Kontrola dostępu do rekordów przy pomocy etykiet bezpieczeństwa LBAC w IDS 11

+ przegląd pozostałych funkcjonalności

Artur Wroński

IBM Information Management

Technical Team Leader

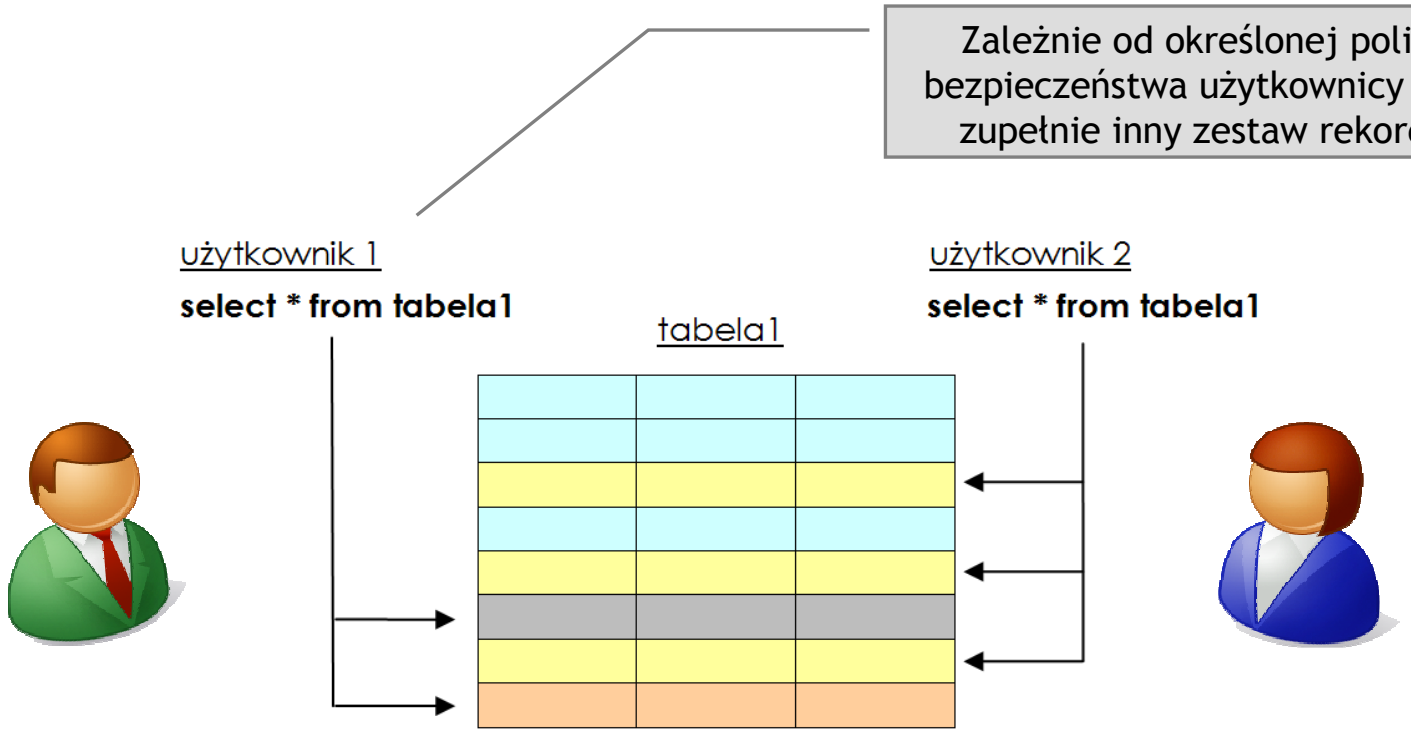
artur.wronski@pl.ibm.com

IBM Information Management software

Zwiększona kontrola dostępu do danych w IDS11

- LBAC – etykiety bezpieczeństwa
- Szyfrowanie komunikacji HDR
- Nowe algorytmy szyfrowania
- Certyfikaty bezpieczeństwa

Label-Based Access Control (LBAC)

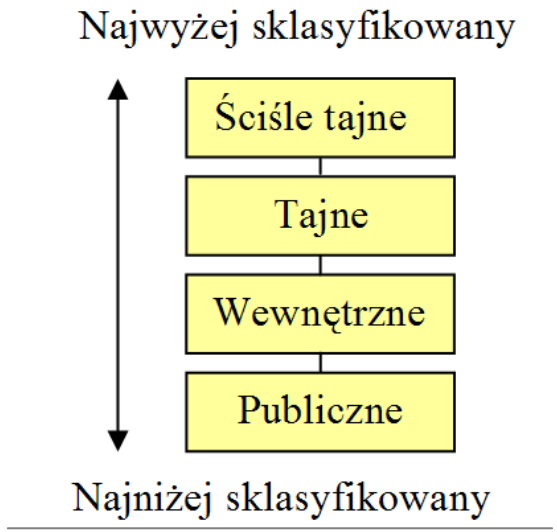


Mechanizm LBAC jest kolejnym przykładem przenikania się technologii pomiędzy serwerami danych IBM (DB2 i rodziną Informix).

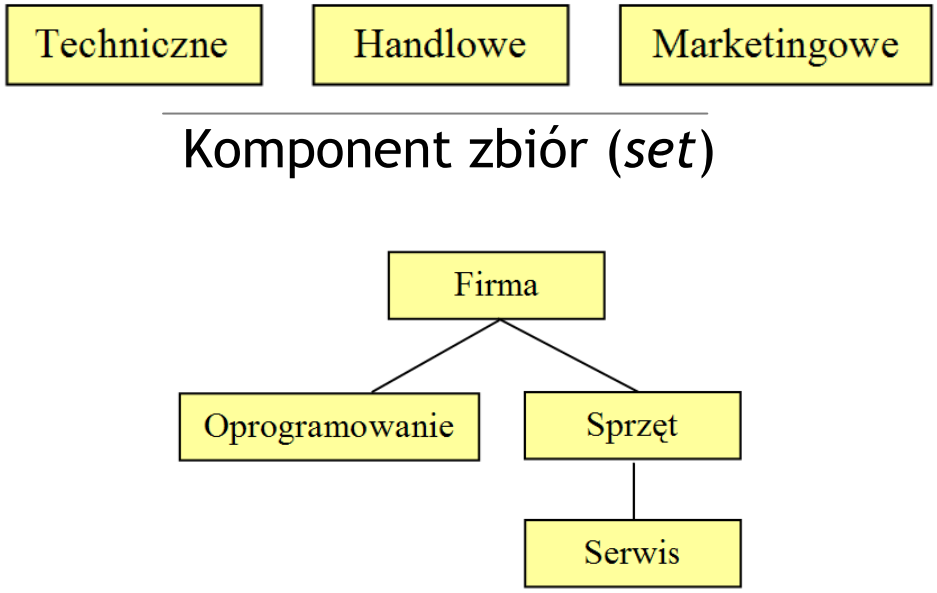
Administrator bezpieczeństwa przygotowuje najpierw **politykę bezpieczeństwa** (*security policy*).

Polityka bezpieczeństwa składa się z jednego, bądź wielu **komponentów etykietowych** (*label component*).

Polityką bezpieczeństwa zabezpiecza się dostęp do tabeli



Komponent szereg (*array*)



Komponent zbiór (*set*)

Komponent drzewo (*tree*)

Utworzenie komponentu etykietowego

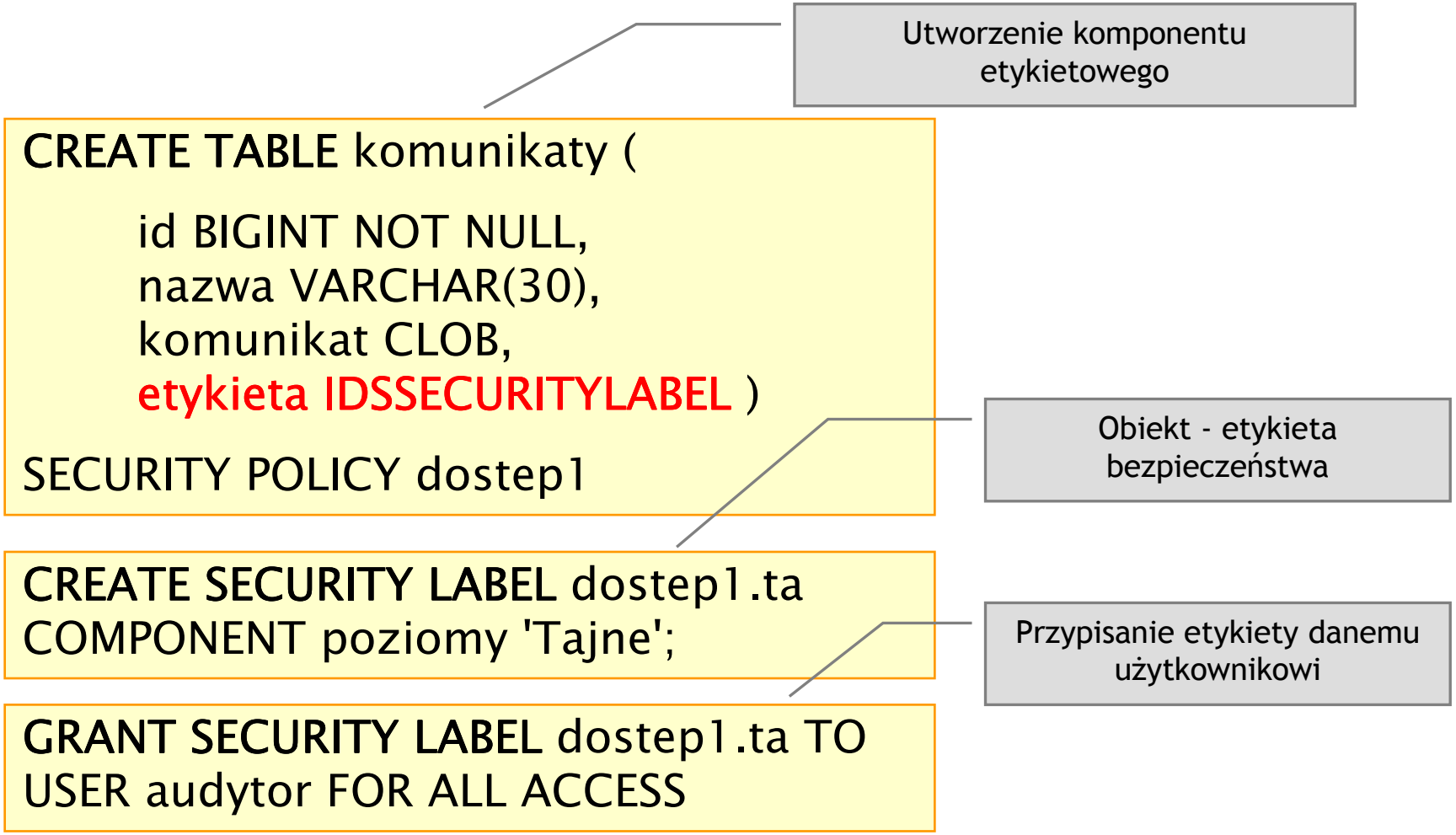
```
CREATE SECURITY LABEL COMPONENT działy TREE ('Firma'
ROOT, 'Oprogramowanie' UNDER 'Firma', 'Sprzęt' UNDER
'Firma', 'Serwis' UNDER 'Sprzęt')

CREATE SECURITY LABEL COMPONENT tematyka SET {
'Handlowe', 'Techniczne', 'Marketingowe' }
```

Utworzenie polityki bezpieczeństwa

```
CREATE SECURITY POLICY dostep1 COMPONENTS działy,
tematyka
WITH IDSLBACRULES
RESTRICT NOT AUTHORIZED WRITE SECURITY LABEL
```

Reguły definiujące porównywanie etykiet. W DB2: DB2LBACRULES


Utworzenie komponentu etykietowego
CREATE TABLE komunikaty (
id BIGINT NOT NULL,
nazwa VARCHAR(30),
komunikat CLOB,
etykieta IDSSECURITYLABEL)
SECURITY POLICY dostep1
Obiekt - etykieta bezpieczeństwa
CREATE SECURITY LABEL dostep1.ta COMPONENT poziomy 'Tajne';
Przypisanie etykiety danemu użytkownikowi
GRANT SECURITY LABEL dostep1.ta TO USER audytor FOR ALL ACCESS

Wstawienie rekordu z domyślną etykietą

```
INSERT INTO komunikaty (id, nazwa, komunikat) VALUES (1,
'Pierwszy', '...')
```

Wstawienie rekordu z etykietą określoną na niższym poziomie.

```
INSERT INTO komunikaty
VALUES (1, 'Pierwszy', '...', SECLABEL_BY_NAME('DOSTEP1',
'PUB'));
```

Oczywiście, jeśli zbiór reguł IDSLBACRULES zezwala na zapisywanie danych na innych poziomach (niższych lub wyższych) .

```
GRANT EXEMPTION ON RULE IDSLBACWRITETREE
FOR dostep1 TO USER audytor
```

Zawsze można zwolnić danego użytkownika z przestrzegania określonych reguł zdefiniowanych w zbiorze reguł.

Label-Based Access Control (LBAC) - podsumowanie

Etykiety zapewniają przeźroczystą dla aplikacji metodę kontroli dostępu do rekordów

Przydatne są wszędzie tam, gdzie wymagana jest ścisła kontrola dostępu do rekordów bez ograniczania metod dostępu do danych

Inne nowe funkcjonalności w IDS 11

Blokady

- Uaktualnione rekordy nie mogą być przeczytane przez innych użytkowników, aż do momentu zatwierdzenia transakcji, chyba, że wykorzystywany jest poziom izolacji READ UNCOMMITTED
- Aplikacja może pracować niewydajnie, jeśli oczekuje na zatwierdzenie transakcji.
- Aplikacja może wykorzystać poziom izolacji READ UNCOMMITTED, lecz może to prowadzić do nieoczekiwanych rezultatów
- Pojawienie się zakleszczeń, może znacznie spowolnić aplikację.

Nowy poziom izolacji
Last Committed Read

Zaprojektowany by zwiększyć współbieżność dostępu do danych i wydajność

Zwraca ostatnią zatwierdzoną wersję rekordu

Przykład *Last Committed Read* - transfer 400 zł z klienta #1234 do klienta #3456

Czas	Transakcja1	Transakcja2 COMMITTED READ (domyślny poziom w bazie transakcyjnej)	Transakcja3 DIRTY READ	Transakcja4 LAST COMMITTED (Nowość w IDS 11.10)
1	-- Bieżące saldo dla klienta #1234 wynosi 1250.00			
2	set isolation to read committed;			
3	begin work;			
4	update cust_tab set balance =balance – 400 where cust_id = 1234;	begin work;	begin work;	begin work;
5	-- Saldo dla klienta #1234 wynosi 850.00			
6	update cust_tab set balance = balance + 400 where cust_id = 3456;	select balance from cust_tab where custid = 1234; -- oczekiwanie na blokadę dla klienta 1234	select balance from cust_tab where custid = 1234; -- bez oczekiwania. zwróci 850.00.	select balance from cust_tab where custid = 1234; -- Bez oczekiwania -- zwróci 1250.00
7	insert into daily_tab(“transfer”, 1234, 3456, 400);	-- Stan: oczekiwanie na blokadę	-- Kontynuacja przetwarzania	-- Kontynuacja przetwarzania
8	Commit work;	-- Stan: oczekiwanie na blokadę	-- można zrobić więcej	-- można zrobić więcej
9		--zwróci 850.00	-- można zrobić więcej	-- można zrobić więcej

Co jest „checkpoint”?

Dla przypomnienia w IDS < 11

- IDS wykorzystuje pamięć (bufferpool) by buforować modyfikacje danych (insert / update / delete)
- Modyfikacje są także odłożone na dysku w celu ewentualnego uspójnienia systemu po awarii (logical log)
- Podczas normalnego przetwarzania pula buforów jest zapisywana okresowo na dysk – tzw. checkpointy.
- Częstsze checkpointy oznaczają krótszy czas odtwarzania

Checkpoint – problem w praktyce

- Większość checkpointów wstrzymuje pracę transakcji (nawet w przypadku rozmytych „fuzzy” checkpointów).
- Rozmyte checkpointy pracują na stronach z danymi, nie indeksowymi
- Wielu klientów stroi agresywne zapisywanie kolejek LRU, by uniknąć blokowania transakcji
  - Zwiększone zużycie CPU
  - Ograniczone wykorzystanie buforów do zapisu

Zbyt częstsze checkpointy oznaczają wstrzymują transakcje, zbyt rzadkie checkpointy mogą doprowadzić do bardzo długiego czasu przywracania spójności bazy po awarii.

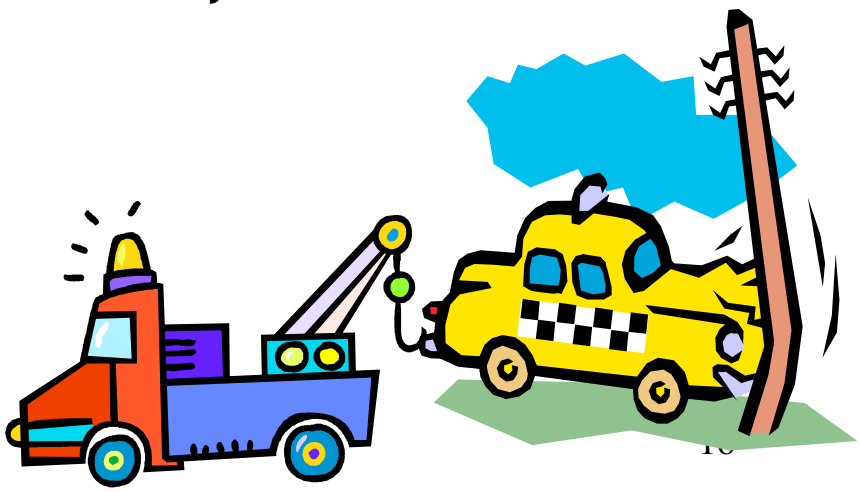
Nowy algorytm checkpointu w IDS 11

- W IDS 11 transakcje nie są wstrzymywane podczas zapisywania stron z pamięci na dysk
- Teraz nie trzeba już tak utrzymywać tak niskich progów LRU
  - Ustawienie kolejek LRU na 60/70 w IDS jest rozsądne
- Kolejki LRU mogą być automatycznie dostrajane (onconfig: AUTO_LRU_TUNING)

Jeśli czyszczenie kolejek LRU było ustawione na bardzo niski poziom, wtedy nowy checkpoint powinien znacznie poprawić wydajność

Ile wynosi czas naprawy bazy po awarii w IDS 10?

- Aktualnie ciężko jest przewidzieć czas przywracania spójności bazy danych po nieprawidłowym zamknięciu bazy
- Rozmyte checkpointy czynią czas naprawy bazy po awarii trudnym do przewidzenia
- CKPTINTVL nie potrafi się dostosować do zmiennych warunków przetwarzania



IDS 11: Reguła określająca czas naprawy po awarii

- Zmienna w ONCONFIG: RTO_SERVER_RESTART - Recovery Time Objective (RTO) określa politykę pracy checkpointów pozwalającą na naprawę bazy w określonym czasie
- Zakres czasu: od 60 do 1800 sekund
- System monitoruje obciążenie i dostosowuje częstość checkpointów, by dostosować się do reguł RTO
- Polityka RTO może być zarządzana dynamicznie (onmode), jak i może być wyłączona

Nowy algorytm checkpointu w IDS 11

Mniej agresywne czyszczenie kolejek LRU

Bardziej przewidywalny czas naprawy po awarii

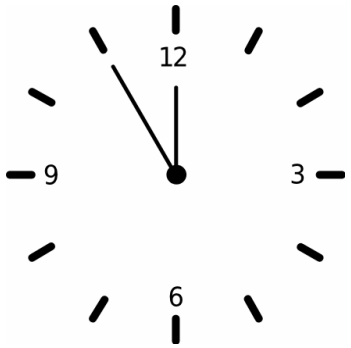
Samostrojenie automatycznych chekpoint-ów

Zwiększenie przepustowości transakcji (10%-500%)

Automatyczna korzyć dla aplikacje OLTP
praktycznie bez jakichkolwiek modyfikacji

Inne nowości w IDS 11
Wbudowane narzędzie harmonogramujące

- Harmonogramuje instrukcje SQL, procedury składowane oraz UDF (także administracyjne API wywoływane z SQL)
- Harmonogram oparty o tabelę ph_task
- 4 rodzaje zadań:



Zadanie	Uruchamiane zadanie o określonym czasie, które nie zbiera danych
Sensor	Zbiera i ewentualnie zapisuje informacje
Zadanie startowe	Wykonywane jednorazowo przy starcie serwera baz danych
Sensor startowy	Wykonywany jednorazowo przy starcie serwera baz danych

Inne nowości w IDS 11

Szczegółowe monitorowanie instrukcji SQL
(SQL Query Drill Down)

OpenAdmin Tool – narzędzie graficzne

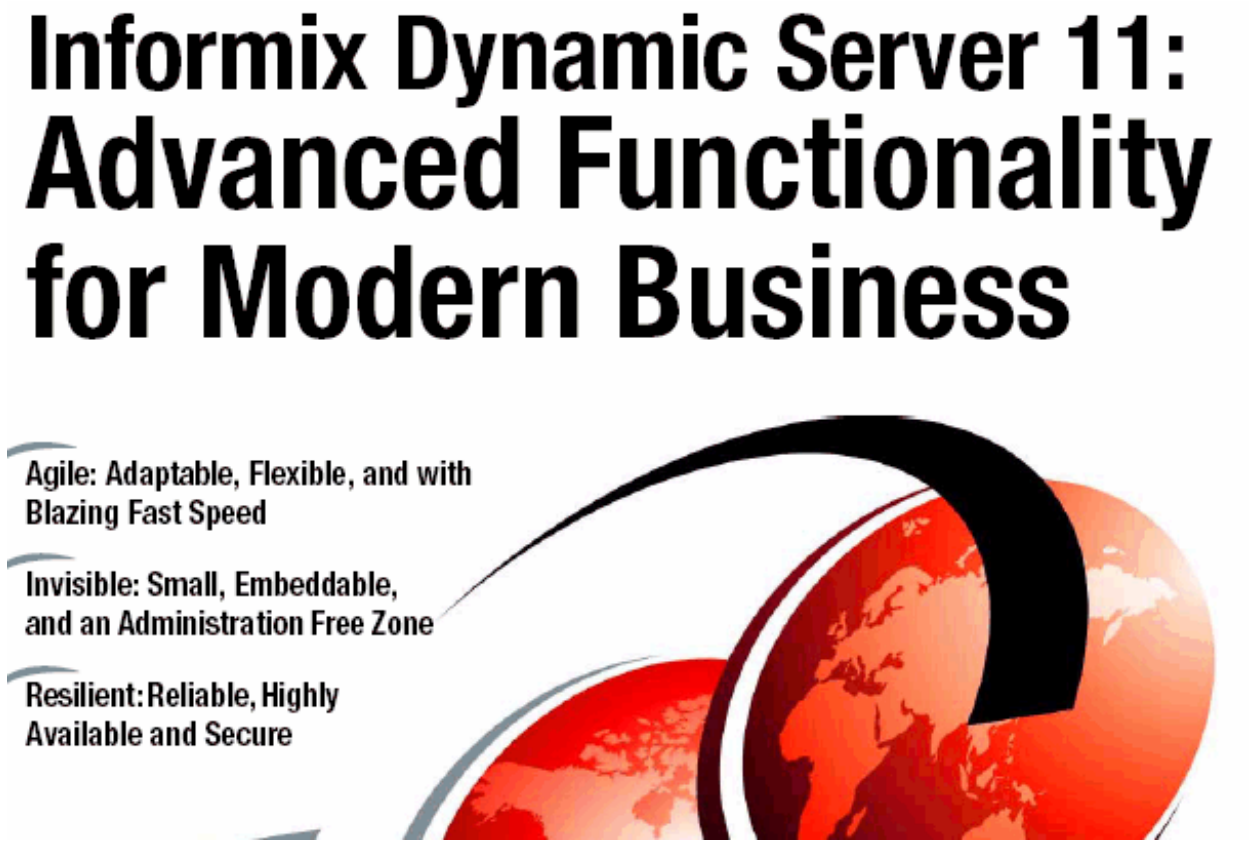
Backup ontape na dysk

Automatyczne zbieranie statystyk podczas tworzenia indeksu

Funkcje publikujące XML

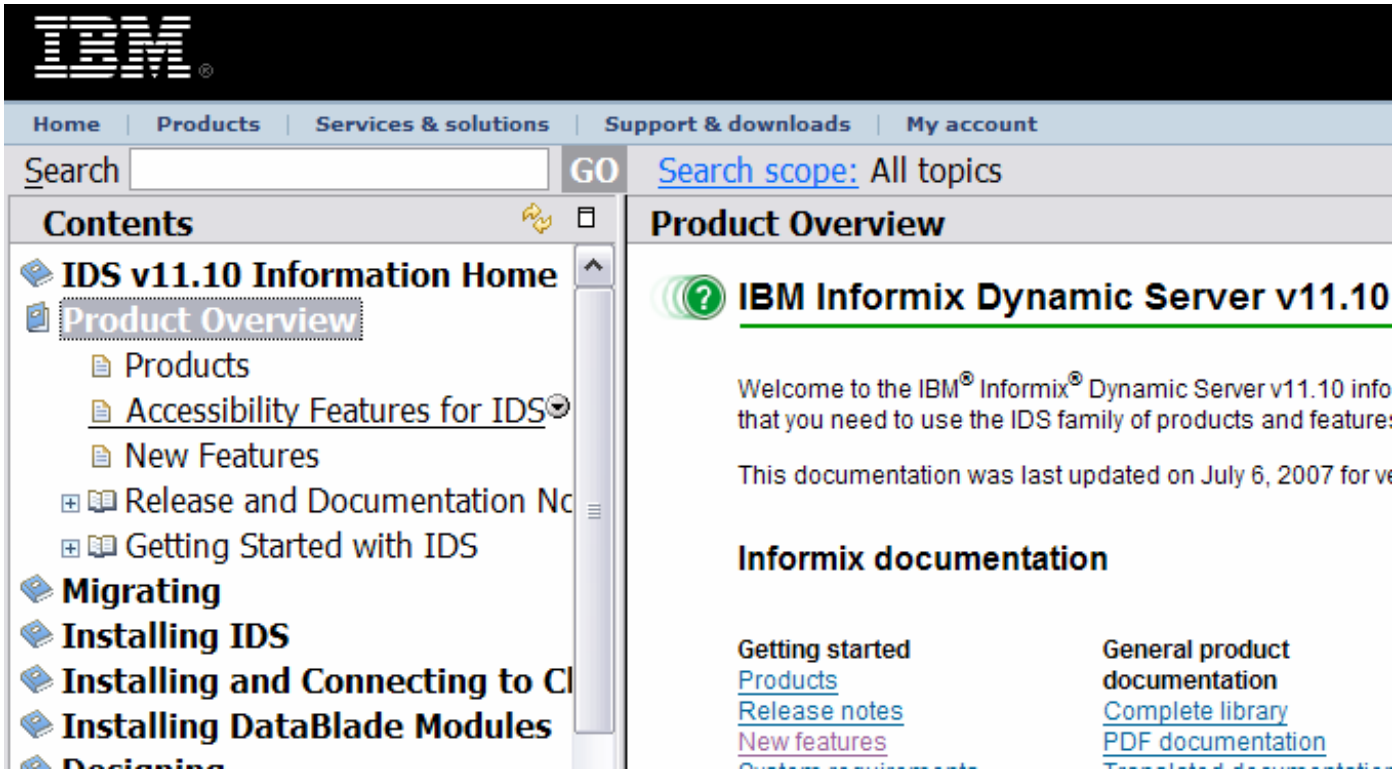
Wbudowane podstawowe wyszukiwanie pełnotekstowe

Web Feature Service DataBlade dla danych przestrzennych



www.ibm.com/redbooks

Informix „Gepard” 2007



Migracja do IDS 11 w kilka minut

Bieżąca wersja	Krok 1	Krok 2
7.24, 7.30, 7.31, 9.20, 9.21, 9.30, 9.40, 10.0	11.0	Nie jest wymagany
9.1x	9.30	11.0
7.22, 7.23	7.31	11.0
Online 5.1x	7.31	11.0

- W pełni zautomatyzowana migracja
- Przy zmianie systemu operacyjnego / platformy sprzętowej wymagany jest dbexport/dbimport
- Automatyczny powrót do starej wersji, jeśli wystąpią błędy migracji
- Dla aplikacji OLTP szacunkowo minimum 10% wzrost wydajności bez strojenia

Pytania?

Artur Wroński

IBM Information Management software

Technical Team Leader

artur.wronski@pl.ibm.com

+603-88-66-49