

Szeroki świat integracji

Dr Tomasz Cieplak

Katedra Organizacji Przedsiębiorstwa

Agenda

- **Pojecie integracji raz jeszcze**
- **Rodzaje integracji**
- **Architektura typu „Luźne sprzężenie”**
- **Architektura typu „Ścisłe sprzężenie”**
- **Przykłady rozwiązań „Luźnego sprzężenia”**
- **Integracja oparta na wymianie komunikatów**

Pojecie integracji raz jeszcze

Integracja w rozumieniu dzisiejszego wykładu będzie oznaczała połączenie ze sobą wielu systemów komputerowych, przedsiębiorstw, koncernów lub ludzi.

Rodzaje integracji

- W świecie integracji istnieje wiele scenariuszy integracji. Jednak najczęściej są one różnymi pochodnymi od sześciu podstawowych:
 - Portal informacyjny,
 - Replikacja danych,
 - Udostępnianie funkcji biznesowych,
 - Architektura zorientowana na serwisy,
 - Rozproszone procesy biznesowe,
 - Integracja B2B

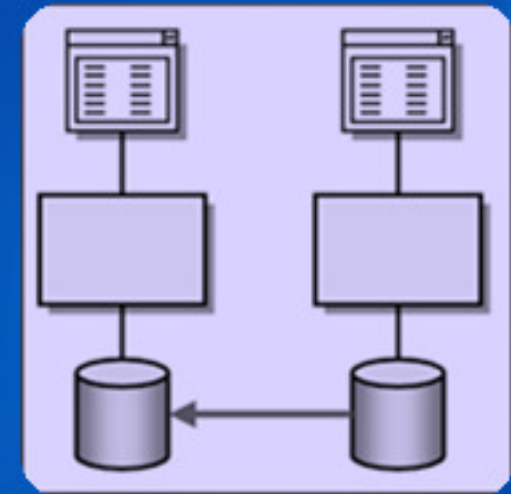
Portal informacyjny

Użytkownik biznesowy musi skorzystać z wielu systemów aby uzyskać odpowiedź na określone pytanie lub wykonać pewną czynność biznesową.



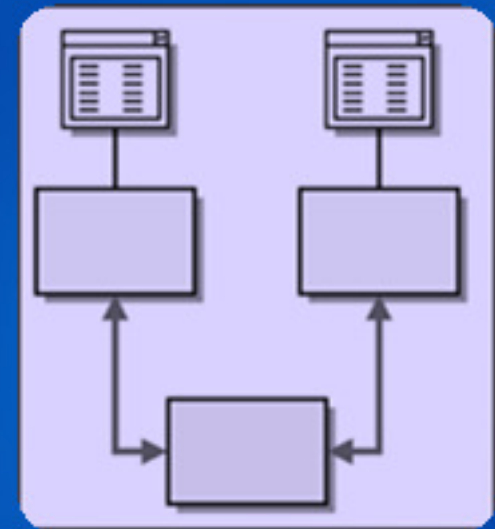
Replikacja danych

- Wiele systemów biznesowych wykorzystuje te same dane, np. NIP, PESEL, adres zamieszkania.
- Systemy przechowują informacje w swoich bazach danych, które wcześniej replikowane.



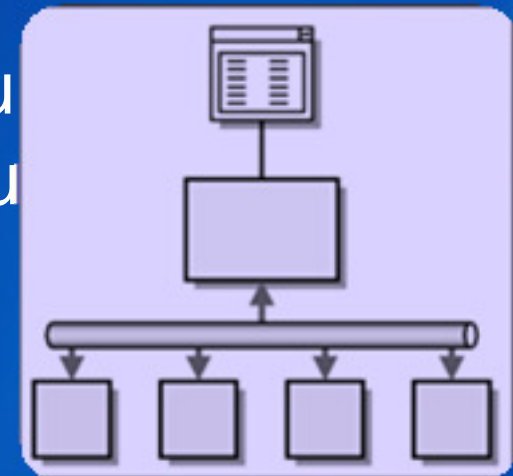
Udostępnianie funkcji biznesowych

- Podobnie jak w przypadku replikacji danych, wiele systemów wykorzystuje nadmierną (redundancję) procesów lub funkcji biznesowych.
- Sprawdzenie poprawności numeru PESEL, kodu miejscowości itp., udostępniane są jako funkcja dla wielu systemów



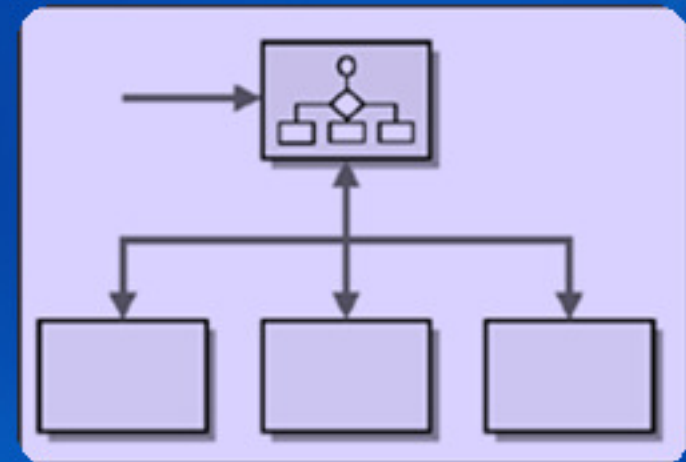
Architektura zorientowana na serwisy

- Udostępnianie funkcji biznesowych często określa się jako serwisy,
- Serwisem nazywany zdefiniowaną funkcję, która jest dostępna ogólnie i odpowiada na pytanie postawione przez klienta serwisu,
- Korzystanie z serwisu przez aplikacje jest możliwe po uzyskaniu opisu serwisu (discovery) i sposobu komunikacji (negotiation).



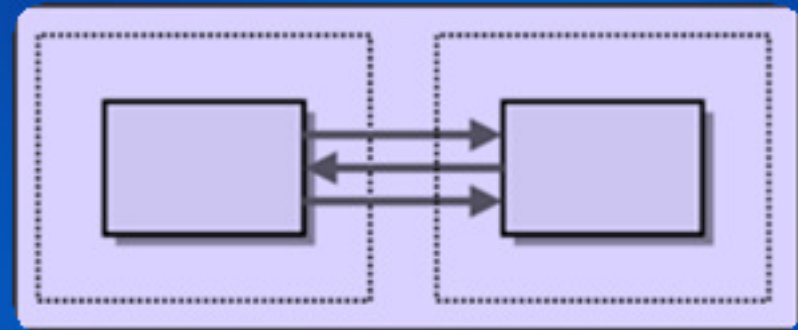
Rozproszone procesy biznesowe

- Komponent typu „business process manager”, którego zadaniem jest uruchamianie funkcji biznesowych zaimplementowanych w wielu działających systemach.
- Granice zastosowań BPM i SOA są bardzo nieokreślone, dla przykładu możliwe jest rozwiązanie, w którym istotne funkcje biznesowe są udostępniane jako serwisy a następnie BPM używa ich za pośrednictwem SOA.



Integracja B2B

- Włączenie funkcjonalności jednej z firm do systemu drugiej: sprzedawca internetowy korzysta z funkcji biznesowych firmy kurierskiej do naliczania kosztu przesyłki i/lub korzysta z funkcjonalności dostawcy towaru – stany magazynowe, cena zakupu.
- Komunikacja poprzez Internet, różne standardy, różnice czasowe, itp.
= problemy



Architektura typu „Luźne sprzężenie” (ang. Loose Coupling)

- Architektura znana od szeregu lat, obecnie bardzo często stosowana ze względu na wszechstronność zastosowań Serwisów Internetowych XML,
- Podstawowe założenie przyjmuje, że zredukowane jest wzajemne powiązania pomiędzy parami (komponenty, aplikacje, serwisy, użytkownicy) wymieniającymi informacje,

Architektura typu „Luźne sprzężenie” (ang. Loose Coupling)

- Inne założenie mówi, że im bardziej pary są ze sobą powiązane i powiązane z protokołem, komunikacja jest bardziej wydajna, jednak całe rozwiązanie jest bardziej podatne na problemy w komunikacji i zmiany w parach.

Architektura typu „Ścisłe sprzężenie” (ang. Tight Coupling)

- Metoda Invoke (wezwanie)– jest lokalną funkcją aplikacji, która aby mogła być wywołana musi być dokładnie określona (parametry) i wywołana w określonym stylu, musi również być napisana w tym samym języku jak obiekt wywoływany.
- Metoda Invoke jest bardzo efektywna i szybka w działaniu lecz nie możliwa do zastosowań w środowiskach rozproszonych, np. Interent

Architektura typu „Ścisłe sprzężenie” (ang. Tight Coupling)

- W celu ułatwienia komunikacji w aplikacjach rozproszonych, powstało wiele rozwiązań, które pakują zdalne dane w semantykę wywoływania metod lokalnych, przykładami są:
 - Remote Procedure Call (RPC),
 - Remote Method Invocation (RMI),
- Platformy i środowiska:
 - CORBA,
 - Microsoft DCOM, .NET Remoting,
 - Java RMI.

Przykład

PROSTA APLIKACJA KIP

Serwis bankowy

- Bank internetowy pozwalający na składanie depozytów przez użytkowników – pieniądze pochodzą z kont istniejących w innych bankach,
- Zastosowanie protokołu TCP/IP,
- Parametrami funkcji zdalnej są tylko imię i kwota 😊,
- KOD NASZEDO BANKU...

Serwis bankowy – kod

```
1. String hostName = "finance.bank.com";
2. int port = 80;
3. IPHostEntry hostInfo =
   Dns.GetHostByName(hostName);
4. IPAddress address = hostInfo.AddressList[0];
5. IPEndPoint endpoint = new IPEndPoint(address,
   port);
   Socket socket = new Socket(address.AddressFamily,
   SocketType.Stream, ProtocolType.Tcp);
   socket.Connect(endpoint);
   byte[] amount = BitConverter.GetBytes(1000);
6. byte[] name = Encoding.ASCII.GetBytes("Joe");
7. int bytesSent = socket.Send(amount);
8. bytesSent += socket.Send(name);
9. socket.Close();
```

Serwis bankowy - efekt

- 7 bitów przesłanych;
- szybkość;
- wieloplatformowość i ogólnodostępność – TCP/IP;
- ALE: protokół TCP/IP nadaje się tylko do przesyłania strumieni bitów (BitConverter),
- PROBLEM: niezgodność typów po konwersji;
- W przykładzie przesyłane są 4 bity reprezentujące 32-bitową liczbę zmiennoprzecinkową, w systemach 64-bitowych jest to 8 bitów,

Serwis bankowy - efekt

- Dlatego system 64-bitowy interpretuje nasze 7 bitów przesłanych jako jedną liczbę (włączając w to imię);
- Różnice w systemach: użycie formatu big-endian lub little-endian. Komputery PC używają tego ostatniego. Co to oznacza?

Serwis bankowy - efekt

- Wejście -> 16 bitowa liczba

00000001 00000010 Big Endian -> 513

00000010 00000001 Little Endian -> 258

- Rozwiązanie: stosowanie funkcji
htons Host to Network Short
htonl Host to Network Long

Serwis bankowy - efekt

- Problem z DNS – mamy nazwę domeny banku, a co jeśli nasza funkcja znajduje się w innej domenie lub poddomenie?
- Czy komunikat mimo braku nazwy zostanie wysłany, a potwierdzenie? Czy w przypadku jego braku system powtórzy transfer danych jeszcze raz?