanie mechanizmów bliźniaczych baz danych - utrzymujemy dwie kopie hurtowni: jedną do ładowania danych, drugą do realizacji zapytań. Jeden serwer wykonuje ekstrakcję danych, drugi obsługuje zapytania. Po zakończeniu danego etapu ładowania, serwery i dane są automatycznie przełączane na hurtownię załadowaną aktualnymi danymi. Na drugim serwerze może się rozpocząć następne ładowanie;
- Zastosowanie mechanizmów bliźniaczych tabel - serwer bazy danych jest jeden, natomiast utrzymujemy dwie kopie tabel. Ładowanie odbywa się do jednej tabeli zwanej tablicą ładowania, a zapytania są kierowane do tablicy zapytań. Po załadowaniu nowych danych do tablicy ładowania, następuje zmiana jej nazwy na tablicę zapytań - poprzednia tablica zapytań jest natomiast usuwana.

Przykładowe rozwiązanie

Twoim zadaniem jest dokonanie ekstrakcji danych z pliku tekstowego. Plik jest w następującym formacie:

```
Imie Nazwisko DataUr
Jan Kowalski 1956
```

Twoim zadaniem będzie dokonanie ekstrakcji danych z pliku tekstowego przy użyciu pakietu Integration Services do tabeli bazy SQL Server 2008.

Po pierwsze w aplikacji Business Intelligence Development Studio tworzysz nowy projekt przy użyciu schematu Integration Services Project. Następnie W zakładce Data Flow przeciągasz komponent Data Source – Flat file source i definiujesz go odpowiednio aby wskazywał na nasz plik źródłowy z danymi.

Następnie aby odpowiednio rozdzielić dane musisz użyć kombinacji komponentów transformujących np Derived Column oraz Conditional Split i przekazujesz dane do komponentu Data Destination – SQL Server Destination.

Dokładny opis rozwiązania podobnego problemu, oraz bardziej skomplikowanych zagadnień omówiony został w podstawowej części laboratorium.

Porady praktyczne

Uwagi ogólne

- Pamiętaj, że przed implementacją konieczne jest drobiazgowo zaprojektowanie procesu ETL. (Szacuje się, że około 80% eksploatowanych hurtowni danych jest mało przydatna z powodu nieefektywnego procesu ETL.
- Bardzo dokładnie sprawdzaj efekty poprawnego procesu ETL – poprawne wykonanie procesu wcale nie znaczy otrzymania prawidłowych wyników.
- Pamiętaj o mechanizmach odtwarzania procesu ETL dla aktywnych hurtowni danych.

Uwagi dla studenta

Jesteś przygotowany do realizacji laboratorium jeśli:

- Rozumiesz ideę i zasady tworzenia procesu ETL
- Rozumiesz różnicę między zwykłą migracją danych a procesem ETL

Pamiętaj o zapoznaniu się z dodatkowymi informacjami z podanych niżej źródeł. Jeżeli coś jest dla Ciebie niejasne bądź dociekliwy i odszukaj dodatkowe informacje, które pogłębią twoją wiedzę.

Dodatkowe źródła informacji

1. Bruckner R., List B., Schiefer J.: *Striving Towards Near Real-Time Data Integration for Data Warehouses*. DaWaK 2002 pp. 317-32
2. Bruckner R., Tjoa A.M.: *Capturing Delays and Valid Times in Data Warehouses Towards Timely Consistent Analyses*. *Journal of Intelligent Information Systems*, 19,2, 2002.
3. Galhardas H., Florescu D., Shasha D., Simon E.: *Ajax: An Extensible Data CleaningTool*. In *Proc. ACM SIGMOD Intl. Conf. On the Management of Data, Teksas (2000)*.
4. Rahm E., Hai Do H., *Data Cleaning: Problems and current approaches*, *Bulletin of the Technical Committee on Data Engineering*, 23/2000
5. Schrefl, M., Thalhammer, T.: *On Making Data Warehouses Active*. *Proceedings of the 2nd International Conference DaWaK, Springer, LNCS 1874, 2000*.
6. Vassiliadis P., Simitsis A., Georgantas P., Terrovitis M.: *A Framework for the Design of ETL Scenarios*. CAiSE 2003.
7. Vassiliadis, P. A. Simitsis, S. Skiadopoulos. *Modeling ETL Activities asGraphs*. In*Proc. 4th Intl. Workshop on Design and Management of Data Warehouses, Canada, (2002)*.
8. Vassiliadis, P., Simitsis, A., Skiadopoulos S.: *Conceptual Modeling for ETL Processes*. DOLAP 2002.
9. Galhardas H., Florescu D., Shasha D., Simon E.: *Ajax: An Extensible Data CleaningTool*. In *Proc. ACM SIGMOD Intl. Conf. On the Management of Data, Teksas (2000)*.
10. Rahm E., Hai Do H., *Data Cleaning: Problems and current approaches*, *Bulletin of the Technical Committee on Data Engineering*, 23/2000
11. Scalzo, B.: *Oracle DBA guide to data warehousing and star schemas.*, NJ: Prentice Hall. 2003.
12. Labio W., Wiener J., Garcia-Molina H., Gorelik V.: *Efficient Resumption of Interrupted Warehouse Loads*, *SIGMOD Conference*, 2000.
13. Raden N.: *Real time: get Real. Take the idea of a real-time data warehouse with a grain of salt, then realize the possibilities*. *Intelligent Enterprise*, vol. 6, no.10, 2003.
14. Rosana L. de B. A. Rocha, Cardoso L., Souza J.: *An Improved Approach in Data Warehousing ETL Process for Detection of Changes in Source Data*. SBBD 2003, pp. 253-266.
15. Gorawski M., Ciepluch M.: *Ocena wydajności komponentów systemu przyrostowej ekstrakcji danych ETL(δ)*. *Bazy danych. Struktury, algorytmy, metody. Praca zbiorowa. Architektura, metody formalne i eksploracja danych*. Red. St. Kozielski [i in.]. Warszawa: Wydaw. Komunikacji i Łączności, 2006, s. 289-298.
16. Gorawski M., Ciepluch M.: *Przyrostowa ekstrakcja danych ETL (δ): aspekty implementacyjno-wydajnościowe*. *Bazy danych. Nowe technologie. Praca zbiorowa. [T. 1]: Architektura, metody formalne i zaawansowana analiza danych*. Red. St. Kozielski [i in.]. Warszawa: Wydaw. Komunikacji i Łączności, 2007, s. 115-124.
17. Gorawski M., Ciepluch M.: *Przyrostowa ekstrakcja danych ETL(δ)*. *Studia Informatica* 2006 vol. 27 nr 1, s. 27-40.
18. Gorawski M., Czmer J.: *Rozbudowa silnika ekstrakcji danych rtetl o mechanizm detekcji zmian źródłowych danych strumieniowych. Modele i zastosowania systemów czasu rzeczywistego*. *Praca zbiorowa*. Pod red. Z. Mazura, Z. Huzara. Warszawa: Wydaw. Komunikacji i Łączności, 2008, s. 220-231
19. Gorawski M., Piekarek M.: *Rozproszony proces ekstrakcji danych z protokołem SimpleRMI*. *Bazy danych. Modele, technologie, narzędzia. Praca zbiorowa. T. 2: Analiza danych i wybrane*






- zastosowania. Red. S. Kozielski [i in.]. Warszawa: Wydaw. Komunikacji i Łączności, 2005, s. 43-50.
20. Gorawski M., Piekarek M.: Rozwojowe środowisko ETL/Java Beans (Development environment ETL/JavaBeans). *Studia Informatica*, 2003, vol. 24, nr 4(56), s. 287-302.
21. Gorawski M., Piekarek M.: Rozwojowe środowisko ETL/JavaBeans wzbogacone o rozproszone sortowanie danych. *Współczesne problemy sieci komputerowych. Nowe technologie. Praca zbiorowa*. Red.: St. Węgrzyn, B. Pochopień, T. Czachórski. Warszawa: Wydaw. Naukowo-Techniczne, 2004, s. 173-180.
22. Gorawski M.: 3 perspektywy procesu ekstrakcji danych. *Strategie informatyzacji i zarządzanie wiedzą*. Red. Z. Szyjewski, J. S. Nowak, J. K. Grabara. Warszawa: Wydaw. Naukowo-Techniczne, 2004, s. 295-341.
23. Gorawski M.: Charakterystyka procesu ekstrakcji danych (The characteristics of data extraction process). *Studia Informatica*, vol. 24, nr 4(56), 2003, s. 211-232.
24. Gorawski M.: Ekstrakcja i integracja danych w czasie rzeczywistym. *Współczesne problemy systemów czasu rzeczywistego. Praca zbiorowa*. Red.: A. Kwiecień, P. Gaj. Warszawa: Wydaw. Naukowo-Techniczne, 2004, s. 435-445.
25. Gorawski M.: Laboratorium hurtowni danych poziomu MS SQL SEVER 2005. *Metody i narzędzia wytwarzania oprogramowania. Konferencja naukowa, Szklarska Poręba, 14-16 maja 2007*. Red. B. Hnatkowska, Z. Huzar. Wrocław: Oficyna Wydaw. Politechniki Wrocławskiej, 2007, s. 583-596.
26. Gorawski M.: Modelowanie procesu ekstrakcji danych (Modeling for extraction data processes). *Metody i systemy komputerowe w badaniach naukowych i projektowaniu inżynierskim. IV Krajowa konferencja, Kraków, 26-28 listopada 2003. Materiały konferencyjne*. Red.: R. Tadeusiewicz, A. Ligęza, M. Szymkat. Akademia Górniczo-Hutnicza, Politechnika Krakowska, Uniwersytet Jagielloński. Kraków: Oprogramowanie Naukowo-Techniczne, 2003, s. 165-170.
27. Gorawski M.: Praktyczne aspekty projektowania hurtowni danych (Practical aspects of data warehouse design) *Studia Informatica*, 2003, vol. 24, nr 4(56), s. 189-210.
28. Gorawski M., Marks P., Gorawski M.J.: Collecting data streams from a distributed radio-based measurement system. *Database systems for advanced applications. DASFAA 2008. 13th International conference, New Delhi, India, March 19-21, 2008*. Eds: J. R. Haritsa, R. Kotagiri, V. Pudi. Berlin: Springer, 2008, *Lecture Notes in Computer Science*. vol. 4947. s. 702-705.

Laboratorium podstawowe

Problem 1 (czas realizacji 30 min)



Dostałeś zlecenie na przygotowanie danych do zaprojektowania hurtowni danych w dziekanacie. Dane przechowywane są obecnie w plikach tekstowych. Twoim zadaniem jest uporządkowanie i oczyszczenie danych oraz przeniesienie ich do nowego pliku tekstowego. Nauczysz się jak używać transformacji (a) Podziału Warunkowego (**Conditional Split**) w celu wydobywania potrzebnych wierszy ze źródła, oraz (b) Formatowania Kolumny (**Derived Column**), w celu uporządkowania wydobytych wierszy.

Zadanie	Tok postępowania
1. Zapoznanie się z plikiem płaskim	<ul style="list-style-type: none">Przy użyciu Eksploratora Windows przejdź do folderu podanego przez prowadzącego np: C:\HD\UID

Studenci_Source.txt	<ul style="list-style-type: none"> Otwórz plik StudenciSource.txt. Plik zawiera listę nazwisk studentów, oraz numery ich indeksów. Dane są przechowywane w surowej formie, plik zawiera nagłówek oraz pusty wiersz przed właściwymi danymi. W tym ćwiczeniu dokonujemy ekstrakcji jedynie imienia i nazwiska studenta; czyli 1) odrzucenie wszystkich wierszy zaczynających się znakiem spacji oraz 2) odrzucenie wszystkich znaków znajdujących się za znakiem otwarcia nawiasu w każdym z wierszy Zamknij plik StudenciSource.txt.  Jeżeli plik nie zostanie zamknięty, przy uruchomieniu pakietu wystąpi błąd.
2. Tworzenie projektu Integration Services o nazwie Studenci	<ul style="list-style-type: none"> Wybierz zakładkę Data Flow (przepływu danych). Z menu głównego wybierz File -> New -> Project. Z listy Project Types wybierz Business Intelligence Projects.  Jeżeli na komputerze jest już zainstalowany Microsoft Visual Studio w wersji 2003, 2005 lub 2008, to oprócz wspomnianego Business Intelligence Projects na liście może pojawić się więcej niż jeden typ projektu. W Templates wskaż Integration Services Project. W polu Name podaj nazwę projektu (np. Studenci). Kliknij OK.  Jeżeli ukaże się powitalny wizard zamknij go.
3. Tworzenie zadania przepływu danych o nazwie Importuj Studentów	<ul style="list-style-type: none"> Wybierz zakładkę Data Flow (przepływu danych). Wybierz link znajdujący się na środku strony, aby dodać nowe zadanie. W panelu właściwości (Properties) zmień wartość wiersza Name na Importuj Studentów
4. Dodanie narzędzia Flat File Source	<ul style="list-style-type: none"> Otwórz Panel Narzędzi (Toolbox) i zapoznaj się z dostępnymi narzędziami  Obiekty narzędzi zorganizowano w trzy główne grupy: Źródła Przepływu Danych (Data Flow Sources), Transformacje Przepływu Danych (Data Flow Transformations), i Źródła Wynikowe Przepływu Danych (Data Flow Destinations). Przeciągnij Flat File Source (Źródłowy Plik Płaski) z Panelu Narzędzi do zakładki przepływu danych  Zwróć uwagę na czerwoną ikonę na tym elemencie. Usługi Integrujące dodają ten znacznik do obiektu, aby powiadomić o konieczności uruchomienia Menadżera Połączeń (Connection Manager), który pozwoli zadaniu połączyć się z zewnętrznym źródłem danych.
5. Dodanie Menadżera Połączeń o nazwie „ Źródło ”	<ul style="list-style-type: none"> Kliknij podwójnie na element Flat File Source, a następnie wybierz przycisk New, aby otworzyć Kreatora Połączeń. Wprowadź nazwę dla menadżera połączeń: Źródło Studentów.



<p>Studentów”</p>	<ul style="list-style-type: none"> • Użyj przycisku Przeszukaj (Browse), aby wybrać plik źródłowy z lokacji podanej przez prowadzącego np: C:\HD\UID \StudentSource.txt • Ustaw liczbę Opuszczonych Wierszy Nagłówek (Header rows to skip) o wartości 1. • Zaznacz pole Nazwy kolumn w pierwszym wierszu danych (Column names in the first data row). • Wybierz Kolumny (Columns) w liście z lewej strony. <p>Tutaj można zobaczyć podgląd wierszy pobieranych z pliku. Obecnie jest to jedna kolumna zdefiniowana jako domyślna. Należy wybrać zakładkę kolumny (Columns) aby łącznik mógł zdefiniować kolumny. Można również zmienić tu ograniczniki wiersza i kolumny, jeżeli jest to potrzebne. Następnie można otworzyć zakładkę Zaawansowane (Advanced) aby zmodyfikować osobno każdą z kolumn.</p> <ul style="list-style-type: none"> • Wybierz zakładkę Zaawansowane (Advanced). <p>Tutaj możesz wybrać właściwości dla każdej z kolumn.</p> <ul style="list-style-type: none"> • Zmień nazwę kolumny 0 na Student. • Zmień OutputColumnWidth na 250. • Wybierz OK a następnie Podgląd (Preview) aby zobaczyć wyniki. <ul style="list-style-type: none"> • Wybierz OK, aby zamknąć okno edytora Menadżera Połączeń.
<p>6. Mapowanie Connection Manager (Menadżera połączeń) do źródła danych (Data Adapter)</p>	<ul style="list-style-type: none"> • W Flat File Source Editor (Edytorze Źródła Pliku Płaskiego) wybierz Columns (kolumny). <p>Mapowanie pomiędzy kolumną zewnętrzną (dla Menadżera Połączeń) oraz kolumną wyjściową (dla Źródła Danych) jest generowane automatycznie.</p> <p>Teraz mamy Źródło Danych powiązane z Menadżerem Połączeń a całość gotowa jest do użycia w transformacjach. Można zauważyć, że proces nie jest zakończony, dopóki nie zmapujemy kolumn pomiędzy Menadżerem Połączeń oraz Źródłem Danych.</p>
<p>7. Dodanie transformacji Conditional Split (Podział Warunkowy)</p>	<p>W pierwszym obowiązkowym kroku następuje oddzielenie wierszy, które mają zostać zachowane, oraz wierszy, które mają zostać odrzucone. W tym przykładzie odrzuca się wiersze zaczynające się pustą przestrzenią (spacją – Nie posiadamy Nazwiska studenta, wyłącznie nr Indeksu). Najlepszym narzędziem do otrzymania tego efektu jest transformacja Podział Warunkowy (Conditional Split). Narzędzie to pobiera dane wejściowe i zależnie od warunku kieruje dane na właściwe wyjście.</p> <ul style="list-style-type: none"> • Na zakładce Data Flow (Przepływ Danych), przeciągnij z Toolbox’a (Panelu Narzędzi) narzędzie Conditional Split (Podział Warunkowy). • Wybierz Flat File Source (Źródłowy Plik Płaski) i przeciągnij jego zieloną strzałkę (Wyjście) do Conditional Split (Podziału Warunkowego). • Kliknij podwójnie na Conditional Split. • Zmień Default Output Name na Odrzucone Wiersze.


	<p>Wszystkie wiersze niespełniające warunku przekazywane są na domyślne wyjście, warto więc nadać mu nazwę. W środkowej części edytora (tabela) stwórz dodatkowe wyjście (wpisz nazwę wyjścia do komórki Output Name):</p> <ul style="list-style-type: none"> Nazwa wyjścia (Output Name): Poprawne Wiersze. W Podziale Warunkowym otrzymujesz kilka kanałów wyjściowych (zielone strzałki). Nadanie im nazw pozwala na prostszy wybór prawidłowego wyjścia. Condition (Warunek): <code>SUBSTRING([Student],1,1) != " "</code> <p>Warunek ten ma na celu stwierdzenie, czy pierwszym znakiem wiersza jest spacja. Jeżeli tak nie jest, (czyli wiersz zawiera imię i nazwisko studenta wraz z jego oceną a nie jedynie numer indeksu), wiersz zostanie wysłany na wyjście nazwane Poprawne Wiersze. Pomocą w pisaniu warunków są foldery dostępne w górnej części edytora transformacji Podział Warunkowy. Po ich rozwinięciu można zobaczyć, jakie funkcje są dostępne oraz zapoznać się z ich składnią. Aby uzyskać na ten temat więcej wiadomości można odwołać się do książki on-line pod tytułem "SSIS Expression Reference"</p> <ul style="list-style-type: none"> Wybierz OK. i zamknij edytor
<p>8. Dodanie transformacji Derived Column (Formatowanie Kolumny) przy użyciu wyjścia Poprawne Wiersze z transformacji Conditional Split</p>	<p>Następnym krokiem jest odrzucenie niepotrzebnych danych z wierszy, które chcemy zatrzymać.</p> <ul style="list-style-type: none"> Przeciągnij narzędzie Derived Column (Formatowanie Kolumny) z Toolbox'a (Panelu Narzędzi). <p>Transformacja Formatowanie Kolumny tworzy z istniejącej kolumny nową kolumnę.</p> <ul style="list-style-type: none"> Przeciągnij zieloną strzałkę wyjścia Podziału Warunkowego do Formatowania Kolumny. Wybierz Poprawne Wiersze z listy możliwych wyjść i wybierz OK. Kliknij podwójnie zadanie Derived Column (Formatowanie Kolumny), aby otworzyć edytor transformacji. Wybierz folder Columns (kolumny) i przeciągnij kolumnę student do pierwszego wiersza listy Derived Column (Formatowanych Kolumn). Ustaw opcję Derived Column: Replace 'Student'. <p>Można zachować wynik warunku w nowej kolumnie bądź też je nadpisać. W tym wypadku źródło nie jest potrzebne, więc nadpisujemy dane.</p>

	<ul style="list-style-type: none"> • Zmień warunek Formatowanej Kolumny na: SUBSTRING([Student],1,FINDSTRING([Student],",",1)) • Wybierz OK.
9. Użycie opcji zaawansowanych, aby zobaczyć właściwości transformacji Derived Column	<ul style="list-style-type: none"> • W panelu Data Flow (Przepływ Danych) kliknij prawym przyciskiem myszy (ppm) na Derived Column (Formatowaną Kolumnę) i wybierz opcję Show Advanced Editor. <p> Tu możesz przeglądać oraz zmieniać zaawansowane właściwości komponentu.</p> <ul style="list-style-type: none"> • Wybierz zakładkę Input and Output Properties. • Rozwiń Derived Column Input, rozwiń Input Columns, i wybierz kolumnę Student. • Zamknij edytor.
10. Dodanie Flat File Destination (Źródła wynikowego jako Plik Płaski) wraz z nowych Menadżerem Połączeń Eksportuj Studentów	<ul style="list-style-type: none"> • Przeciągnij Flat File Destination (Źródła wynikowego jako Plik Płaski) z Toolbox'a (Panelu Narzędzi). • Wybierz Derived Column (Formatowanie Kolumny) i przeciągnij zieloną strzałkę wyjścia Flat File Destination. • Kliknij podwójnie na Flat File Destination i wybierz przycisk New, aby stworzyć nowy Connection Manager (Menadżer Połączeń). • Wybierz domyślny format pliku: Delimited (Odgraniczany). • Wpisz nazwę Menadżera Połączeń: Eksportuj Studentów. • Wpisz nazwę pliku wraz ze ścieżką dostępu np: C:\HD\UID\StudentiDestination.txt. • Wybierz OK. <p> Zwróć uwagę na pole informujące o nadpisaniu danych w pliku wynikowym. Pole to powinno zostać zaznaczone.</p> <ul style="list-style-type: none"> • W edytorze Flat File Destination Editor (Źródła Wynikowego jako Plik Płaski) wybierz Mappings (Mapowania), aby powiązać kolumny wejściowe z wyjściowymi. • Zamknij edytor.
11. Uruchomienie pakietu integracji danych	<ul style="list-style-type: none"> • Kliknij ppm na powstały pakiet i wybierz Execute Package. • Z menu Debug wybierz Stop Debugging. • Otwórz plik C:\HD\UID\Studenti Destination.txt, aby sprawdzić czy dane zostały prawidłowo oczyszczone (plik powinien zawierać listę imion i nazwisk studentów). • Widzimy błąd, kolumna zawiera oprócz imienia i nazwiska również znak spacji i nawias, aby naprawić ten błąd należy zmodyfikować następująco warunek w komponencie Formatowana Kolumna (Derived Column): SUBSTRING([Student],1,FINDSTRING([Student],",",1) - 2) • Kliknij ppm na powstały pakiet, wybierz Execute Package i ponownie sprawdź plik wynikowy.

Problem 2 (czas realizacji 15min) - kontynuacja problemu 1

W dalszej części zlecenia z problemu drugiego otrzymałeś zadanie przeniesienia danych nie tylko do plików tekstowych, lecz także do bazy SQL Server 2008. Aby było to możliwe użyjesz transformacji Multiprzesyłu (**Multicast**). Dodasz również zadanie Wykonania SQL (**Execute SQL Task**) aby oczyścić tabelę wynikową przed załadowaniem danych, aby możliwe było uruchamianie pakietu kilka razy bez konieczności ręcznego czyszczenia tabeli.

Zadanie	Tok postępowania
1. Stworzenie nowej bazy danych SQL Server o nazwie Studenci z Simple Recovery Model (Modelem Prostego Odtwarzania)	<p>Aby użyć danych, które właśnie stworzyłeś jako źródło przeszukiwania rozmytego (Problem.1/2 Moduł2), dane muszą znajdować się w tabeli a nie w pliku płaskim. Zamiast usuwać obiekt Źródło Wynikowe jako Plik Płaski, dane wyjściowe mogą zostać przesłane do dodatkowego źródła wynikowego. Trzeba jedynie stworzyć bazę danych, która przechowa wyniki.</p> <ul style="list-style-type: none"> W Object Explorer (Eksploratorze Obiektów) w SSMS (SQL Server Management Studio), kliknij ppm na folder Databases (Bazy Danych) i wybierz New Database (Nową Bazę Danych). <p> SSMS musi być podłączony do serwera Baz Danych (DataBase Engine). Częstym błędem jest połączenie do np. Analysis Services lub Reporting Services.</p> <ul style="list-style-type: none"> Wprowadź nazwę bazy danych: Studenci. Na stronie Options (Opcje), Zmień model Odzyskiwania na Simple (Prosty).
2. Dodanie transformacji Multicast (Multiprzesyłu) pomiędzy komponentami Derived Column (Formatowana Kolumna) a Flat File Destination (Płaski Źródło Wynikowe jako Plik)	<p>Teraz można przesłać wyniki z komponentu Derived Column do dwóch źródeł wynikowych pliku tekstowego oraz nowostworzonej bazy danych.</p> <ul style="list-style-type: none"> W module SSBIDS, przeciągnij narzędzie Multicast (Multiprzesył) z Toolbox 'a (Panelu Narzędzi) do zakładki Data Flow (Przepływ Danych) w pakiecie Studenci. Multicast (Multiprzesył) podobnie jak Conditional Split (Podział Warunkowy) ma wiele wyjść z tym, że w Multicast (Multiprzesyłu) na każdym z wejść znajdują się dokładnie te same dane. Usuń połączenie pomiędzy Derived Column (Formatowaną Kolumną) a Flat File Destination (Źródłem wynikowym). Przeciągnij zieloną strzałkę wyjścia Derived Column (Formatowanej Kolumny) do Multicast (Multiprzesyłu). Przeciągnij zieloną strzałkę wyjścia Multicast (Multiprzesyłu) do Flat File Destination (Źródła Wynikowego jako Plik Płaski). <p> Dopóki nie zmienia się "rodowodu" kolumn, możesz przerywać oraz dodawać kolejne przepływy danych bez ponownego mapowania. Jeżeli jednak w jakikolwiek sposób modyfikuje się kolumny należy usunąć poprzednie mapowania i stworzyć nowe.</p>
3. Dodanie SQL Server Destination (Źródła	<ul style="list-style-type: none"> Przeciągnij SQL Server Destination (Źródła Wynikowego jako SQL Server) z Toolbox 'a (Panelu Narzędzi). Przeciągnij zieloną strzałkę wyjścia Multicast (Multiprzesyłu) do SQL Server Destination (Źródła Wynikowego jako SQL Server).

<p>Wynikowego jako SQL Server) wraz z nowym Menadżem Połączeń podłączonym z bazą danych Studenci i nową tabelą, Imie i Nazwisko</p>	<ul style="list-style-type: none"> • Kliknij podwójnie SQL Server Destination (Źródła Wynikowego jako SQL Server) następnie wybierz New, aby stworzyć nowego Menadżera Połączeń. • Wybierz New na wyborze połączenia do danych (Select Data Connection), aby zdefiniować nowe połączenie: • Ustaw local SQL Server (lokalny serwer SQL) jako źródło danych, wybierz Windows Authentication, i wybierz bazę danych Studenci. • Wybierz przycisk Test Connection, aby sprawdzić połączenie. • Dwukrotnie wybierz OK. • Wybierz New w SQL Destination Editor (Use a table or view), aby stworzyć nową tabelę. • Zmień nazwę tabeli, na ImieNazwisko i wybierz OK. <p>Menadżer Połączeń stworzy nową tabelę, w której brak jest pola nadpisania, które było obecne przy Pliku Płaskim. W związku z tym niezbędne jest dodanie dodatkowego komponentu czyszczącego tabelę, co ujęte jest w dalszej części ćwiczenia.</p> <ul style="list-style-type: none"> • Wybierz Mappings (Mapowania) w SQL Destination Editor po to, aby ustawić prawidłowe mapowania pomiędzy kolumną wejściową i wyjściową. • Wybierz OK i zamknij edytor.
<p>4. Dodanie Execute SQL Task (Wykonaj Zadania SQL)</p>	<p> Następnym krokiem jest dodanie zadania czyszczącego tabelę wynikową przed wykonaniem zadania przepływu danych.</p> <ul style="list-style-type: none"> • Na zakładce Control Flow (Kontroli Przepływu) dodaj Execute SQL Task (Wykonaj Zadania SQL): <ul style="list-style-type: none"> • Nazwa WyczyśćImieNazwisko. • Połączenie: localhost.Studenci. • Zadanie SQL (SQL Statement): Delete from ImieNazwisko.
<p>5. Dodanie Precedence Constraint (Wymuszenia Pierwszeństwa), aby uruchomić komponent <i>Wyczyść ImieNazwisko</i> przed <i>Importuj Studentów</i></p>	<ul style="list-style-type: none"> • Przeciągnij strzałkę wyniku z elementu WyczyśćImieNazwisko do Importuj Studentów
<p>6. Uruchomienie pakietu integracji danych i sprawdzenie wyników</p>	<ul style="list-style-type: none"> • Kliknij ppm na pakiet i wybierz Execute Package. • Z menu Debug wybierz Stop Debugging. • Za pomocą SSMS, rozwiń Databases, następnie Studenci, oraz rozwiń tabelę. Możliwe, że koniecznym będzie odświeżenie, aby tabela ImieNazwisko była dostępna. • Kliknij ppm na tabelę ImieNazwisko, i wybierz Open Table aby potwierdzić poprawność danych obecnych w tabeli.

Laboratorium rozszerzone

Zadanie 1 (czas realizacji 30 min)

Prosty projekt na laboratorium podstawowym zakłada oczyszczenie danych ograniczające się do otrzymania kolumny Imię nazwisko. Twoim zadaniem będzie stworzenie zarówno w pliku wynikowym jak i w naszej bazie docelowej dodatkowej kolumny, która będzie zawierać Numer Indeksu. Wiedza przedstawiona w laboratorium podstawowym powinna zostać dokładnie przyswojona i zastosowana w rozwiązaniu. Zauważ, że modyfikacje należy zacząć od odpowiedniego warunku w komponencie **Derived column**, pamiętaj również, że po każdej zmianie rodzaju kolumn należy zmodyfikować ich mapowanie oraz **Connection Manager** (Menadżer połączeń).

Zadanie 2 (czas realizacji 60 min)

Twoim zadaniem jest zaprezentowanie uzyskanych kompetencji podczas projektowania pilotowego wdrożenia prostej hurtowni danych w dziekanacie twojej uczelni. W katalogu z danymi znajdziesz pliki potrzebne do wykonania tego zadanie. Pierwszym twoim krokiem powinno być uruchomienia SQL Server Management studio, w którym utworzysz nową bazę „System Nauczania”. Następnie wykonaj tu zapytanie SQL „lms_skrypt.sql” dostarczone w pliku danych. To zapytanie stworzy tabele, do których musisz wyeksportować dane załączone w pliku. Przeanalizuj dokładnie tabele i zaprojektuj proces ETL, który pozwoli na wypełnienie nowej bazy odpowiednimi danymi.