

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

Kamil Napiórkowski

Nr albumu: 134462

Rola procedur składowanych i wyzwalaczy w aplikacjach
bazodanowych wykorzystujących bazę danych MySQL.

The role of stored procedure procedures and triggers in the
database applications that use MySQL database.

Praca licencjacka
napisana pod kierunkiem
dr Marcina Płonkowskiego
w katedrze Systemów
Operacyjnych i Sieciowych.

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

Spis treści

1. Wstęp	3
2. Teoretyczne aspekty związane z bazą danych.	4
3. Elementy programowalne po stronie serwera.....	14

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

1. Wstęp

W niniejszej pracy zostało omówione znaczenie procedur składowanych i wyzwalaczy. Zagadnienia dotyczą czynności wykonywanych w aplikacjach bazodanowych po stronie serwera. Głównym powodem stosowania procedur składowanych jest po prostu wydajność, czyli szybsze wykonywanie poleceń oraz oszczędność pamięciowa. Wyzwalacze i procedury składowane są automatycznie wykonywane, bez żadnej ingerencji użytkownika. Wyzwalacze stosuje się w wielu systemach bazodanowych, ale w tej pracy zagadnienia zostały ograniczone jedynie do systemu bazodanowego MySQL, choć trzeba pamiętać, że ten system jest integralny z SQL. MySQL została kupiona w 2008 roku przez firmę Sun, a następnie w 2010 roku firmę tę przejął ORACLE a wraz nią prawa do MySQL.

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

2. Teoretyczne aspekty związane z bazą danych.

Bazy danych są to odpowiednio uporządkowane zbiory danych, zorganizowany zgodnie z modelem danych, który jest notacją służącą do opisu danych. Struktura danych i powiązania pomiędzy nimi są opisane przez schemat bazy danych, natomiast dane to zbiór reguł określających strukturę danych i operacje na nich w bazie (tabele, obiekty), z kolei informacja różni się od danych tym że to jest to co uzyskujemy z bazy danych. Integralność danych określa, które stany bazy danych są poprawne. Formalna poprawność bazy danych, oznacza jej fizyczną organizację, zgodność ze schematem bazy danych i regułami dostępu. Umieszczone informacje w bazie danych mają określać dany fragment świata realnego, który nazywamy obszarem analizy, a pojedynczy, wyróżniony obiekt, który ma być odzwierciedlony w bazie danych nazywamy encją (lub inaczej rekordem bazy danych). Najmniejszą strukturą w bazie danych są pola, który odwzorowuje pewną cechę, którym zadaniem jest przechowywanie danych.

Integralność obejmuje dwa rodzaje spójności:

- wewnętrzna – schematem bazy danych oraz innymi
- zewnętrzna – stane świata rzeczywistego

Ważnym czynnikiem przekładającym się na pracę z bazą danych jest dobrze zaprojektowana struktura. Jeśli tak nie jest to można mieć problemy nawet z tak elementarnymi operacjami jak pozyskiwanie informacji, czy brak integralności danych.

Optymalizacja pól:

- Nazwa powinna być zrozumiała
- Nazwa powinna być jednoznaczna i przystępna
- Nie używaj akronimów
- Użyta nazwa nie powinna odnosić się do więcej niż cechy
- Nazwa powinna być w liczbie pojedynczej

Więzy integralności - w kontrolowaniu integralności trzeba przestrzegać więzów nakładanych na dane, aby uniemożliwić powstanie niespójnego stanu bazy danych.

Rodzaje więzów integralności:

- Dane wymagane
- Więzy dziedzinowe
- Integralność encji
- Integralność referencyjna
- Więzy ogólne

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

Cechy bazy danych:

- Modelowanie świata rzeczywistego
- Spójność
- Bezpieczeństwo danych
- Współdzielenie danych
- Efektywne przetwarzanie danych
- Trwałość danych

Bazy danych składają się z części:

- Intensjonalnej – zbiór definicji, które określają strukturę danych
- Ekstensjonalnej – łączny zbiór danych

Model bazy danych to specyfikacja opisująca strukturę danych w bazie danych.

Modele baz danych:

- Hierarchiczny – dane gromadzone są na kolejnych poziomach drzewa.
- Sieciowy – struktura danych przypominająca sieć
- Relacyjny – każda tabela opisana jest przez klucz danych, który jednoznacznie określa dany rekord w tabeli.
- Obiektowy – dane są zdefiniowane według zasad programowania obiektowego.

Podstawowe zadania bazy danych:

- Odzworowanie części świata realnego
- Gromadzenie danych
- Długoterminowe przechowywanie danych

Ważnym zagadnieniem w bazach danych jest klucz. Rodzaj tego klucza decyduje o jego funkcji. Mimo że tabela może zawierać kilka rodzajów kluczy to najważniejsze są dwa: klucz główny i klucz obcy. Pierwszy rodzaj klucza to pole lub ich grupa, które w jednoznaczny sposób określają każdą encję w tabeli. Jego wartość identyfikuje konkretny rekord, a jego pole daną tabelę w całej bazie danych. Rola klucza głównego jest bardzo ważna, bo zapewnia integralność tabel i definiowanie relacji dla tabel jest łatwiejsze. Z kolei klucz obcy to kolumna lub zbiór kolumn, która czerpie wartości z tej samej dziedziny co klucz główny tabeli powiązanej z nią w bazie danych. Klucze obce pokazują relacje między danymi z różnych tabel.

Jako fundament relacyjnych baz danych uznaje się teorię przedstawioną przez Franka Edgara Codd'a w 1970 roku wraz z publikacją *A Relational Model of Data for Large Shared Data Banks* (Relacyjny model str. 5

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

danych dla dużych współdzielonych banków danych). Relacyjna baza danych (ang. *Relational Database*) to baza danych w której zależności między danymi opisane są w postaci relacji (tabeli), która jest podstawową i jedyną strukturą danych. Jej podstawą jest algebra relacyjna. Schemat relacji to informacja o strukturze, atrybutach które opisują daną relację. Relacje zawsze występują w pierwszej postaci normalnej.

System zarządzania bazami danymi (ang. *Database Management System*) jest zorganizowanym zbiorem narzędzi umożliwiającym dostęp i zarządzanie bazami danymi.

Ograniczenia pozwalają wprowadzać na kolumny warunki. Określają jakie wartości są dopuszczalne w danym polu i umożliwiają dodatkową kontrolę danych.

- NOT NULL – w danej kolumnie pola nie mogą mieć wartości NULL
- PRIMARY KEY – wartości nie mogą się powtarzać. Wartości są nie powtarzające się i różne od NULL
- DEFAULT – określa domyślną wartość wprowadzanych danych
- FOREIGN KEY – klucz obcy, integralność referencyjna
- UNIQUE – Wartości wpisywane do tabeli są niepowtarzające się
- CHECK – warunek, sprawdzający czy wprowadzana wartość jest poprawna
- ENUM – umożliwia na prowadzenie jednej wartości z określonego zbioru
- SET – umożliwia na wprowadzeniu co najmniej jednej wartości z określonego zbioru

Postulaty Codda:

- Informacyjna – dane muszą być reprezentowane jako wartości atrybutów tabelach
- Dostępność – istnieje dostęp do danych w bazie za pomocą nazwy tabeli, atrybutu i wartości klucza podstawowego
- Postulat dla wartości NULL – istnieje specjalna wartość NULL, która może reprezentować nieokreślone wartości
- Postulat dotyczący katalogu – potrzebne jest, aby system obsługiwał wbudowany katalog relacyjny
- Postulat języka danych - system musi udostępniać język przetwarzania danych
- Postulat modyfikacji perspektyw – system musi pozwalać na modyfikację widoków
- Modyfikacja danych - system musi umożliwiać modyfikację danych
- Postulat fizycznej niezależności danych – zmiany fizycznej postaci danych i organizacji dostępu nie wpływają na programy
- Niezależność więzów spójności – więzy spójności są określone w bazie i nie zależą od programu
- Postulat niezależności dystrybucji – działanie programu nie zależy od modyfikacji i dystrybucji bazy
- Bezpieczeństwo względem operacji niskiego poziomu – operacje niskiego poziomu nie mogą w żaden sposób modyfikować model relacyjny i więzów spójności

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

Funkcje systemu zarządzania bazą danych:

- Definiowanie struktur
- Wykonywanie operacji CRUD
- Wykonywanie zapytań
- Tworzenie raportów i zestawień
- Administrowanie bazą
- Zarządzanie plikami

SQL to strukturalny język zapytań (ang. Structured Query Language) używany do tworzenia, modyfikowania baz danych oraz do umieszczania i pobierania z nich danych. Należy do języków czwartej generacji. Nie jest językiem proceduralnym, lecz deklaratywnym. Polega to na tym, że język definiuje informację wynikową i nie określa sposobu wykonania polecenia, a decyzję o sposobie wykonania zadania pozostawia systemowi zarządzania bazą danych. Jest obsługiwany przez większość systemów bazodanowych, m.in. MySQL, Oracle, PostgreSQL, Sybase, Microsoft SQL Server. Standardem SQL jest ANSI obsługiwany przez MySQL, natomiast MySQL jest systemem zarządzania relacyjnymi bazami danych o dostępie swobodnym. MySQL należy do firmy Oracle.

Podstawowe zagadnienia dotyczące języka SQL:

- Zapytania muszą kończyć się średnikiem
- Wartości muszą znajdować się między średnikami lub cudzysłowami, za wyjątkiem wartości liczbowych
- Nazwy funkcji zapisujemy bez cudzysłowów

Polecenia języka SQL można podzielić na trzy główne grupy:

- Język Definiowania Danych – DDL (ang. Data Definition Language) – umożliwia tworzenie struktur baz danych i ich modyfikowanie (CREATE, DROP, ALTER).
- Język Manipulacji Danymi – DML (ang. Data Manipulation Language) – umożliwia dokonywanie operacji na danych (INSERT, UPDATE, DELETE)
- Język Kontroli Danych – DCL (ang. Data Control Language) - (GRANT, REVOKE, DENY)

Kwerenda to zapytanie do bazy danych.

Indeksy są to pomocnicze struktury dostępowe, używane w celu zmniejszenia czasu pobierania konkretnych danych w tabeli, czyli zwiększenie szybkości wykonywania zapytań. Indeksy mogą być proste, składające się z jednej kolumny lub złożone, składające się z wielu kolumn. Jeżeli tabela nie zawiera indeksu to w czasie wykonywania zapytania SELECT wiersze tabeli przeglądane są sekwencyjne i wybierane są te, które spełniają warunki zawarte w klauzuli WHERE. Jeśli zostanie zdefiniowany odpowiedni indeks to najpierw zostanie przeszukany indeks, który jest posortowany, czyli odczyt indeksu odbywa się na zasadzie skoków do bliskiego sąsiada. Ogólnie ujmując mechanizm indeksów porównywalny jest do indeksów dołączanych na końcu książek.

Stworzenie i aktualizacja indeksu oraz korzystanie z niego wymaga określonego czasu procesora i pamięci. Najlepiej definiować indeksy tylko tam, gdzie są naprawdę potrzebne oraz nie należy

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

definiować indeksów typu B-drzewo na kolumnach o małym zakresie wartości i nie tworzyć indeksów dla małych tabel. Najbardziej efektywne używanie indeksów jest dla zapytań, które zwracają mało wierszy. Indeksy tworzone są między innymi automatycznie na kolumnach primary key i unique.

Transakcja jest sekwencją logicznych powiązanych operacji na bazie danych, operacje wstawiania, usuwania, modyfikacji danych. Transakcja przeprowadza bazę danych z jednego stanu spójnego w inny stan spójny, przykładem transakcji może być przelew bankowy, czyli trzeba pobrać określoną kwotę z jednego konta, a na drugim dodać tą kwotę, jeśli jakiegokolwiek instrukcja byłaby nie wykonana to automatycznie anulowane są także poprzednie instrukcje zawarte jako transakcja.

W latach osiemdziesiątych XX wieku pojawił się akronim, pozwalający w zwięzły i łatwy do zapamiętania sposób podstawowe właściwości transakcji:

- Atomicity (atomowość)
- Consistency (spójność)
- Isolation (izolacja)
- Durability (trwałość)

Atomowość – zbiór operacji wchodzących w skład transakcji jest niepodzielny albo zostaną wykonane wszystkie operacje transakcji albo żadna. Grupa instrukcji SQL w transakcji stanowi jedną jednostkę pracy.

Spójność – operacje wykonywane na bazie danych muszą zachowywać spójność bazy danych, czyli transakcja przekształca bazę z jednego stanu spójnego w inny spójny

Izolacja – transakcje są odseparowane jedna od drugiej. Jeśli kilka transakcji przebiega równocześnie ich wykonywanie nie może ze sobą kolidować. Współbieżnie wykonywanie transakcje widza stan bazy danych, jak gdyby były wykonywane w sposób sekwencyjny

Trwałość – zmiany dokonywane w bazie przez zatwierdzone transakcje są trwałe, nie mogą zostać utracone nawet w wyniku awarii systemu.

Cechy te gwarantują, że dane będą spójne i niezależnie od okoliczności nasza baza danych będzie zawsze w prawidłowym stanie. To jest bardzo ważne szczególnie w biznesie, na przykład w systemach bazodanowych w bankach, mimo wszystko pojawia się wiele różnych problemów, które w różny sposób negatywnie wpływają na bazę danych, takich jak wydajność.

Punkty bezpieczeństwa:

- Umożliwiają podział transakcji na etapy
- Dzięki nim istnieje możliwość cofnięcia się do określonego momentu w historii bieżącej transakcji
- Cofnięcie powoduje anulowanie zmian, które zostały wprowadzone przez operacje transakcji wykonywane od momentu utworzenia punktu bezpieczeństwa do którego nastąpiło cofnięcie, do momentu zarządzania przez użytkownika cofnięcia do punktu bezpieczeństwa

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

COMMIT i ROLLBACK:

- Tylko użytkownik wprowadzający zmiany w ramach transakcji może je zobaczyć
- Inni użytkownicy nie mogą zobaczyć zmian wprowadzonych w ramach niezatwierdzonych transakcji
- Stan bazy danych może zostać przywrócony
- Wiersze, które zostały zablokowane nie mogą być modyfikowane przez innych użytkowników

Po wykonaniu polecenia ROLLBACK:

- Następuje anulowanie wszystkich zmian wprowadzonych przez transakcje
- Wszystkie wiersze, które zostały zablokowane zostają odblokowane
- Zostają usunięte punkty bezpieczeństwa

Po wykonaniu polecenie COMMIT:

- Następuje utrwalenie zmian, czyli poprzedni stan bazy danych zostaje nadpisany
- Zmiany wprowadzone przez transakcje są widoczne dla wszystkich użytkowników
- Wszystkie wiersze, które zostały zablokowane zostają oblokowane
- Zostają usunięte punkty bezpieczeństwa
- Zostaną usunięte punkty bezpieczeństwa
- Rozpoczyna się nowa transakcja

Zakończenie transakcji:

- Niejawne – wykonanie polecenia DDL i DLC, awaria
- Jawne – wykonanie polecenia COMMIT lub ROLLBACK

Jawne zakończenie transakcji wykonywane jest przez żądanie użytkownika czyli wykonanie polecenia COMMIT zatwierdzającego transakcję lub nie. Nie jawne zakończenie transakcji wykonywane jest gdy użytkownik zażąda zakończenia sesji, czy też dojdzie do awarii lub wykonania polecenia DDL lub DLC. Po tych poleceniach sesja zostaje zatwierdzona, a po zakończeniu zaczyna się nowa sesja, a podczas awarii bieżąca transakcja zostaje wycofana.

Widok , inaczej perspektywa to po prostu wirtualna tabela, która nie posiada własnych danych, tylko pobiera je z innych tabel.

Rodzaje perspektyw:

- Proste – bazują na jednej tabeli, nie grupują danych, nie wykorzystują funkcji, może wykorzystywać DML
- Złożone – bazują na przynajmniej jednej tabeli, może wykorzystywać funkcje oraz grupuje dane, nie zawsze może wykorzystywać DML

Jeżeli widok wykorzystuje funkcje grupujące, zawiera klauzulę GROUP BY lub słowo kluczowe DISTINCT to nie można za jej pomocą usuwać wierszy. Jeśli perspektywa funkcje grupujące, zawiera klauzulę GROUP BY lub słowo kluczowe DISTINCT i wykorzystuje wyrażenia do definiowania kolumn to nie można

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

za jej pomocą modyfikować wierszy, a jeśli wykorzystuje funkcje grupujące, zawiera klauzulę GROUP BY lub słowo kluczowe DISTINCT, nie zawiera kolumn, które zostały zdefiniowane w tabeli jako NOT NULL to nie można za jej pomocą wstawiać wierszy do bazy danych.

Zarządzanie wymaganiami odnosi się do programów łączących mnóstwo widoków:

- Scentralizowane podejście – wymagania obejmujące poszczególne widoki użytkowników są scalane w jeden zbiór wymogów dotyczących definiowania bazy danych. Podczas projektowania bazy danych tworzony jest jeden model bazy dla wszystkich elementów zbioru wymagań.
- Podejście integrujące widoki – każdy zdefiniowany widok jest podstawą do tworzenia

Korzyści z korzystania z widoków:

- Wygoda – upraszczanie zapytań
- Stworzenie dodatkowego systemu zabezpieczeń
- Minimalizowanie ilości dostępu do kolumn lub wierszy w bazowej tabeli
- Warstwa abstrakcyjna

Widoki można definiować z dodatkowymi opcjami:

- Schemabinding – powoduje, że system zarządzania bazą danych zablokuje modyfikację tabel, które mogą spowodować błąd realizacji perspektyw
- Encryption – powoduje zablokowanie dostępu do definicji perspektyw.
- With Check Option – w sytuacjach, gdy perspektyw umożliwia modyfikację danych w tabeli macierzystej i w definicji perspektywy występuje klauzula WHERE.

Wiersze migrujące to takie wiersze, które albo pojawiają się w perspektywie, albo z niej znikają. Gdy wiersze spełnia warunek WHERE zapytania tworzącego perspektywę to jest widoczny w perspektywie do momentu gdy przestaje spełniać wymogi podane w klauzuli WHERE. Aby zapobiegać takim sytuacjom dodaje się do perspektywy klauzulę WITH CHECK OPTION, zabezpiecza to że użytkownik dokonuje zmiany tylko w obrębie danych do których ma uprawnienia.

Zalety stosowania perspektyw:

- Niezależność danych
- Zmiany dokonane w tabelach od razu rzutują

Język PL/SQL jest proceduralnym rozszerzeniem języka SQL. Pozwala na korzystanie z takich struktur programistycznych podobnych do tych jakie występują na przykład w C++: pętli, zmiennych i instrukcji warunkowych. Pozwala na tworzenie programów do operacji na danych.

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

Kursor to struktura w której przechowujemy wyniki zapytania SQL. Pozwala pobierać rekordy w sposób sekwencyjny w dowolnej ich ilości. Jego zaletą jest to że może działać zarówno po stronie serwera i jak klienta.

Normalizacja – proces upraszczania struktury bazy danych, w taki sposób aby osiągnęła ona postać optymalną.

Korzyści wynikające z normalizacji:

- Pewność, że każda tabela ma prawidłową strukturę
- Brak anomalii

Większości baz danych wystarczy normalizacja do trzeciej postaci normalnej, np. obsługa zamówień sklepu komputerowego.

Pierwsza postać normalna – każda wartość atrybutu jest wartością elementarną.

Druga postać normalna – każdy atrybut niekluczowy zależy w pełni funkcyjnie od klucza głównego (żaden atrybut nie będący kluczem nie zależy od części klucza głównego). Aby stwierdzić, czy tabela jest w drugiej postaci normalnej trzeba wyznaczyć zależności funkcyjne i klucz główny tabeli.

Trzecia postać normalna – Atrybuty niekluczowe zależą jedynie od klucza głównego. Aby stwierdzić, czy tabela jest w trzeciej postaci normalnej trzeba sprawdzić, czy żaden element nie zależy od atrybutów, które nie są kluczami.

Model ER - Diagram związków encji pozwalają w sposób graficzny opisywać model konceptualny. Model ER (ang. Entity relationship) służy do nieformalnego przedstawiania projektu baz danych. Stosowany do komunikowania się z użytkownikiem

Obszar analizy reprezentuje:

- Encja
- Atrybut
- Związek

Model ERD:

Zbiór encji – to grupa obiektów o tym samym właściwościach w rzeczywistości uznawane za niezależne byty.

Wystąpienie encji to unikalny i rozpoznawalny obiekt ze zbioru encji.

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

Atrybut to cecha encji. Istnieje wiele rodzajów atrybutów:

- Atrybut prosty to atrybut zawierający tylko jedną składową
- Atrybut złożony zawiera wiele składowych
- Atrybut kluczowy to atrybut jednoznacznie identyfikujący daną encję
- Atrybut pochodny – używany do reprezentacji informacji, które mogą być obliczane na podstawie danych zawartych w bazie

Związek to powiązanie między zbiorem encji. Stopień związku określa liczbę encji w związku.

Rodzaje związków:

- Związek binarny (linarny)
- Związek potrójny

Związkom mogą być przypisane nazwy pól, co jest istotne przy związkach rekurencyjnych.

Silny zbiór encji – to zbiór encji, którego istnienie nie jest zależne od innych zbiorów encji

Słaby zbiór encji – to zbiór encji, którego klucze składają się (przynajmniej częściowo) z atrybutów kluczowych innych encji. Istnienie stałego zbioru encji zależy od innych zbiorów encji.

Liczność opisuje maksymalną liczbę możliwych wystąpień związku dla encji uczestniczącej w tym związku:

- Jeden do jednego
- Jeden do wielu
- Wiele do wielu

Uczestnictwo określa czy w pewnym związku biorą udział wszystkie, czy tylko niektóre wystąpienia encji.

- Opcjonalnie – określa czy w pewnym związku biorą udział wszystkie, czy tylko niektóre wystąpienia encji
- Obowiązkowe – wszystkie instancje muszą brać udział w związku

Projektowanie baz danych umożliwia lepsze zaplanowanie struktury i budowy systemu bazodanowego, pozwala na minimalizację potencjalnych błędów, czyli mamy pewność że baza danych będzie w pełni zoptymalizowana, spójna i integralna.

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

Metody projektowania baz danych:

- Wstępujące – rozpoczyna się od zdefiniowania wszystkich atrybutów, a następnie poprzez analizę powiązań łączy się je w encje i związki między nimi. Metoda znajduje zastosowanie przy projektowaniu prostych baz danych o małej liczbie atrybutów
- Zstępująca – rozpoczyna się od tworzenia modeli danych zawierających niewielką liczbę ogólnych encji, atrybutów i związków między nimi, a następnie stosując metodę kolejnych uściśleń wprowadza się encję wprowadza się encję, związki i atrybuty niższych poziomów. Właściwa strategia projektowania złożonych baz danych.

Cele modelowania danych:

- Ułatwienie zrozumienia znaczenia danych
- Pomoc w porozumiewaniu się w sprawie wymagań informacyjnych

Kryteria optymalności modeli:

- Poprawność – zgodność ze sposobem definiowania i organizacji danej instancji
- Prostota – zrozumiałość zarówno dla informatyków, jak i nieprofesjonalnych informatyków
- Wyrażalność – zdolność odróżnienia różnych typów danych, związków pomiędzy danymi oraz węzłów
- Brak redundancji – eliminacji niepotrzebnych informacji
- Uniwersalność – brak powiązań modelu z jakąś konkretną aplikacją czy technologią
- Rozszerzalność – zdolność rozbudowy o elementy realizujące nowe wymagania przy minimalnych skutkach dla istniejących użytkowników
- Integralność – zgodność ze sposobem wykorzystywania informacji i zarządzania nią w instytucji

Etapy projektowania bazy danych:

- Konceptualne (pojęciowe) projektowanie bazy danych – jest procesem konstrukcji modelu, który jest całkowicie niezależny od narzędzi
- Logiczne projektowanie bazy danych – jest procesem konstrukcji modelu opartego na konkretnym modelu danych
- Fizyczne projektowanie bazy danych – jest procesem opisu implementacji bazy danych, który zawiera powiązania, organizację plików i indeksów.

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

3. Elementy programowalne po stronie serwera.

Procedury składowane (ang. stored procedure) jest to element języka SQL przedstawiony jako zestaw instrukcji wywoływany pojedynczą nazwą, umieszczony na serwerze bazodanowym. Wykorzystuje się to do sprawdzania walidacji danych lub jako mechanizm kontroli dostępu. Jest to bardzo przydatny mechanizm na przykład gdy chcemy wstawić do pola nie odpowiednie dane, wtedy przy próbie wprowadzenia danej niepoprawnego typu wykonywane są instrukcje zawarte w procedurze. Ich stosowanie zwiększa wydajność systemu bazodanowego i bezpieczeństwo oraz szybkość działania poprzez minimalizację ruchu w sieci, a także zabezpiecza bazę danych przed atakami, które polegają na iniekcji kodu SQL. Taki atak pozwala nieupoważnionej osobie poznać strukturę bazy danych, rozszerzyć swoje uprawnienia, wykonać dowolne zapytanie/instrukcje w bazie danych i wywołać procedurę składową, która zawiera jakiś skrypt zwiększający funkcjonalność serwera, a wyniku tego otrzymać dostęp do danych serwera. Bronić się można przez sprawdzanie wprowadzanych danych i filtrowaniu informacji, które są wysyłane do bazy przez klienta. Jako zalety należy uznać to że zastosowanie procedur składowanych umożliwia redukcję nie potrzebnego kodu.

Plan wykonania procedury składowanej będzie wygenerowany w momencie pierwszego wywołania procedury, następnie będzie buforowany i teoretycznie może być wykorzystywany nieskończoność. Cechą procedur składowanych jest ich łatwość i przenoszeniu, można uruchomić na każdym systemie korzystającym z MySQL i jak również łatwo ulegają migracji między różnymi wersjami. Możemy również kontrolować działanie procedur składowanych za pomocą cech, czyli atrybutów z którymi są wywoływane te procedury. Procedury składowane w bazie MySQL mają zablokowany dostęp do danych w bazie danych. Ważną informacją na temat procedur jest to, że nie wolno im zmieniać innych procedur. Procedury za to mogą tworzyć, modyfikować i usuwać definicje tabel, widoków i baz danych. Bardzo naturalnym zastosowaniem procedur jest obsługiwanie błędów, które mogą powstać podczas działania systemu, na przykład gdy węzły zostaną naruszone.

Funkcje składowane, podobnie jak procedury składowane, są obiektami bazy danych składającego się z kodu w języku SQL. Podstawowa różnica polega na tym, że funkcje posiadają zdefiniowany typ zwracanej wartości i mogą być wykorzystywane w zapytaniach SELECT.

Katolicki Uniwersytet Lubelski Jana Pawła II

Wydział Matematyki, Informatyki i Architektury Krajobrazu

Procedury umożliwiają:

- Przekazywanie parametrów wywołań
- Mogą wykonać niemalże wszystkie instrukcje języka SQL oraz inne procedury składowane (co pozwala na uniknięciu powtarzania nie potrzebne kodu)
- Zwracać wyniki wywołań do podprogramu, który go wywołał
- Zwracać informację o prawidłowym lub nie wykonaniu procedury składowanej

Procedury składowane można podzielić na dwa typy:

- SQL procedury składowane – umożliwiają łatwiejszy import procedur składowanych z innych DBMS
- Zewnętrzne procedury składowane

Wyzwalacz(ang. triggers) to podprogram powiązany z daną tabelą, który uaktywnia się podczas jakiegoś zdarzenia w bazie na przykład, podczas wykonania zapytań INSERT, DELETE oraz UPDATE. Ściśle rzecz ujmując wyzwalacz to specjalny rodzaj procedur składowanych. Wyzwalacze w MySQL są dostępne od wersji 5.0.

Wyzwalacze służą do:

- Uniemożliwieniu wprowadzenia nie poprawnych danych
- Wymusza integralność danych