



## GOLD SPONSORS



Volvo Group IT

## SILVER SPONSORS



## BRONZE SPONSOR



## STRATEGIC PARTNER



# Nietrywialne podejście do partycjonowania tabel

Tomasz Waloszek

# Agenda

- Co to jest partycjonowanie
- Ograniczenia
- Rodzaje partycjonowania
- Partition function i partition scheme
- Sliding window scenario
- Partycjonowanie a colunstore index
- Partycjonowanie 2016 vs Partycjonowanie 2014
- Czy partycjonowanie to dobra metoda na optymalizację zapytań?

Agenda!



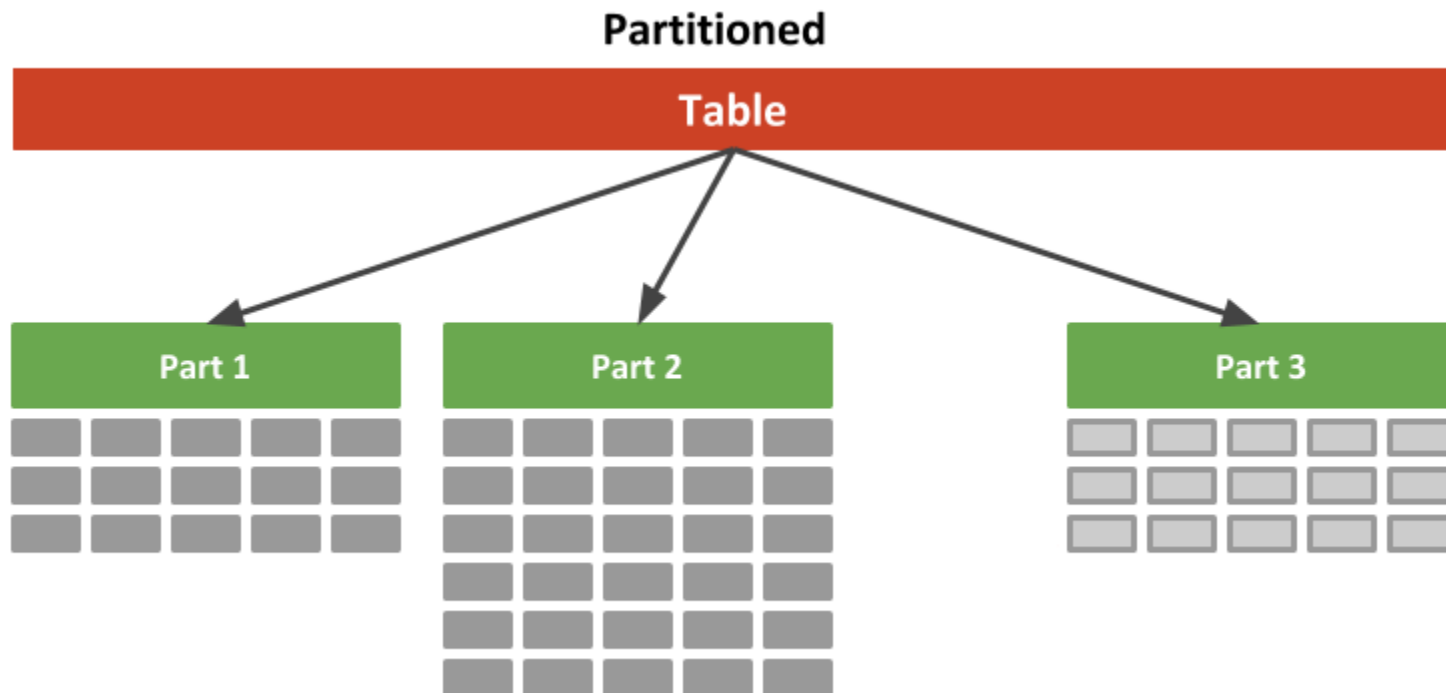
# Tomasz Waloszek

- Lider bydgosko-toruńskiej grupy Data Community
- MCP | MCSA | MCSE Data Management and Analytics
- Bussiness Inteligence Consultant
- [tomasz.waloszek@datacommunity.pl](mailto:tomasz.waloszek@datacommunity.pl)
- <https://www.linkedin.com/in/tomaszwaloszek/>
- Asystent Centrum Kettlebell Polska  
<http://www.centrumkb.pl/tomasz-waloszek/>
- Prywatnie miłośnik sportów walki oraz treningów siłowych metodyką hardstyle



# Co to jest partycjonowanie

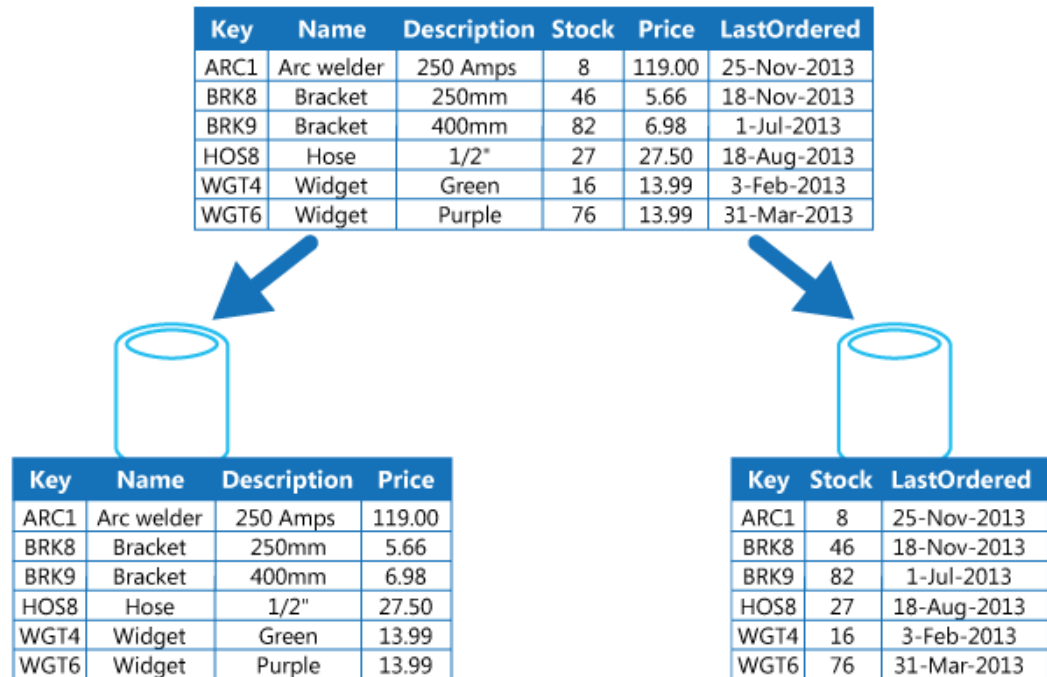
Partycjonowanie tabeli generalnie polega na podzieleniu jej na mniejsze części. Dzięki temu zamiast jednego dużego miejsca do przechowywania danych możemy podzielić je na mniejsze części.



# Vertical Partitioning

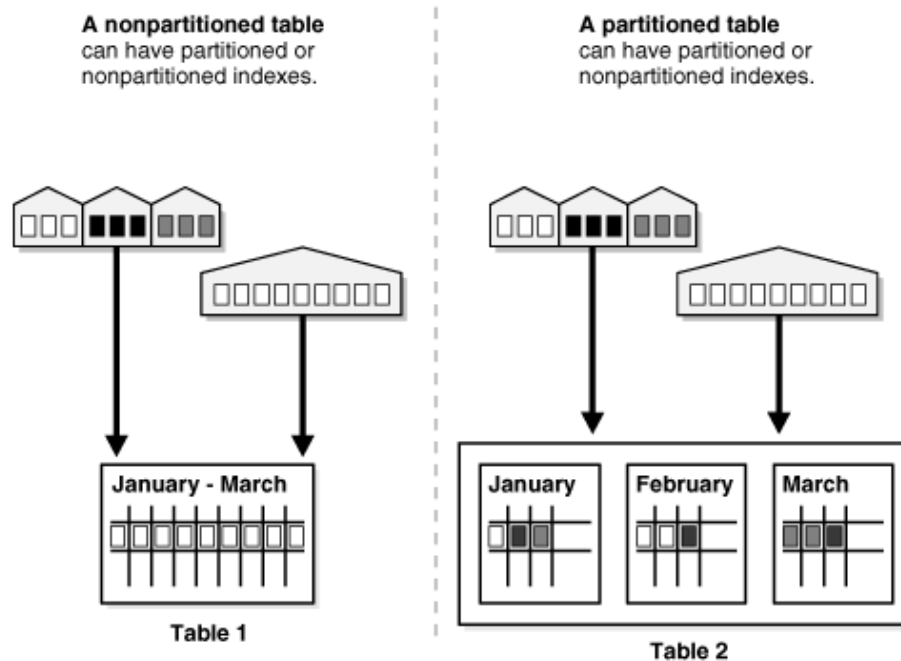
Vertical partitioning polega na podziale tabeli zawierającej dużą ilość kolumn na kilka tabel o mniejszej ilości kolumn. Rodzaje:

- normalizacja pionowa,
- dzielenie wierszy



# Horizontal Partitioning

- Horizontal Partitioning (partycjonowanie poziome) dzieli tabelę na kilka tabel. Każda tabela zawiera taką samą liczbę kolumn, ale mniejszą liczbę wierszy.



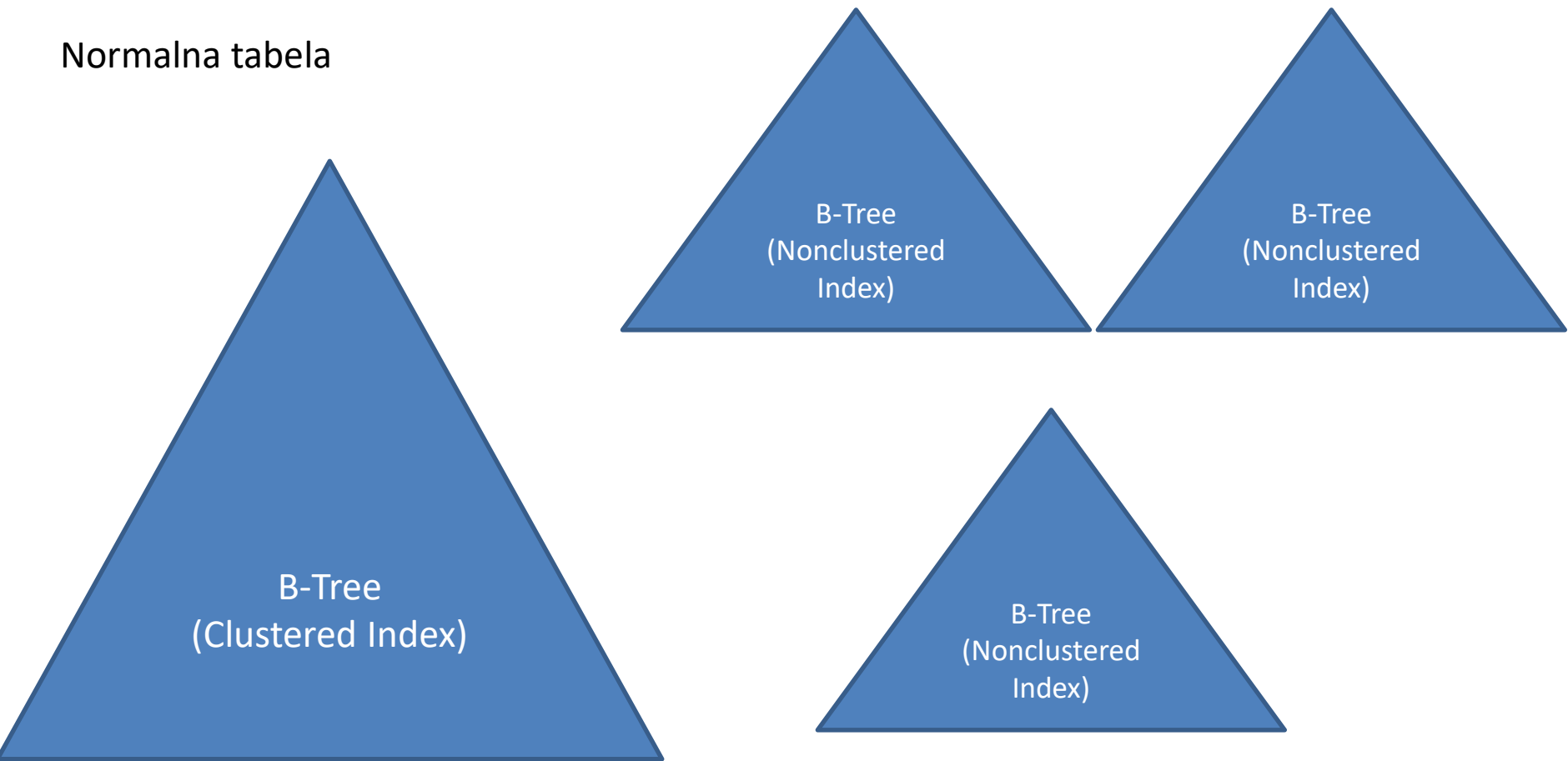
# Ograniczenia

- „Persisted SKU Feature”
- Ograniczenia ilości partycji:
  - SQL 2008 i wcześniejsze **Max 1000**
  - SQL Server 2008 SP2 i nowsze **Max 15000**
- **Jedna kolumna!**
- Typ danych dowolny, za wyjątkiem:
  - Timestamp
  - Image, text, ntext
  - Varchar(max), varbinary(max)

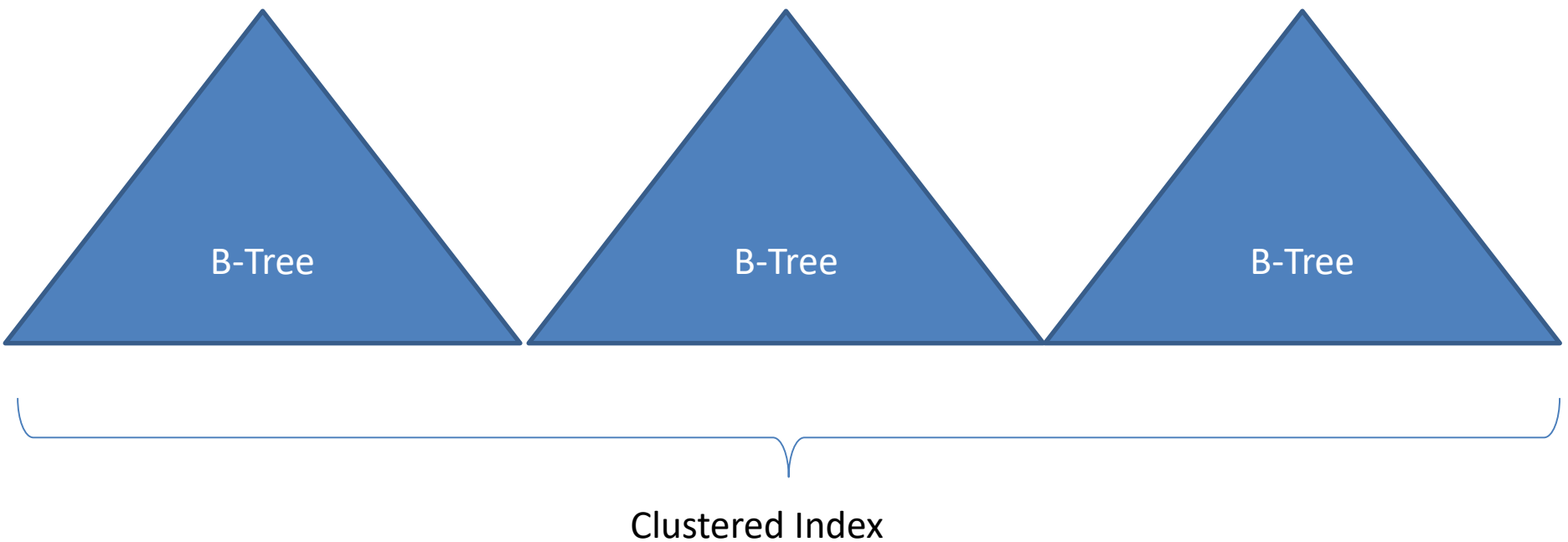


# Horizontal Partitioning

Normalna tabela

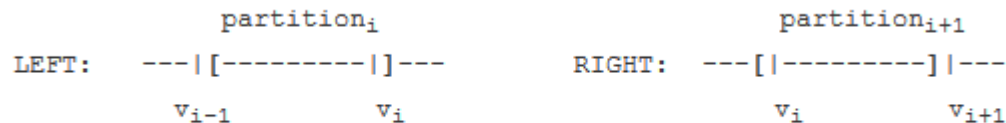


# Horizontal Partitioning

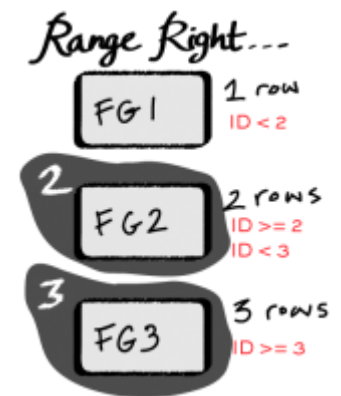


# Partition Function

- Wyznacza wartości graniczne partycji
- Generuje N+1 partycji
- Może być „lewa” albo „prawa”



```
CREATE PARTITION FUNCTION Partition_Function (INT)
AS
RANGE RIGHT FOR VALUES (2,3)
```



# Partition Scheme

- Mapuje partycje na fizyczny storage
- Tworzona dla konkretnej funkcji
- Partition -> Filegroup

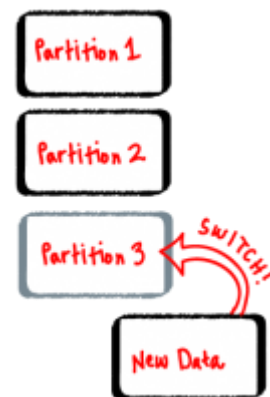
```
CREATE PARTITION SCHEME Partition_Scheme AS PARTITION Partition_Function  
TO ([FG1], [FG2], [FG3])
```



# Switch

Operacja dzięki której w efektywny sposób możemy transferować dane, robiąc to następująco:

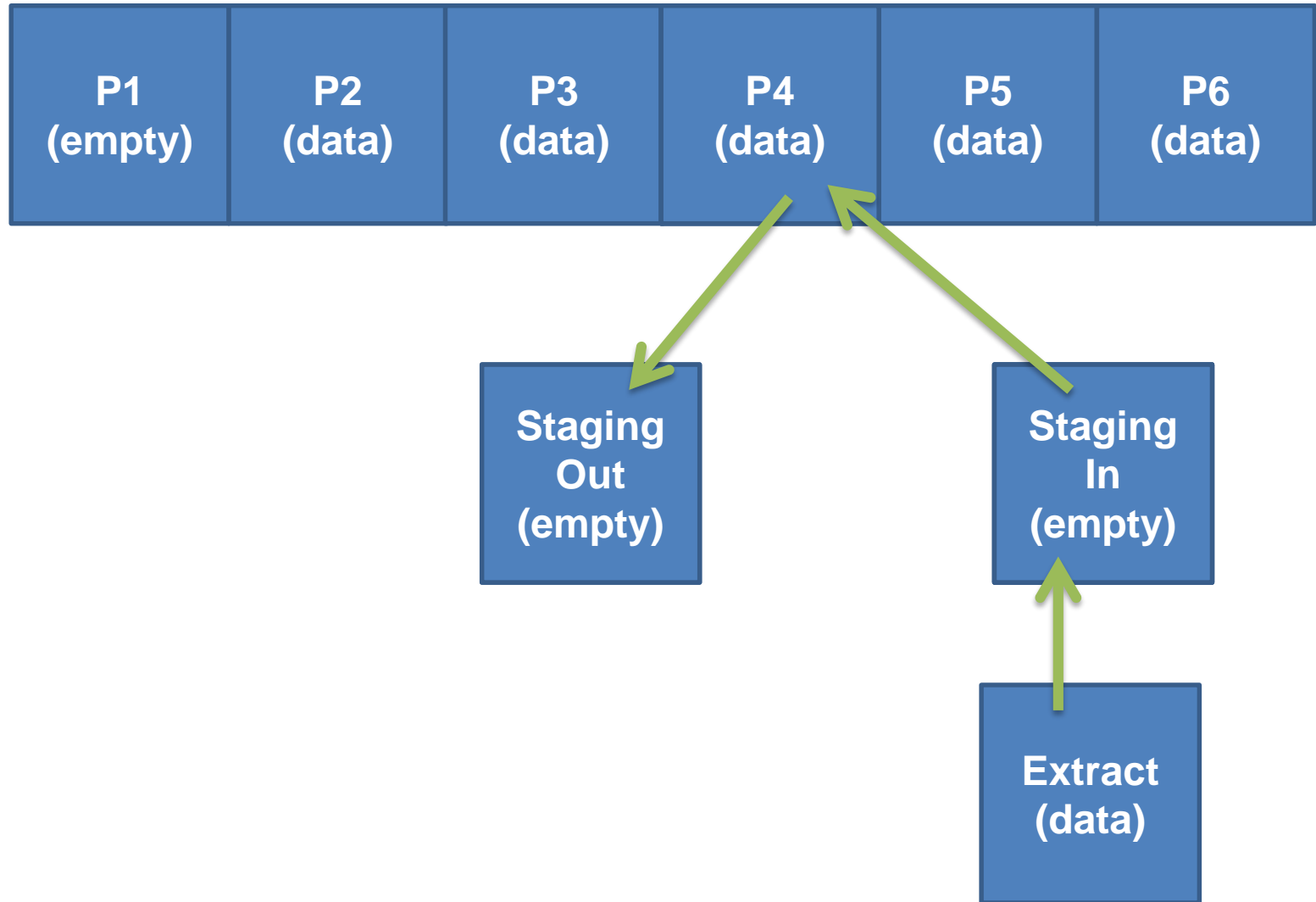
- Tworzymy tabelę o identycznej strukturze
- Przepisujemy dane z wybranej partycji do nowej tabeli



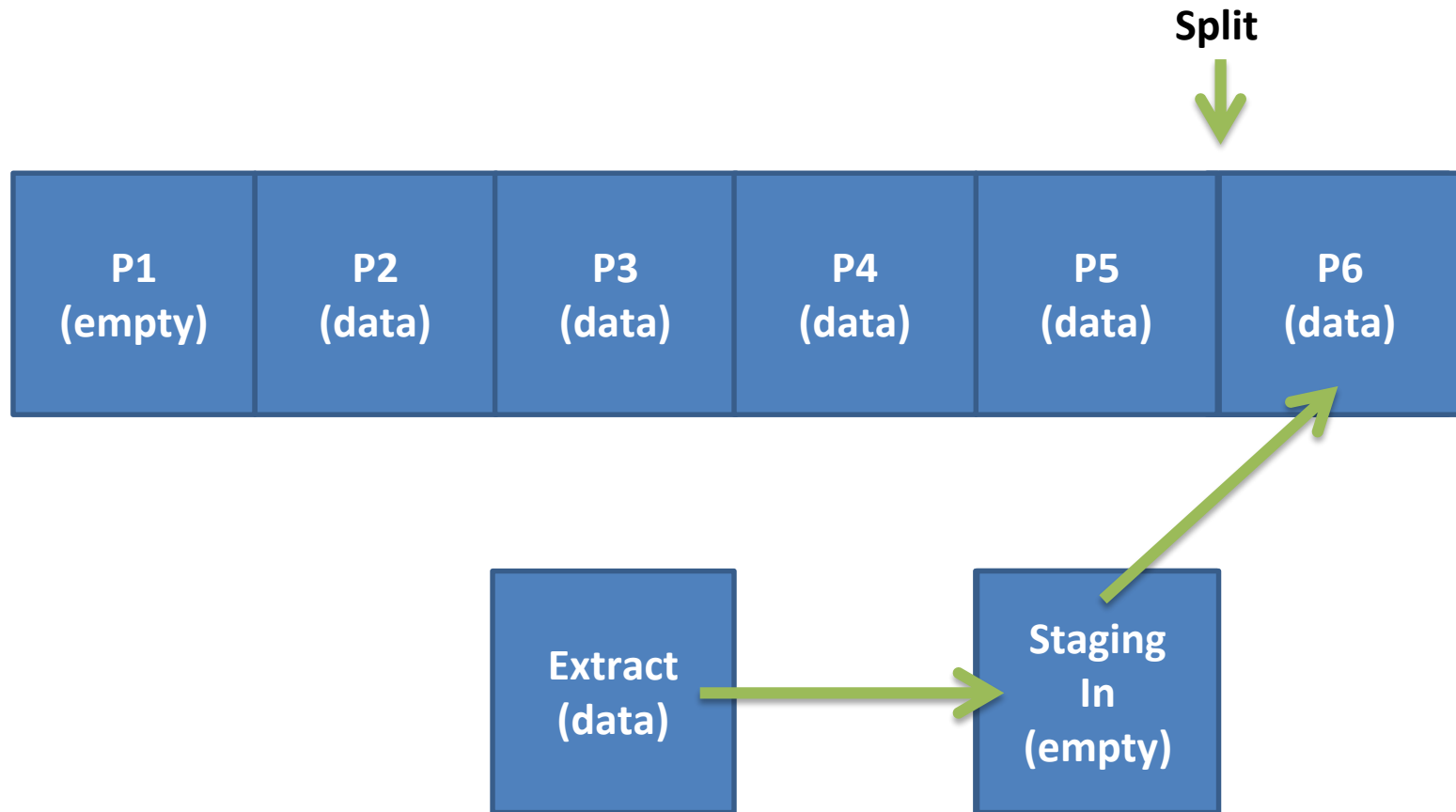
Warunki by dokonać switch partycji:

- Obie tabele muszą istnieć przed wykonaniem operacji przełączenia
- Partycja, do której przełączamy musi istnieć i musi być pusta
- Partycje muszą być wykonane po tej samej kolumnie
- Źródła i miejsce docelowe tabel muszą być w tej samej (File Group) grupie plików

# Switch

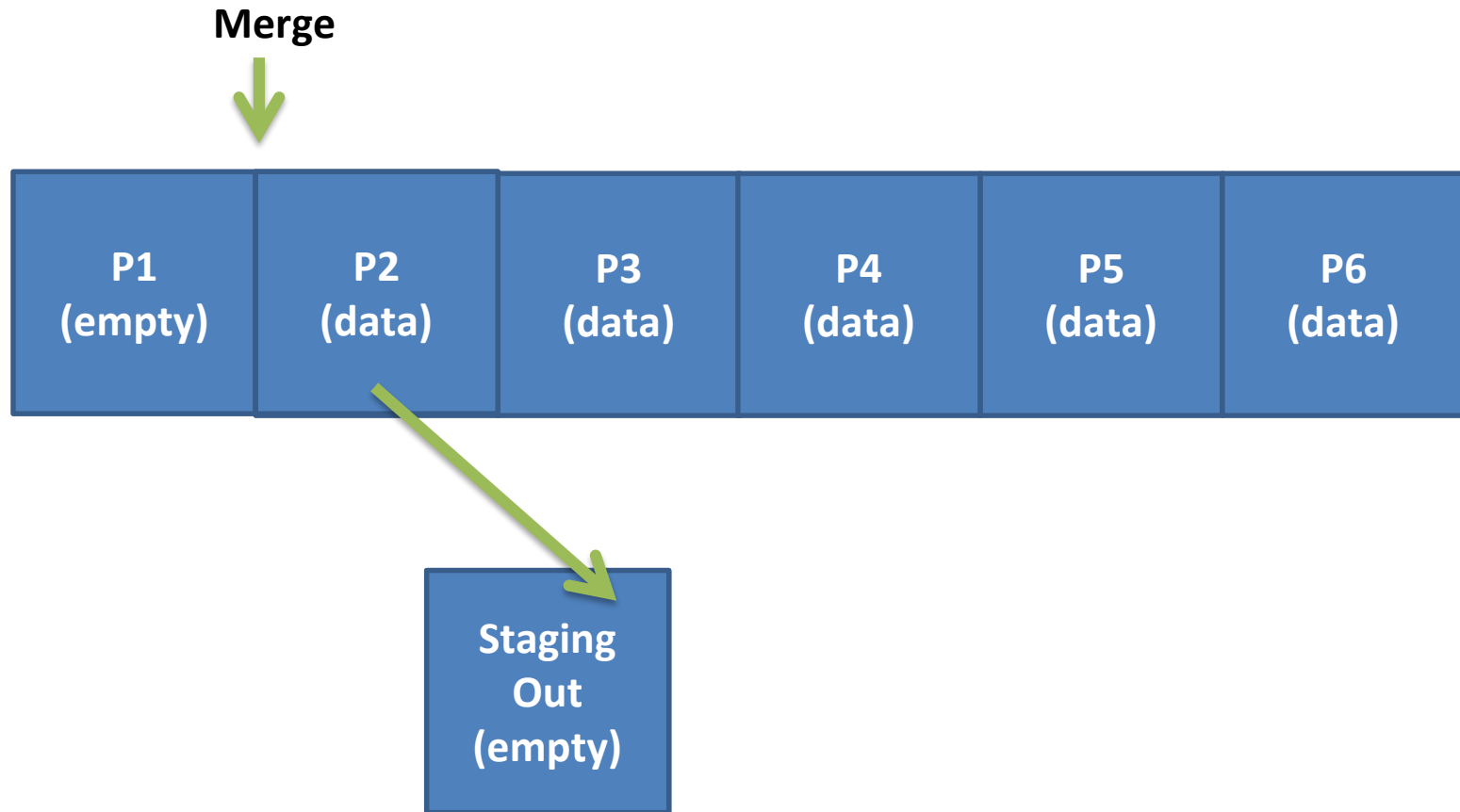


# Split



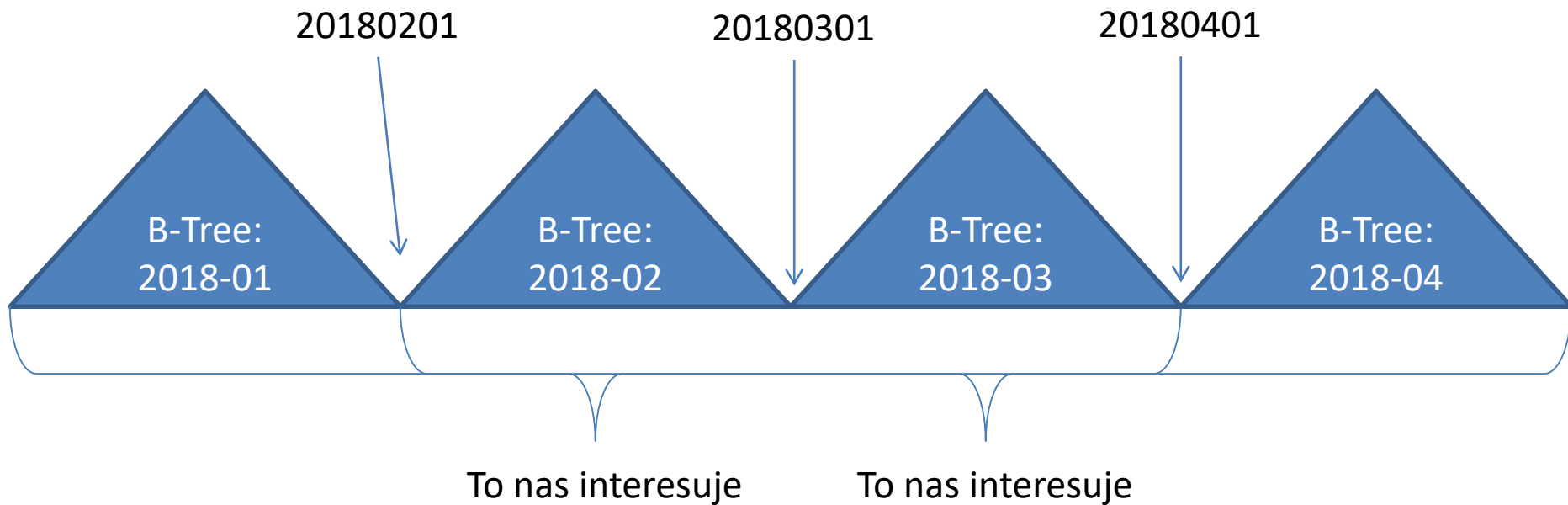


# Merge



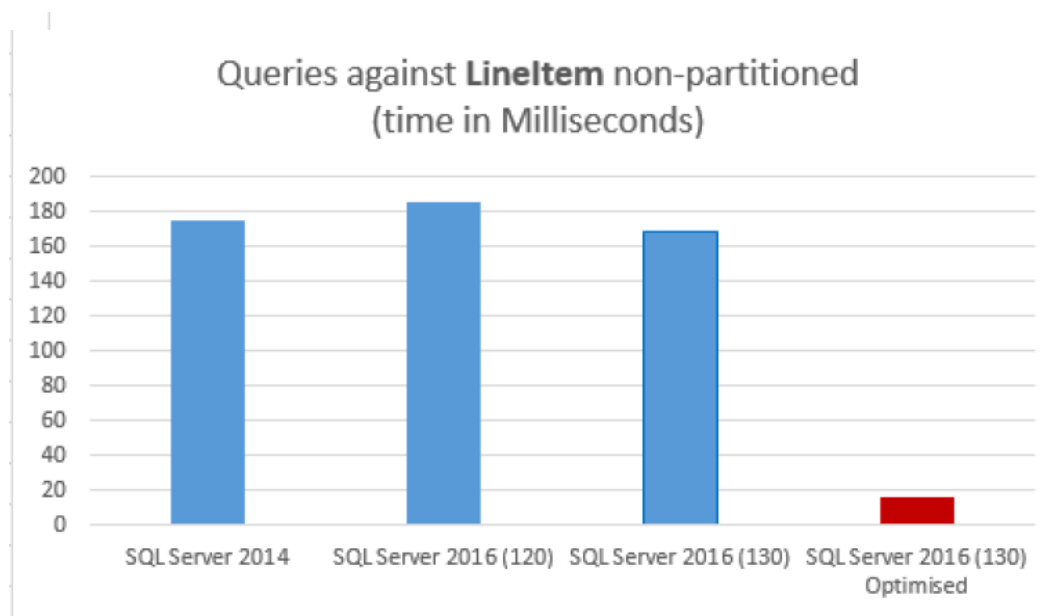
# Sliding window scenario

Jeden z popularniejszych scenariuszy wdrażania partycjonowania, w którym potrzebujemy dane tylko z ostatnich X miesięcy wstecz. W tym celu dodajemy partycję z jednej strony, "usuwamy" partycję z drugiej strony.





# Partycjonowanie 2016 vs Partycjonowanie 2014



Według Sunila Agarwal (ex Program Manager responsible for the SQL Server 2016):  
W SQL Server 2016 zostało usunięte przejście do wcześniej wyeliminowanych partycji.

# Podsumowanie

## Plusy:

- Transparentność tabeli dla użytkownika
- Niski narzut (locks, speed) na przesuwanie dużych ilości danych do osobnej tabeli, do obcinania danych
- Maintenance
  - Data loads
  - Usuwanie/archiwizacja danych
  - Index rebuilds etc
- Zwiększenie wydajności zapytań
- Zrównoleglenie zapytań
- SQL Server automatycznie zarządza rozmieszczeniem danych w odpowiednich partycjach
- Łatwiejsze zarządzanie i administracja dużymi tabelami

## Minusy:

- Nie możemy usunąć ani zmieniać typu danych kolumny partycjonującej
- Dla replikacji tabeli nie możemy korzystać z opcji alter ani switch
- Możemy potrzebować większą liczbę indeksów gdy dane są tylko do odczytu i mamy różne statystyki dla tych danych

# PYTANIA



# Materialy

- <https://tinyurl.com/ybzkdyr9>
- <https://www.brentozar.com/archive/2013/01/best-practices-table-partitioning-merging-boundary-points/>
- <https://blogs.msdn.microsoft.com/hanspo/2009/08/21/inside-of-table-and-index-partitioning-in-microsoft-sql-server/>
- <https://www.brentozar.com/sql/table-partitioning-resources/>
- <https://davidpeterhansen.com/2012/09/29/view-partitions-in-sql-server/>
- <https://improve.dk/deciphering-a-sql-server-data-page/>



## GOLD SPONSORS



Volvo Group IT

## SILVER SPONSORS



## BRONZE SPONSOR



## STRATEGIC PARTNER

