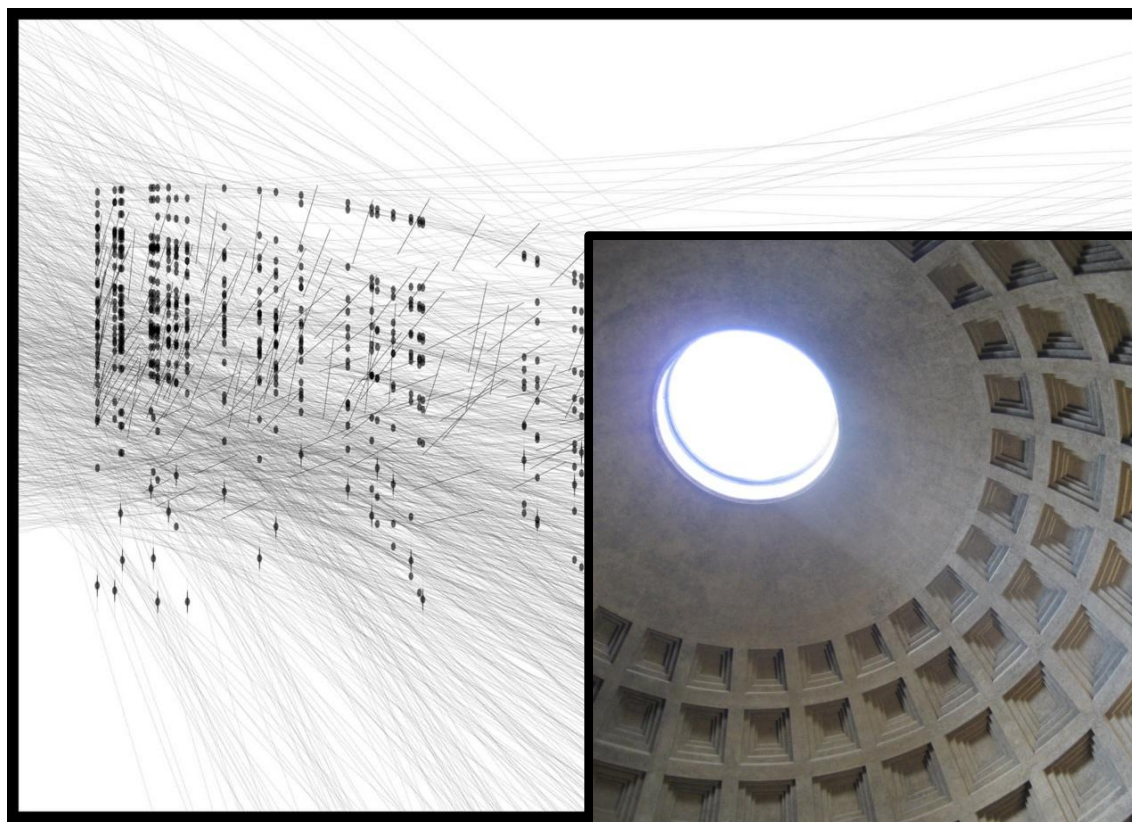


# Architektura i integracja systemów.

Andrzej Ratkowski

Wykład I. Wprowadzenie



# Informacje organizacyjne

Andrzej Ratkowski, A.Ratkowski@elka...

pok. 555, konsultacje: środy 17:00-18:00

strona: [andrzejratkowski.blogspot.com](http://andrzejratkowski.blogspot.com)

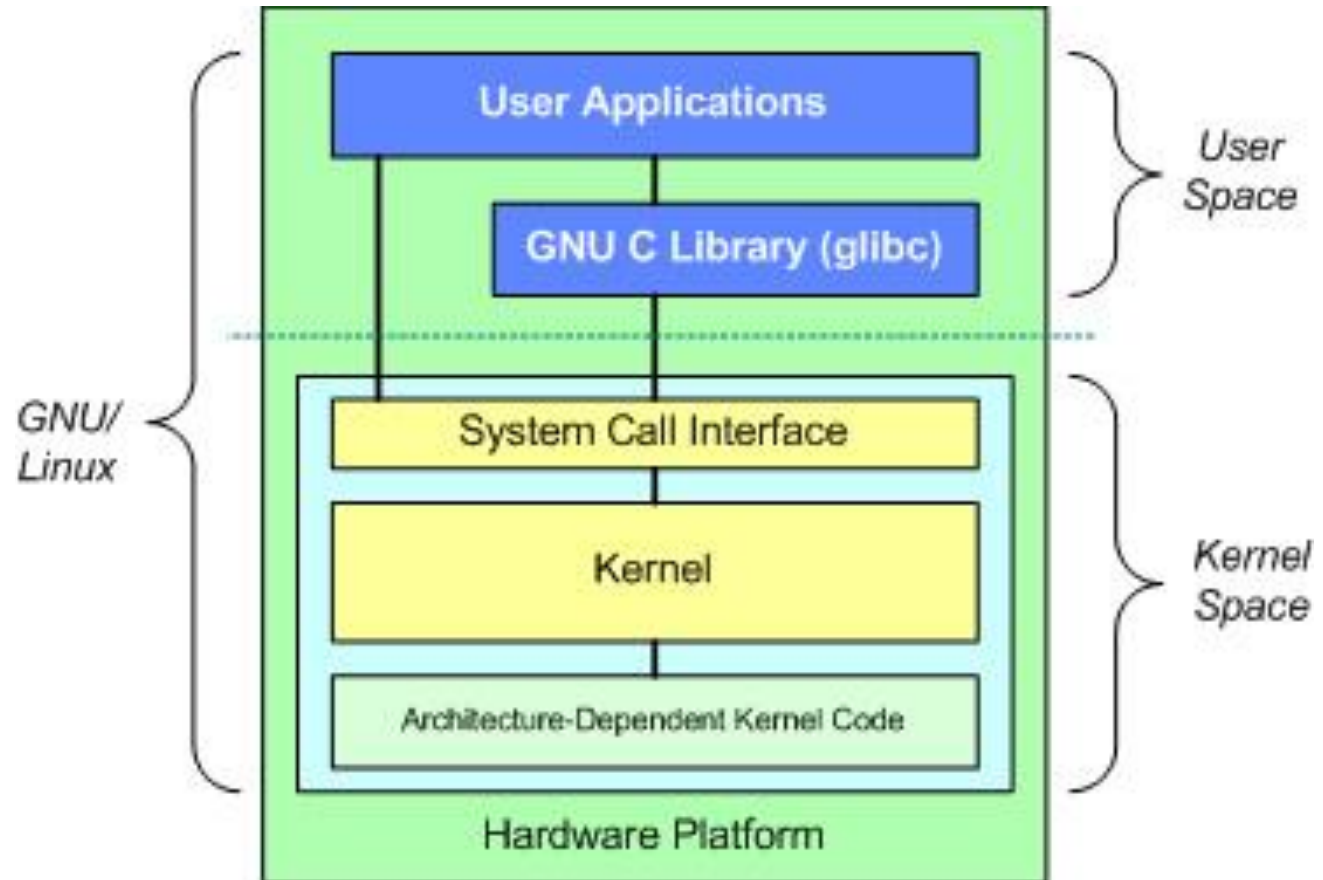
Zaliczanie: 2 kolokwia x 15 punktów +

Projekt: 30 punktów =

60 punktów

Projekt podzielony jest na 2 etapy po 20 i 10 punktów

# Architektura - przykład



# Architektura

Architektura programu lub systemu informatycznego to struktura lub struktury systemu na które składają się elementy oprogramowania, zewnętrzne właściwości tych elementów i relacje między nimi.

[Bass, Clements, Kazman]

# Co należy do domeny architektury?

## Należy

- Struktura systemu
- Procesy biznesowe
- Interface użytkownika
- Wymagania
- Technologie

## Nie należy

- Algorytmy
- Proces wytwarzania oprogramowania
- Model obiektowy oprogramowania

# Architektura

Architektura programu lub systemu informatycznego to struktura lub struktury systemu na które składają się **elementy oprogramowania**, zewnętrzne właściwości tych elementów i relacje między nimi.

[Bass, Clements, Kazman]

# Elementy oprogramowania

- Aplikacje (programy wykonywalne)
- Oprogramowanie systemowe (w tym systemy baz danych)
- Komponenty „software’owe”
- Moduły kodu źródłowego
- Usługi (services)
- Funkcje
- Klasy
- Interfejsy
- Obiekty
- Powiązania między obiektami
- Bazy danych (architektura danych)
  - Tablice, widoki etc.

# Elementy sprzętowe

- Urządzenia sieciowe
- Łączy
- Serwery
  - Serwery wirtualne
- Macierze dyskowe
- Systemy składowania danych (taśma)
- Stacje robocze
- Urządzenia wydruku



# Elementy logiczne...

- Tzn. **grupujące** (organizujące) pojęciowo i funkcjonalnie
  - systemy = wyróżniony zbiór komponentów
  - podsystemy = systemy
  - oprogramowanie = zbiór komponentów oprogramowania
  - aplikacje = zbiór komponentów oprogramowania realizujących pewien zbiór funkcji
  - środowiska (np. wykonawcze) = oprogramowanie oraz sprzęt niezbędny działania danego oprogramowania
  - platformy = sprzęt + oprogramowanie + cel, któremu służą dwa wcześniejsze
  - sieci SAN, LAN, WAN
  - procesy biznesowe (sekwencje funkcji /czynności/)

# Architektura

Architektura programu lub systemu informatycznego to struktura lub struktury systemu na które składają się elementy oprogramowania, zewnętrzne właściwości tych elementów i **relacje** między nimi.

[Bass, Clements, Kazman]

# Relacje

- Relacje=
  - Oddziaływanie (fizyczne, społeczne, wydawania poleceń)
  - Wymiana/przepływ informacji
  - Przynależność / wchodzenie w skład większego komponentu
  - Wskazywanie jednej danej przez inną daną

# Po co architektura?

1. Wysokopoziomowy opis systemu.
2. Komunikacja pomiędzy techniką (IT) a użytkownikami (biznes).
3. Podejmowanie wczesnych decyzji.
4. Możliwość stosowania wzorców (architektonicznych).

# Właściwa architektura (struktura) ?

- Dostosowana do potrzeb użytkownika/klienta
- Jednoznaczne przypisanie funkcji do modułów (*no overlapping*)
- Wspiera strategię biznesową (przedsiębiorstwa, produktu, oprogramowania).
- Właściwie wyważa pomiędzy elastycznością a stabilnością wymaganą przez organizację
- Właściwie opisana.

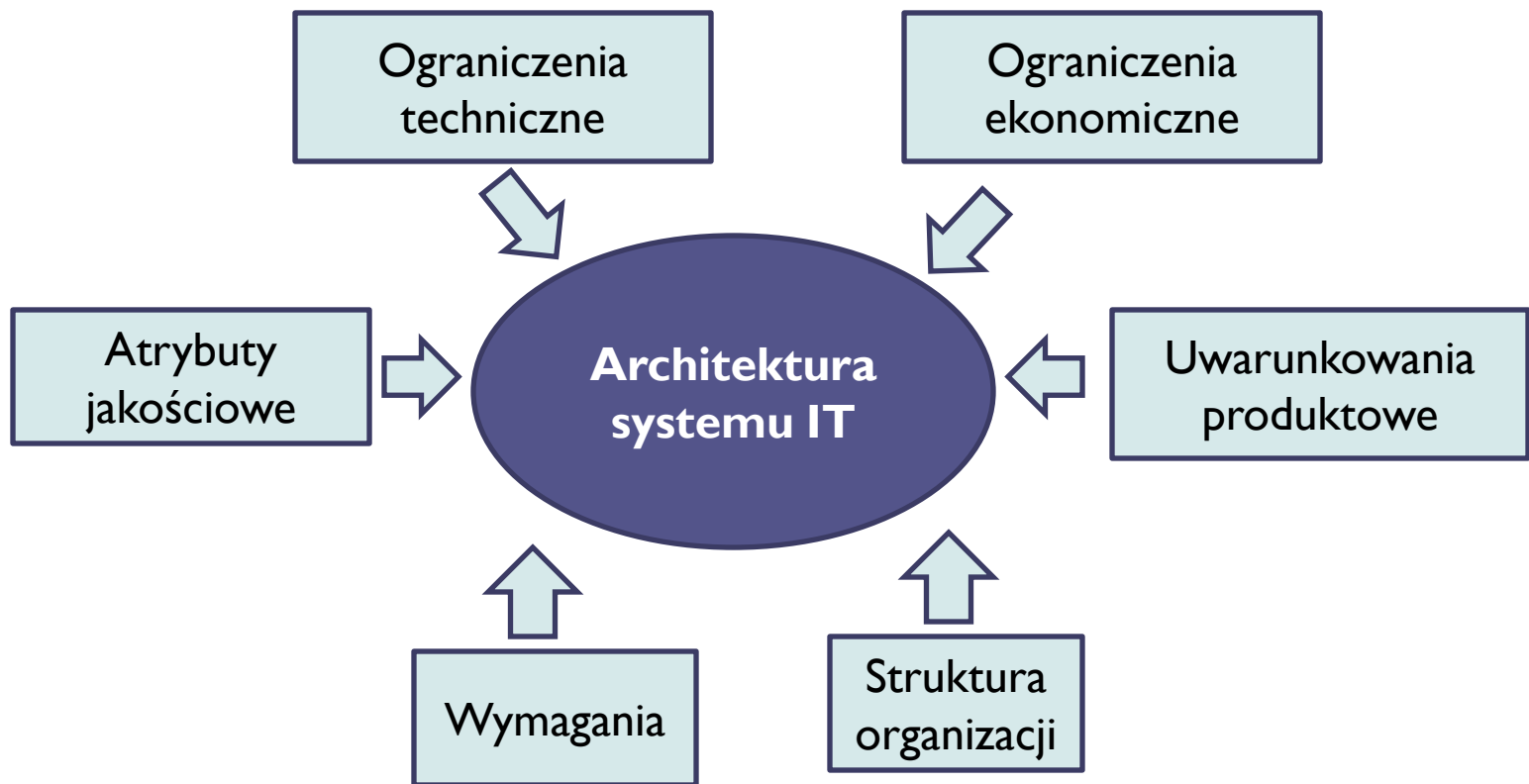
# Właściwa architektura(opis) ?

- Czytelna i zrozumiała
- Właściwy poziom abstrakcji (dla właściwego poziomu architektonicznego)
- Wewnętrznie spójna
- Jednoznaczny słownik pojęć
- Znana w organizacji na wszystkich poziomach

# Dlaczego architektura jest ważna?

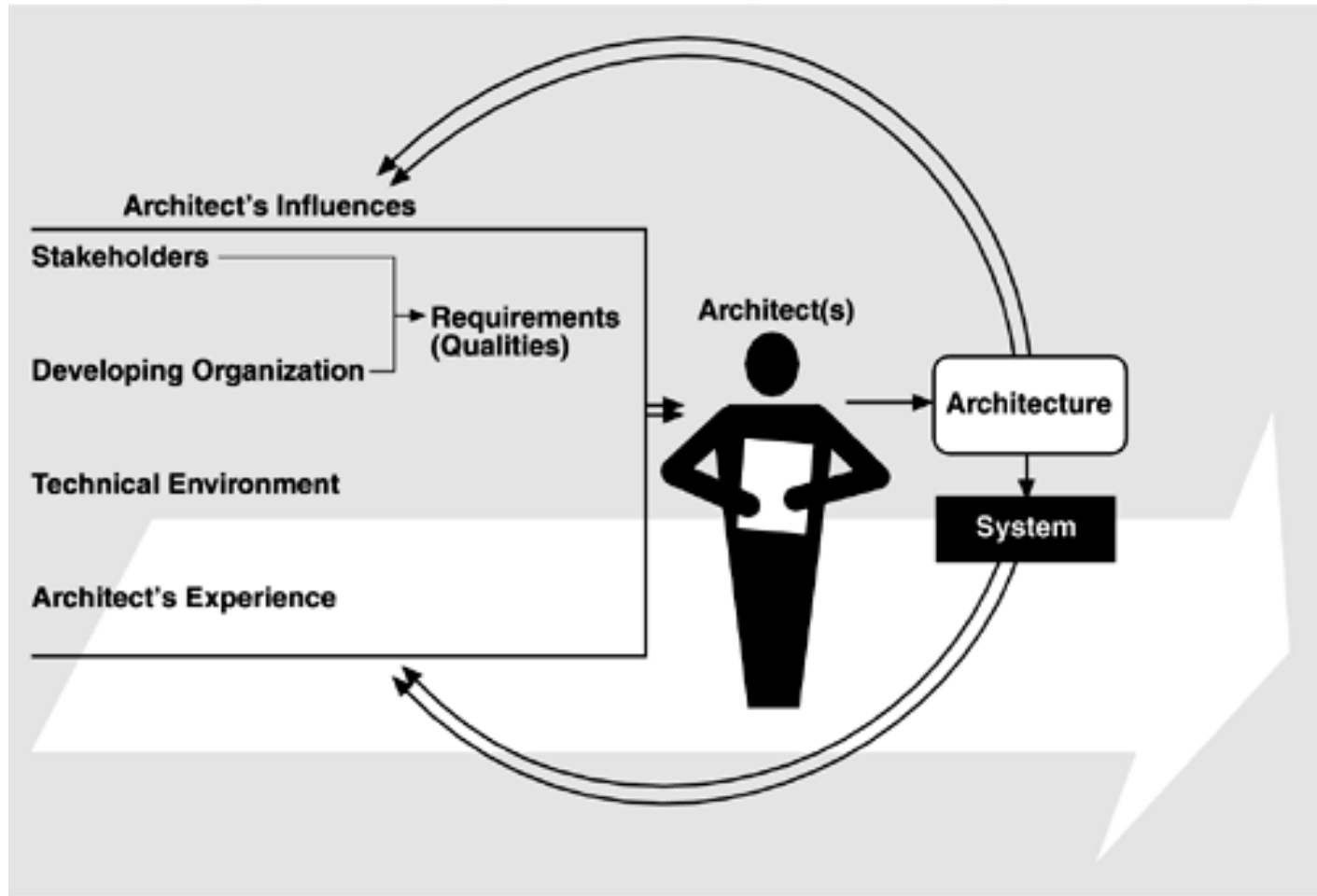
- Można podejmować właściwe decyzje na wczesnym etapie, kiedy decyzje są „tanie”.
- Można wnioskować o cechach systemu na podstawie architektury.
- Zapewnia język komunikacyjny pomiędzy użytkownikiem a wytwórcą oprogramowania (dostawcą rozwiązań).
- Umożliwia zarządzanie złożonymi systemami.

# Czynniki wpływające na architekturę





# Cykl biznesowo architektoniczny



# Role architektoniczne

Procesy Biznesowe

Oprogramowanie

Infrastruktura

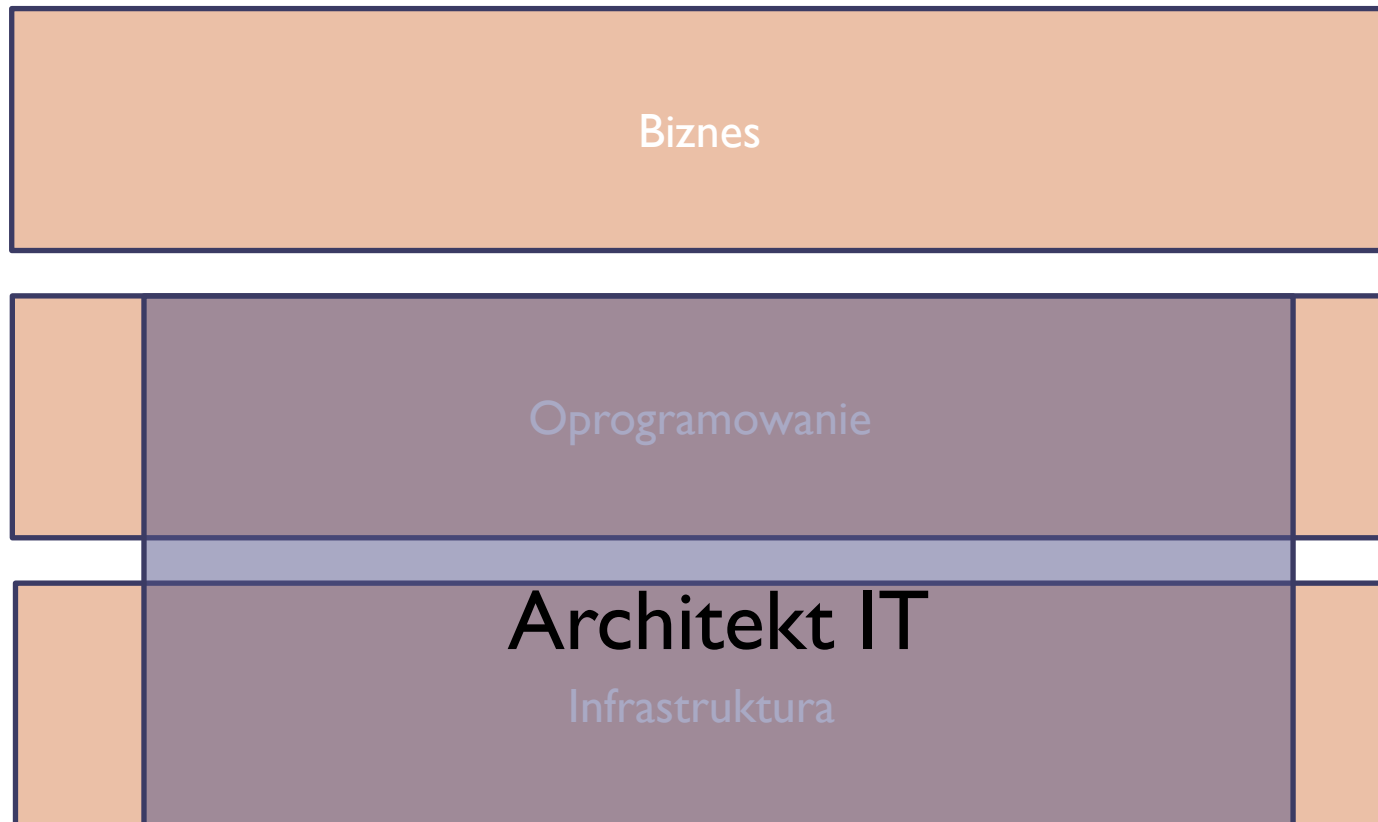
# Role architektoniczne

Biznes  
Architekt Biznesowy

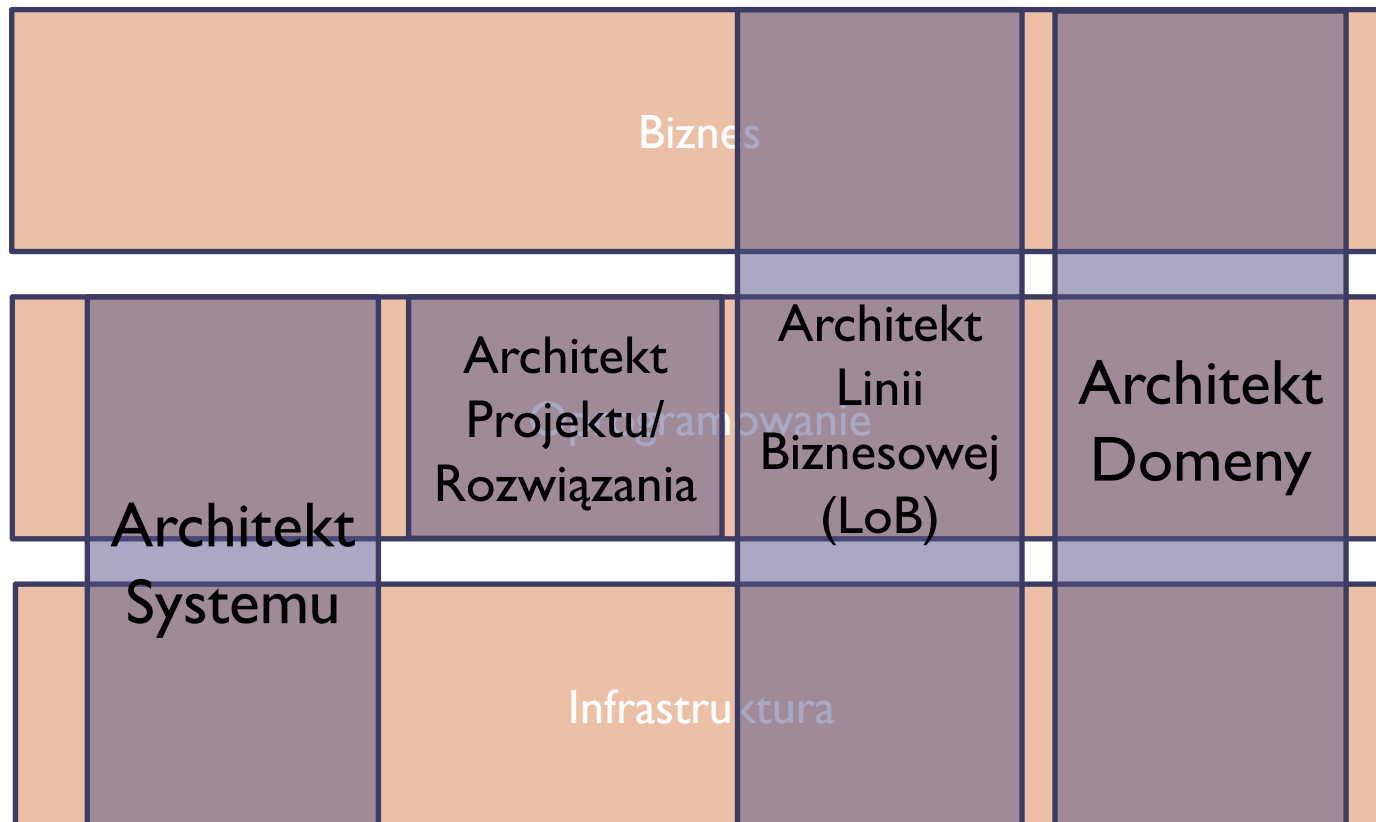
Oprogramowanie  
Architekt Oprogramowania

Infrastruktura  
Architekt infrastruktury

# Role architektoniczne



# Role architektoniczne



# Role architektoniczne



# Zadania architekta

- Tworzenie architektury
- Modyfikowanie architektury
- Nadzorowanie zgodności systemu z założoną architekturą
- Transformacje architektury związane z dużymi zmianami
- Ocena i wybór rozwiązań
- Utrzymywanie spójnego języka pojęć
- Realizacja strategii rozwoju systemu
- Tworzenie strategii

# Umiejętności architekta

- **Komunikacja**
- Umiejętności analityczne
- Umiejętność poszukiwania kompromisów
- Mediacja
- Dyplomacja
- Umiejętność prezentacji



# Projektowanie architektury

- Brak ugruntowanych metod projektowania architektury i standardów przemysłowych
- Co ma do dyspozycji architekt
  - Rozum, wiedza, doświadczenie
  - Komponenty, powiązania, notacje (UML i inne, w tym notacje ad hoc)
  - Taktyki architektoniczne
  - Style/wzorce architektoniczne
  - Zasadnicze elementy architektur systemów IT
  - Zasadnicze dylematy architektoniczne
- Literatura:
  - Bass L. i in.: *Architektura oprogramowania w praktyce*. WNT.
  - Hofmeister Ch. i in.: *Tworzenie architektury oprogramowania \ Architektura programowania w praktyce*. WNT

# Standardy przemysłowe związane z architekturą

- IEEE 1471
- TOGAF – wytwarzanie architektury korporacyjnej
- Siatka Zachmana
- Przemysłowe wzorce architektoniczne
  - Frameworx (eTOM, SID, TAM) – telekomunikacja
  - DoDAF – wojsko
  - ....

# Metody modelowania i opisu architektury

- tablica + kreda
- Powerpoint
- UML
- Archimate
- Decyzje Architektoniczne (Architectural Decissions)

# Integracja systemów w przedsiębiorstwie

*Enterprise Integration*

Sytuacja przedsiębiorstwa w kontekście integracji IT:

- wiele systemów (aplikacji) do obsługi różnych funkcji organizacji,
- skomplikowana organizacja systemów z wieloma zawikłanymi powiązaniem,
- potrzeba szybkich zmian systemów i ich powiązania,
- różnorodne technologie, modele danych, interfejsy użytkownika - środowisko heterogeniczne.

**Problem: jak zorganizować współpracę aplikacji żeby osiągnąć spójny system (*system of systems*) ?**

# Integracja

Metody radzenia sobie z integracją:

- Integracja point-to-point
- wzorzec architektoniczny szyny komunikacyjnej (EAI, ESB),
- Architektura usługowa SOA



# Literatura

**Architektura oprogramowania w praktyce.**

Len Bass, Paul Clements, Rick Kazman