

Problemy optymalizacji, rozbudowy i integracji systemu Edu wspomagającego e-nauczanie i e-uczenie się w PJWSTK



Paweł Lenkiewicz

Polsko – Japońska Wyższa Szkoła Technik Komputerowych



Plan prezentacji

- PJWSTK i system Edu.
- Integracja platformy LMS z innymi systemami działającymi na uczelni.
- Problemy wydajnościowe platform e-learningowych.

PJWSTK

- Założona w 1994 roku w ramach porozumienia między rządami Polski i Japonii.
- Kształci informatyków na stopnie inżyniera, magistra i doktora.
- Prowadzi studia podyplomowe.





Nauczanie na odległość w PJWSTK

- 2000-01 Kursy realizowane na odległość wspólnie z Uniwersytetem w Charlotte, USA – na poziomie magisterskim.
- 2001 Zbudowanie własnego systemu zarządzającego studiowaniem na odległość o nazwie Edu.
- 2002 Rozpoczęcie zaocznych studiów przez Internet na stopień inżyniera informatyki.
- 2006 Rozpoczęcie dwuletnich uzupełniających studiów magisterskich przez Internet.
- 2008 Uruchomienie internetowych studiów podyplomowych.



System Edu

- Nauka przez Internet odbywa się przy pomocy platformy Edu.
- Studenci opanowują jeden wykład z 15 do każdego przedmiotu w ciągu tygodnia.
- Po każdym wykładzie należy rozwiązać zadania i umieścić je w systemie lub rozwiązać test.
- Platforma ułatwia komunikację pomiędzy wykładowcą i studentami (baza adresów, chat, forum dyskusyjne, wymiana plików).



Moduły systemu

- Ogłoszenia
- Chat
- Forum dyskusyjne
- Obszar roboczy do wymiany plików
- Kalendarz
- Foldery zadań domowych
- Wykłady
- Lekcje
- Testy
- Oceny



Użytkownicy systemu

- Studenci internetowi
- Studia otwarte
- Indywidualny tok nauczania
- Studenci studiów stacjonarnych, powtarzający przedmioty w trybie internetowym
- Przedmioty studiów stacjonarnych, wspomagane platformą Edu

Problem: Bardzo duża liczba użytkowników i duża różnorodność ich praw dostępu. Częste zmiany.

Inne systemy używane przez studentów

- Wirtualny dziekanat
- FTP
- E-mail
- Serwer WWW
- System elektronicznego obiegu dokumentów
- i wiele innych...

Problem: Wiele kont, konieczność pamiętania dużej liczby haseł.



Jedno hasło do wszystkich usług

Rozwiązanie: Active Directory

Podstawowe zastosowanie:

- Dostęp do konta z dowolnego komputera.

Możliwe inne zastosowanie:

- Integracja usług szkolnych (jedno hasło do wszystkich usług).

Zalety:

- Stosunkowo prosta implementacja i administracja.
- Możliwość integracji z bardzo dużą liczbą narzędzi, zwykle przy niewielkim nakładzie pracy.



Konieczność integracji systemów

Synchronizacja baz danych systemu obsługującego dziekanat z platformą nauczania na odległość:

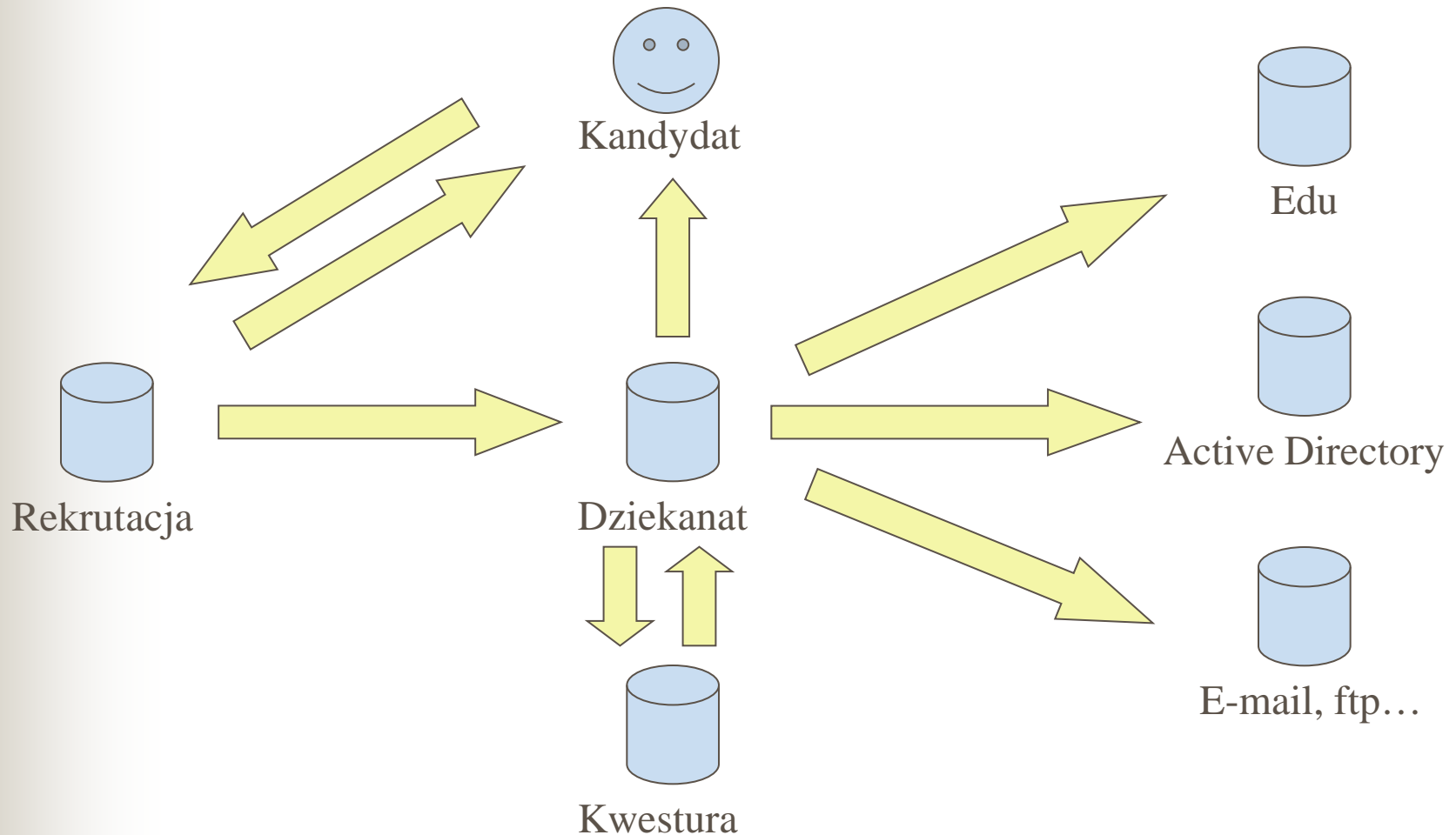
- Automatyczne zakładanie kont.
- Automatyczne ustalanie praw dostępu do kursów.
- Powiadamianie o zmianach.
- Blokowanie / odblokowywanie kont.



Inne korzyści z integracji

- Możliwość wystawiania ocen przez platformę.
- Raportowanie oparte na zintegrowanych bazach danych.

Przykładowy przepływ informacji





Rozwiązania techniczne

Replikacja baz danych

Zalety:

- Możliwość łatwego określenia, co i kiedy będzie replikowane.
- Łatwe zachowanie spójności danych.
- Natychmiastowa lub opóźniona synchronizacja.

Wady:

- Zbyt „ciężkie” narzędzie do tego zastosowania.
- Duże nakłady administracyjne.



Rozwiązania techniczne

Wyzwalacze bazy danych

Zalety:

- Zmiana wprowadzona w jednej bazie powoduje automatyczne wprowadzenie zmian do baz pozostałych.
- Łatwa implementacja przy podobnym schemacie.
- Spójność danych.

Wady:

- Negatywny wpływ na wydajność.
- W większości przypadków natychmiastowa aktualizacja nie jest konieczna.



Rozwiązania techniczne

Integration Services

Zalety:

- Ogromne możliwości wymiany i transformacji danych.
- Możliwość wymiany danych z serwerami innymi niż MS SQL Server.

Wady:

- W większości zastosowań jest to zbyt „ciężkie” narzędzie.



Rozwiązania techniczne

Połączone serwery (linked servers) i procedury składowane

Zalety:

- Wygodny dla programisty język tworzenia programów działających bezpośrednio na połączonych bazach danych.
- Bardzo łatwa administracja (po utworzeniu programów).
- Wysoka wydajność.

Wady:

Możliwość łączenia wyłącznie kompatybilnych serwerów.

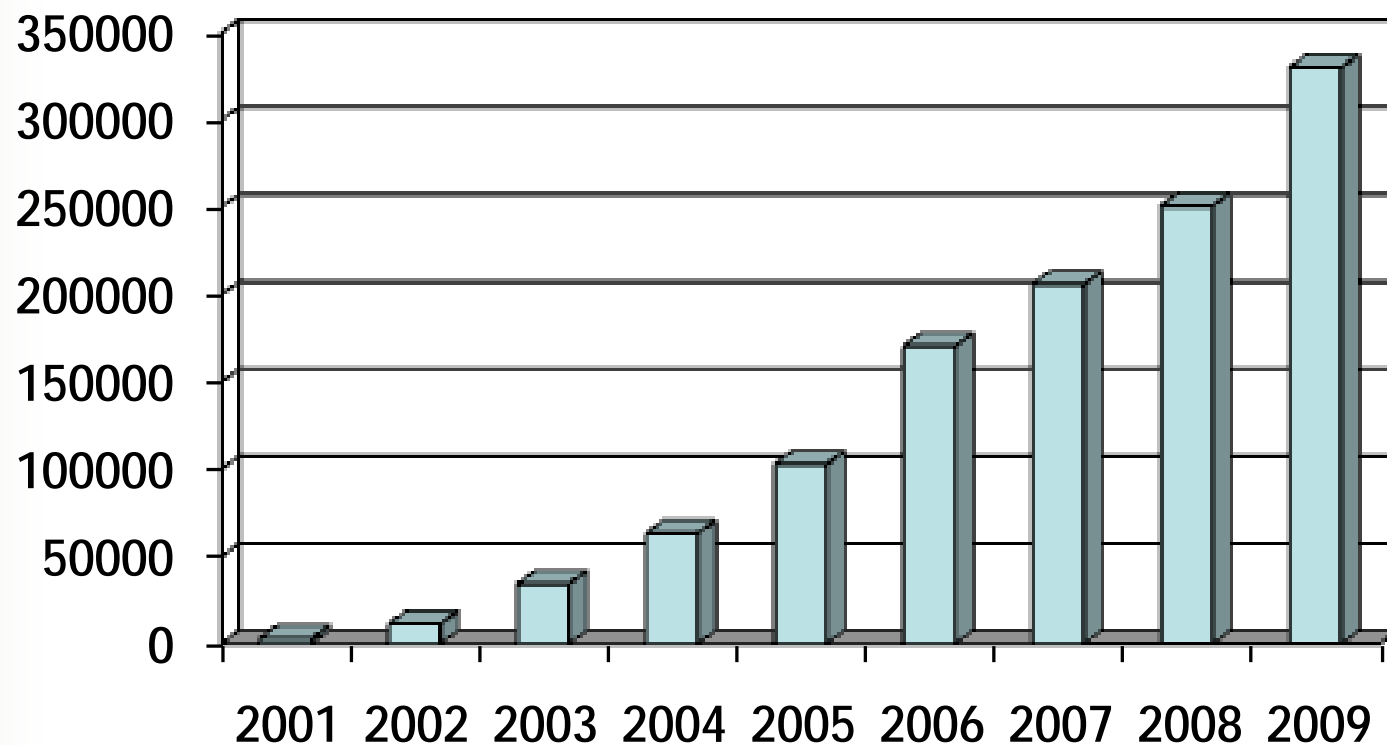
Kod jest wywoływany na żądanie lub według harmonogramu i za każdym razem weryfikuje całość wymienianych danych.



Rozwiązanie optymalne

**Zintegrowana baza danych modelująca
całą uczelnię**

Liczba logowań





Problemy wydajnościowe

- Czas odpowiedzi
- Czas oczekiwania na wykonanie złożonych operacji (rzadziej)



Analiza pliku śladu bazy danych

- Dziennik, w którym zapisane są wszystkie wywołania SQL wraz z dodatkowymi informacjami.
- Ogromna ilość danych.
- Potencjalnie bardzo dużo przydatnych informacji.
- Dane te są trudne w analizie.

Profil użycia bazy danych platformy e-learningowej

- Dominują proste instrukcje SQL, ale jest ich bardzo wiele.
- Złożone operacje na bazie danych są rzadkie: Mniej niż 10% instrukcji.
- Typowe operacje: złączenia (niewielkie liczby tabel), filtrowanie rekordów (wysoka selektywność), sortowanie.



Wskazówki wydajnościowe

- Indeksy na kluczach obcych.
- Odpowiednie indeksy lub struktury danych na kolumnach, które są używane do sortowania.
- Identyfikacja kolumn używanych w warunkach wyszukiwania. Zalecane indeksy wspierające wyszukiwanie w danych o dużej selektywności.
- Wydzielenie dużych tabel często odczytywanych/zapisywanych równoległe na osobne dyski.
- Partycjonowanie dla tabel, w których potrafimy wydzielić dane odczytywane często i takie, które są używane rzadziej (np. dane archiwalne).



Wskazówki wydajnościowe

- Przy złożonych operacjach poprawa wydajności może być osiągnięta głównie przez optymalizację kodu.
- Mechanizm wielowersyjności – zastosowania raportowe.
- Duża liczba małych plików – zalecane bardziej wydajne systemy plików.