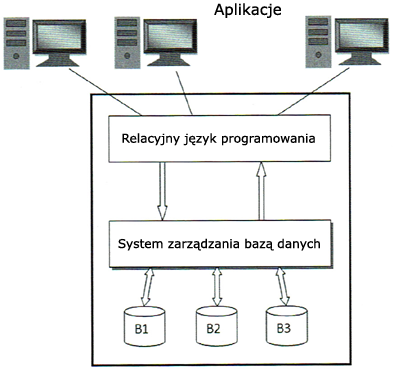
# Architektura systemu baz danych[[1]](#footnote-1)

## System zarządzania bazą danych wraz z bazami danych i językiem komunikowania się tworzą system baz danych.



Interakcja programu użytkowego (aplikacji) z bazą danych odbywa się najczęściej za pomocą języka SQL. **Jest to jedyny sposób komunikowania się aplikacji z bazą danych**.

Język SQL jest narzędziem dostępu do bazy danych stosowanym głównie przez projektantów aplikacji, projektantów baz danych i administratorów baz danych. Użytkownicy komunikują się z bazą danych za pomocą aplikacji i dopiero aplikacje komunikują się z bazą danych za pomocą poleceń SQL.

## **System baz danych – co to jest?**

Mianem systemu baz danych określa się – mówiąc najogólniej – system przechowywania danych zorganizowanych w pliki. Jego elementami są:

* same dane, które mogą mieć różną postać i wymagać ujednolicenia,
* sprzęt, a zatem komputery i serwery, gdzie są przechowywane informacje,
* programy służące do zarządzania bazą daną oraz jej obsługi,
* język komunikowania się z bazą danych,
* ludzie, a zatem użytkownicy bazy danych.

Warto jednak pamiętać, że to pojęcie jest stosowane bardzo często zamiennie z systemem zarządzania bazą danych. Jest to system, który pozwala wykonywać operacje na bazie danych i, zgodnie z nazwą – zarządzać nią.

## **Architektura systemu baz danych – rodzaje**

O architekturze systemu baz danych można mówić w dwóch ujęciach Pierwsze to to związane ze standardem ANSI/SPARC – opisujące w ogólnym ujęciu poziomy systemu baz danych. Drugie to architektura komunikacyjna.

**Architektura systemu baz danych wg standardu ANSI/SPARC**

Na tych schemacie, zaproponowanym po raz pierwszy w 1975 roku, opiera się zdecydowana większość stosowanych współcześnie baz danych, bez względu na ich rodzaj czy spectrum zastosowania. W tej perspektywie wyróżnia się trzy poziomy architektury systemu ba zdanych.

1. **Poziom wewnętrzny** – inaczej jest określany fizycznym, ponieważ jest związany z „fizycznym” sposobem przechowywania danych oraz tym, jak wygląda dostęp do nich. Przykładowo, w lokalnych bazach danych cały zbiór informacji jest przechowywany na jednym komputerze. Istnieją jednak również np. chmurowe bazy danych, w których informacje są przechowywane w modelu rozproszonym, a więc na co najmniej dwóch (a zazwyczaj więcej) urządzeniach fizycznych.
2. **Poziom zewnętrzny** – jest określany także poziomem użytkownika. W tym kontekście istotne jest to, w jaki sposób dane są prezentowane konkretnym odbiorcom. Na tym poziomie nie są prezentowane zbędne dane oraz te, do których użytkownik końcowy nie ma prawa dostępu.
3. **Poziom pojęciowy, czyli koncepcyjny** – to aspekt związany z opisem zawartości bazy danych oraz powiązań i relacji pomiędzy poszczególnymi jej elementami.

## **Architektura komunikacyjna systemu baz danych**

W tym ujęciu chodzi o zróżnicowanie sposobu, w jaki baza danych komunikuje się z klientem. Składają się na nią opis elementów stanowiących system baz danych, sposób ich powiązana, jak również logika stojąca za takim, a nie innym sposobem organizacji. W tym ujęciu wyróżnia się:

* **architekturę jednowarstwową**
* **architekturę dwuwarstwową**
* **architekturę trójwarstwową**

## **Architektura jednowarstwowa**

„Baza danych, system zarządzania baza danych i interfejs użytkownika udostępniony przez aplikację znajdują się na tym samym komputerze, tzn. baza danych znajduje się na tym samym komputerze, na którym jest przetwarzana i wykorzystywana.  
  
W praktyce stosuje się dwa sposoby komunikacji z bazą danych:

* **architektura klient-serwer**
* **architektura 3-warstwowa**

**Architektura klient-serwer**

W architekturze klient-serwer aplikacje zainstalowane na stacjach użytkowników komunikują się z bazą danych, wykorzystując sieciowe oprogramowanie dedykowane do komunikacji z systemem zarządzania bazą danych.

## C:\Users\Kasia\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\14B6D44B.tmp

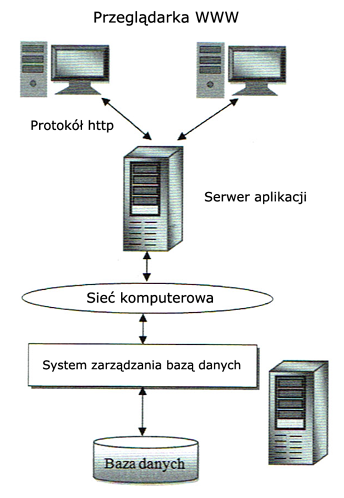
## **Jak działa architektura klient-serwer?**

W przypadku systemów baz danych najczęściej stosowana jest architektura klient-serwer. W tym modelu jeden klient może jednocześnie korzystać z usług wielu serwerów, a jeden serwer – być użytkowany przez wielu użytkowników tym samym czasie. Oznacza to podział ról pomiędzy klienta a serwer – tak, że spoczywają na nich inne obowiązki. Jak dokładnie wygląda podział ról? Otóż:

* klient jest stroną żądającą dostępu do bazy danych – to on aktywnie wysyła żądanie do serwera i oczekuje na jego odpowiedź. Korzysta on z dedykowanego interfejsu. Nie ma jednak wpływu na połączenie z serwerem, ochronę znajdujących się na nich danych ani na ich przetwarzanie;
* po stronie serwera jest ustanowienie połączenia, przetwarzanie danych oraz ich ochrona. Pełni on jednak rolę pasywną – świadczy usługę (a więc dostarcza przetworzonych w odpowiedni sposób danych) na żądanie.

**Architektura 3-warstwowa**

W architekturze 3-warstwowej pomiędzy użytkownikami a serwerem bazy danych znajduje się tzw. serwer aplikacji, który udostępnia umieszczone na nim aplikacje. Jest to architektura typowa dla aplikacji WWW. Aplikacje są udostępniane przez serwer aplikacji w postaci stron internetowych. Użytkownik komunikuje się z bazą danych przez przeglądarkę stron WWW. W odpowiedzi na polecenia użytkownika serwer aplikacji wysyła odpowiednie żądania do systemu zarządzania bazą danych, który wykonuje polecenia i przesyła ich wyniki do serwera aplikacji. Serwer aplikacji przesyła te wyniki do aplikacji użytkownika.



1. <https://summ-it.pl/architektura-systemu-baz-danych/> [↑](#footnote-ref-1)