HURTOWNIE DANYCH

FUNKCJE OKIENKOWE

OCENA JAKOŚCI DANYCH

DR INŻ. BERNADETTA MALESZKA

POWTÓRZENIE

- CASE, COALESCE, NULLIF
- CAST
- CTE
- PIVOT
- ZAPYTANIA Z PODSUMOWANIAMI – OPERATORY:
  - ROLLUP, CUBE, GROUPING SETS

```
select s.SalesPersonID, s.CustomerID, sum(s.subTotal) as suma
from [Sales].[SalesOrderHeader] s
group by cube(s.SalesPersonID, s.CustomerID)
order By 3 desc;
```

100 %

Results | Messages

	SalesPersonID	CustomerID	suma
1	NULL	NULL	109846381.4039
2	NULL	NULL	29358677.2207
3	276	NULL	10367007.4265
4	277	NULL	10065803.5404
5	275	NULL	9293903.0046
6	289	NULL	8503338.6457
7	279	NULL	7171012.7501
8	281	NULL	6427005.554
9	282	NULL	5926418.3555
10	290	NULL	4509888.9311
11	283	NULL	3729945.349
12	278	NULL	3609447.2148
13	280	NULL	3325102.5941
14	284	NULL	2312545.69
15	288	NULL	1827066.7118
16	286	NULL	1421810.9242
17	274	NULL	1092123.8561
18	NULL	29818	877107.1923
19	276	29818	856562.4908
20	279	29715	853849.1795
21	NULL	29715	853849.1795
22	NULL	29722	841908.7707
23	NULL	30117	816755.5763
24	NULL	29614	799277.895
25	NULL	29639	787773.0438
26	NULL	29701	746317.5293
27	NULL	29617	740985.8338
28	287	NULL	732759.1841
29	NULL	29994	730798.7139
30	NULL	29646	727272.6493
31	283	29580	724299.6365
32	NULL	29580	724299.6365
33	NULL	29827	711864.7624
34	NULL	29497	700803.7854
35	NULL	29716	693502.4852
36	276	29913	671618.0295
37	NULL	29913	671618.0295
38	290	30103	643745.8958
39	NULL	30103	643745.8958
40	276	29957	636226.4698
41	NULL	29957	636226.4698

```sql
select s.SalesPersonID, s.CustomerID, sum(s.subTotal) as suma
from [Sales].[SalesOrderHeader] s
group by cube(s.SalesPersonID, s.CustomerID)
order By 3 desc;
```

100 %

Results | Messages

	SalesPersonID	CustomerID	suma
1	NULL	NULL	109846381.4039
2	NULL	NULL	29358677.2207
3	276	NULL	10367007.4265
4	277	NULL	10065803.5404
5	275	NULL	9293903.0046
6	289	NULL	8503338.6457

FUNKCJE OKIENKOWE

- GROUP BY – DLA KAŻDEJ GRUPY JEDNA WARTOŚĆ WYRAŻENIA
- OVER – WGLĄD PRZEZ „OKNO REKORDÓW”

```
<FUNKCJA AGREGUJĄCA> OVER (
        [<KLAUZULA PARTYCJONOWANIA>]
        [<KLAUZULA ORDER> [<KLAUZULA RAMKI>]]
)
```

FUNKCJE OKIENKOWE

- GROUP BY – DLA KAŻDEJ GRUPY JEDNA WARTOŚĆ WYRAŻENIA
- OVER – WGLĄD PRZEZ „OKNO REKORDÓW”

```
<FUNKCJA AGREGUJĄCA> OVER (
        PARTITION BY <LISTA KOLUMN>
        ORDER BY <LISTA PORZĄDKOWANIA>
)
```

OVER

- FUNKCJE AGREGUJĄCE:
  - COUNT, SUM, AVG, MIN, MAX
- FUNKCJE SZEREGUJĄCE:
  - ROW_NUMBER – NR POZYCJI
  - RANK – RANKING (TO SAMO MIEJSCE DLA TYCH SAMYCH WARTOŚCI)
  - DENSE_RANK – RANKING, NUMEROWANIE CIĄGŁE
  - NTILE – GRUPUJE REKORDY POPRZEZ PRZYPISANIE TEJ SAMEJ WARTOŚCI SZEREGUJĄCEJ CZŁONKOM GRUPY

OVER - PRZYKŁAD

```
SELECT DISTINCT CustomerID,
      SUM(TotalDue) OVER(Partition BY CustomerID) "Suma zakupów",
      AVG(TotalDue) OVER(Partition BY CustomerID) "Średnia zakupów"
FROM Sales.SalesOrderHeader;
```

CustomerID	Suma zakupów	Średnia zakupów
15254	1795,0173	897,5086
16321	107,4613	53,7306
18727	95,3284	47,6642
24930	71,7919	71,7919
18470	6653,8902	3326,9451

OVER - PRZYKŁAD

```
SELECT SalesOrderID, CustomerID,
      RANK() OVER(ORDER BY CustomerID) AS Ranking
FROM Sales.SalesOrderHeader;

SELECT SalesOrderID, CustomerID,
      DENSE_RANK() OVER(ORDER BY CustomerID)  AS
                        "Dense ranking"
FROM Sales.SalesOrderHeader;
```

OVER - WYNIK

SalesOrdesID	Average	Ranking RANK	Dense Ranking DENSE_RANK
51131	151704,9021	1	1
55282	151704,9021	1	1
61324	151704,9021	1	1
65322	151704,9021	1	1
54356	123246,6745	5	2
52456	123246,6745	5	2

OVER - PRZYKŁAD

```
SELECT SalesOrderID, CustomerID, SubTotal,
   SUM(SubTotal) OVER(PARTITION BY CustomerID) AS Sprzedaż,
   SUM(SubTotal) OVER() AS "Całkowita sprzedaż",
   SUM(SubTotal) OVER(PARTITION BY CustomerID)
         / SUM(SubTotal) OVER()*100.   AS "Udział klienta %"
FROM Sales.SalesOrderHeader
```

SalesOrderID	CustomerID	SubTotal	Sprzedaż	Całkowita sprzedaż	Udział klienta %
51783	29818	107270,1188	877107,1923	109846381,4039	0.79
50270	29818	78041,5646	877107,1923	109846381,4039	0.79
57105	29818	110050,8354	877107,1923	109846381,4039	0.79
45577	29715	58704,1822	853849,1795	109846381,4039	0.77
44801	29715	48156,2081	853849,1795	109846381,4039	0.77
44133	29715	25686,5186	853849,1795	109846381,4039	0.77

NTILE - PRZYKŁAD

```
WITH Sprzedaz AS
(
    SELECT [SalesPersonID] emp, AVG([SubTotal]) srednia
    FROM [Sales].[SalesOrderHeader]
    WHERE [SalesPersonID] IS NOT NULL
    GROUP BY [SalesPersonID]

)
SELECT srednia, emp
       , NTILE(5) OVER(ORDER BY srednia DESC) AS grupy
FROM Sprzedaz;
```

NTILE - WYNIK

```
WITH Sprzedaz AS
(
    SELECT [SalesPersonID] emp, AVG([SubTotal]) sre
    FROM [Sales].[SalesOrderHeader]
    WHERE [SalesPersonID] IS NOT NULL
    GROUP BY [SalesPersonID]

)
SELECT srednia, emp
       , NTILE(5) OVER(ORDER BY srednia DESC) AS gr
FROM Sprzedaz;
```

srednia	emp	grupy
35001,0799	280	1
26557,8741	281	1
25770,7938	290	1
24801,4531	276	2
24434,8811	289	2
22752,5803	274	2
21868,7024	282	3
21280,7685	277	3
20653,1177	275	3
19735,1605	283	4
18788,697	287	4

średnia	emp	Ocena pracownika
35001,0799	280	Wyróżniony
26557,8741	281	Wyróżniony
25770,7938	290	Wyróżniony
21868,7024	282	Dobry
21280,7685	277	Dobry
16518,1835	284	Kandydat do zwolnienia
15424,988	278	Kandydat do zwolnienia

```
WITH Sprzedaz AS
(
      SELECT [SalesPersonID] emp, AVG([SubTotal]) srednia
      FROM [Sales].[SalesOrderHeader]
      WHERE [SalesPersonID] IS NOT NULL
      GROUP BY [SalesPersonID]
)
SELECT srednia, emp,
      CASE NTILE(3) OVER(ORDER BY srednia DESC)
         WHEN 1 THEN 'Wyróżniony'
         WHEN 2 THEN 'Dobry'
         WHEN 3 THEN 'Kandydat do zwolnienia'
      END  AS "Ocena Pracownika"
FROM Sprzedaz
```

OVER

```
SELECT *
FROM
 (SELECT DISTINCT CustomerID, -- SalesOrderID,
    SUM(SubTotal) OVER(PARTITION BY CustomerID) Suma,
    AVG(SubTotal) OVER(PARTITION BY CustomerID) AS
                                          Srednia,
    ROW_NUMBER() OVER(ORDER BY CustomerID) AS RNUM,
    DENSE_RANK() OVER(ORDER BY CustomerID) AS
                                     denseRANKtest,
    RANK() OVER(ORDER BY CustomerID) AS RANKtest,
    NTILE(1000) OVER(ORDER BY CustomerID) AS NTILEtest
  FROM Sales.SalesOrderHeader)  T1
```

OVER - WYNIK

CustomerID	Suma	Srednia	RNUM	denseRANK test	RANKtest	NTILEtest
11000	8248,99	2749,6633	1	1	1	1
11000	8248,99	2749,6633	2	1	1	1
11000	8248,99	2749,6633	3	1	1	1
11001	6383,88	2127,96	4	2	4	1
11001	6383,88	2127,96	5	2	4	1
11001	6383,88	2127,96	6	2	4	1
11002	8114,04	2704,68	7	3	7	1
11002	8114,04	2704,68	8	3	7	1
11002	8114,04	2704,68	9	3	7	1
11003	8139,29	2713,0966	10	4	10	1

POZYCJONOWANIE WYNIKU

- LAG, LEAD, FIRST_VALUE, LAST_VALUE
  - znajdują wiersz w podziale i odczytują jego wartość

- OVER (PARTITION BY (YEAR(ORDERDATE)) ORDER BY MONTH(ORDERDATE) ROWS BETWEEN 1 PRECEDING AND CURRENT ROW
  - w bieżącym i poprzednim miesiącu

- CUME_DIST, PERCENT_RANK
  - wyznacza percentyle

PODSUMOWANIE SQL - ZADANIA

BAZA: ADVENTUREWORKS

1. Jaka była łączna suma transakcji (SalesOrderHeader.SubTotal) w poszczególnych latach dla kolejnych dni tygodnia?
2. Zaproponuj podział klientów na 3 rozłączne grupy wiekowe. Ilu różnych klientów dokonało zakupów w kolejnych miesiącach roku w każdej z grup? Ilu klientów w poszczególnych grupach wykonało zakup dokładnie jeden raz?
3. Przygotuj zestawienie produktów, których sprzedaje się miesięcznie min. 20 sztuk. Dla każdego produktu podaj jego kategorię.

Przeanalizuj uzyskane wyniki.

Jeśli w zapytaniu warto użyć CTE, to porównaj efektywność swojego rozwiązania z wersją bez CTE.

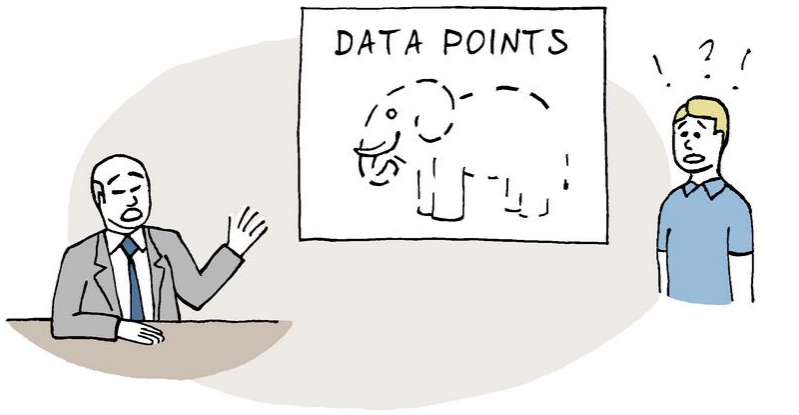
PODSUMOWANIE SQL - ZADANIA

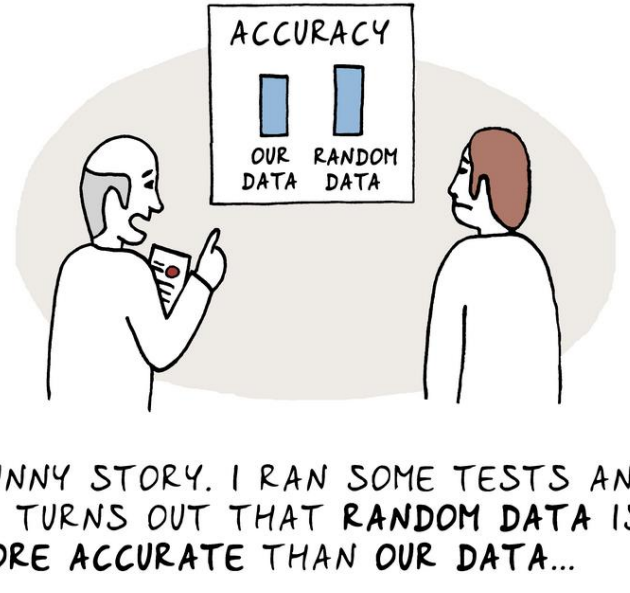
4. Przygotuj zestawienie, w którym przeanalizujesz, ilu jest różnych klientów dla każdej płci w kolejnych miesiącach (05.2011 – 06.2024)? Jak procentowo rozkłada się ich udział w całkowitej wartości sprzedaży (Sales.SalesOrderHeader.TotalDue)?
5. Przeanalizuj udział sprzedanych produktów w poszczególnych podkategoriach w stosunku do całych kategorii (zarówno pod względem liczbowym jak i wartościowym).
6. Przygotuj zestawienie, w którym możliwa będzie analiza regionalna z uwzględnieniem lokalnej waluty (kwoty sprzedaży w zależności od waluty i regionu).

Przeanalizuj uzyskane wyniki.

ANALIZA JAKOŚCI DANYCH

JAKOŚĆ DANYCH

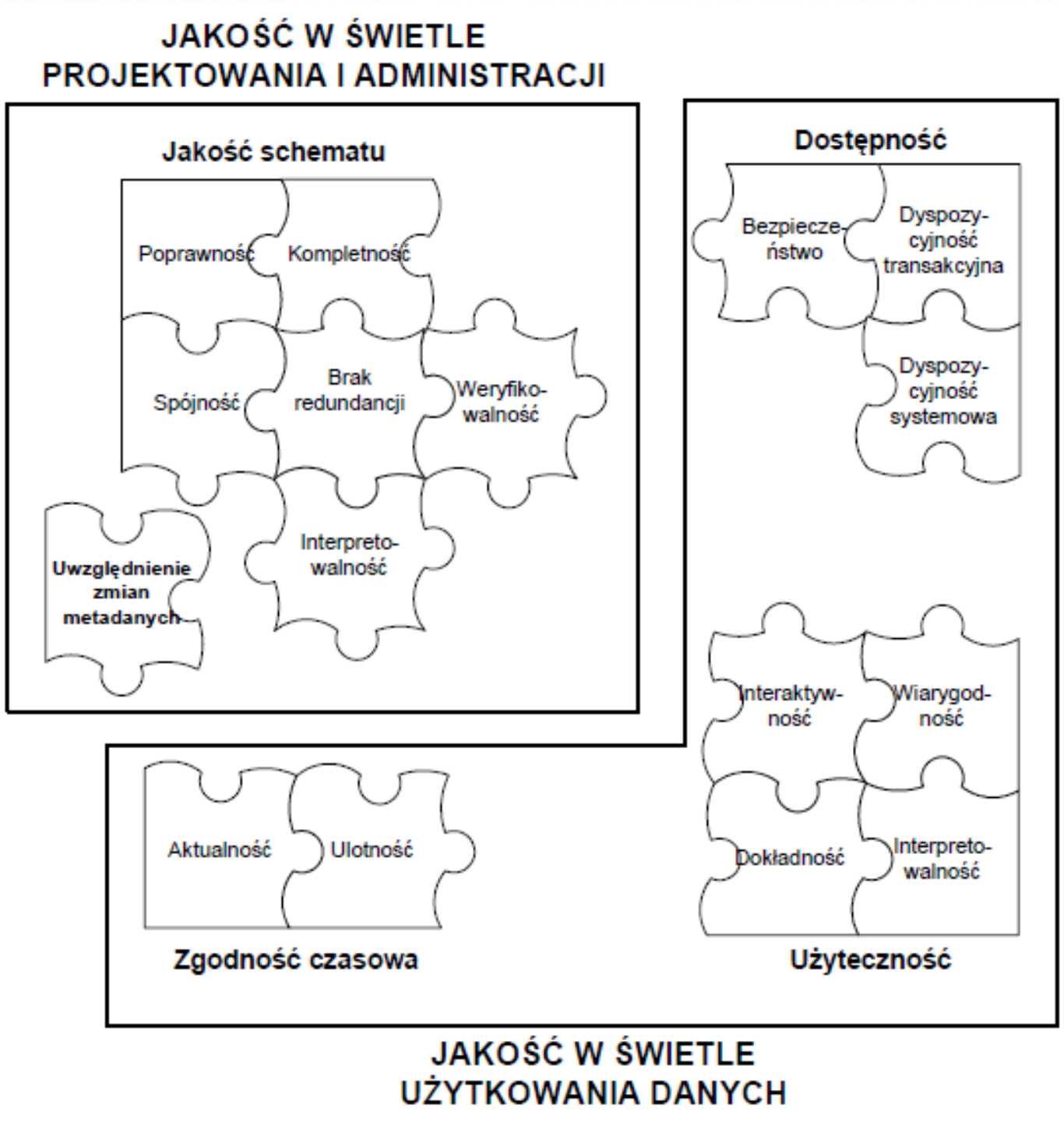




https://dataedo.com/cartoon

JAKOŚĆ DANYCH

- „Jakość danych należy rozumieć jako kwalifikację poprawności danych, ale także ich przydatności”

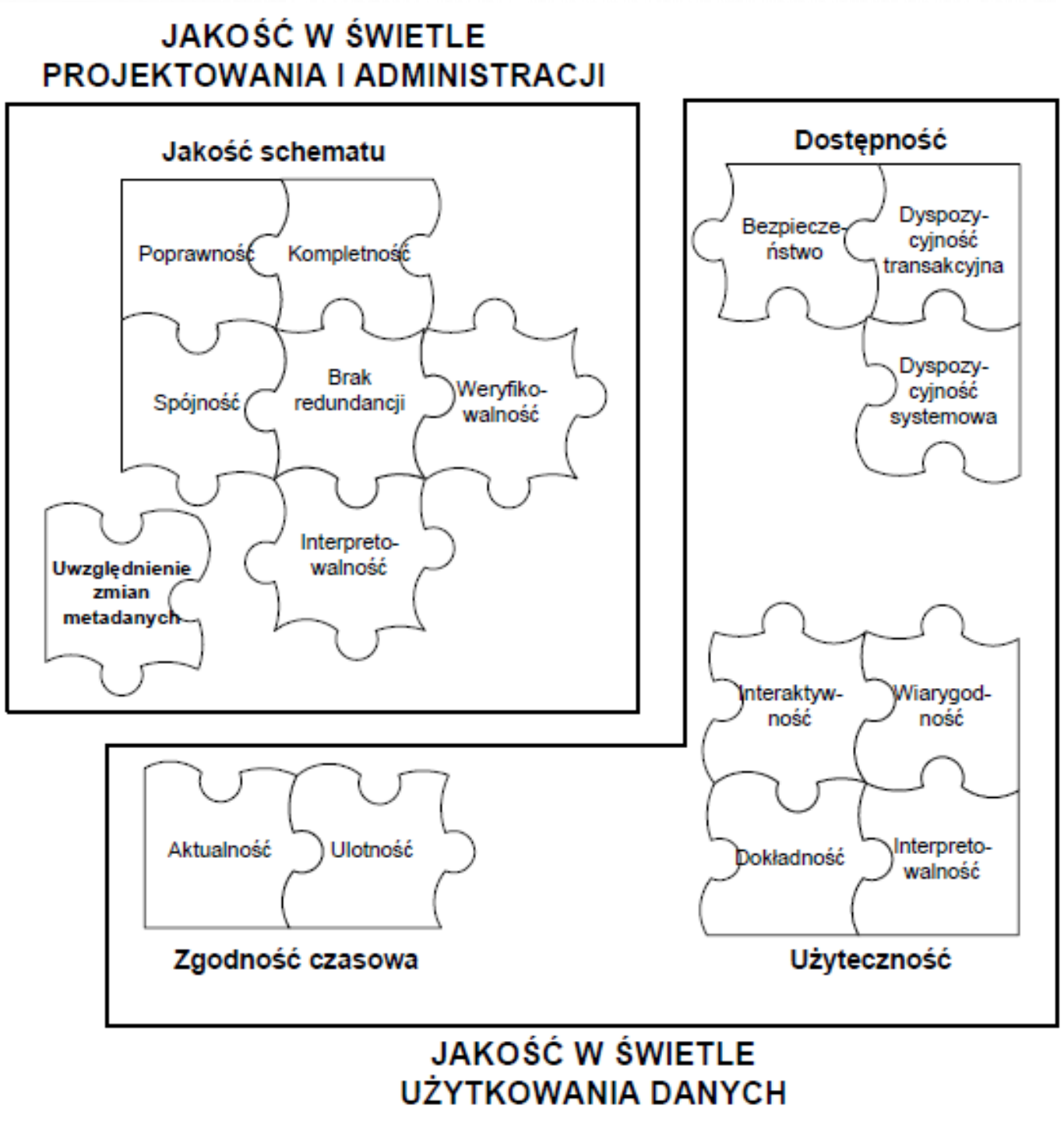


Błaszczyk K., Knosala R. (2006): Problem Jakości Danych w Hurtowniach. Systemy Wspomagania Organizacji SWO 2006, s. 45-54

JAKOŚĆ DANYCH

- Jakość schematu:
  - poprawność (validation)
  - kompletność (completeness)
  - spójność (consistency)
  - brak redundancji (reduction of recurrence)
  - weryfikowalność (traceability)
  - interpretowalność (interpretability)
  - uwzględnienie zmian metadanych

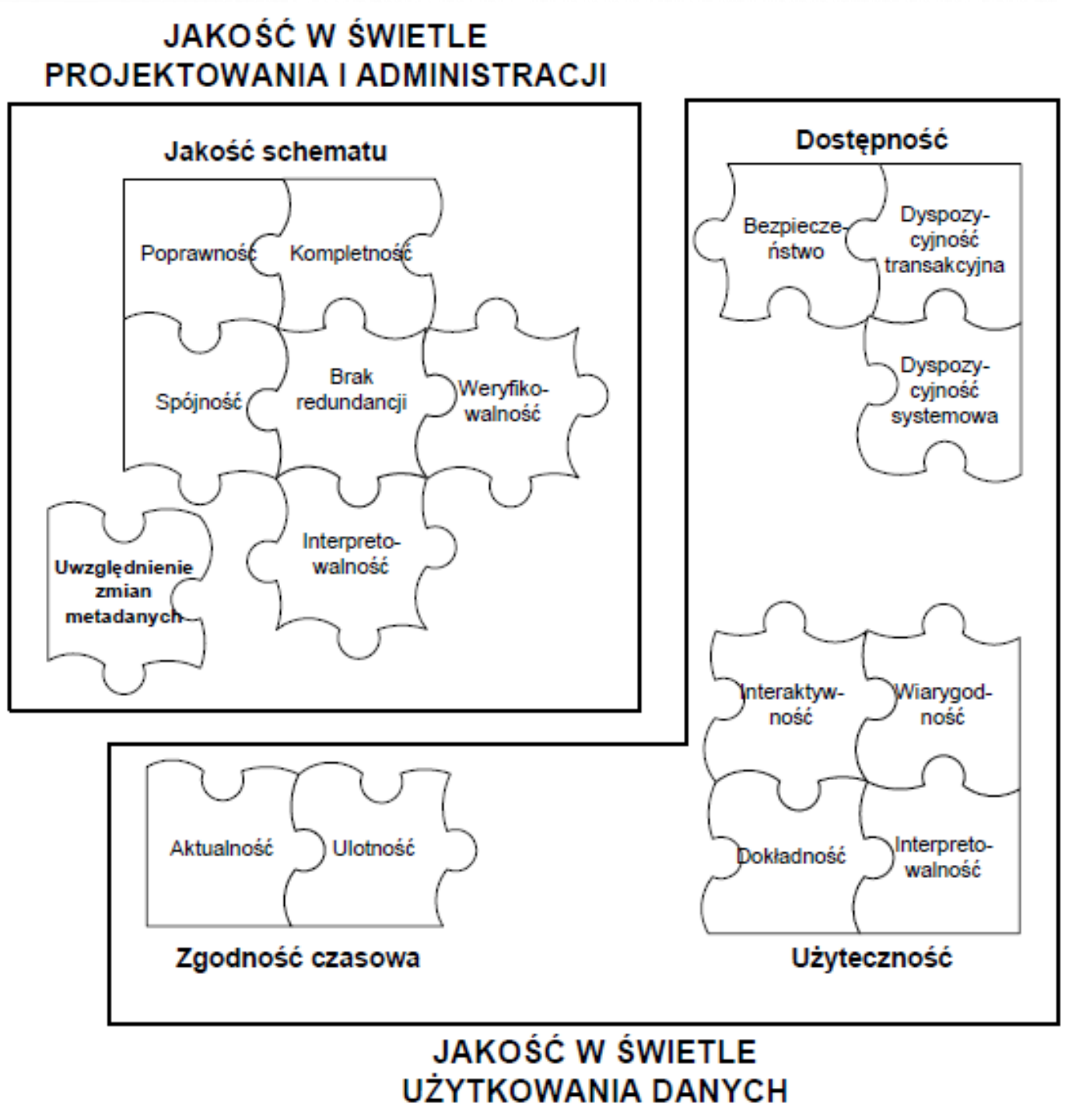
Błaszczyk K., Knosala R. (2006): Problem Jakości Danych w Hurtowniach. Systemy Wspomagania Organizacji SWO 2006, s. 45-54



JAKOŚĆ DANYCH

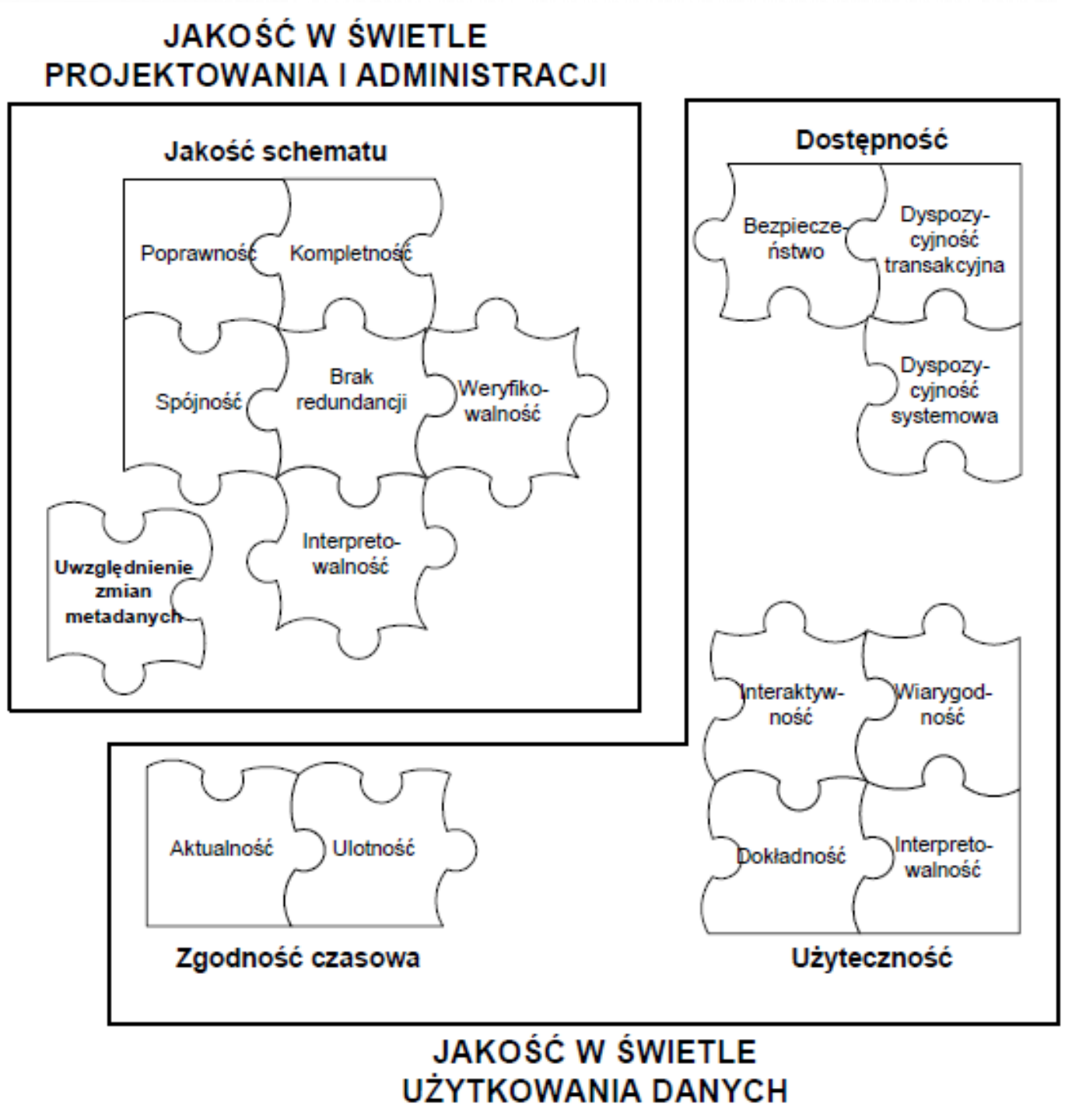
- dostępność (accessibility):
  - bezpieczeństwo (security)
  - dyspozycyjność transakcyjna (transactional availaibility)
  - dyspozycyjność systemu (system availability)
- zgodność czasowa (timelines):
  - aktualność (currency)
  - ulotność (volatility)

Błaszczyk K., Knosala R. (2006): Problem Jakości Danych w Hurtowniach. Systemy Wspomagania Organizacji SWO 2006, s. 45-54



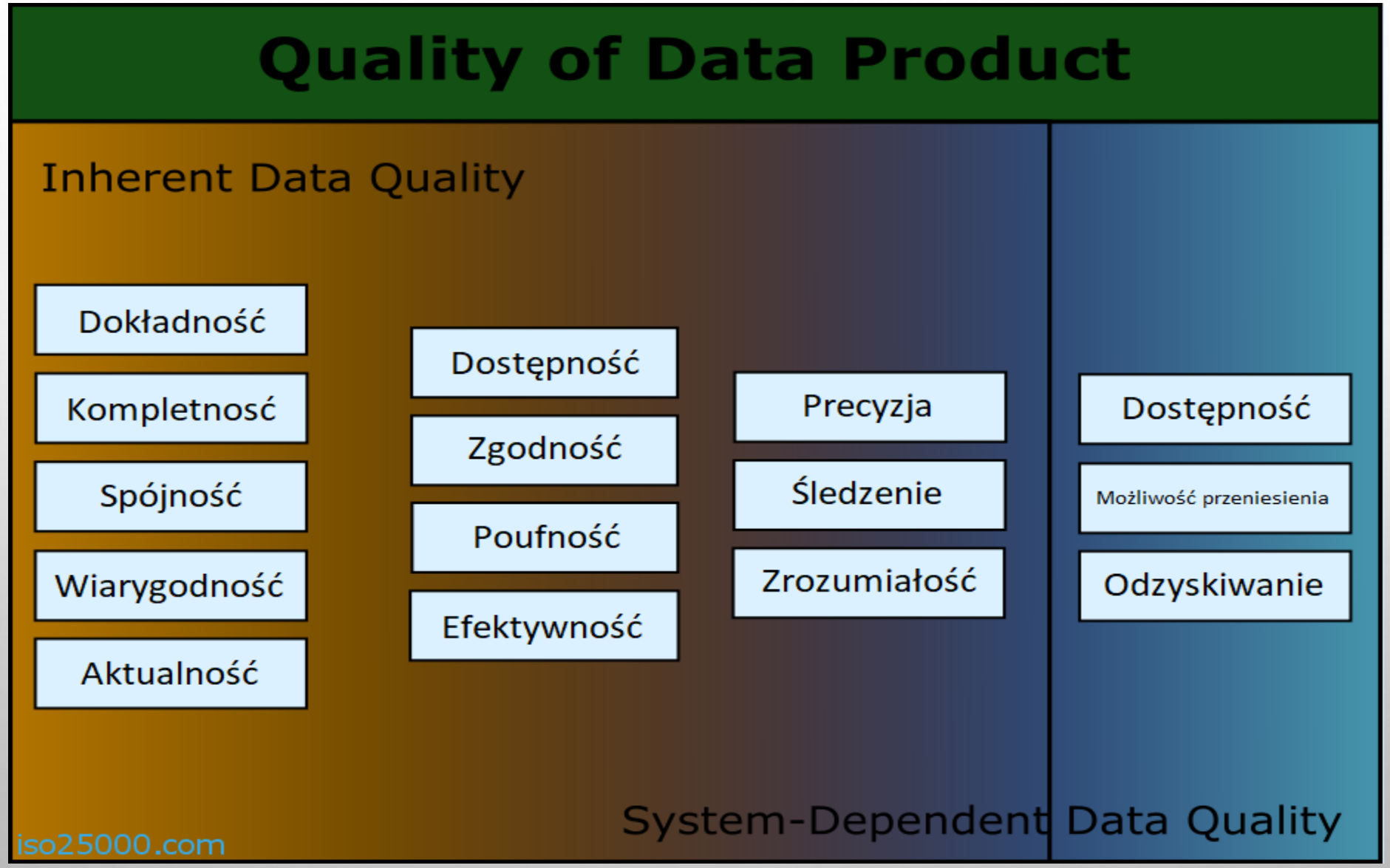
JAKOŚĆ DANYCH

- użyteczność (usefulness):
  - interaktywność (interactivity)
  - interpretowalność (interpretability)
  - wiarygodność (credibility)
  - dokładność (accurancy)



Błaszczyk K., Knosala R. (2006): Problem Jakości Danych w Hurtowniach. Systemy Wspomagania Organizacji SWO 2006, s. 45-54

NORMA ISO 25000



6 WYMIARÓW JAKOŚCI DANYCH

Kompletność	Dokładność	Spójność
Poprawność	Unikalność	Integralność

https://www.collibra.com/blog/the-6-dimensions-of-data-quality

JAKOŚĆ DANYCH DLA KLIENTA

Gartner (2020): Data Quality Fundamentals for Data & Analytics Technical Professionals

TRIAGE DANYCH

- proces identyfikacji, ustalania priorytetów i rozwiązywania problemów z jakością danych
- pomaga zapewnić dokładność, kompletność i spójność danych.

- Kroki:
- identyfikacja problemów z jakością danych
- ustalenie priorytetów problemów z jakością danych (selekcja danych)
- rozwiązywanie problemów z jakością danych: oczyszczanie danych, walidację danych i wzbogacanie danych
- regularne przeglądanie jakości danych i rozwiązywanie wszelkich pojawiających się problemów

KORZYŚCI Z SELEKCJI DANYCH

- poprawa jakości danych
- większa dokładność danych
- zmniejszone ryzyko naruszeń danych i innych incydentów związanych z danymi
- lepsza zgodność (np. z przepisami prawa)
- większa wydajność

PROFILOWANIE DANYCH

- profilowanie kolumn
- profilowanie międzykolumnowe
- profilowanie międzytabelowe
- walidacja reguł i ograniczeń
- integralność kluczy
- liczność
- wzorce i rozkłady częstości

https://dataedo.com/cartoon

PROFILOWANIE DANYCH

- profilowanie kolumn
  - statystyki: min, max, średnia
  - typ danych, rozmiar
  - liczność
  - rozkład częstości
  - wartości unikalne
  - kandydaci na klucze
  - procentowa zawartość wartości NULL

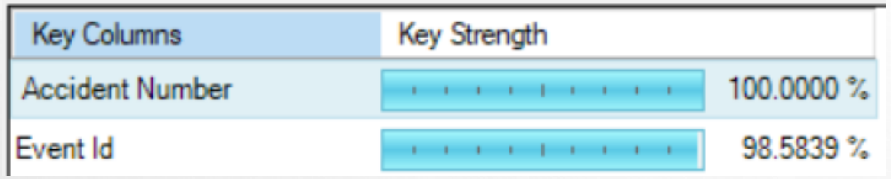
https://dataedo.com/cartoon

PROFILOWANIE DANYCH

- profilowanie międzykolumnowe
  - analiza klucza
  - analiza zależności pomiędzy kolumnami -> zależności funkcyjne
- profilowanie międzytabelowe
  - analiza danych rozproszonych
  - analiza więzów integralności

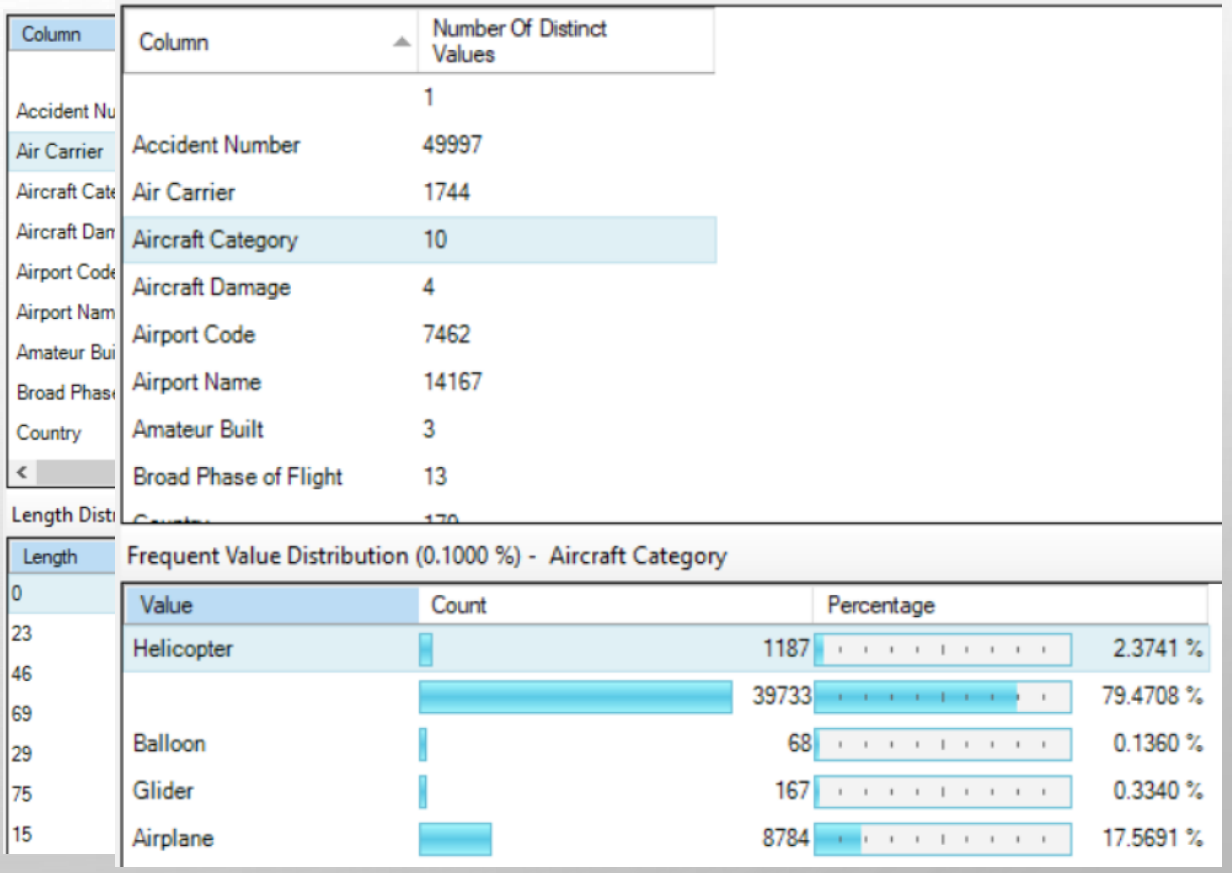
https://datascientyst.com/data-cleaning-steps-python-example/

PROFILOWANIE DANYCH



Key Columns	Key Strength
Accident Number	100.0000 %
Event Id	98.5839 %

- klucze kandydujące
- procent brakujących danych
- rozkład długości ciągów znakowych w danych
- rozkład częstości występowania wartości
- wartości unikalne



Column	Number Of Distinct Values
	1
Accident Number	49997
Air Carrier	1744
Aircraft Category	10
Aircraft Damage	4
Airport Code	7462
Airport Name	14167
Amateur Built	3
Broad Phase of Flight	13

Frequent Value Distribution (0.1000 %) - Aircraft Category

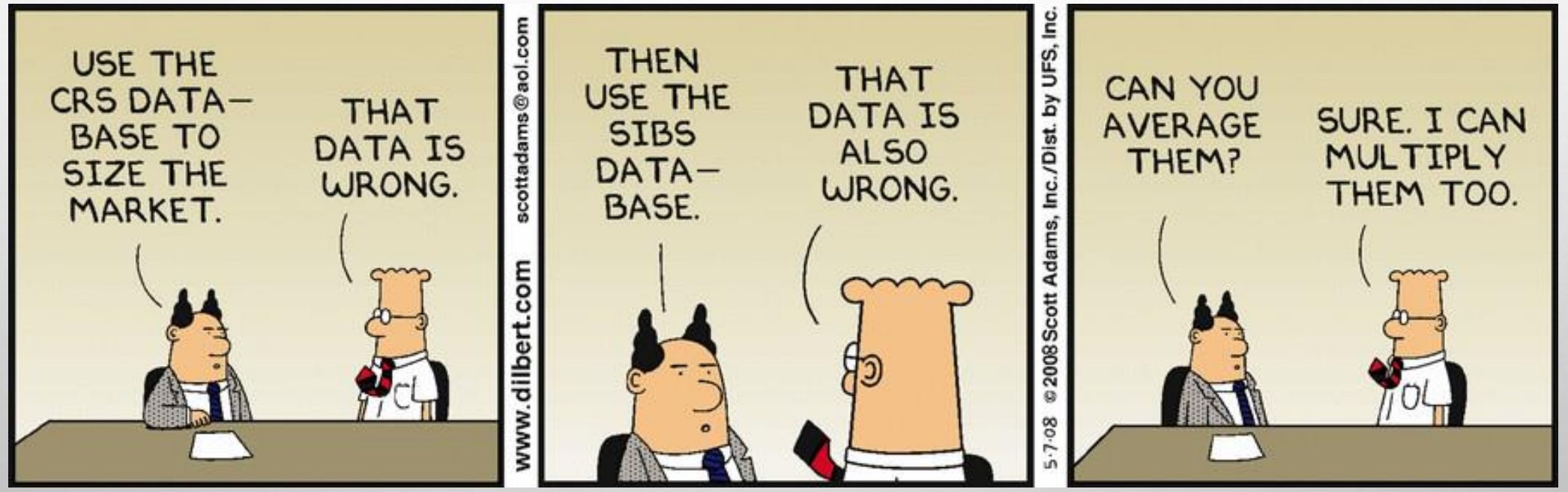
Value	Count	Percentage
Helicopter	1187	2.3741 %
	39733	79.4708 %
Balloon	68	0.1360 %
Glider	167	0.3340 %
Airplane	8784	17.5691 %

CZYSZCZENIE DANYCH

1. Zidentyfikuj problematyczne dane
2. Wyczyść dane
3. Usuń, zakoduj, uzupełnij wszelkie brakujące dane
4. Usuń wartości odstające lub przeanalizuj je osobno
5. Oczyść zanieczyszczone dane, sprawdź proces pobierania danych
6. Standaryzacja niespójnych danych
7. Sprawdź, czy twoje dane mają sens (są prawidłowe)
8. Zapobiegaj problemom z typami danych
9. Usuń błędy inżynieryjne (strukturalne)
10. Powtarzaj kroki 1 - 9 aż jakość danych będzie wystarczająca

https://dataedo.com/cartoon

DZIĘKUJĘ ZA UWAGĘ



https://www.linkedin.com/pulse/clean-tidy-data-gooddata-linda-warnier