

Strona danych

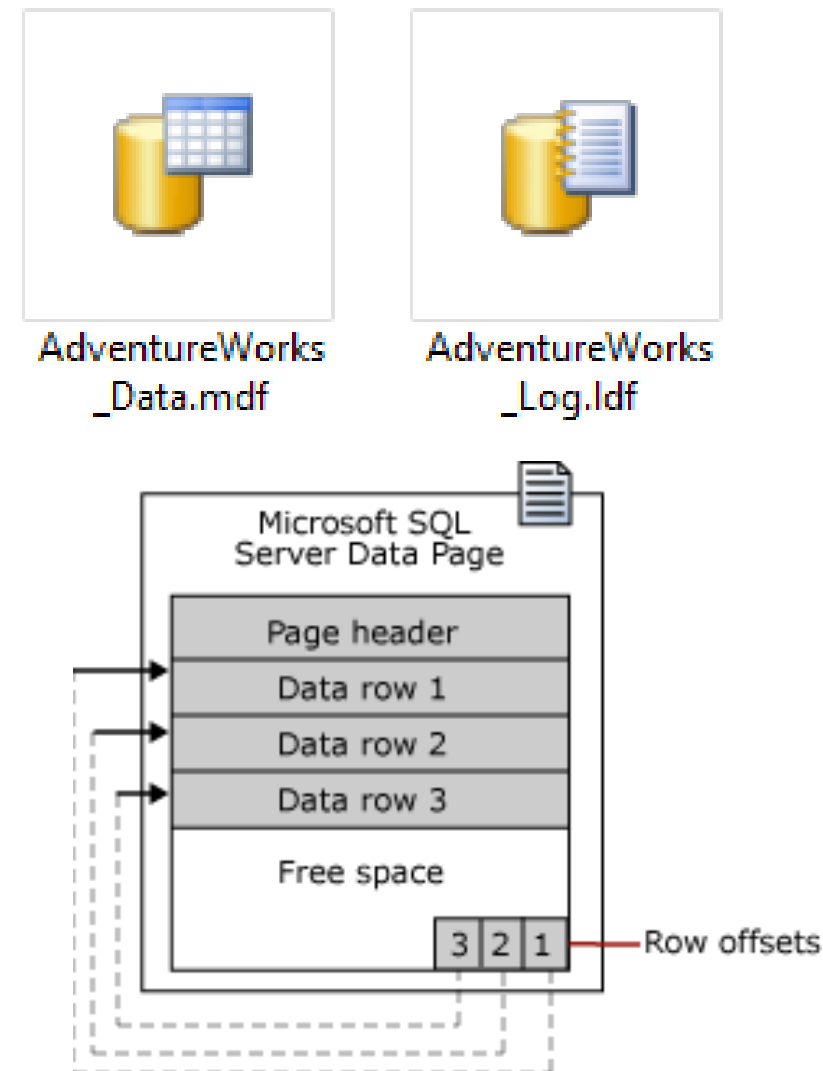
Jedna baza danych = min 2 pliki:

mdf – plik z danymi

ldf – log transakcyjny

Plik z danymi podzielony jest na strony danych

Jedna strona = 8 KB



Statystyki IO/TIME

```
138 SET STATISTICS IO ON
139
140 SELECT ID, FirstName, LastName
141 FROM People
142 WHERE City = 'Croix'
143
```

149 %

Results Messages Execution plan

(30 rows affected)
Table 'People'. Scan count 1, logical reads 102,

```
138 SET STATISTICS TIME ON
139
140 SELECT ID, FirstName, LastName
141 FROM People
142 WHERE City = 'Croix'
143
```

149 %

Results Messages

SQL Server parse and compile time:
CPU time = 0 ms, elapsed time = 17 ms.
SQL Server parse and compile time:
CPU time = 0 ms, elapsed time = 0 ms.

(30 rows affected)

SQL Server Execution Times:
CPU time = 15 ms, elapsed time = 1 ms.

Plan wykonania

Przepis na optymalne wykonanie zapytania biorąc pod uwagę strukturę danych i indeksów, liczbę danych itd

Czytamy od prawej do lewej

Przykładowe operatory:

Index scan

Index seek

Sort

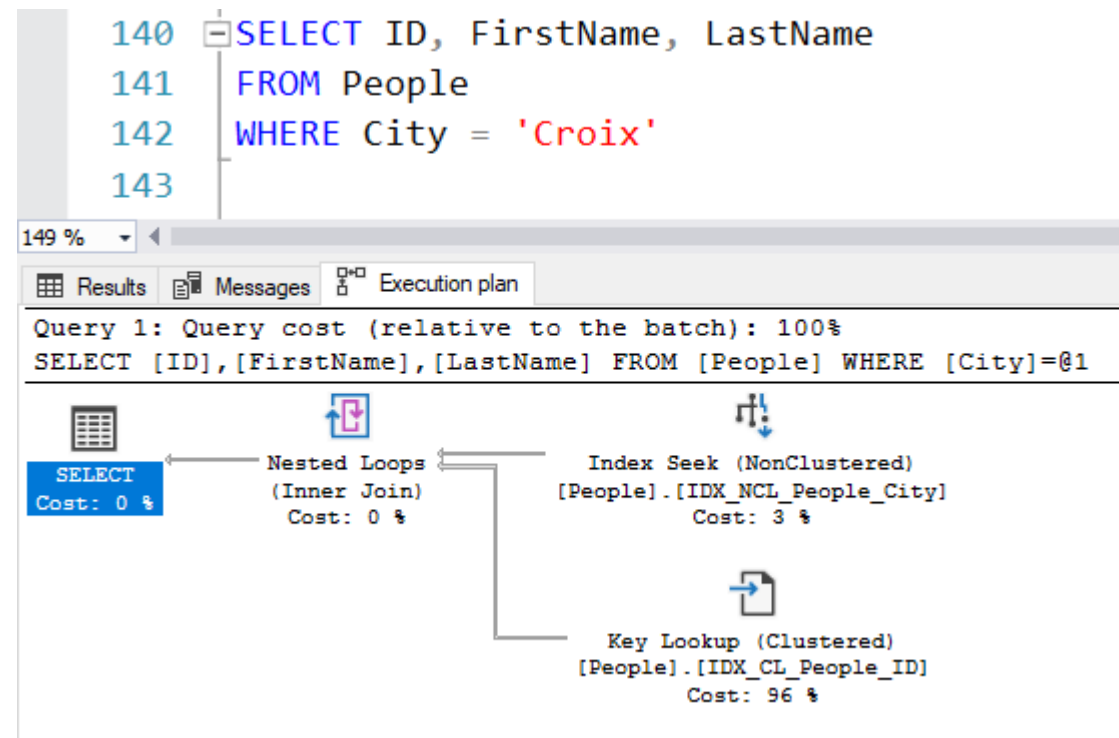
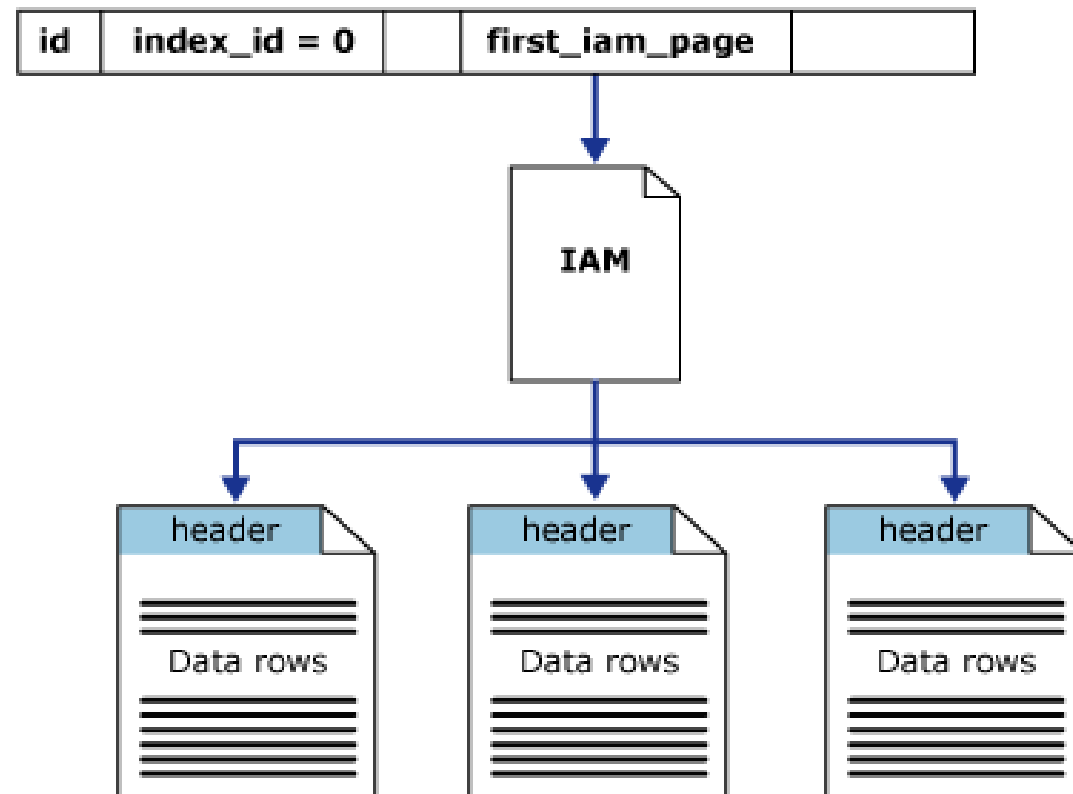


Tabela bez indeksów

Dane zapisywane na kolejnych stronach zgodnie z kolejnością wstawiania do tabeli

IAM (Index Allocation Map) zawiera informację, które strony zawierają wiersze należące do tabeli

Tabela danych będzie poprawnie zwracać dane jeśli skasujemy z niej indeksy (tyle, że dużo wolniej)



Index klastrowy

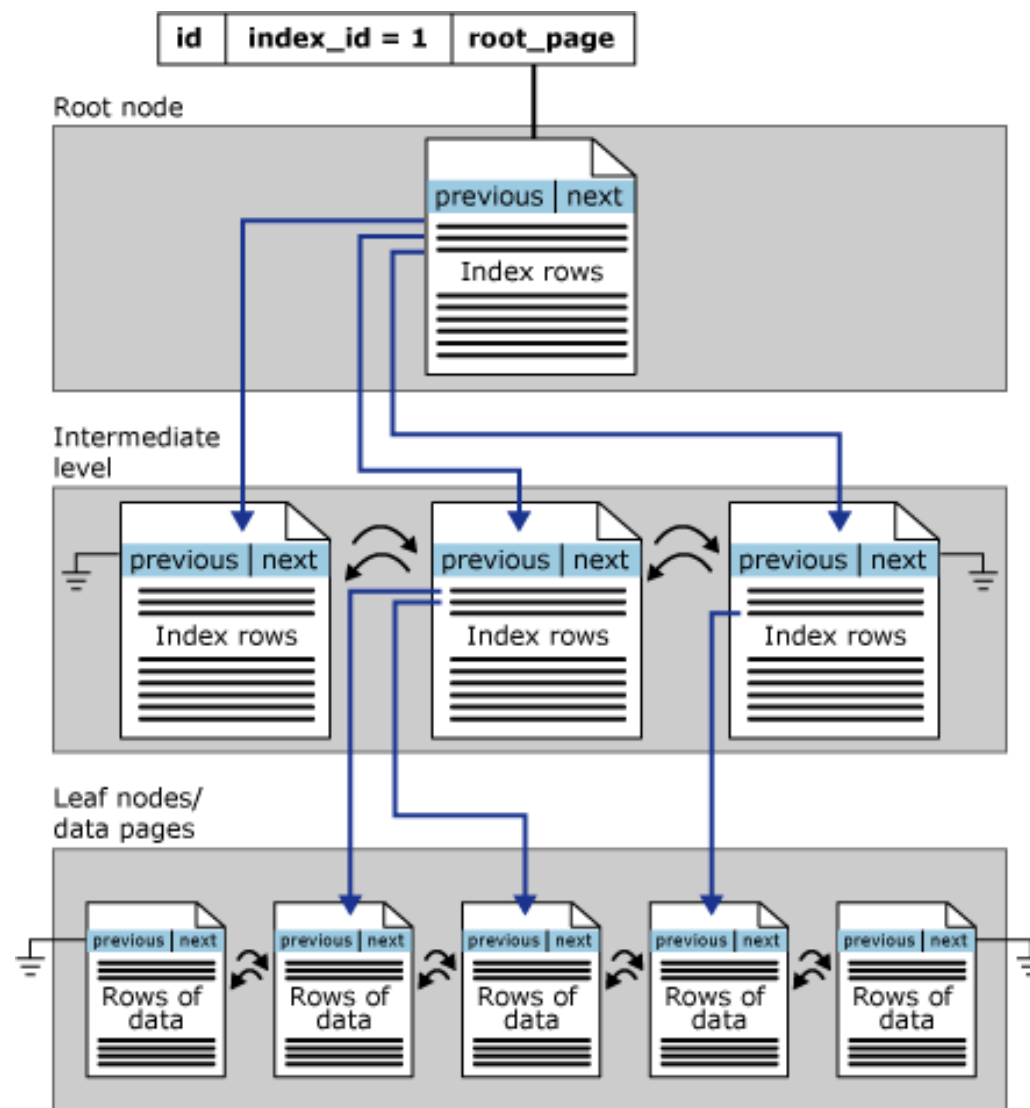
Clustered index

Może być 1 w tabeli

Wymusza logiczny porządek stron tabeli (wg klucza)

Na poziomie liści znajdują się dane

Domyślnie w SQL Server
PRIMARY KEY = Clustered Index



Klucz indeksu klastrowego

Maksymalnie 16 kolumn, 900 bajtów

Wąski (liczba kolumn, typy danych)

Przyrostowy (identity/ sequence) – kolejne wartości - rosnące

Statyczny (niezmienny)

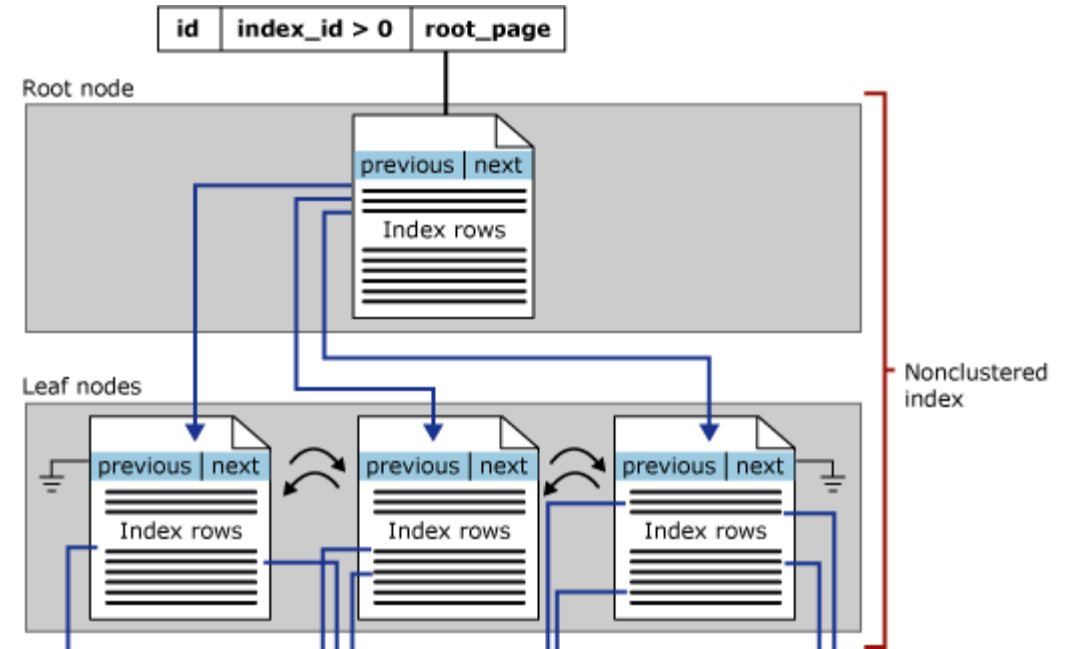
Unikalny (jeśli CI nie jest unikalny –
SQL Server dodaje 4B do każdego
klucza żeby zagwarantować
wewnętrznie unikalność)

Index nieklastrowy

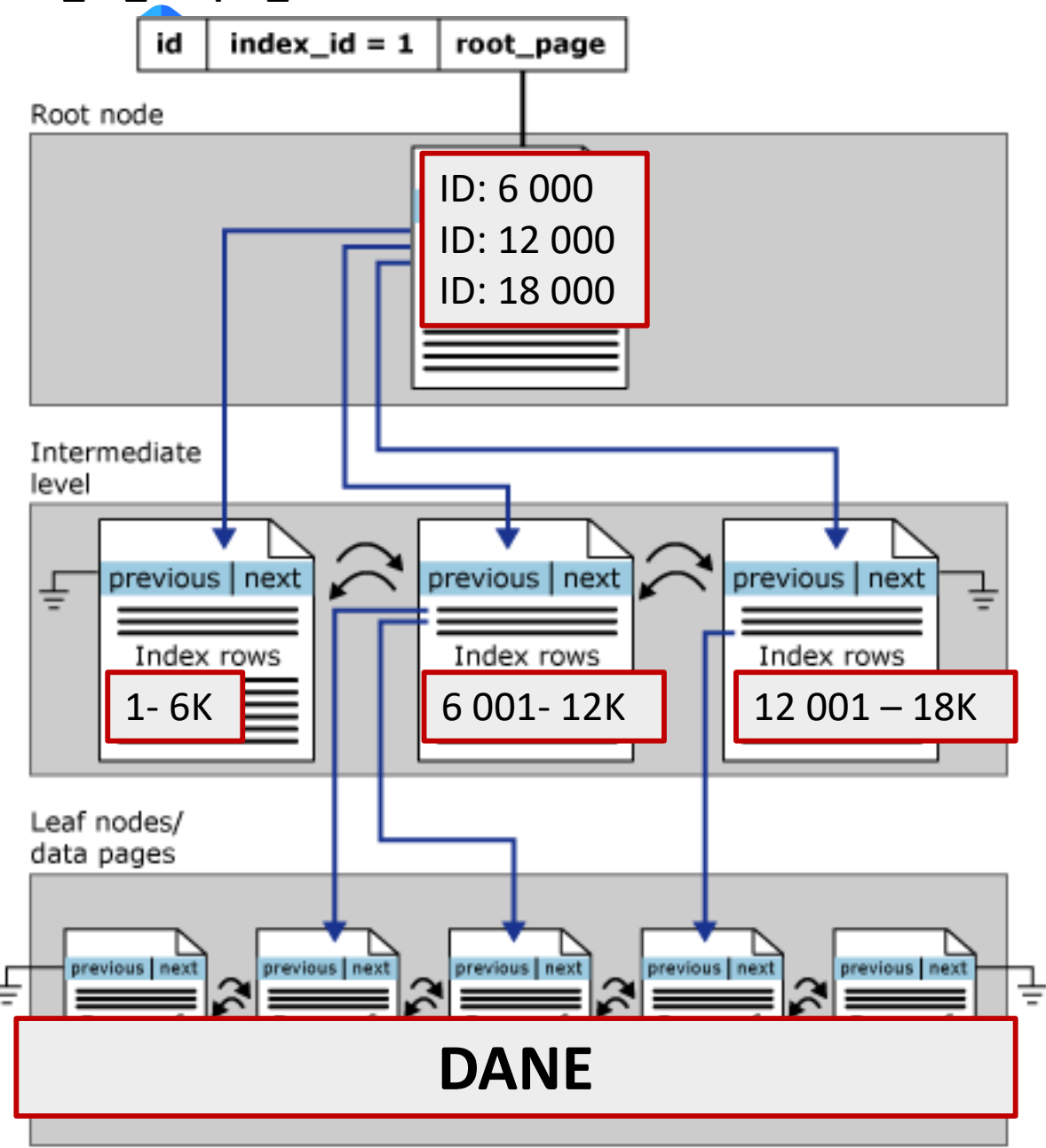
Nonclustered index

999 w jednej tabeli

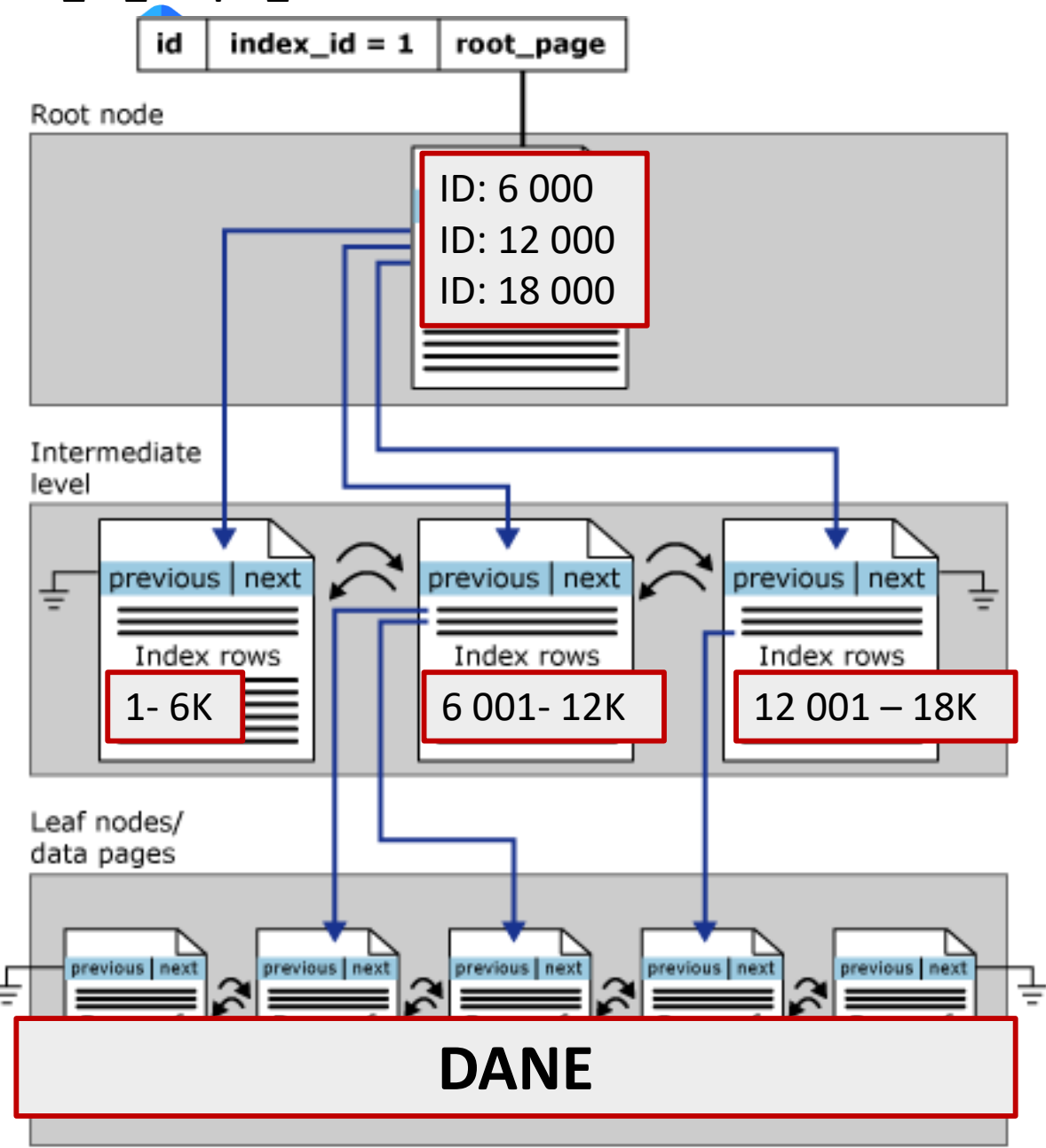
Na poziomie liści znajdują się klucze indeksu klastrowego/ numery stron na stercie



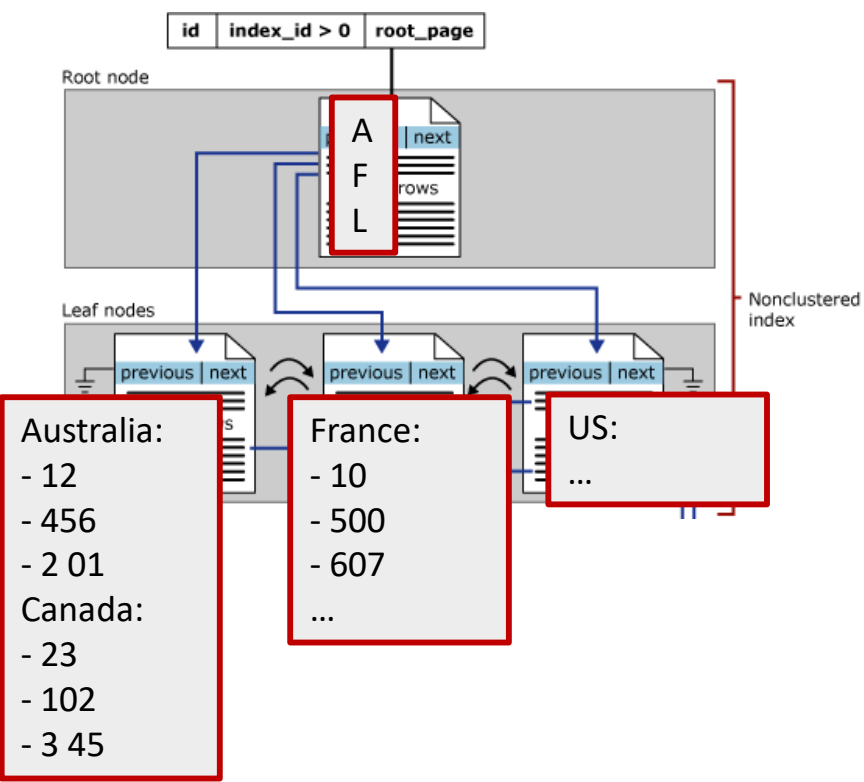
IDX_CL_People_ID



IDX_CL_People_ID

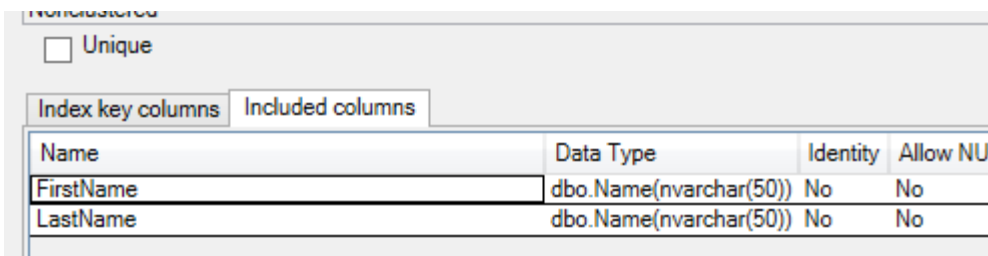


IDX_NCL_People_Country



Indeks pokrywający

- Indeks zawierający wszystkie kolumny wymagane przez zapytanie
- Brak konieczności wyszukiwania danych w klastrze/ stercie



The screenshot shows the 'Index Properties' dialog box for a non-clustered index. The 'Index key columns' tab is selected, showing a table with columns: Name, Data Type, Identity, and Allow NU. The 'Name' column is split into 'FirstName' and 'LastName'. Both are of type 'dbo.Name(nvarchar(50))', have 'No' for Identity, and 'No' for Allow NU.

Name	Data Type	Identity	Allow NU
FirstName	dbo.Name(nvarchar(50))	No	No
LastName	dbo.Name(nvarchar(50))	No	No

```
SELECT ID, FirstName, LastName  
FROM People  
WHERE City = 'London'
```

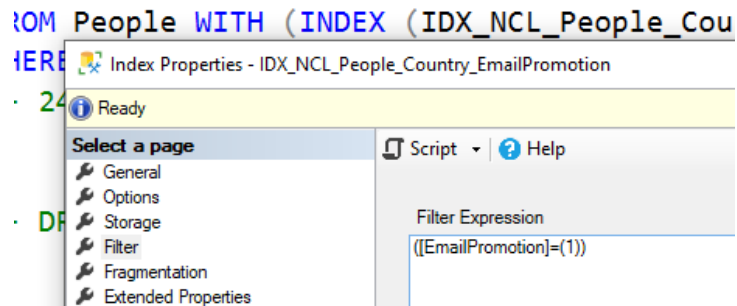
```
CREATE NONCLUSTERED INDEX IDX_NCL_People_City ON People (City)  
INCLUDE (Firstname, Lastname)
```

Indeks filtrujący

Indeks zawierający podzbiór wierszy
na podstawie warunku filtrującego WHERE

Zazwyczaj zawiera ten sam warunek, co większość zapytań

Jest mniejszy od indeksu, który zawierałby wszystkie dane – a więc bardziej efektywny



```
CREATE NONCLUSTERED INDEX IDX_NCL_People_Country_EmailPromotion
ON People (Country)
INCLUDE (Firstname, Lastname)
WHERE EmailPromotion = 1
GO
```