

Przetwarzanie informacji

Rafał Adamczak

http://www.fizyka.umk.pl/~raad/Przetwarzanie_informacji.pdf

Informacja

Informacja (Wikipedia) (łac. informatio - wyobrażenie, pojęcie) to pojęcie o wielu definicjach w różnych dziedzinach. Zasadniczo istnieją dwa podstawowe sposoby rozumienia tego terminu. Pierwszy, który można nazwać obiektywnym i wywodzi się z fizyki i matematyki, gdzie informacja oznacza pewną własność fizyczną lub strukturalną obiektów, i drugi, subiektywny (kognitywistyczny), gdzie informacja jest tym, co umysł jest w stanie przetworzyć i wykorzystać do własnych celów.

Wiele wysiłku włożono w rozwinięcie pojęcia informacji w wyniku czego powstało wiele różnych definicji (Belkin 1978; Belkin & Robertson 1976; Brier 1998; Brookes 1975, 1980; Buckland 1991; Budd 2001; Day 2001; Derr 1985; Dervin 1977, 1983, 1999; Dretske 1981; Floridi 2002; Fox 1983; Goonatilake 1991; Hjørland 2002b; Losee 1990, 1997; MacKay 1969; Meadow & Yuan 1997; Pratt 1977; Raber 2003; Shannon & Weaver 1975; Thompson 1968 i wielu innych!!

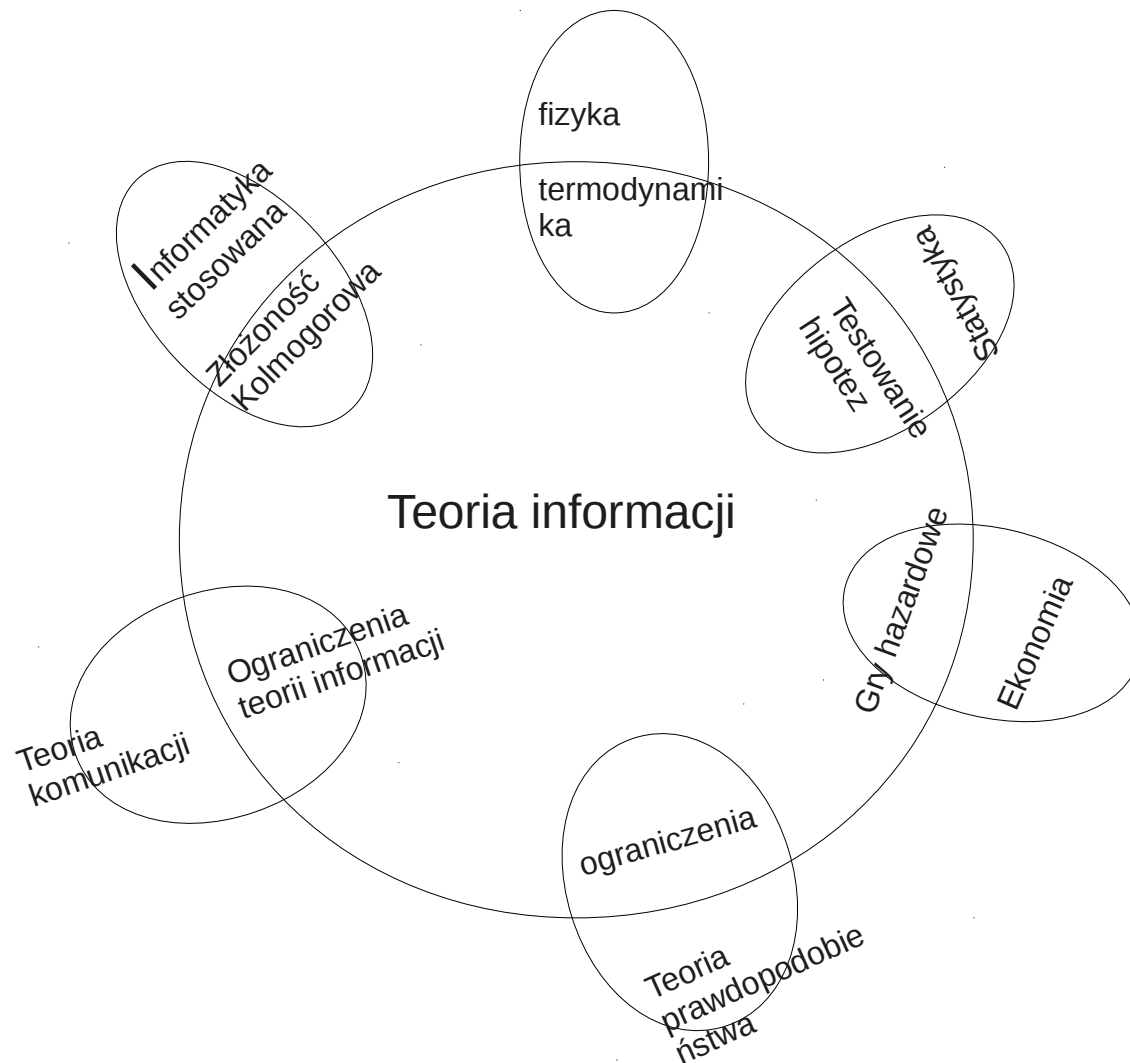
Źródło różnorodnych definicji: różne dyscypliny z których pochodzą twórcy tych definicji
(<http://www.success.co.il/is/dik.html>)

Ścisła definicja informacji zawiera pięć różnych poziomów. Informacja posiada następujące cechy:

1. Statystyka - informacja wyróżnia się specyficznym uporządkowaniem swych elementów składowych.
2. Syntaktyka - informacja jest podana w logicznym systemie, który można odczytać - np. reguły gramatyczne języka, czy alfabet Morse'a.
3. Semantyka - informacja posiada logiczne znaczenie jako przekaz treści - np.: pociąg do Wrocławia odjedzie o 16:00, z peronu czwartego.
4. Pragmatyka - informacja nadaje się do praktycznego wykorzystania np. w działaniach i procesach w systemach fizycznych, biologicznych.
5. Apobetyka - wykorzystanie informacji w procesach skutkuje w osiągnięciu logicznego celu.

Jeśli brakuje jednej z tych cech w danym kodzie, to kod ten nie jest informacją.

Teoria informacji a inne dziedziny nauki



Najważniejsze wielkości : entropia, informacja wzajemna

$$H(X) = - \sum_{x \in X} p(x) \log(p(x))$$

Entropia

Entropia od greckiego en-trope przemieniać się. W termodynamice oznacza prawdopodobieństwo stanu cieplnego materii, w matematyce poziom nieokreśloności sytuacji lub zadania. W informatyce natomiast entropia charakteryzuje możliwość oddawania informacji przez źródło.

Przykład: rzucamy 8-ścienną kostką i przekazujemy wynik

Entropia:

$$H(X) = - \sum_{i=1}^8 p(i) \log_2 p(i) = - \sum_{i=1}^8 \frac{1}{8} \log_2 \frac{1}{8} = - \log_2 \frac{1}{8} = \log(8) = 3(\text{bity})$$

1=001, 2=010, ...

Przykład: jakiś język polinezyjski

p	t	k	a	i	u
1/8	1/4	1/8	1/4	1/8	1/8

Entropia (przesłanie jednej litery):

$$H(X) = - \sum_{i \in \{p,t,k,a,i,u\}} p(i) \log_2 p(i) = - \left(4 \frac{1}{8} \log_2 \frac{1}{8} + 2 \frac{1}{4} \log_2 \frac{1}{4} \right) = 2 \frac{1}{2} \text{ (bitu)}$$

Kodowanie liter (dla częściej występujących liter używamy mniejszej liczby bitów)

p	t	k	a	i	u
100	00	101	01	110	111

Entropia – miara chaosu, stopnia nieuporządkowania

Fizyka – entropia układu rośnie, lub pozostaje stała, jeśli nie zostanie dostarczona energia

Miara niepewności:

Niska entropia – wysoka pewność, przewidywalność

Wysoka entropia – niska pewność, ale także ilość informacji jaką możemy uzyskać przeprowadzając eksperyment

Miara ilości informacji

$$I(X, Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \left(p \frac{(x, y)}{p(x) p(y)} \right)$$

Informacja wzajemna. Intuicyjnie informacja wzajemna mierzy, ile informacji o X można poznać, znając Y, czyli o ile poznanie jednej z tych zmiennych zmniejsza niepewność o drugiej.

$$I(Y; X) = H(Y) - H(Y|X)$$

Wzrost informacji

Przykłady zastosowania informacji wzajemnej

pojemność kanału komunikacyjnego, która jest maksymalną możliwą do uzyskania wzajemną informacją pomiędzy wejściem a wyjściem z kanału

w **kryptologii teoretycznej i kwantowej** przy ocenie bezpieczeństwa bezwarunkowego systemów szyfrowania

w **uczeniu maszynowym** przez zastosowanie **ukrytych modeli Markowa (HMM)**

porównywanie modeli językowych w **lingwistyce komputerowej**

rekonstrukcja obrazu w **tomografii komputerowej** dla zastosowań medycznych

nakładanie obrazów (ang. image registration).

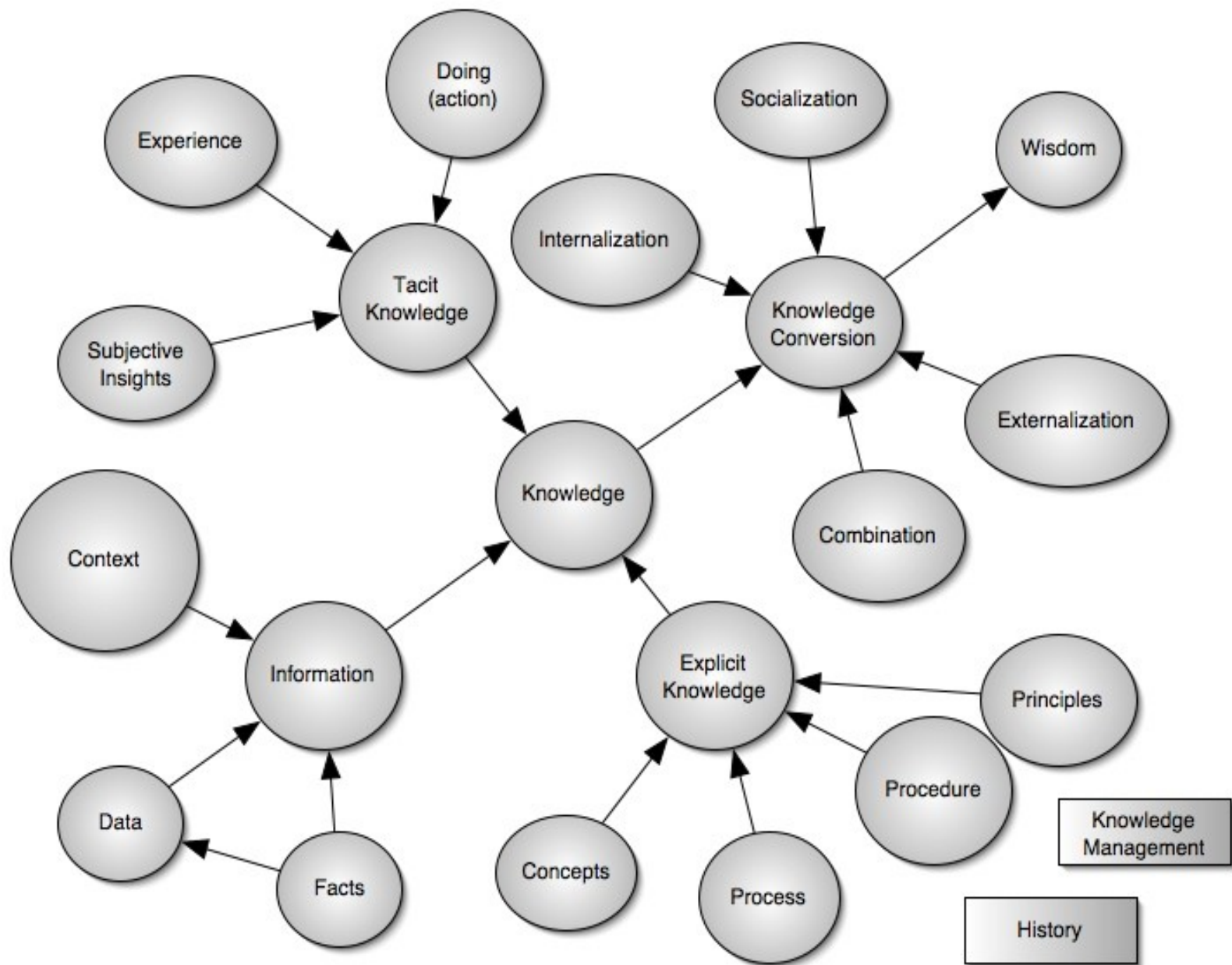
Czym jest informacja? Wbrew pozorom pojęcie to nie jest wcale tak łatwo zdefiniować. Człowiek posiada pewne wyobrażenia o świecie, ucząc się nabywa nie tylko informacje ale i wiedzę. Wiemy na przykład jak prowadzić samochód. Wiedza jest czymś bardziej ogólnym niż informacja. Informacją jest stwierdzenie: maksymalna szybkość osiągnięta przez ten samochód wynosi 160 km/h. Informacja jest pojęciem dość abstrakcyjnym, gdy podanie maksymalnej szybkości jako 100 mil/h lub 44.4 m/sek zawiera te same informacje, chociaż liczby są inne. Liczby te moglibyśmy dodatkowo zapisać w zupełnie inny sposób, alfabetem arabskim lub pismem Braila, nie zmieniając wcale informacji. Konkretna reprezentacja informacji to właśnie dane.

Są cztery prawa natury w odniesieniu do informacji:

1. Informacja jest fundamentalną wielkością niematerialną a nie właściwością materii.
2. Czynniki materialne takie jak procesy fizyczne czy chemiczne nie mogą stworzyć wielkości niematerialnej.
3. Informacja nie może powstać w procesach statystycznych, gdyż nie zawierałaby wtedy m.in. syntaktyki, semantyki.
4. Informacja może pochodzić tylko od inteligentnego nadawcy, tzn. twórczego, samodzielnie myślącego, działającego w określonym celu i posiadającego własną wolę.

Z czwartego prawa wynikają prawa dodatkowe:

- 4a. Każdy kod oparty jest na wzajemnej umowie pomiędzy nadawcą i odbiorcą dotyczącej zgodnego rozumienia pięciu zasadniczych cech informacji.
- 4b. Nie ma nowej informacji bez inteligentnego nadawcy.
- 4c. Cały łańcuch przekazu informacji można prześledzić w tył, aż do źródła tej informacji, czyli do inteligentnego nadawcy. Np. gdy słuchamy radia jesteśmy świadomi, że to nie radio jest źródłem informacji, nie masz radiowy, ale nadawca programu – konkretna osoba. Nadawca nie zawsze musi być widzialny, ale zawsze musi być, jeśli mamy do czynienia z informacją.



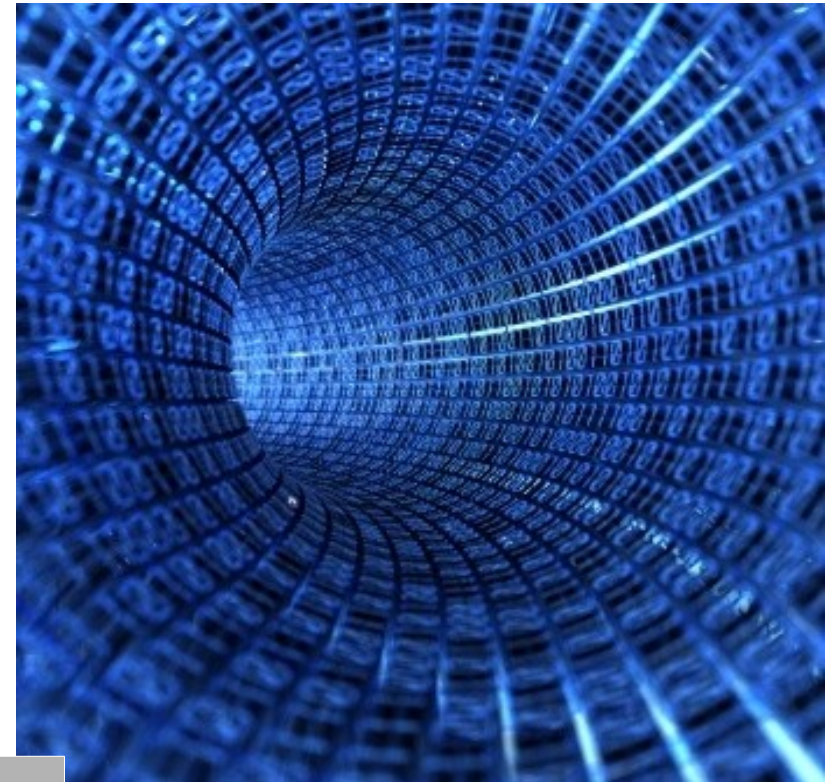
Rysunek zze strony

http://www.nwlink.com/~Donclark/knowledge/knowledge_typology.html

Dane, informacja, wiedza i mądrość

- Dane

- Dane są trzy rzeczy, jedna jest czerwono brązowa, dwie są szare
- jedna z rzeczy waży 5 kb, druga 20g I trzecia 2 tony.
- Jedna z rzeczy ma trąbę, druga ogon a trzecia ma managera



- Informacja

ID	KOLOR	WAGA	INNE
1	czerwono-brązowy	5kg	Ma managera
2	szary	20g	Ma ogon
3	szary	2 tony	Ma trąbę

Wiedza

ID	KOLOR	WAGA	INNE	TYP
1	czerwono-brązowy	5kg	Ma managera	kot
2	szary	20g	Ma ogon	mysz
3	szary	2 tony	Ma trąbę	słoń

Mądrość

Najwyższy stopień zrozumienia. Poziom ten osiąga się po poznaniu dostatecznie dużej ilości wzorców i meta wzorców, które następnie jesteśmy w stanie zsyntetyzować i użyć w nowy nie spotykany dotąd sposób

Pewien garncarz odwiedzał dwór chińskiego cesarza i spotkał tam czytającego jakąś księgę urzędnika. Pytając się o drogę, zapyta zaciekawiony: o czym, panie, czytasz? Urzędnik na to: to księga mądrości (Tao Te King). Garncarz zaśmiał się na to. Czemu się śmiejesz – pyta rozzłoszczony urzędnik. Czcigodny pan musi żartować. Ja nawet swojego czeladnika nie mogę nauczyć z książek lepienia garnków, jak więc mądrość zawrzeć można w książce?

Gra w 20 pytań

<http://www.20q.net/>

Bity, bajty czyli wewnętrzna reprezentacja liczb

bit: 1 0

bajt: 8 bitów możliwa jest więc za pomocą jednego bajta przedstawienie 2^8 różnych liczb

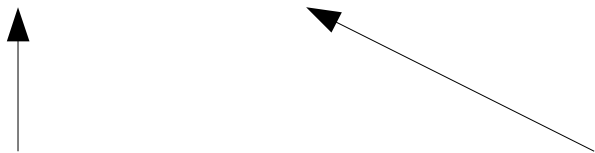
1 kB (kilobajt) – 2^{10} - 1024 bity

1MB (megabajt) – 2^{20} – 1048576

1GB (gigabajt) – 2^{30} - 1073741824

$$00001100 - 0*2^7+0*2^6+0*2^5+0*2^4+1*2^3+1*2^2+0*2^1+0*2^0=12$$

$$00100001 - 0*2^7+0*2^6+1*2^5+0*2^4+0*2^3+0*2^2+0*2^1+1*2^0=33$$



bit
najbardziej
znaczący

bit najmniej
znaczący

Dec	Hex	Char
128	80	À
129	81	Á
130	82	Â
131	83	Ã
132	84	Ä
133	85	Å
134	86	Æ
135	87	Ç
136	88	È
137	89	É
138	8A	Ê
139	8B	Ë
140	8C	Ì
141	8D	Í
142	8E	Î
143	8F	Ï
144	90	Ï
145	91	ð
146	92	É
147	93	Ò
148	94	Ó
149	95	Ô
150	96	Õ
151	97	Ö
152	98	×
153	99	Ø
154	9A	Ù
155	9B	Ú
156	9C	Û
157	9D	Ü
158	9E	Ý
159	9F	ÿ

Dec	Hex	Char
160	A0	à
161	A1	á
162	A2	â
163	A3	ã
164	A4	ä
165	A5	å
166	A6	æ
167	A7	ç
168	A8	è
169	A9	é
170	AA	ê
171	AB	ë
172	AC	ì
173	AD	í
174	AE	î
175	AF	ï
176	B0	ï
177	B1	ð
178	B2	é
179	B3	ò
180	B4	ó
181	B5	ô
182	B6	õ
183	B7	ö
184	B8	×
185	B9	ø
186	BA	ù
187	BB	ú
188	BC	û
189	BD	ü
190	BE	ý
191	BF	ÿ

Dec	Hex	Char
192	C0	Ĺ
193	C1	ł
194	C2	Ł
195	C3	ł
196	C4	ł
197	C5	ł
198	C6	ł
199	C7	ł
200	C8	ł
201	C9	ł
202	CA	ł
203	CB	ł
204	CC	ł
205	CD	ł
206	CE	ł
207	CF	ł
208	D0	ł
209	D1	ł
210	D2	ł
211	D3	ł
212	D4	ł
213	D5	ł
214	D6	ł
215	D7	ł
216	D8	ł
217	D9	ł
218	DA	ł
219	DB	ł
220	DC	ł
221	DD	ł
222	DE	ł
223	DF	ł

Dec	Hex	Char
224	E0	Ā
225	E1	ā
226	E2	Ă
227	E3	ă
228	E4	Ą
229	E5	ą
230	E6	Ȧ
231	E7	ȧ
232	E8	Ȧ
233	E9	ȧ
234	EA	Ȧ
235	EB	ȧ
236	EC	Ȧ
237	ED	ȧ
238	EE	Ȧ
239	EF	ȧ
240	F0	Ī
241	F1	ī
242	F2	Ĭ
243	F3	ĭ
244	F4	Ĳ
245	F5	ĳ
246	F6	Ĳ
247	F7	ĳ
248	F8	Ĳ
249	F9	ĳ
250	FA	Ĳ
251	FB	ĳ
252	FC	Ĳ
253	FD	ĳ
254	FE	Ĳ
255	FF	ĳ

$2^{10}=1024=1\text{K}$, kilobajt, typowa strona tekstu to kilka KB;
 $2^{20}=1024\text{K}=1\text{M}$, megabajt, książka bez grafiki lub minuta muzyki;
 $2^{30}=1024\text{M}=1\text{G}$, gigabajt, film cyfrowy, sporo grafiki, ludzki genom;
 $2^{40}=1024\text{G}=1\text{T}$, terabajt, duża biblioteka, szerokoekranowy film w kinie;
 $2^{50}=1024\text{T}=1\text{P}$, petabajt, ludzka pamięć;

1 książka = $70*50*300=1\,050\,000 \sim 1\text{MB}$

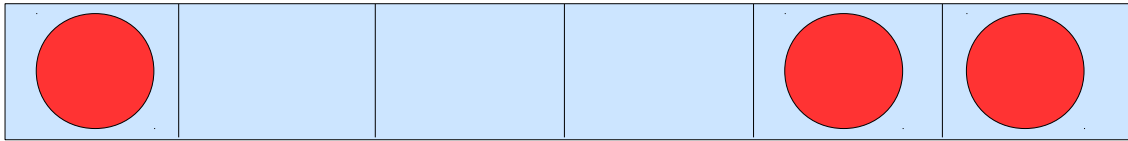
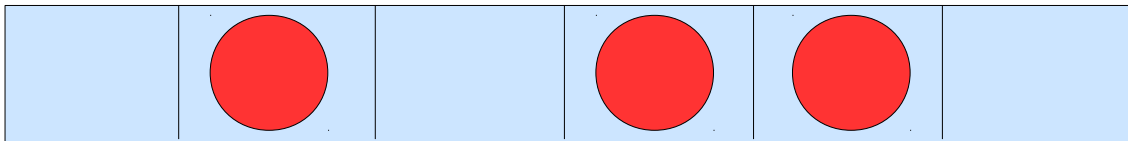
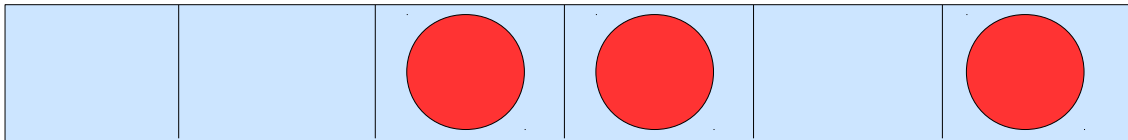
1 zdjęcie = $800*600*3=1\,440\,000 \sim 1.4\text{MB}$

1 film (HDMI – $1920*1200$, 1.5h) = $1920*1080*3*24*60*90=806\text{ GB}$

Biblioteka Kongresu USA zawiera około 20 TB informacji tekstowej, ale filmy i informacja graficzna dużo więcej. Film z jakością wystarczająca dla kin zawiera setki gigabajtów
Eksperymenty naukowe dostarczają terabajtów danych dziennie.

Bramki i logika kombinatoryczna

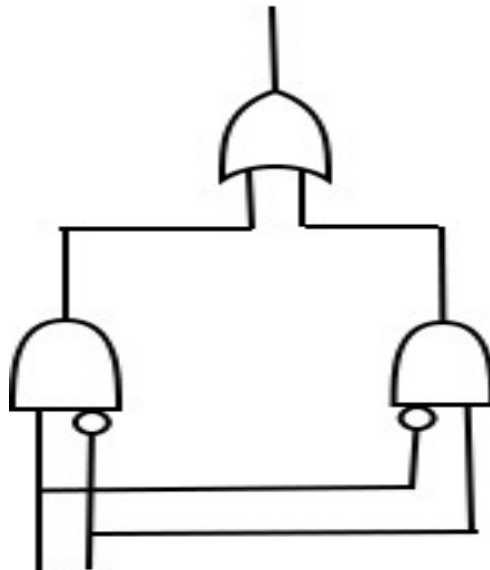
Ograniczamy się do układu dwójkowego, jedynymi cyframi są 0 i 1.



0	+	0	=	0
0	+	1	=	1
1	+	0	=	1
1	+	1	=	0 plus bit przeniesienia

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

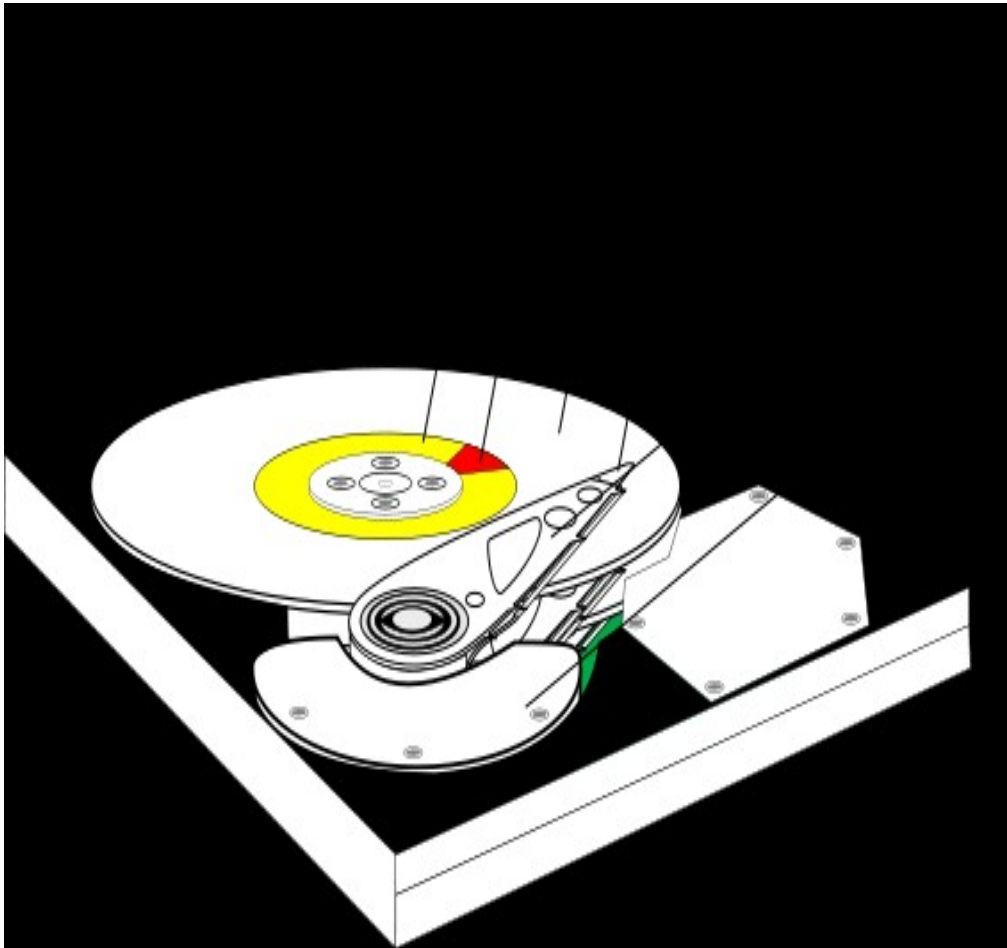
Mając do dyspozycji bramki AND, OR i NOT możemy stworzyć dowolną inną bramkę.



Przechowywanie danych



Zapisywanie informacji na dysku



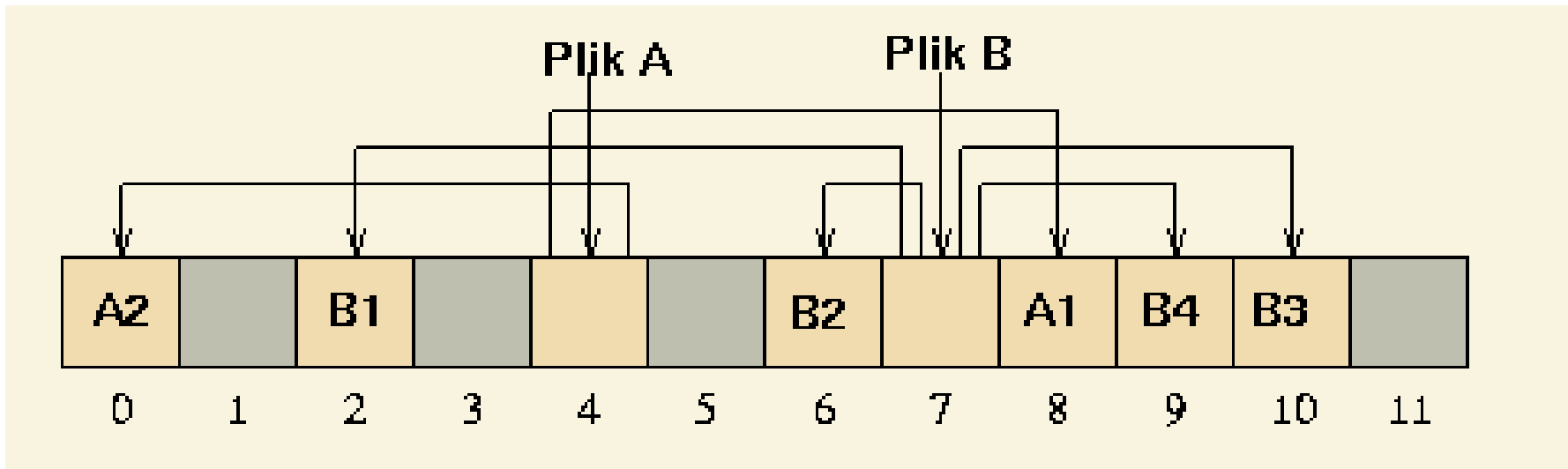
Dla dysków twardych najważniejsze są parametry: pojemność, szybkość transmisji danych (np. 300MB/s), czas dostępu (np. 10 ms), prędkość obrotowa talerzy (np. 7200 obr/min.) oraz MTBF.

RAM – czas dostępu 5 ns,

Tablica FAT (File Allocation Table)

Tablica informuje o tym, do jakich plików należą poszczególne obszary dysku. Zależnie od swojego rozmiaru plik może rozciągać się na większą liczbę jednostek alokacyjnych. Każdy obszar dysku (klaster) posiada zapis w FAT wyszczególniający numer klastera, gdzie kontynuowany jest dany plik.

Plik na dysku twardym



Przechowywanie informacji

Przechowywanie informacji miało ogromne znaczenie zanim wynalezione zostały komputery:

- Przechowywanie informacji rządowych
- DDC (Dewey Decimal System) system wynaleziony przez Melvil Dewey w 1876 roku przeznaczony do organizacji książek w bibliotekach. Używany w 200000 bibliotek w 135 krajach (wikipedia)
 - 000 – Computer science, information & general works
 - 100 – Philosophy and psychology
 - 200 – Religion
 - 300 – Social sciences
 - 400 – Language
 - 500 – Science (including mathematics)
 - 600 – Technology/Applied Science
 - 700 – Arts and recreation
 - 800 – Literature
 - 900 – History, geography, and biography

Bazy danych – uporządkowany zbiór danych, który przetwarzany jest za pomocą specjalnych programów DBMS (Database Management System). Podwaliny do teorii baz danych zostały utworzone w latach 60

Dostępne bazy danych

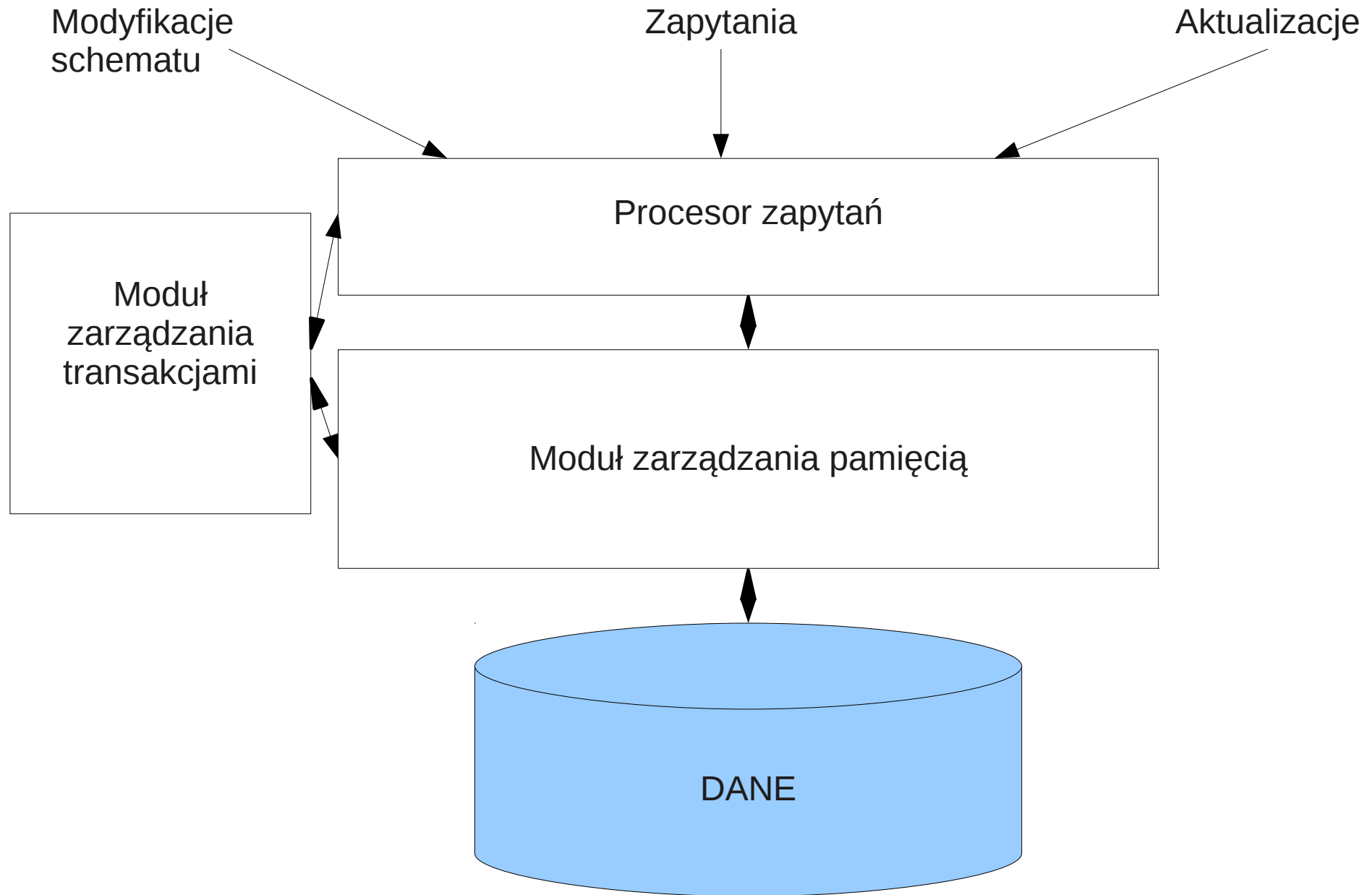
Komercyjne

- Oracle
- IBM
 - DB2
 - Informix
- Microsoft
 - SQL Server
- Sybase
 - Adaptive server

Niekomercyjne

- MySQL
- PostgreSQL
- FireBird

Architektura systemu DBMS



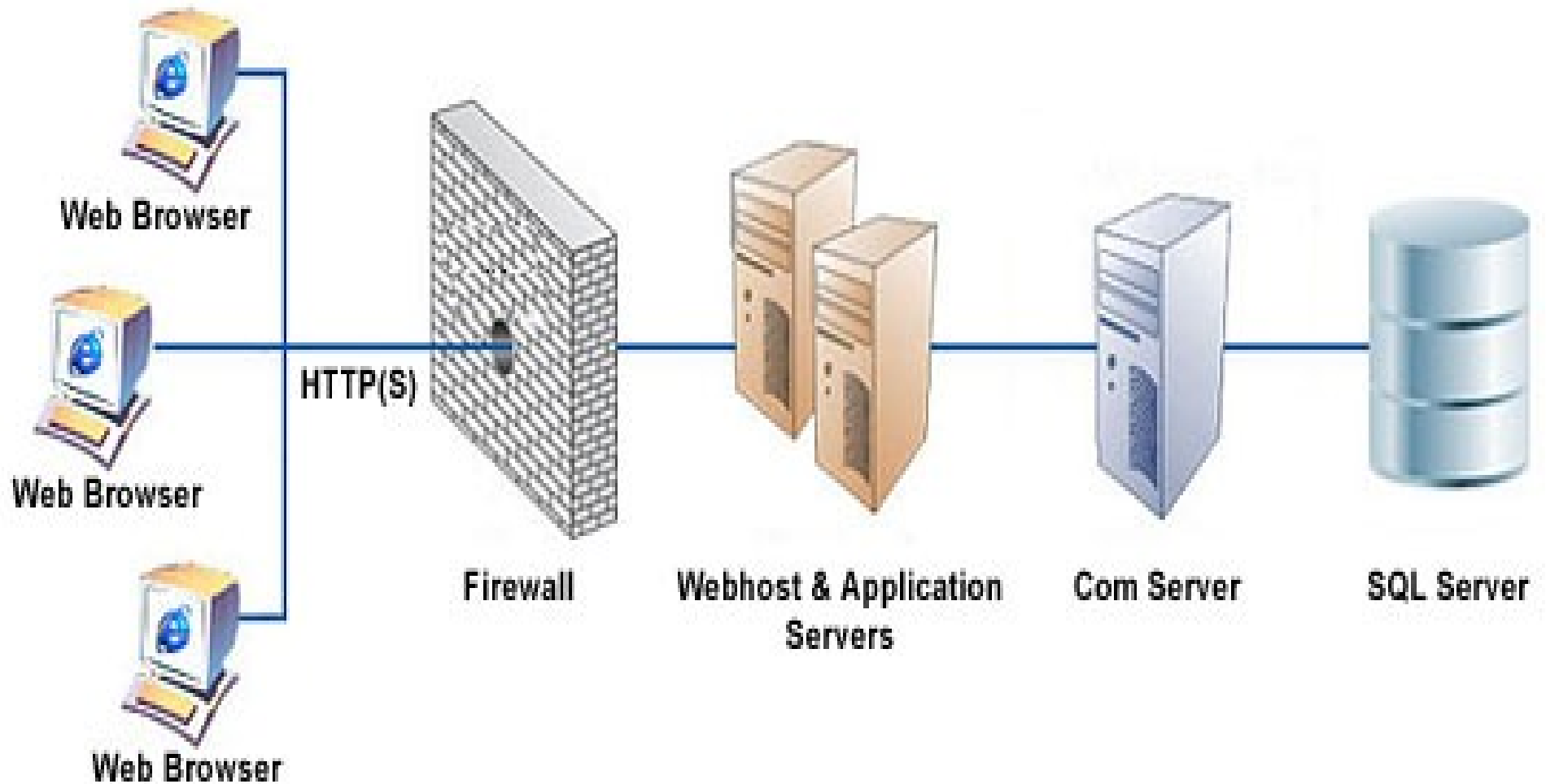
Zadaniem modułu zarządzania zapytaniami jest przekształcenie zapytania lub operacji w bazie danych wyrażanych zazwyczaj w języku bardzo wysokiego poziomu (np. SQL) w ciąg poleceń żądających dostarczenia określonych danych.

Jest sekwencją logicznie powiązanych operacji na bazie danych, która przeprowadza bazę danych z jednego stanu spójnego w inny stan spójny. Systemy bazy danych umożliwiają łączenie operacji w transakcje i gwarantują poprawne zarządzanie transakcjami.

Właściwości:

- Niepodzielność. Żądamy aby cała transakcja została przeprowadzona.
- Spójność. Baza danych musi stwarzać warunki spójności.
- Izolacja. Gdy dwie transakcje przeprowadzane są równocześnie ich działania nie mogą na siebie wpływać
- Trwałość. Jeśli transakcja się zakończy to jej wynik nie może zostać utracony z powodu awarii, nawet jeśli awaria występuje zaraz po zakończeniu transakcji.

Architektura klient-serwer



Bazy danych

Bazy danych można podzielić według struktur organizacji danych, których używają:

* Bazy proste :

- bazy kartotekowe (złożona z jednej lub kilku tablic zawierających rekordy, z których każdy zawiera identyczną strukturę pól. Każda tablica danych jest samodzielnym dokumentem i nie może współpracować z innymi tablicami)
- sieciowe
- hierarchiczne bazy danych (np. struktura katalogów na dysku)

* Bazy złożone :

- bazy relacyjne
- bazy obiektowe
- bazy relacyjno-obiektowe
- strumieniowe bazy danych
- temporalne bazy danych

Bazy danych - wymagania

- Spójność bazy danych
 - Baza danych jest spójna jeśli jej stan jest zgodny ze stanem reprezentowanego przez nią fragmentu świata rzeczywistego.
 - Baza danych jest spójna jeśli są w niej spełnione wszystkie ograniczenia referencyjne i integralnościowe.
- Poprawne modelowanie świata rzeczywistego
- Autoryzacja dostępu do danych
- Współbieżność dostępu do danych
- Metadane

Bazy Kartotekowe

Pole

Jan	Kowalski	056676766773	Toruń
Jan	Nowak	048938989	Toruń
Bonifacy	Nowak	982839289	Warszawa
Damazya	Kowalski	289239898	Włocławek
Damian	Nowacki	2893989833	Gdańsk

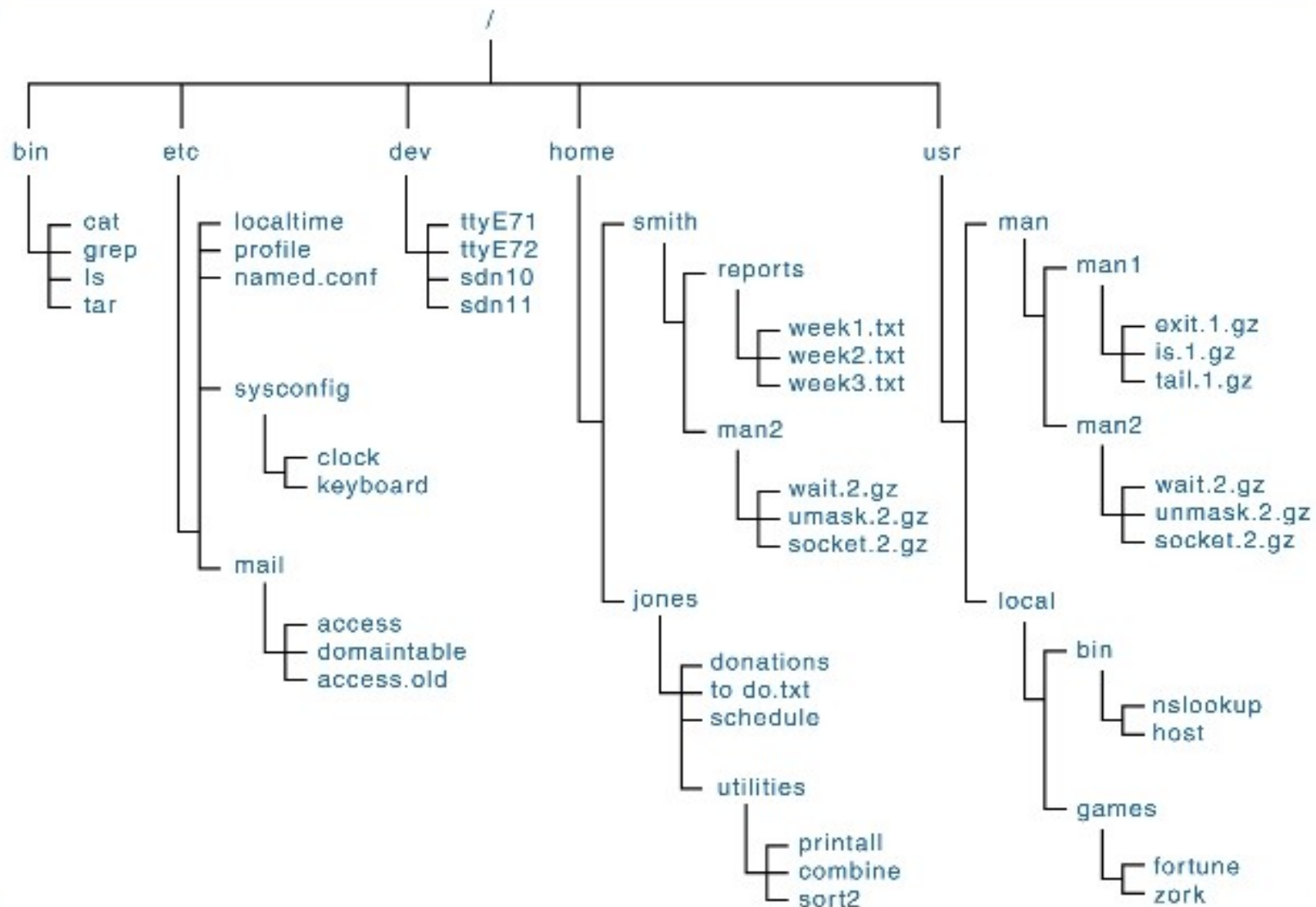
wartość

rekord

Typy danych

- * Tekst - typ tekstowy. Maksymalny rozmiar pola 255 bajtów.
- * Memo - typ notatnikowy. Maksymalny rozmiar pola 64 000 bajtów.
- * Liczba - typ liczbowy. Rozmiar pola od 1 do 4 bajtów.
- * Waluta - typ kwotowy, zarezerwowany dla pieniędzy. Rozmiar pola wynosi 8 bajtów.
- * Data/Godzina - umożliwia zapisywanie w polu daty i godziny. Rozmiar pola wynosi 8 bajtów.
- * Tak/Nie - typ logiczny, prawda lub fałsz. Rozmiar pola wynosi 1 bit.
- * Obiekt OLE - służy do przechowywania tzw. BLOBów. Maksymalny rozmiar pola - do 1 GigaBajta.
- * Skrót BLOB pochodzi od Binary Large Object (duży obiekt binarny) i oznacza plik binarny, który DBMS przechowuje jako całość w jednym polu rekordu. Dzięki stworzonemu mechanizmowi można w rekordzie przechowywać dowolnego typu informacje, takie jak grafika, teledyski i pliki zawierające dźwięk.
- * Licznik - Dane tego typu to kolejne liczby naturalne, generowane w sposób automatyczny przez Access. Rozmiar pola wynosi 4 bajty.

A Unix Directory Tree



Model sieciowy bazy danych

Założmy, że mamy 3 typy rekordów:

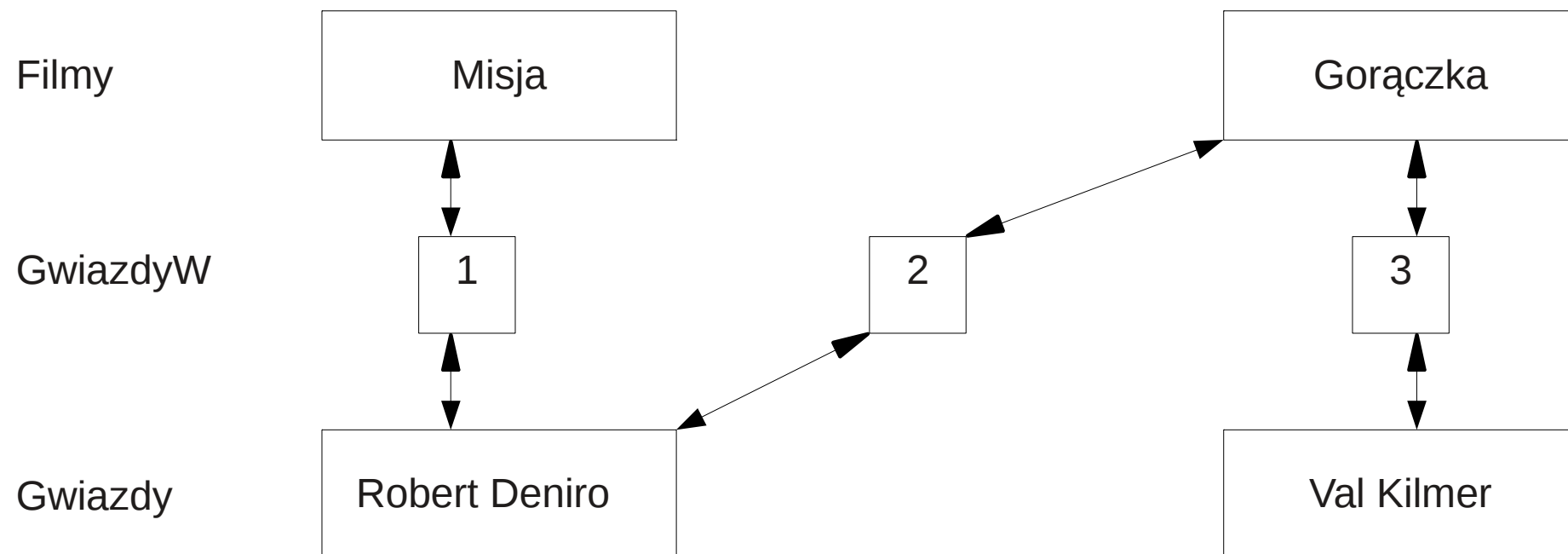
Gwiazdy (nazwisko, adres)

Filmy (tytuł, rok, długość, typ filmu)

GwiazdyW – typ łączący

Model sieciowy jest uogólnieniem modelu hierarchicznego.

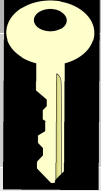
Ma on, podobnie jak model hierarchiczny, dwie struktury danych: typy rekordów i powiązania. Jednak powiązania między tabelami nie muszą być jednokierunkowe, a tabele mogą mieć więcej niż jedno powiązanie.



Klucz w bazie danych

Głównymi zastosowaniami kluczy w bazie danych są:

- poprawienie wydajności działania bazy danych
- zapewnienie automatycznej kontroli poprawności operacji przetwarzania danych
- Klucz główny – Primary key
 - – Jednoznaczny
 - – Unikalny
- Klucz obcy
 - – Odwołujący się do Primary Key z innej tabeli

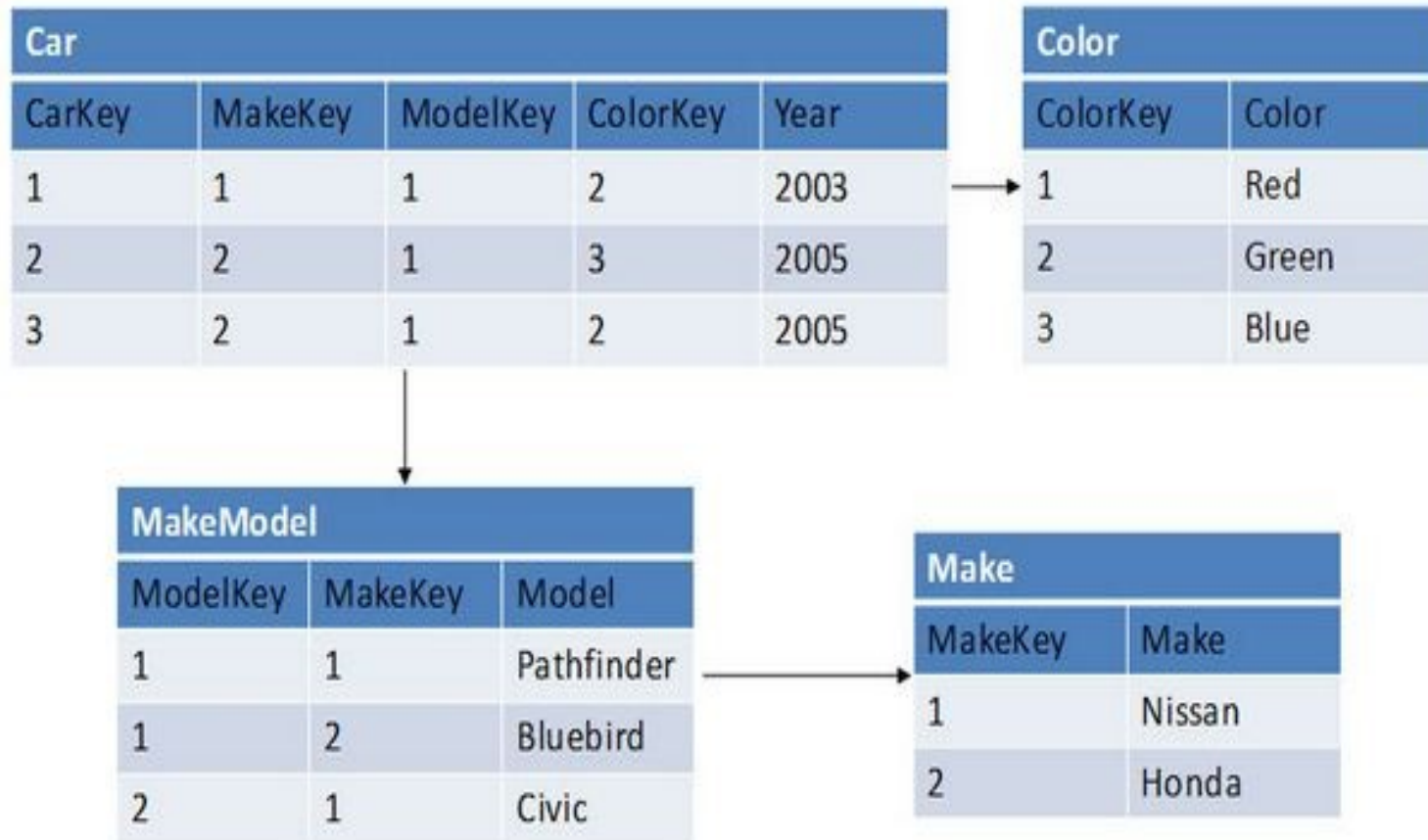


1	Jan	Kowalski	056676766773	Toruń
2	Jan	Nowak	048938989	Toruń
3	Bonifacy	Nowak	982839289	Warszawa
4	Damazy	Kowalski	289239898	Włocławek

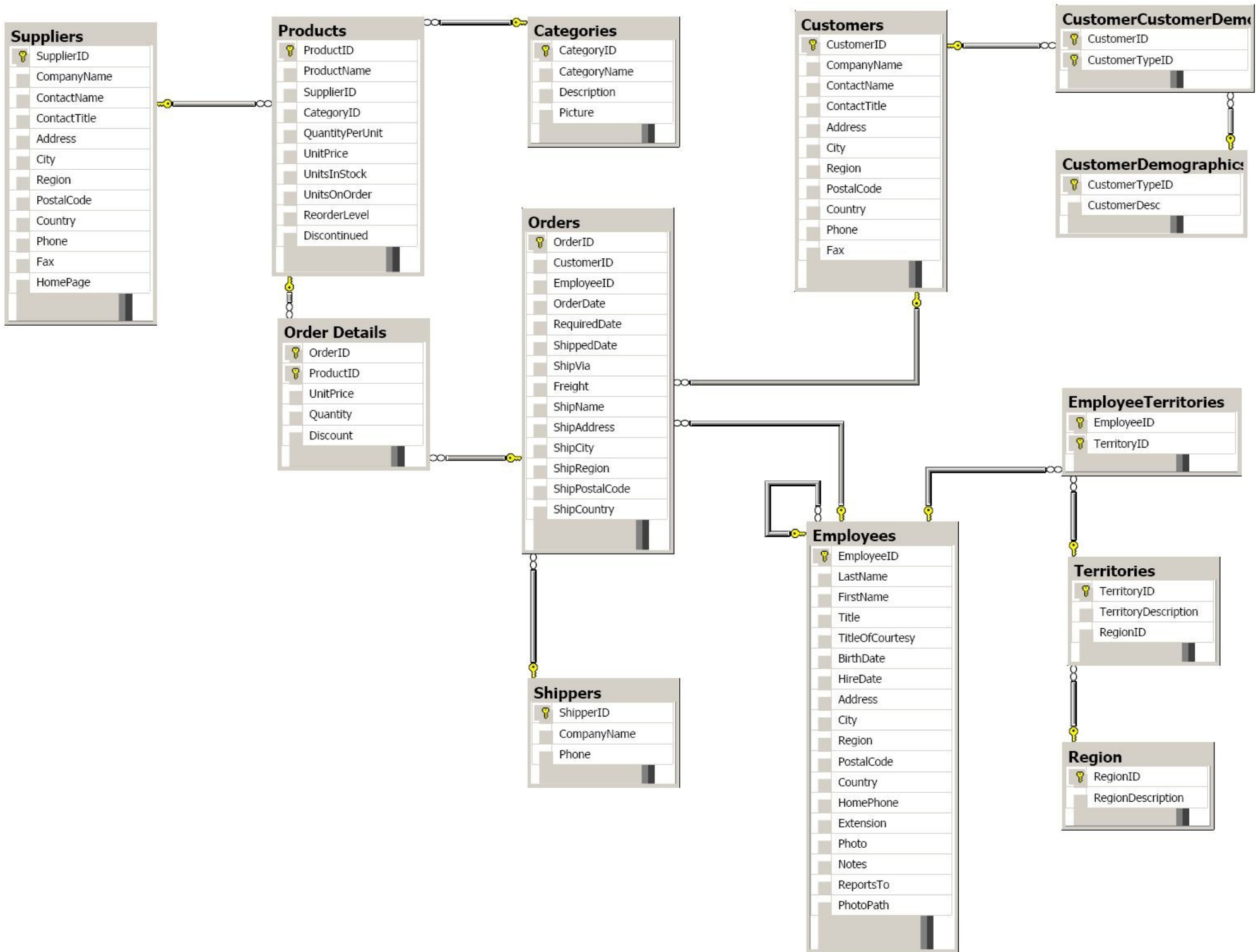
Płaskie bazy danych

ID	Klient	Adres	Telefon	Data sprzedaży
1	Darek	Ul Łódzka 20	078374883	12/01/2008
2	Darek	Ul Łódzka 20	078374883	14/03/2008
3	Matylda	Poznańska 10	566265622	01/04/2008
4	Wiesław	Mickiewicza 12	676727333	05/04/2008
5	Marta	Słowackiego 20	561526552	23/06/2008
6	Marta	Słowackiego 20	561526552	23/06/2008
7	Darek	Ul Łódzka 20	078374883	10/08/2008
8	Michał	Habrowa 5	255615652	11/09/2008
10	Michał	Habrowa 5	255615652	12/10/2008

Relacyjne bazy danych

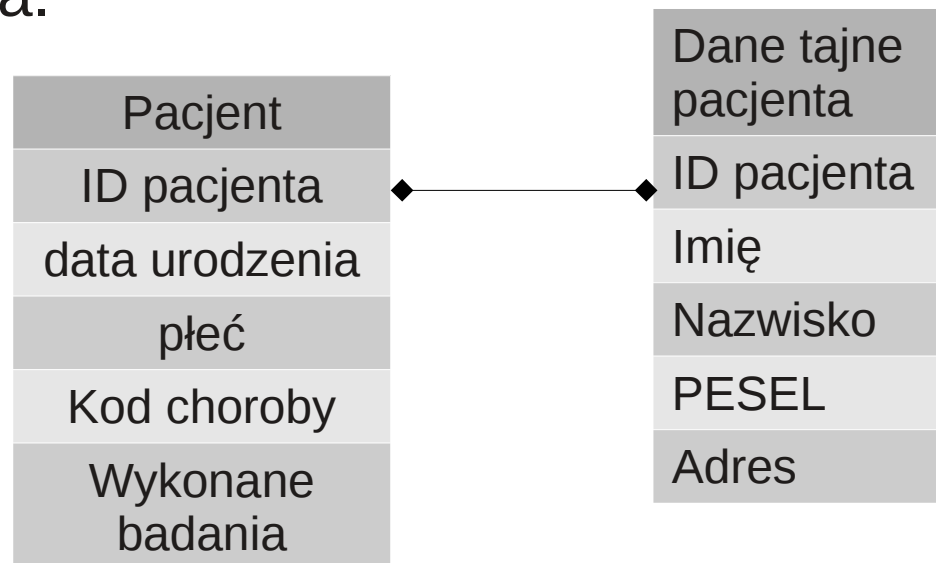


Example of a Typical Relational Data Model

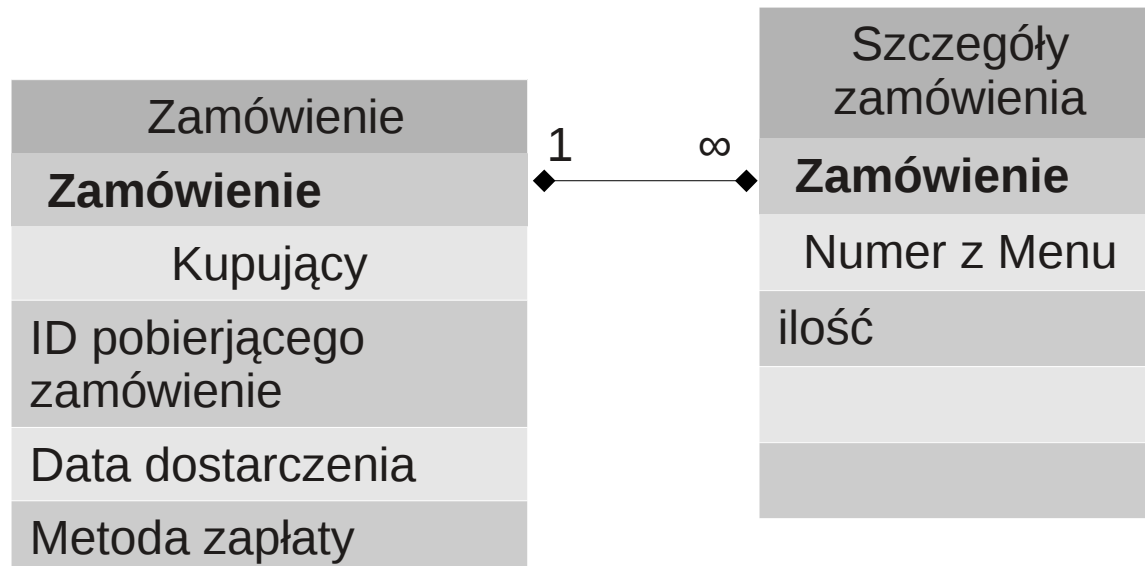


Relacje w relacyjnych bazach danych

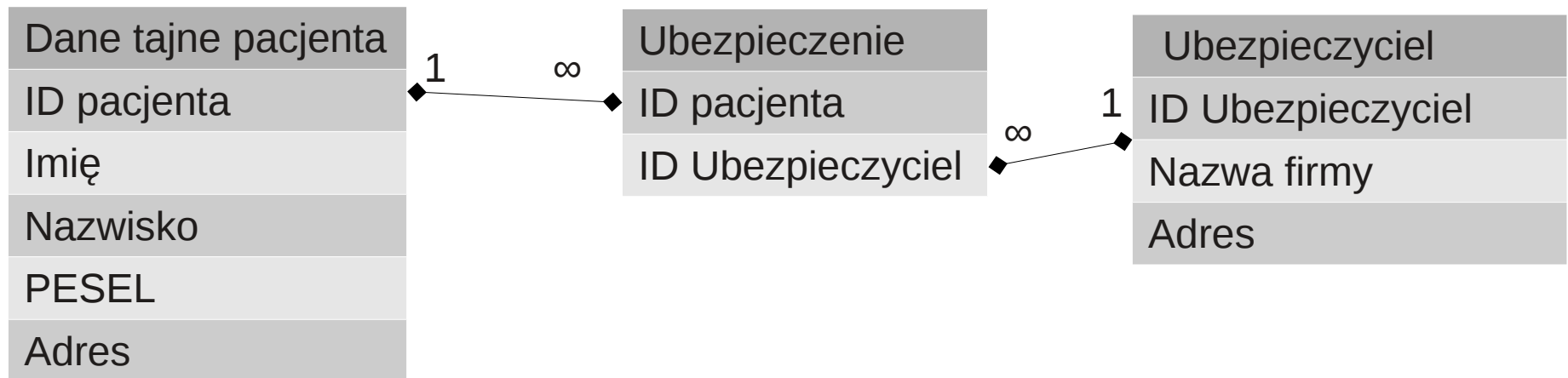
Dwie tablice są w relacji 1-1 gdy dla każdego wiersza w jednej tablicy występuje co najwyżej jeden wiersz drugiej tablicy. W świecie rzeczywistym takiej relacje występują jednak rzadko. Stosowane są natomiast w sytuacjach rozdzielenia tablicy na dwie np. z powodów bezpieczeństwa.



Dwie tablice są w relacji 1-n jeżeli dla każdego wiersza w jednej tablicy istnieje zero, jeden lub wiele wierszy w drugiej tablicy, ale dokładnie jeden wiersz w pierwszej tablicy.



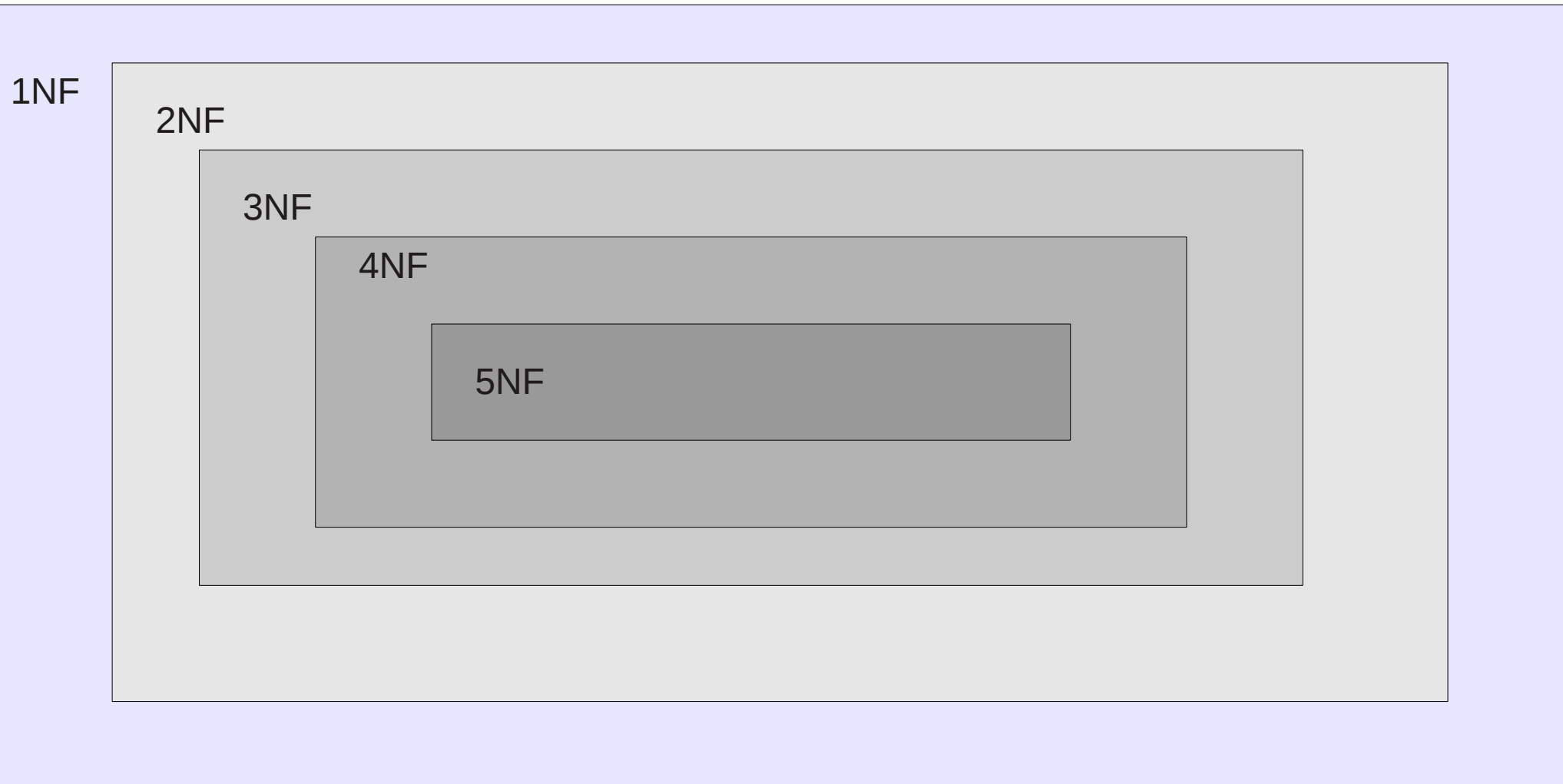
Dwie tablice są w relacji n-n gdy dla każdego wiersza w pierwszej tablicy występuje możliwość istnienia wielu wierszy w drugiej tablicy. Relacje tego typu nie mogą być realizowane w programach obsługujących relacyjne bazy danych. Dlatego relację tą tworzy się poprzez utworzeniu kilku relacji 1-n.



Normalizacja relacyjnych baz danych

Normalizacja bazy danych jest to proces efektywnej organizacji danych w bazie danych, Ma na celu eliminację redundantnych danych oraz upewnienie się że przechowywane relacje mają sens.

Występuje 5 typów normalizacji bazy danych.



- 1NF (normal form)
 - Eliminacja identycznych kolumn
 - Utworzenie oddzielnych kolumn dla każdej grupy powiązanych ze sobą danych i określenie kluczy podstawowych dla każdej tabeli (wartości atrybutów mają być atomowe :-))
- 2NF
 - Wypełnij wszystkie wymagania dla poprzednich typów normalizacji
 - Usuń podzbiór danych występujących w wielu wierszach i przenieś je to innej tabeli (rekordy nie powinny zależeć od niczego innego tylko od klucza podstawowego tabeli)
 - Utwórz relacje pomiędzy tymi nowo utworzonymi tabelami.
- 3NF
 - Wypełnij wszystkie wymagania dla poprzednich typów normalizacji.
 - Kolumny są wzajemnie niezależne (Wartości rekordu, które nie są częścią jego klucza, nie należą do tabeli. Zazwyczaj, jeśli zawartość grupy pól odnosi się do więcej niż jednego rekordu tabeli, należy rozważyć umieszczenie tych pól w oddzielnej tabeli.).
- 4NF
 - Wypełnij wszystkie wymagania dla poprzednich typów normalizacji.
 - Muszą zostać zlikwidowane wszystkie wielowartościowe zależności.

1NF

Te tabele nie spełniają wymogów 1NF

Zamówienie ID	Kupujący ID	Zakup
1	4	5 młotków, 3 śrubokręty, 1 prostownik
2	8	1 zestaw kluczy
3	1	1 młotek
4	1	3 klucze uniwersalne
5	4	0.5 kg gwoździ

Zamówienie ID	Kupujący ID	Zakup1	Ilość 1	Zakup2	Ilość 2	Zakup 3	Ilość 3
1	4	młotek	5	śrubokręt	3	prostownik	1
2	8	Zestaw kluczy	1				
3	1	młotek	1				
4	1	Klucz uniwersalny	3				
5	4	gwoździe	0.5				

Zamówienie ID	Kupujący ID	Zakup ID	Ilość 1	Produkt
1	4	1	5	młotek
1	4	2	1	śrubokręt
1	4	3	1	prostownik
2	8	1	3	Zestaw kluczy
3	1	1	1	młotek
4	1	1	3	Klucz uniwersalny
5	4	1	0.5	gwoździe

2NF

Zamówienie ID	Kupujący ID	Data zakupu	Zakup ID	Produkt ID	Ilość 1	Produkt
1	4	5/1/94	1	1	5	młotek
1	4	5/1/94	2	2	1	śrubokręt
2	8	2/3/95	1	3	3	Zestaw kluczy
3	1	12/6/95	1	1	1	młotek
3	1	12/6/95	2	4	1	wiertarka
4	1	28/7/98	1	5	3	Klucz uniwersalny
5	4	28/7/98	1	6	0.5	gwoździe

Zamówienie ID	Kupujący ID	Data zakupu
1	4	5/1/94
2	8	2/3/95
3	1	12/6/95
4	1	28/7/98
5	4	28/7/98

Zamówienie ID	Zakup ID	Produkt ID	Ilość 1	Produkt
1	1	1	5	młotek
1	2	2	1	śrubokręt
2	1	3	3	Zestaw kluczy
3	1	1	1	młotek
3	2	4	1	wiertarka
4	1	5	3	Klucz uniwersalny
5	1	6	0.5	gwoździe

3NF

Zamówienie ID	Zakup ID	Produkt ID	Ilość 1
1	1	1	5
1	2	2	1
2	1	3	3
3	1	1	1
3	2	4	1
4	1	5	3
5	1	6	0.5

Produkt ID	Produkt
1	młotek
2	śrubokręt
3	Zestaw kluczy
4	wiertarka
5	Klucz uniwersalny
6	gwoździe

Formularze

Formularze stosuje się głównie do: przeglądania, wprowadzania, usuwania oraz modyfikowania danych, otwierania innych formularzy i raportów, tworzenia okienek dialogowych, pól wyliczalnych i wielu innych operacji. Większość informacji zawartych w formularzu pochodzi z odpowiedniego źródła danych, którym jest rekord z konkretnej tabeli.

Kwerendy

Kwerenda bazy danych (ang. query) - zapytanie do bazy danych, umożliwiające znalezienie i wyświetlenie informacji żądanych przez użytkownika bazy.

W programach do obsługi baz danych stosuje się rozmaite typy kwerend. Przykładowo, w Microsoft Access typy te obejmują m.in.:

Kwerenda wybierająca dane - prosta kwerenda wybierająca tabele i pola

kwerenda krzyżowa – tworzy arkusz kalkulacyjny oparty na danych z trzech lub więcej pól.

Kwerenda tworząca tabele – wybiera rekordy i zapisuje ich kopie z nowej tabeli.

Kwerenda aktualizująca – wyszukuje informacje i zmienia zawartość wskazanych pól.

Kwerenda usuwająca – wyszukuje informacje i usuwa je z bazy.

Kwerenda dołączająca – pobiera rekordy ze wskazanych tabel i dołącza je do innych tabel.

W MS Access istnieją dwa języki definiowania kwerend, język QBE (ang. Query By Example) oraz język SQL (ang. Structured Query Language).

SQL

- SQL - Structured Query Language -strukturalny język zapytań
 - Światowy standard przeznaczony do definiowania, operowania i sterowania danymi w relacyjnych bazach danych
 - Powstał w firmie IBM pod koniec lat 70-tych
 - Występuje w produktach większości firm produkujących oprogramowanie do zarządzania bazami danych
 - Polecenia SQL mają postać podobną do zdań w języku angielskim
 - Komentarze zaczynają się od znaków --
 - Pomimo prób standaryzacji istnieje szereg różnych dialektów SQL
-
- Język SQL nie rozróżnia wielkości liter.

SQL w pigułce

```
CREATE TABLE osoba(imie TEXT, nazwisko TEXT);  
ALTER TABLE osoba ADD wiek INT NOT NULL DEFAULT 0;  
INSERT INTO osoba VALUES('Jan', 'Kowlaski', 26);  
SELECT * FROM osoba;  
SELECT * FROM osoba ORDER BY wiek;  
SELECT * FROM osoba ORDER BY wiek LIMIT 1;  
SELECT nazwisko, COUNT(*) FROM osoba GROUP BY nazwisko;  
SELECT nazwisko,AVG(wiek) AS t1 FROM osoba GROUP BY nazwisko  
ORDER BY t1;
```

SELECT instrukcja (klauzula) pozwalająca wydobyć z bazy danych odpowiednie informacje, rezultat zapisywany jest w tablicy o nazwie result-set

Składnia:

SELECT nazwa_pola (lub pól w postaci listy przecinkowej) FROM nazwa tablicy

SELECT * FROM nazwa tablicy

WERE - instrukcja pozwalająca na wyselekcjonowanie tych rekordów, które spełniają wyszczególniony warunek

SELECT nazwa_pola (lub pól) FROM nazwa tablicy WHERE nazwa_pola wartość operatora

Operator	Opis
=	równy
<>	nierówny
>	większy
<	mniejszy
>=	Większy lub równy
<=	Mniejszy lub równy
BETWEEN	Wartość z przedziału
LIKE	Wartość zgodna z wzrociem
IN	Wartość w zbiorze danych

Operatory logiczne

AND OR NOT

AND

1	1	1
1	0	0
0	1	0
0	0	0

OR

1	1	1
1	0	1
0	1	1
0	0	0

Negowanie złożonych wyrażeń I prawa de Morgana

NOT (A AND B)
NOT (A OR B)

NOT A OR NOT B
NOT A AND NOT B

DISTINCT – wybierane są tylko te rekordy które mają różne wartości

```
SELECT DISTINCT nazwa_pola FROM nazwa_tablicy
```

ORDER BY – pozwala posortować elementy w wybranie kolumnie (lub kolumnach przedstawionych w postaci listy przecinkowej)

Składnia:

```
SELECT nazwa_pola (lub pól) FROM nazwa tablicy ORDER BY nazwa_pola ASC|DESC
```

nazwa_pola po ORDER BY nie musi występować wśród nazw pól po SELECT!

GROUP BY – grupuje elementy mające tą samą wartość w zadanej kolumnie, używane jest w połączeniu z funkcją agregującą

Składnia:

```
SELECT nazwa_pola (lub pól) FROM nazwa tablicy WHERE nazwa_pola wartość operatora  
GROUP BY nazwa_pola
```

Wbudowane funkcje języka SQL

Funkcje agregujące w języku SQL:

AVG() - wyznacza wartość średnią kolumny

COUNT() - Zlicza liczbę elementów nie pustych w kolumnie

FIRST() - zwraca pierwszą wartość

LAST() - Zwraca ostatnią wartość

MAX() - Zwraca największą wartość w kolumnie

MIN() - zwraca najmniejszą wartość w kolumnie

SUM() - Sumowanie wartości zadanej kolumny

Funkcje SQL:

UCASE() - Konwertuje zadane pole na duże litery

LCASE() - Konwertuje zadane pole na małe litery

MID() - Wydobywa znaki z pola tekstowego

LEN() - Zwraca długość pola tekstowego

ROUND() - Zaokrągla wartość pola zmiennoprzecinkowego zgodnie z zadany formatem

FORMAT() - Definiuje w jaki sposób dane pole powinno być wyświetlane

ACTIVE, ADD, ALL, AFTER, ALTER, AND, ANY, AS, ASC, ASCENDING, AT, AUTO, AUTOINC, AVG

2. BASE_NAME, BEFORE, BEGIN, BETWEEN, BLOB, BOOLEAN, BOTH, BY, BYTES

3. CACHE, CAST, CHAR, CHARACTER, CHECK, CHECK_POINT_LENGTH, COLLATE, COLUMN, COMMIT, COMMITTED, COMPUTED, CONDITIONAL, CONSTRAINT, CONTAINING, COUNT, CREATE, CSTRING, CURRENT, CURSOR

4. DATABASE, DATE, DAY, DEBUG, DEC, DECIMAL, DECLARE, DEFAULT, DELETE, DESC, DESCENDING, DISTINCT, DO, DOMAIN, DOUBLE, DROP

5. ELSE, END, ENTRY_POINT, ESCAPE, EXCEPTION, EXECUTE, EXISTS, EXIT, EXTERNAL, EXTRACT

6. FILE, FILTER, FLOAT, FOR, FOREIGN, FROM, FULL, FUNCTION

7. GDSCODE, GENERATOR, GEN_ID, GRANT, GROUP, GROUP_COMMIT_WAIT_TIME

8. HAVING, HOUR

IF, IN, INT, INACTIVE, INDEX, INNER, INPUT_TYPE, INSERT, INTEGER, INTO, IS, ISOLATION

10. JOIN

11. KEY

12. LONG, LENGTH, LOGFILE, LOWER, LEADING, LEFT, LEVEL, LIKE, LOG_BUFFER_SIZE

13. MANUAL, MAX, MAXIMUM_SEGMENT, MERGE, MESSAGE, MIN, MINUTE, MODULE_NAME, MONEY, MONTH

14. NAMES, NATIONAL, NATURAL, NCHAR, NO, NOT, NULL, NUM_LOG_BUFFERS, NUMERIC

15. OF, ON, ONLY, OPTION, OR, ORDER, OUTER, OUTPUT_TYPE, OVERFLOW

16. PAGE_SIZE, PAGE, PAGES, PARAMETER, PASSWORD, PLAN, POSITION, POST_EVENT, PRECISION, PROCEDURE, PROTECTED, PRIMARY,

17. PRIVILEGES

18. RAW_PARTITIONS, RDB\$DB_KEY, READ, REAL, RECORD_VERSION, REFERENCES, RESERV, RESERVING, RETAIN, RETURNING_VALUES, RETURNS, REVOKE, RIGHT, ROLLBACK

19. SECOND, SEGMENT, SELECT, SET, SHARED, SHADOW, SCHEMA, SINGULAR, SIZE, SMALLINT, SNAPSHOT, SOME, SORT, SQLCODE, STABILITY, STARTING, STARTS, STATISTICS, SUB_TYPE, SUBSTRING, SUM, SUSPEND

20. TABLE, THEN, TIME, TIMESTAMP, TIMEZONE_HOUR, TIMEZONE_MINUTE, TO, TRAILING, TRANSACTION, TRIGGER, TRIM

21. UNCOMMITTED, UNION, UNIQUE, UPDATE, UPPER, USER

22. VALUE, VALUES, VARCHAR, VARIABLE, VARYING, VIEW

23. WAIT, WHEN, WHERE, WHILE, WITH, WORK, WRITE

24. YEAR

Obsługa wielu tabel w SQL

- Iloczyn kartezjański

```
SELECT * FROM Klienci, Zamowienia
```

- Unikanie ilocznu kartezjańskiego

- ```
SELECT * FROM Klienci, Zamowienia WHERE
Klienci.ID=Zamowienia.IDKlienci
```

- Aliasy tabel

- ```
SELECT * FROM Klienci k, Zamowienia z WHERE  
k.ID=z.IDKlienci
```

Historia Obiektowych baz danych

1987: Pierwszy system z obiektową bazą danych: SmallTalk

GemStone, C++ Ontos Vbase

1991: ObjectStore, Objectivity/DB, O2

2004: odb4o, DTS/S1, Perst

Języki programowania używane w obiektowych bazach danych :

SmallTalk, Lisp, COP, C++, Java, C#

Cechy obiektowych baz danych

Zalety:

- Dostęp do danych jest szybszy od dostępu w relacyjnych bazach danych bo nie ma potrzeby łączenia tabel
- Dane multimedialne mogą być używane, gdyż to metody klasy związane z tymi danymi zajmują się ich obsługą i interpretacją
- OODBMS może być zmieniana (oprogramowana) bez istotnych zmian w systemie jako całości

“Wady”:

- Technika wskaźnikowa wyszukiwania danych, może przy ogólnych zapytaniach być powolna
- Hermetyzacja, powoduje ukrywanie pewnych obszarów danych, przez co nie są one dostępne dla zapytań

Definicja obiektu

Obiekt – zdefiniowany przez użytkownika złożony typ danych

- Posiada strukturę lub stan (zmienne)
- Metody, opisujące zachowanie obiektu

Obiekt charakteryzowany jest przez 4 cechy:

- Identyfikator - unikalny numer identyfikacyjny
- Nazwa – Obiekt może posiadać unikalną nazwę w systemie bazodanowym
- Czas życia – określa czy obiekt jest trwały, czy chwilowy
- Strukturę

Koncepcje związane z obiektami

Abstrakcyjne typy danych:

- Definicje klas, które reprezentują pojęcia abstrakcyjne nie definiują żadnych znanych obiektów, mają sens jako klasa podstawowa.

Hermetyzacja danych i operatorów:

- Implementacja operatorów i struktury, która jest niewidzialna na zewnątrz klasy

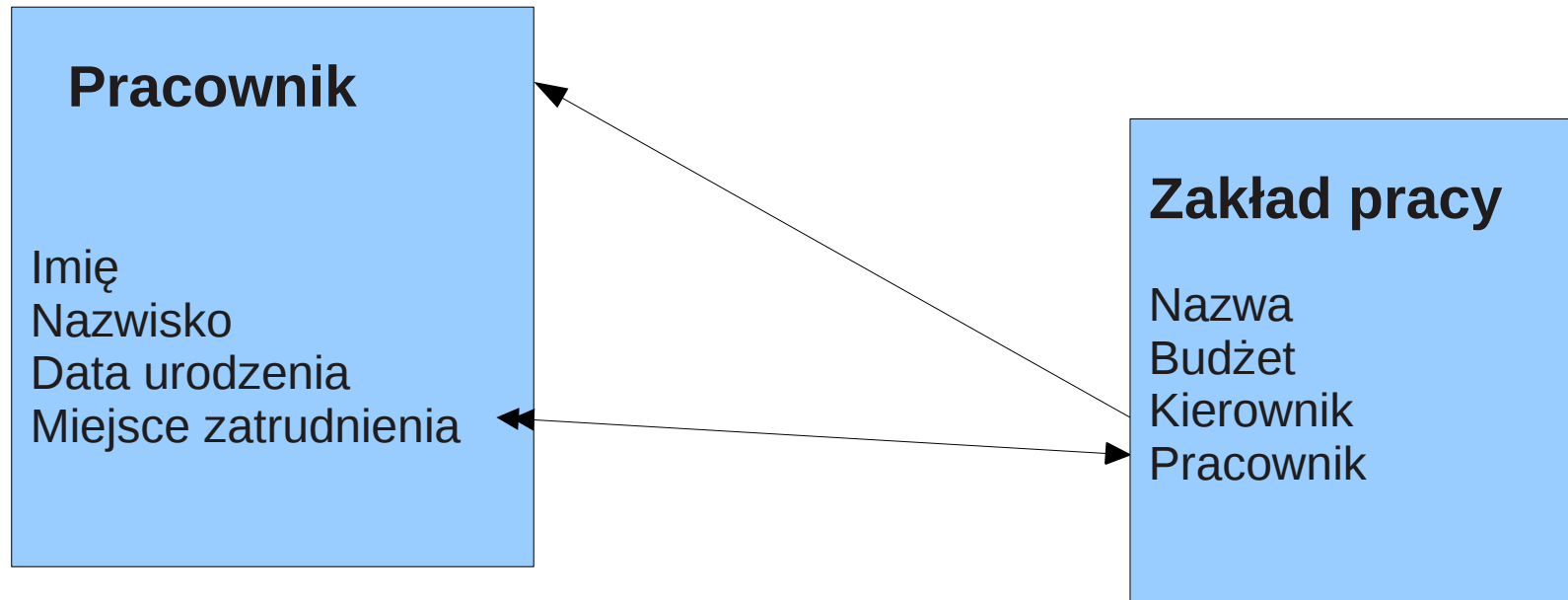
Dziedziczenie:

- Współdzielenie danych z innymi klasami.

Polimorfizm:

- Mechanizm pozwalający używać zmiennych i metod na wiele różnych sposobów

Obiektowe bazy danych



```
Class Pracownik
{
    String Imie;
    String Nazwisko;
    Date Data_urodzenia;
    .....
```

Metody

```
Oblicz_srednia_place()
};
```

```
Class Zaklad_pracy
{
    String Nazwa;
    float Budzet;
    Pracownik Kierownik;
    Set <Pracownik> zatrudnieni;
    .....
```

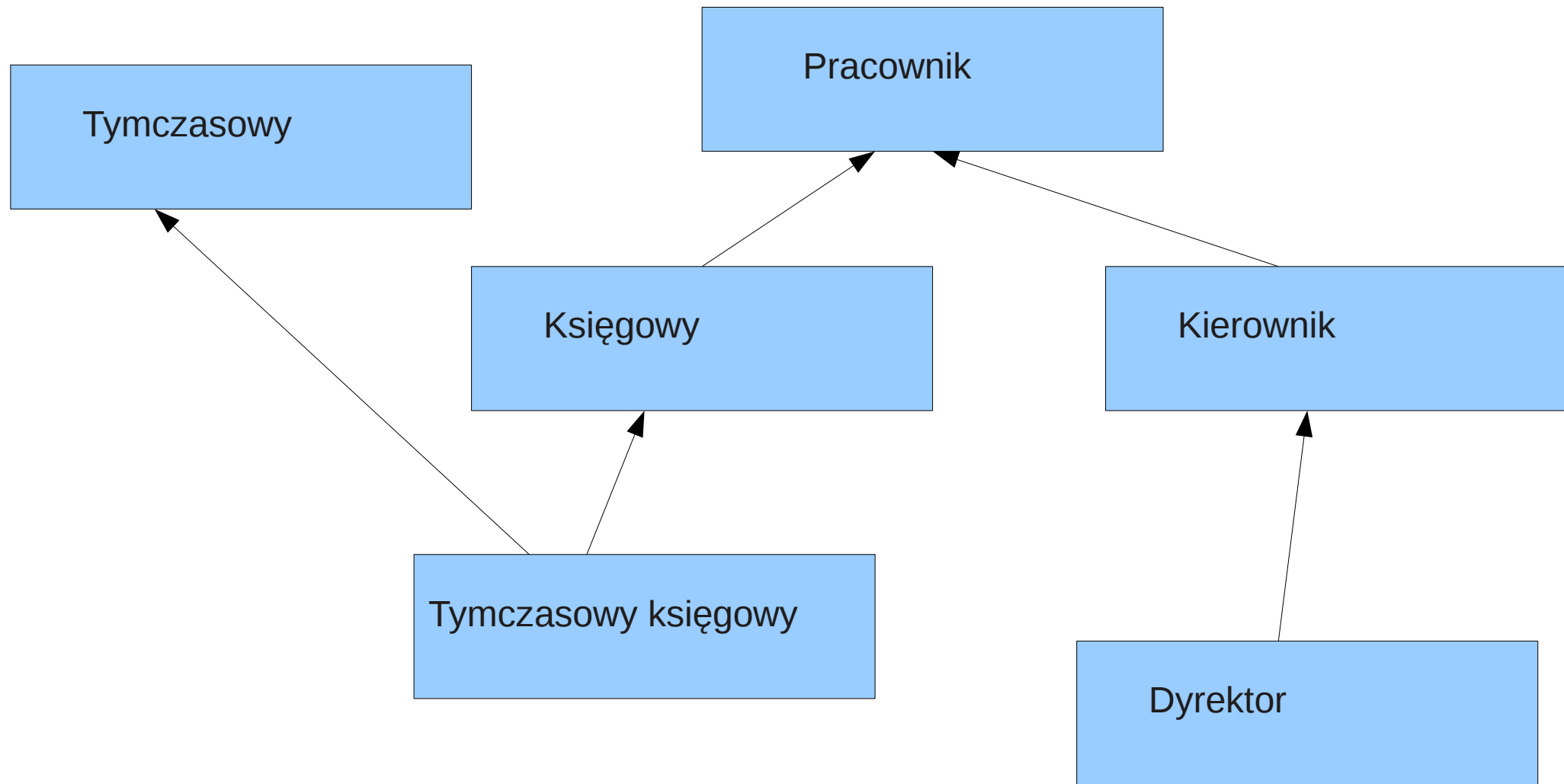
Metody

```
Oblicz_srednia_place()
};
```

Dziedziczenie



Każdy kierownik jest pracownikiem, ma jako obiekt prawie identyczne cechy jak pracownik!!



Class Pracownik

```
{  
    Nazwisko;  
    Adres;
```

```
.....  
}
```

Class Kierownik:Pracownik

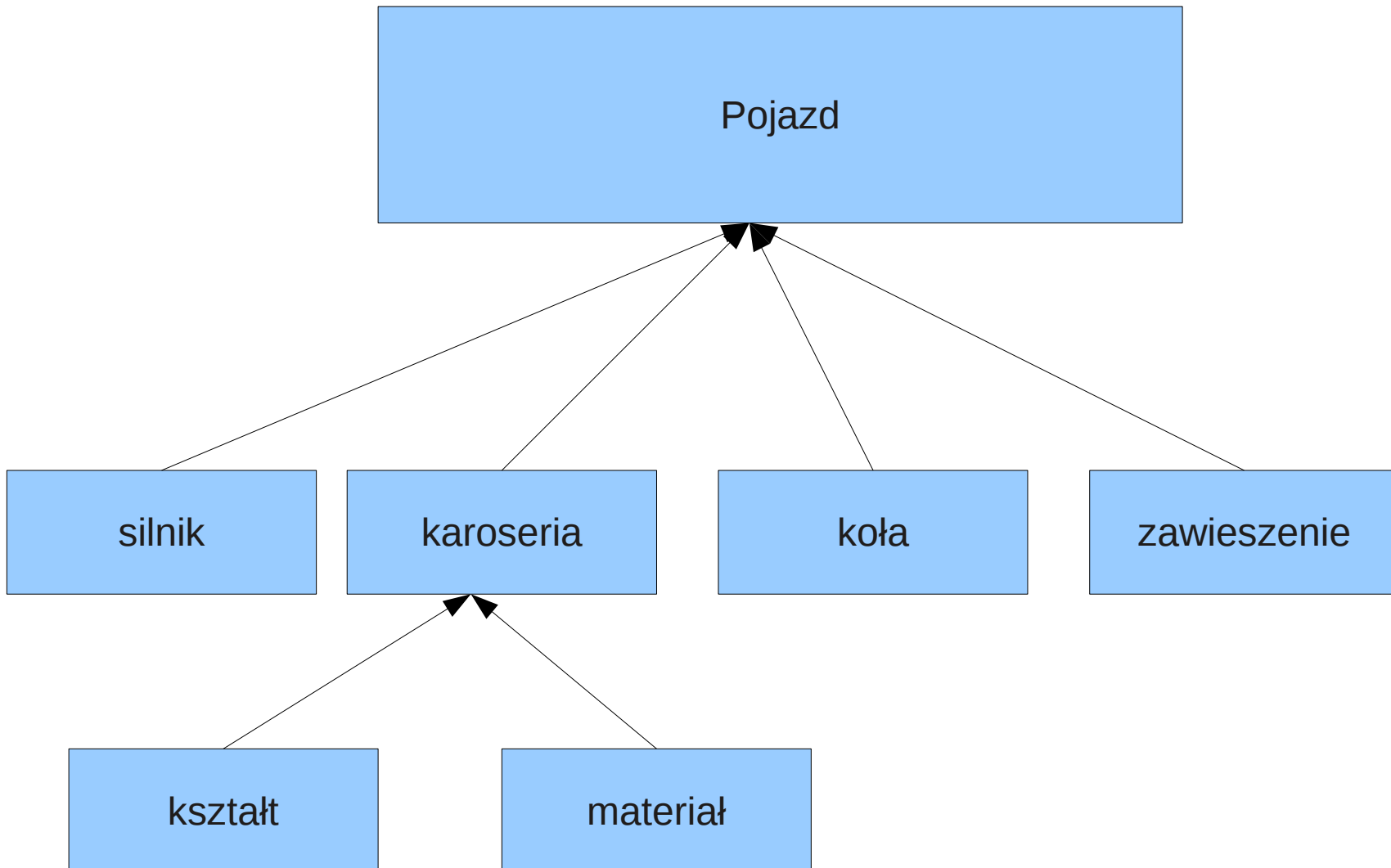
```
{  
}  
Class Księgowy:Pracownik
```

```
{  
}  
Class Dyrektor:Kierownik
```

```
{  
}  
Class Księgowy_tym:Księgowy
```

```
{  
}
```

Wielokrotne dziedziczenie



Hermetyzacja

Class

{

Private:

Public:

}

Przykład relacyjnej bazy danych i obiektowej

Pracownik

ID	Nazwisko	Dat. urodzenia	Zakład pracy

Projekt

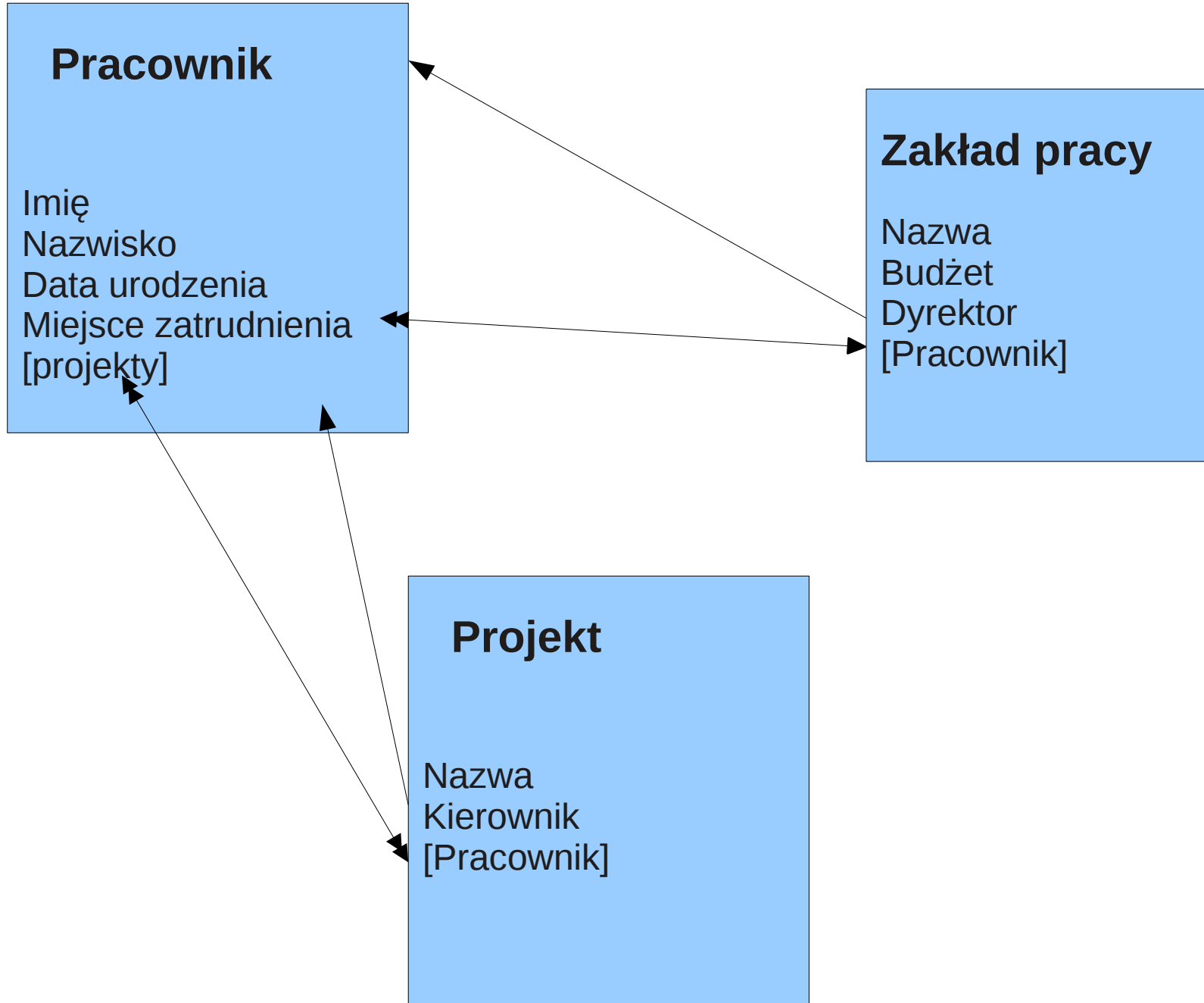
ID	Nazwa	Kierownik

Projekt pracownika

ID_pracownika	ID_Projektu

Zakład pracy

ID	Nazwa	Dyrektor



Internet i jego historia

1969

Powstaje ARPAnet, sieć czterech komputerów stworzona przez amerykańską agencję rządową ARPA. W 1971 sieć ta liczyła sobie 13 węzłów, a w 1973 roku - już 35.

1971

Początki poczty elektronicznej. Ray Tomlinson wysyła pierwszą wiadomość elektroniczną.

1973

Do ARPANETu włączone zostają pierwsze instytucje spoza Stanów Zjednoczonych: University College of London w Wielkiej Brytanii i Royal Radar Establishment w Norwegii.

1979

Powstaje Usenet, tekstowe grupy dyskusyjne - stworzony przez studentów Toma Truscotta, Jima Ellisa i Steve Bellovina. Dziś Usenet to ponad 50 tysięcy grup i miliony użytkowników, czytających i biorących udział w dyskusjach.

1983

Od ARPANET odłączona zostaje jej część wojskowa, tworząc MILNET. W ARPANET hosty i sieci zaczynają używać protokołu TCP/IP. Powstaje właściwy Internet.

1989

Tim Berners-Lee oraz Robert Cailliau złożyli do CERN-u projekt stworzenia sieci dokumentów hipertekstowych, o nazwie World Wide Web. Miał to być zbiór dokumentów hipertekstowych, który miał ułatwić pracę w CERN-ie.

1990

Tim Berners-Lee stworzył podstawy HTML i pierwszą stronę internetową. Projekt World Wide Web powstaje na komputerze NeXT, w pierwszej odsłonie umożliwia jednocześnie przeglądanie i edycję hipertekstowych dokumentów. W rok później zostaje zainstalowany na serwerach CERN, a z nich rozpowszechnia się na cały świat.

1991

Zniesiono zakaz używania Internetu do celów komercyjnych

1993

Pojawia się Mosaic, pierwsza graficzna przeglądarka World Wide Web. Tworzy ją zespół: Marc Andreessen, Eric Bina i inni studenci NCSA. Dzięki niej znacznie wzrasta popularność Internetu i World Wide Web.

1994

David Filo i Jerry Yang tworzą Yahoo! Jako spis interesujących ich miejsc w Internecie; z czasem serwis ten rozwija się w najślynniejszy katalog zasobów internetowych na świecie, a jego twórcy zostają milionerami.

1995

Marc Andreessen tworzy Netscape Navigator, w swoim czasie najpopularniejszą przeglądarkę internetową, zdobywającą w swoim czasie do 80 procent rynku.

1998

Powstaje spółka Google Technology Inc. (obecnie Google Inc.)

WWW jako baza danych

GOOGLE:

- * Ponad 4 miliardy zindeksowanych stron. Każda z nich ma średnio 10 kB
- * Do 2000 komputerów w jednym klastrze
- * Ponad 30 klastrów
- * Obsługa 104 języków
- * W jednym klastrze składowanych jest ponad 1 Petabajt danych czyli milion gigabajtów
- * Pojedynczy klaster ma wydajność zapisu/odczytu na poziomie 2 Gbit/s
- * Komputerów w klastrze jest tak dużo że dziennie psują się średnio 2 serwery
- * Brak poważniejszej awarii od lutego 2000
- * Google obsługuje średnio 1000 zapytań w ciągu sekundy.

Jak działa szukarka Google

Można wyróżnić 3 poziomy:

- Googlebot – roboty sieciowe wyszukujące stron
 - Poprzez formę URL www.google.com/addurl.html
 - Poprzez “pełzacz”
- Indeksowanie
 - Indeksowany jest cały tekst na stronie, ignorowane są słowa “stopu”
- Procesor zapytań
 - Do oceny istotności zapytania stosowany jest tzw. PageRank, strona o większej wartości PageRank pojawia się wyżej na liście. Do określenia PageRank używane jest ponad 100 parametrów między innymi: popularność strony, położenie poszukiwanego ciągu znaków na stronie, bliskość położenia poszukiwanych łańcuchów na stronie

BIGTABLE

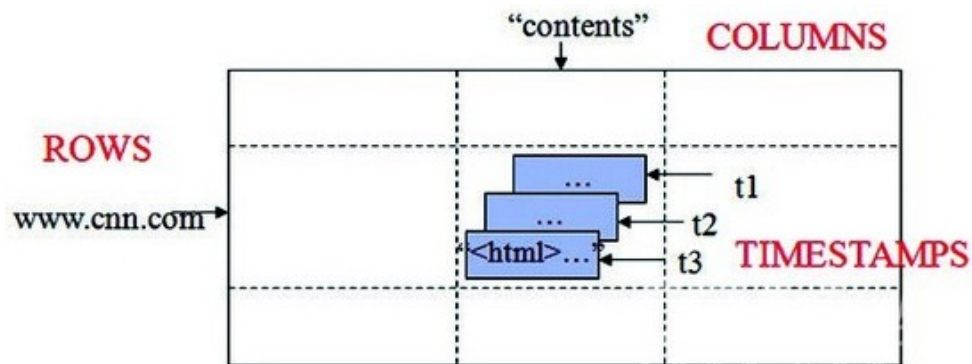
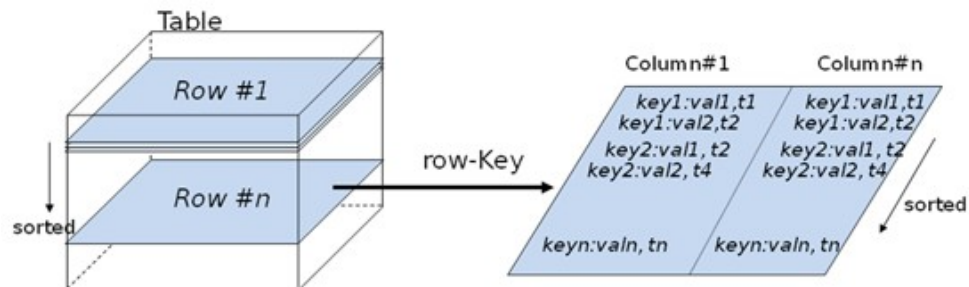


图2 Google BigTable的数据模型



GOOGLE: Problem z wyszukiwaniem słów niejednoznacznych!

Michał oglądał telewizję

Paweł patrzył na lecące ptaki.

Monika przyglądała się szybko biegnącym
telewizorom.

Dla poprawnego zinterpretowania tych zdań potrzebna jest rozległa wiedza, która obecnie jest trudno dostępna dla komputerów

Konieczna jest reprezentacja podstawowych pojęć i własności: czas, przestrzeń, materia, wydarzenie, kolekcja, rodzaj substancji itp.

Sieci semantyczne

Sieć semantyczna to reprezentacja wiedzy za pomocą węzłów połączonych przy użyciu łuków (krawędzi). Komputerowe implementacje sieci semantycznych miały zastosowanie w sztucznej inteligencji i translacji maszynowej.

$$S = \langle P, T, R \rangle$$

Gdzie

P zbiór pojęć (wierzchołki grafu, węzły)

T zbiór typów relacji (jest elementem, należy do, posiada itp)

R zbiór wszystkich relacji występujących w danej sieci semantycznej

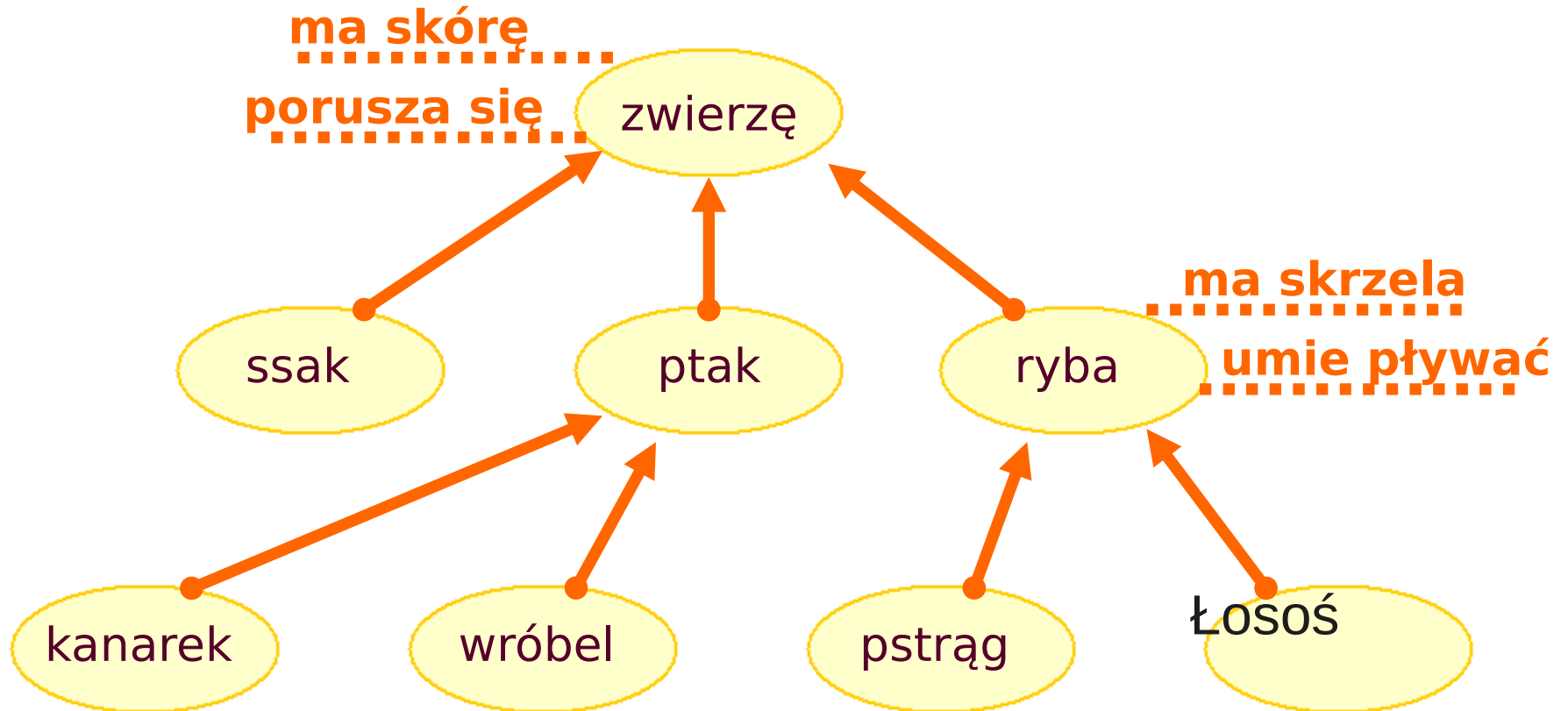
Sieci semantyczne są od dawna używane w filozofii, psychologii, lingwistyce.

Sieci semantyczne mogą być używane zarówno to reprezentacji wiedzy jak i do wspomaganie automatycznych systemów wnioskowania

6 typów sieci semantycznych:

- definiujące
- twierdzeń
- implikacyjne
- wykonawcze
- uczące się
- hybrydowe

Sieci semantyczne



Collins & Quillian, 1969

Drzewo Porfiriusza (definiująca)

Najwyższy rodzaj

fundamentum

Rodzaj I gatunek pośredni

fundamentum

Rodzaj I gatunek pośredni

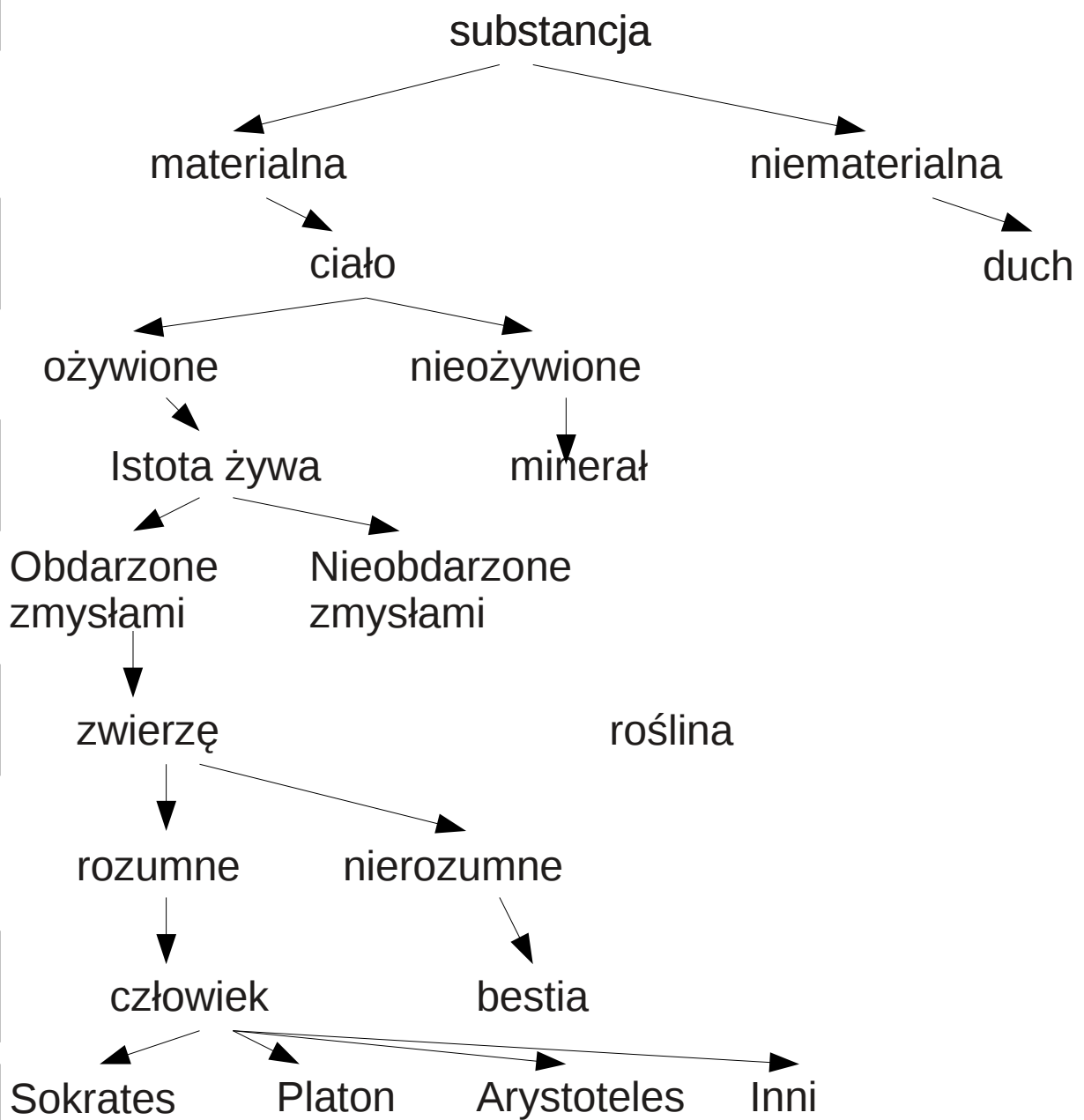
fundamentum

gatunek pośredni
Rodzaj najniższy

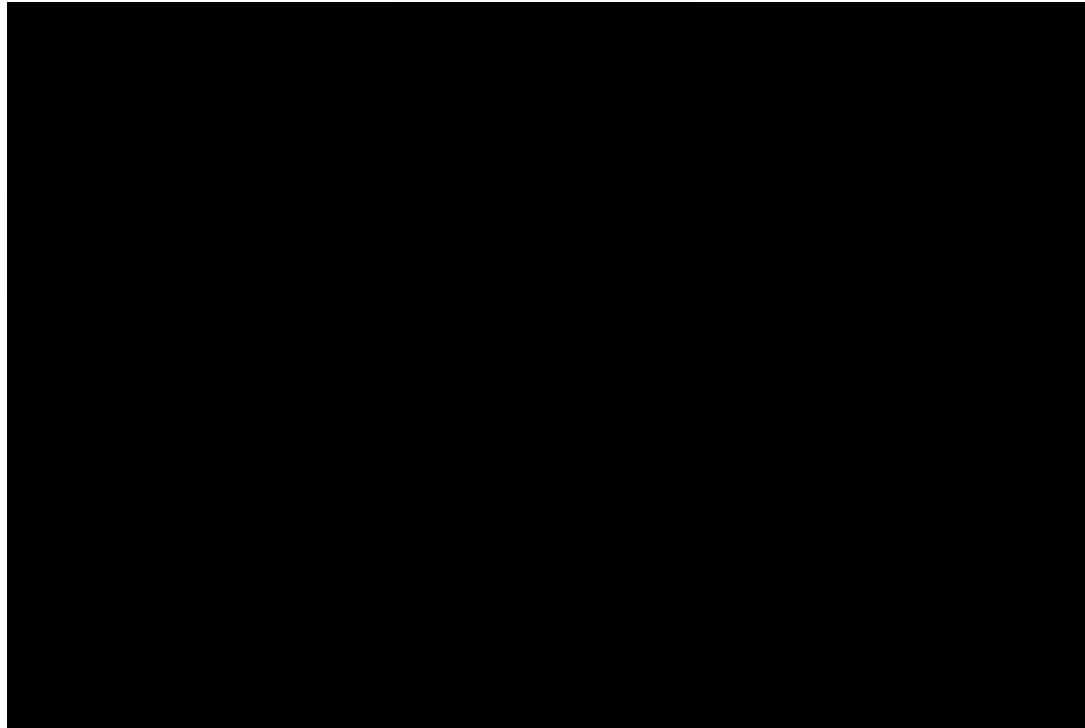
fundamentum

Gatunek najniższy

Jednostki (indywidua)



Sieci Implikacyjne



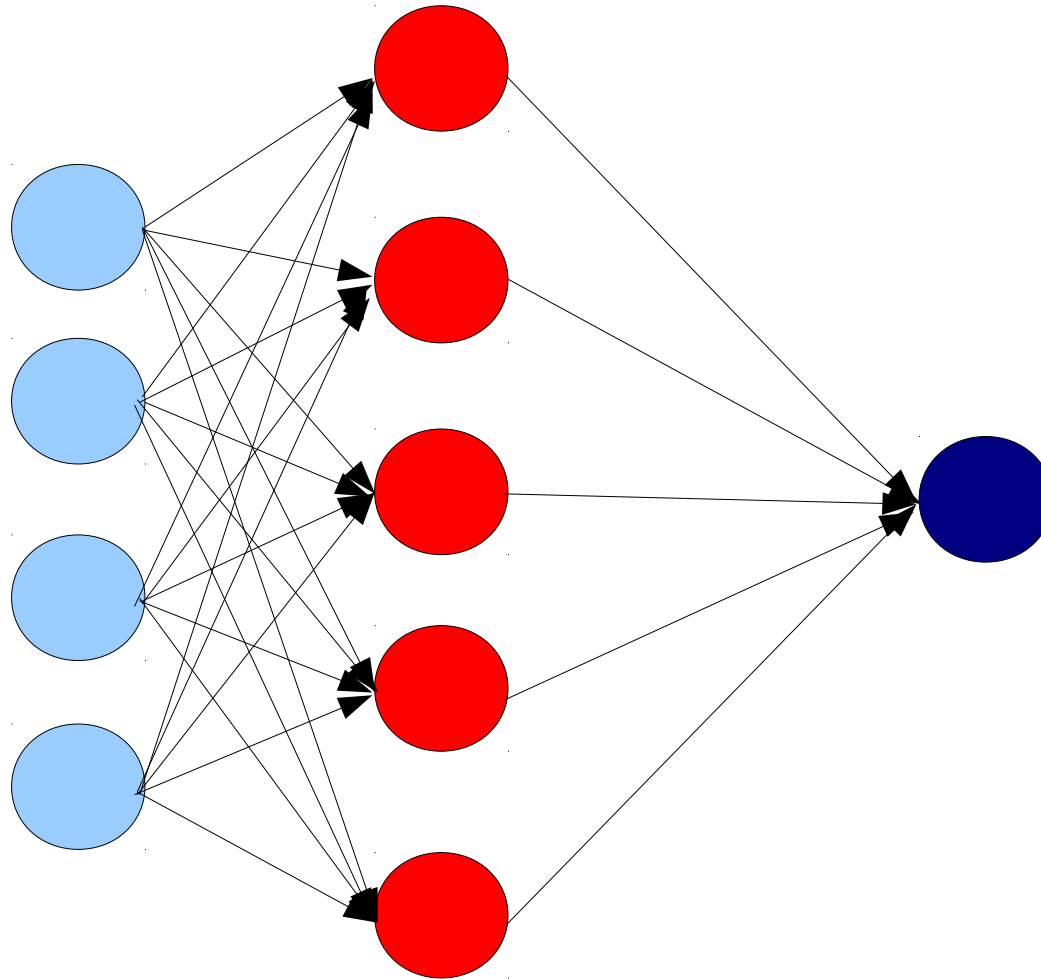
Sieci wykonywalne

W tego typu sieciach występują 3 mechanizmy:

- Komunikaty sieć może przesyłać dane z jednego węzła do drugiego, mogą to być pojedyncze bity tzw. markery, tokeny, triggery, albo duża wartość numeryczna
- Procedury dołączane – programy dołączone są do węzłów, które realizują jakąś akcję, wykonują obliczenia na danych w tym węźle, lub węzłach sąsiadujących
- Transformacje grafów -kombinują grafy, modyfikują,



Sieci uczące się



Ontologia

Ontologia lub metafizyka (por. metafizyka klasyczna) – podstawowy obok epistemologii dział filozofii starający się badać strukturę rzeczywistości i zajmujący się problematyką związaną z pojęciami bytu, istoty, istnienia i jego sposobów, przedmiotu i jego własności, przyczynowości, czasu, przestrzeni, konieczności i możliwości.
(źródło Wikipedia.pl)

Informatyka

Ontologia zajmuje się odkrywaniem i opisywaniem „tego co jest”, pewnym fragmentem rzeczywistości, mniej lub bardziej dokładnie określonym. Aby zapewnić jednoznaczność przekazu wiedzy na temat określonej rzeczywistości, wykorzystuje się kategoryzację oraz hierarchizację.

class-def zwierzę	%zwierzęta są klasą
class-def roślina	%rośliny są klasą
subclass-of NOT zwierzę	%wyłączenie ze zwierząt
class-def drzewo	
subclass-of roślina	%drzewa są rodzajami roślin
class-def gałąź	
slot-constraint is-part-of has-value drzewo	%gałęzie są częścią drzew
class-def liść	
slot-constraint is-part-of has-value gałąź	%liście są częścią gałęzi
class-def definicja drapieżników	%drapieżniki są zwierzętami
subclass-of zwierzę	
slot-constraint zjada value-type zwierzę	%które zjada tylko inne zwierzęta
class-def definicja roślinożernych	%roślinożerne są zwierzętami
subclass-of zwierzę	
slot-constraint zjada value-type roślina	
OR (slot-constraint is-part-of has-value roślina)	%które zjada tylko rośliny lub części roślin
class-def żyrafa	%żyrafy są zwierzętami
subclass-of zwierzę	
slot-constraint zjada value-type liść	%oraz jedzą liście
class-def lew	
subclass-of zwierzę	%lwy też są zwierzętami
slot-constraint zjada value-type roślinożerne	%ale zjadają one roślinożerne
class-def rośliny jadalne	%rośliny jadalne to rośliny zjadane
subclass-of roślina	%przez drapieżników i roślinożerców
slot-constraint zjadany has-value roślinożerny,	
drapieżnik	

Semantyczne szukarki

<http://www.hakia.com>

<http://swoogle.umbc.edu>

- `url:foaf` search documents having "foaf" as part of their URLs
- `url:"http://www.w3.org/2000/01/rdf-schema"` search a particular SWD with the URL "<http://www.w3.org/2000/01/rdf-schema>"
- `desc:timbl` search documents having "timbl" in their document annotations
- `def:food` search documents explicitly defining the term(classes/properties) that include a token "food". Note that the term food is case-insensitive.
- `ref:food` search documents implicitly defining the term(classes/properties) that include a token "food". That is, the term being a class/property is deduced by the domain and range definition of RDF/RDFS and OWL predicates. Note that in the results returned by Swoogle search, only the fields 'desc' and 'def' are highlighted, the field 'ref' and 'pop' are neglected for a neat output.
- `pop:person` search documents that populate the class with a token of 'person' with instances OR use the property with a token of 'person' as predicates

XML

I just got a new pet dog.

```
<sentence>  
<person href="http://aaronsw.com/">I</person> just got a new pet  
<animal>dog</animal>.  
</sentence>
```

```
<sentence>  
<person href="http://aaronsw.com">I</person> just got a new pet <animal type="dog"  
href="http://aaronsw.com/myDog">dog</animal>.  
</sentence>
```

Konieczność istnienia przestrzeni nazw!

Semantyczny internet

Sieć semantyczna WWW siatka powiązanych ze sobą informacji. Powiązanych w taki sposób aby informacje te były do przetwarzania przez komputery.

Narzędzia semantycznych sieci WWW:

- RDF (Resource Description Framework)
- OWL (Web Ontology Language)

Celem zastosowania tych języków jest przedstawianie wiedzy w postaci łatwo przetwarzanej przez programy komputerowe, nie zaś ich wyświetlanie użytkownikom.

```
<?xml version="1.0"?>
<RDF>
<Description about="http://www.lis.uw.edu/default.asp">
<autor>Jan Abacki</autor>
<utworzono>1 Maj 1999</utworzono>
<zmodyfikowano>1 Luty 2004</zmodyfikowano>
</Opis>
</RDF>
```

Wiedzę w RDF przedstawia się w postaci trójki: podmiot, orzeczenie, dopełnienie (lub: obiekt, rodzaj powiązania, wartość cechy).

W przedstawionym przykładzie:

- <http://www.lis.uw.edu/default.asp> jest podmiotem (obiektem),
- element <autor> jest orzeczeniem (rodzajem powiązania, predykatem),
- zaś wartość “Jan Abacki” jest dopełnieniem (wartością cechy).

Reprezentacja Wiedzy

Główną siłą sprawczą wyznaczającą zakres i kierunek prac nad reprezentowaniem wiedzy jest to, do czego owa reprezentacja

ma być stosowana oraz w - w pewnym stopniu – to, w jaki sposób wiedza będzie pozyskiwana.

Nie istnieje zatem jedna, akceptowana przez wszystkich definicja terminu **reprezentacja wiedzy**.

Reprezentacja wiedzy

Opisy, Fakty

- są to zestawy cech i pojęć pierwotnych,
- służą do identyfikacji i rozróżniania obiektów i klas

Relacje:

- zapisane w bazie wiedzy zależności i skojarzenia pomiędzy faktami

Wiedza symboliczna

- deklaratywna – „wiem, że” – łatwa w opisie i formalizacji; trudności z reprezentacją sekwencji działań; nie podaje się sposobów rozwiązywania problemów; wnioskowanie oparte o skojarzenia faktów

Lub też polegająca na określeniu zbioru specyficznych dla reprezentowanej dziedziny faktów, stwierdzeń i reguł

- proceduralna – „wiem jak” – duża efektywność reprezentowania procesów; trudności z modyfikacją; bez korzystania ze skojarzeń
Lub też wiedza polegająca na określeniu zbioru procedur, działanie których reprezentuje wiedzę o dziedzinie

Wiedza niesymboliczna

odwołująca się do doświadczeń zbieranych na podstawie obserwacji środowiska nas otaczającego

Metody reprezentacji wiedzy symbolicznej

- rachunek zdań
- rachunek predykatów
- wykorzystujące zapis stwierdzeń
- wykorzystujące systemy regułowe
- sieci semantyczne
- ramy przypadków
- używające modeli obliczeniowych

Rachunek zdań

Zdania mogą być prawdziwe lub nie, reprezentują fakty.

Zdania stanowią zbiór, do którego należą również zdania otrzymane przez zastosowanie operatorów logicznych:

- \sim nieprawda
- \wedge koniunkcja, i
- \vee alternatywa
- \Rightarrow implikacja

Do zdań można stosować kwantyfikatory:

- \forall dla wszystkich
- \exists istnieje

Rachunek zdań pozwala na wnioskowanie na kilka sposobów.

Logika predykatów

Logiczna reprezentacja stwierdzeń o obiektach.

Logika stwierdzeń mających za argumenty obiekty,
np. *ja*, *człowiek*, *kartka*.

Predykaty mają argumenty i wartość logiczną.

Predykat *jest-czerwony*(x), *większy-od*(x,y),
lżejszy-od(x,y)

Predykat *isa*, czyli „jest członkiem”.

Logika predykatów dopuszcza kwantyfikatory.

Reprezentacja logiczna

$\exists x, Ptak(x)$, czyli istnieje przynajmniej jedno takie x , że $Ptak(x)=T$.

„Każdy ptak ma skrzydła” można zapisać jako:

$$\forall x, Ptak(x) \Rightarrow MaSkrzydła(x)$$

Wnioskowanie:

z prawdziwych faktów \Rightarrow nowe, prawdziwe fakty.

$$\forall x. Wróbel(x) \Rightarrow Ptak(x)$$

$$\forall x. Wróbel(x) \Rightarrow MaSkrzydła(x)$$

Reguły wnioskowania nie zależą od konkretnej wiedzy.

Jeśli $P \rightarrow Q$, oraz $Q \rightarrow R$ to $P \rightarrow R$

\Rightarrow

\Rightarrow

\Rightarrow

Przykład

- „Każdy Polak zna jakiegoś Francuza” (język naturalny)
- „Dla każdego x , jeśli x jest Polakiem, istnieje takie y , że y jest Francuzem i x zna y ”

Typy, Nazwy i Miary

- Klaudiusz jest słoniem
 - Słoń to gatunek
 - Klaudiusz to gatunek (???)
-
- W pierwszym zdaniu „słoń” dotyczy konkretnego obiektu
 - W drugim zdaniu „słoń” dotyczy opisu gatunku a więc wszystkich słoni
 - Rozwiązaniem jest wprowadzenie typów (uwaga! Typy należą do logiki drugiego rzędu)

Metody Regułowe

- JEŚLI przesłanka, TO konkluzja
- oraz
- JEŚLI przesłanka, TO działanie
- lub inaczej za pomocą zdania
- IF przesłanka THEN konkluzja
- oraz
- IF przesłanka THEN działanie

Reguły proste i złożone

IF spełnione są wszystkie warunki podpisanej umowy THEN towar można odebrać z magazynu

Ramy

Wprowadzone przez Marviną Minsky'ego w 1975 r.

Operatory: indukują zmiany, ale większość obiektów się nie zmienia i może być ujęta w „ramy”.

Ramy to złożone struktury powstałe w wyniku nagromadzenia się wcześniejszych doświadczeń; ich zadaniem jest opisać obiekt lub koncepcje i możliwości jej użycia.

Ramy mają „szufladki” lub „haczyki” (slots, hooks) na fakty lub procedury.

Rozumowanie polega na zapełnianie szufladek.

Wiedza deklaratywna i proceduralna może zostać umieszczona w dobrze zdefiniowanych ramach.

- Rama: pojedyncza struktura zawierająca klatki.
- Klatka (slot): element ramy który zawiera jedną lub więcej faset.
- Faseta: element opisujący coś w klatce.
 - Fasety mogą być: liczbami, tekstami, piktogramami, lub nawet kolejnymi ramami.
 - Wartość wpisaną w klatce nazywamy wartością klatki. Jest ona zapisana w fasetce jako wartość typu VALUE. Każdej klatce jest przypisany pewien określony zbiór faset.
 - W fasetach tych mogą być zapisane np. warunki dotyczące uznania wartości atrybutów jako wartości dopuszczalnych, procedury pozyskiwania tej wartości, itp. Zbiór ten może być dowolnie uzupełniany dodatkowymi informacjami.
- Demon: procedura dołączona do szczeliny odpalana warunkowo.
- Instancja: przykład ramy.

Ramę można opisać jako pewnego rodzaju reprezentację wiedzy o następujących właściwościach:

1. Rama jest strukturą danych, opisującą pewien obiekt albo klasę obiektów i oferującą dostęp do pełnej informacji o tym obiekcie.
2. Rama jest zbiorem klatek. Całość informacji o obiekcie, zawarta w ramie, jest dzielona na części będące wartościami klatek ramy.
3. Każda klatka odpowiada pewnej właściwości danego obiektu i jest określonego rodzaju, tzn. ma zdefiniowaną dziedzinę wartości, które mogą być w niej umieszczone.
4. W zależności od dziedziny wartości zawartych w klatce istnieją różne rodzaje klatek. Klatka jest jednoznacznie zdefiniowana przez podanie jej rodzaju, jak również do jakiej ramy należy.
5. Wartości klatki mogą być różne. W szczególności wartością klatki może być odwołanie do innej ramy albo dowolna procedura lub funkcja.
6. Klatki są dodatkowo dzielone na fasety, zawierające wybrane wartości klatki.
7. Ramy, klatki i fasety są identyfikowane za pomocą nazw.

„**DEFAULT**” (domyślnie) - fasetka ta zawiera tzw. domyślną wartość klatki. Jest to wartość stereotypowa dla obiektów opisywanych przez daną ramę. Fasetka ta jest brana pod uwagę jedynie wówczas, gdy fasetka VALUE nie jest wypełniona lub nie istnieje.

„**REQUIRE**” (wymagać) – fasetka ta zawiera ograniczenie dla elementu „VALUE”, to jest określającą dopuszczalną wartość tego elementu. Opisuje ona dziedzinę wartości możliwych do wpisania do klatki i najczęściej sprowadza się do określenia górnej i dolnej wartości granicznej. Zawartość tej fasetki jest wykorzystywana przez fasetkę IF-ADDED.

„**COMMENT**” (komentarz) - zawiera tekst będący opisem danej klatki.

„**CARDINALITY**” (główny) - podaje ona liczbę fasetek typu VALUE występujących w klatce.

„**RANGE**” (zasięg) - zawiera ona listę lub zakres dopuszczalnych wartości fasetki VALUE.

„**IF-NEEDED**” (jeśli potrzebne) – inicjowane przez system w momencie pobierania wartości z klatki. Zawiera funkcję, która wyznacza nieznaną wartość klatki poszukując jej w ramach nadrzędnych w stosunku do ramy wyjściowej. Jest wyszukiwana w klatce, jeżeli fasetki VALUE i DEFAULT są niewypełnione lub nie istnieją. Fasetki IF-NEEDED i DEFAULT mają podobne znaczenie. IF-NEEDED oblicza wartość a DEFAULT znajduje domyślną.

10

„**IF-ADDED**” (jeśli dodane) – inicjowane przez system w momencie modyfikacji wartości w klatce czyli zawiera funkcję, która wpisuje wartości do klatki i jest poszukiwana w klatce wówczas, gdy próbujemy wstawić do klatki nową wartość.

„**IF-REMOVED**” (jeśli usunięte) - inicjowane przez system w momencie modyfikacji wartości w klatce czyli zawiera funkcję usuwania wartości z klatki i jest poszukiwana w klatce, jeśli tylko próbujemy wymazać jej wartość.

„**CLEAN-UP**” (sprzątnięcie) - jest inicjowana przez system w momencie kasowania ramy.

„**RULES**” (reguła) – są to elementy zawierające listy wszystkich reguł wnioskowania których użycie może zmienić wartość elementu „VALUE” tej klatki.

Ramy – prosty przykład

Ogólna ramka PIES

Co to: ZWIERZĘ; ZWIERZĄTKO DOMOWE

Rasa: ?

Właściciel: OSOBA (jeśli-potrzebna: znajdź OSOBA z ZWIERZĄTKO DOMOWE =ja)

Imię: NAZWA WŁASNA (DEFAULT = Reks)

....

Ramka PIES-SAŚIADA

Co to: PIES

Rasa: kundel

Właściciel: Józek

Imię:

Schemat Ramy

Rama: Publikacja

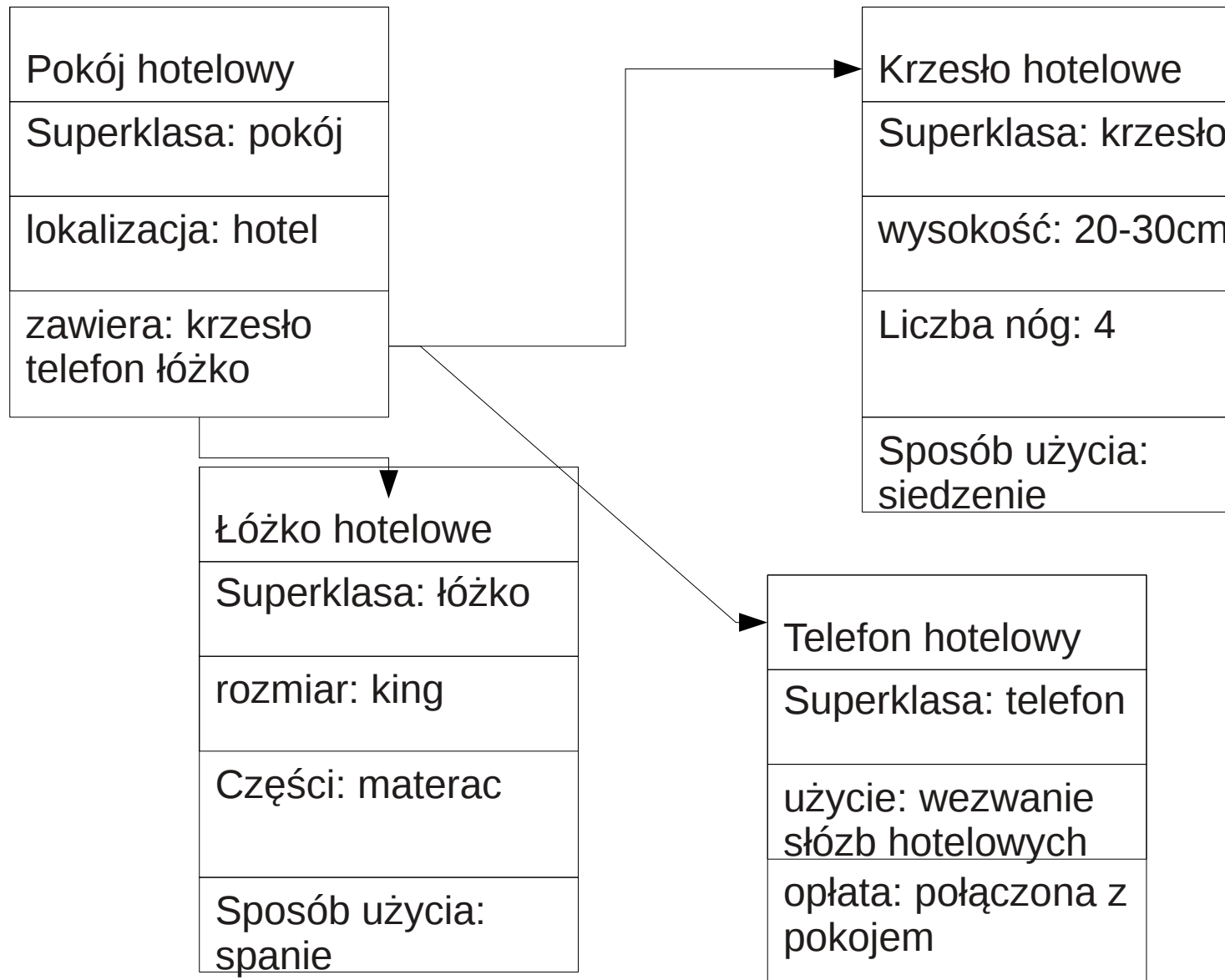
Klatka 1: Tytuł

Faseta Value: Tytuł
Faseta 2: Podtytuł

Klatka 2: Autorzy

Faseta Value: Autor_1
Faseta j: Autor_j

Ramy - przykład



Skrypty

JEDZENIE-W-RESTAURACJI (skrypt)

Obiekty: (restauracja, pieniądze, jedzenie, menu, stoliki, krzesła)

Role: (klienci, kelnerzy, kucharze)

Punkt-widzenia: klient

Czas-zdarzenia (godziny otwarcia restauracji)

Miejsce-zdarzenia (położenie restauracji)

Sekwencje zdarzeń:

najpierw: Wchodzimy do restauracji (skrypt)

potem if (znak rezerwacja lub prosimy-czekać-na-miejsce)

then (zwrócić-uwagę-kelnera skrypt)

potem Prosimy-usiąść skrypt

potem Zamawiamy-jedzenie skrypt

Skrypty cd

potem Spożywamy-jedzenie skrypt unless (długie-czekanie) when
Wychodzimy-zdenerwowani-z-restauracji skrypt

potem if (jedzenie-bardzo-dobre)
then Gratulacje-dla-kucharza skrypt

potem Płacimy-za-jedzenie skrypt

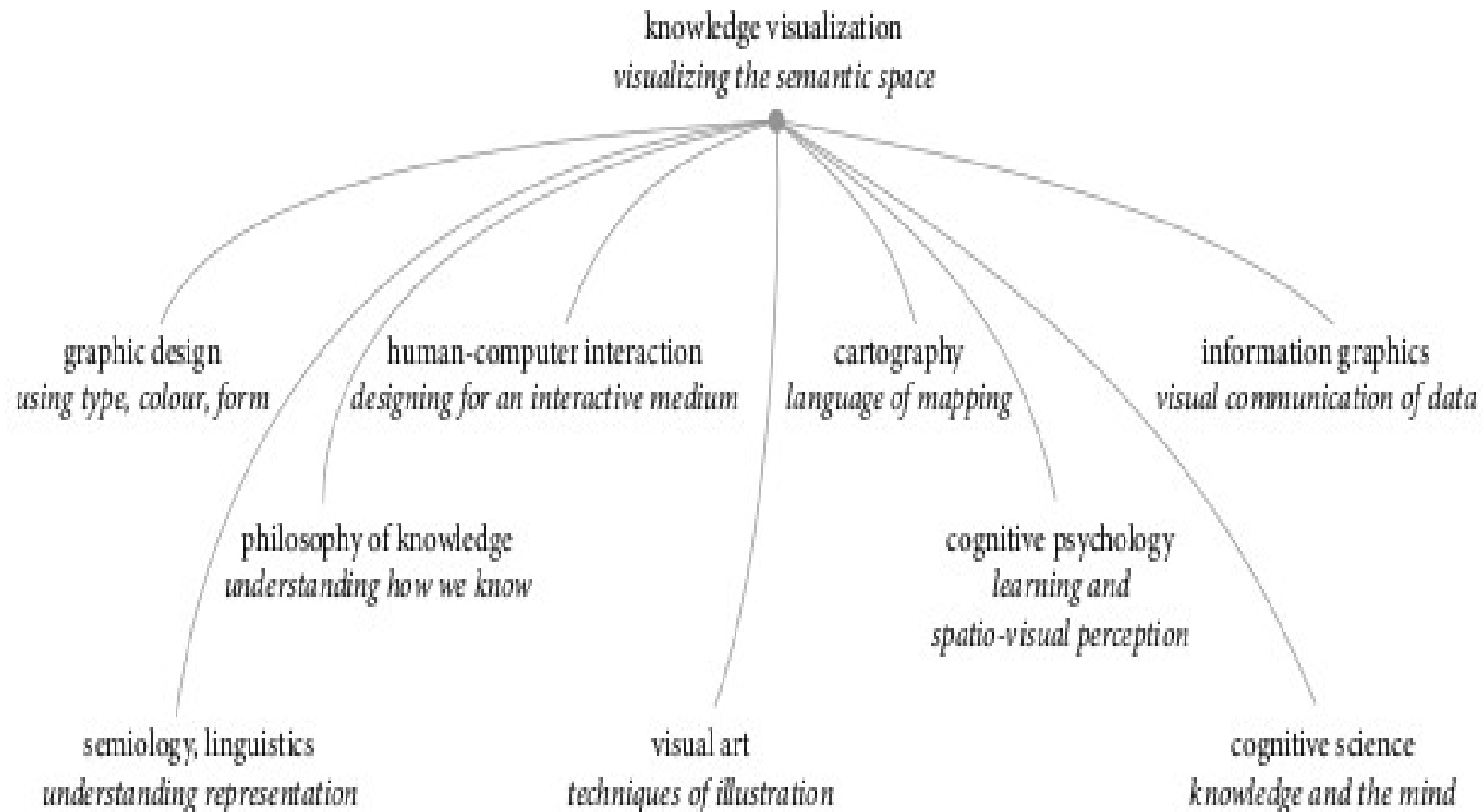
koniec:

Opuszczamy-restaurację skrypt

Skrypty umożliwiają analizę typowych historyjek, np. wycinków z gazet opisujących przejmowanie banków, fuzje firm itp.

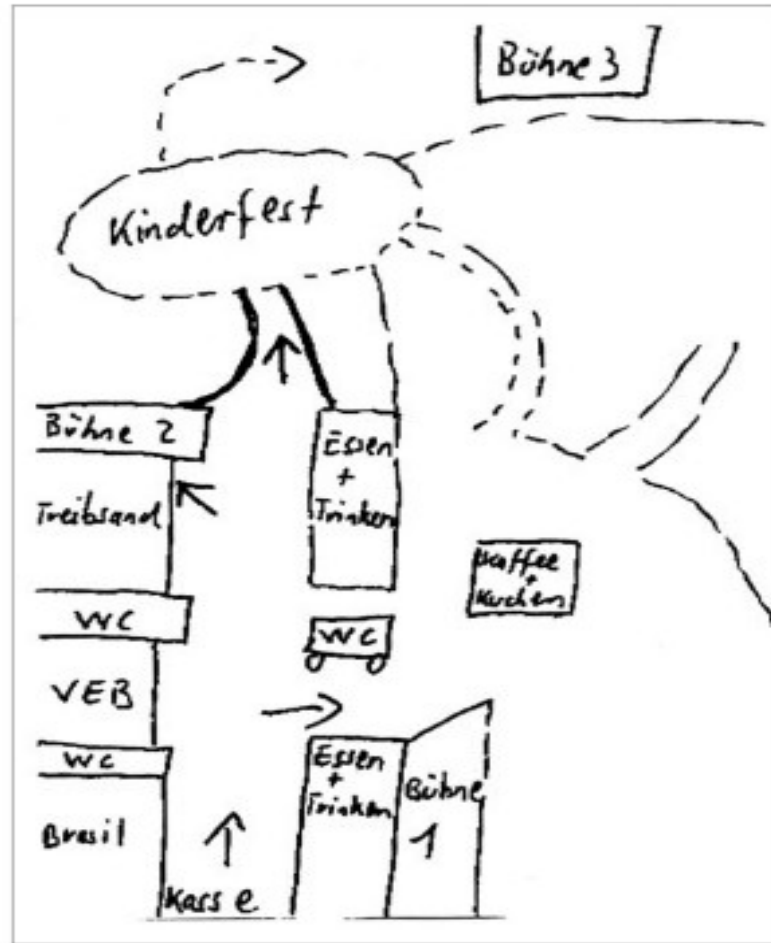
Dzięki temu można łatwo dopasować formę pytań i odpowiedzi do tekstu

Wizualizacja wiedzy



Mapy

Graficzne ilustracje struktur, relacji i atrybutów przestrzeni.



Upside down World Map

© Copyright Hema Maps Pty Ltd 2003

LEGEND

Capital = City, Town =

Hema Maps Pty Ltd
Ph: +61 7 3340 0000 Fax: +61 7 3340 0099
Web: www.hemamaps.com
Email: manager@hemamaps.com.au

Hema Maps NZ Limited
Ph: +64 9 273 6459 Fax: +64 9 273 6479
Email: sales.hema@clear.net.nz



GREENLAND

GREENLAND



Mapowanie mediów cyfrowych

Cyfrowe media wprowadzają cechy, które nie były przewidziane przez kartografów:

- Animacja

- Integracja tekstu, obrazu i dźwięku

- Interaktywność

- Usieciowienie

- 3D i światy wirtualne

Przeгляд najlepszych metod wizualizacyjnych

