# Pliki danych: konfiguracja, przenoszenie, zmiana nazwy

**Poziom: operacyjny** **Dotyczy: Oracle Database 12c–19c** **Czas: ~20 min**

*Datafile trzeba czasem przenieść na inny dysk albo poprawić mu nazwę po migracji — te fiszki pokazują, jak sprawdzić jego bieżącą konfigurację i jak bezpiecznie zmienić mu adres, offline i online.*

**Skróty użyte na tej stronie (2)**

| Skrót | Co znaczy |
|---|---|
| **SQL** | Structured Query Language: język zapytań do relacyjnych baz danych |
| **WMS** | Warehouse Management System: system zarządzania magazynem |

# Fiszka

**ZMIENIA SYSTEM**

## Sprawdzenie konfiguracji wszystkich datafile: rozmiar, autoextend, MAXSIZE

```
SELECT tablespace_name, file_name, bytes/1048576 file_size_mb,
       autoextensible,
       increment_by*8192/1048576 AS increment_by_in_mb,
       TRUNC((CASE WHEN bytes > maxbytes THEN bytes ELSE maxbytes END)/(1024*1024))
maxsize_of_datafile_in_mb
FROM dba_data_files
ORDER BY file_id;
```

Wyciągamy pełny obraz wszystkich plików danych w bazie naraz — rozmiar bieżący, czy autoextend jest włączony, o ile rośnie za jednym razem i jaki jest jego realny sufit. Kolumna

`maxsize_of_datafile_in_mb` bierze większą z dwóch wartości, bo `MAXSIZE` bywa ustawiony niżej niż faktyczny bieżący rozmiar pliku, jeśli plik urósł ręcznie po ustawieniu limitu.

Przykład:

```
TABLESPACE_NAME  FILE_NAME                                          FILE_SIZE_MB AUT
INCREMENT_BY_IN_MB MAXSIZE_OF_DATAFILE_IN_MB
---------------- ------------------------------------------------ ------------ ---
------------------ --------------------------
SYSTEM           /oracle/db/wmsprd/oradata/system01.dbf                1600
NO                    0                         1600
UNDOTBS1         /oracle/db/wmsprd/oradata/undotbs01.dbf               6600
YES                 100                         8192
INTG             /oracle/db/wmsprd/oradata/intg07.dbf                 32000
YES                 100                        32000
```

**ZMIENIA SYSTEM**

## Sprawdzenie konfiguracji tempfile

```
SELECT tablespace_name, file_name, bytes/1048576 file_size_mb,
       autoextensible,
       increment_by*8192/1048576 AS increment_by_in_mb,
       TRUNC((CASE WHEN bytes > maxbytes THEN bytes ELSE maxbytes END)/(1024*1024))
maxsize_of_datafile_in_mb
FROM dba_temp_files
ORDER BY file_id;
```

To samo zapytanie, tylko po widoku `dba_temp_files` zamiast `dba_data_files` — sprawdzamy rozmiar i autoextend plików tymczasowych, na przykład zanim zdecydujemy o przebudowie tablespace TEMP opisanej na stronie UNDO i TEMP.

Przykład:

```
TABLESPACE_NAME  FILE_NAME                                          FILE_SIZE_MB AUT
INCREMENT_BY_IN_MB MAXSIZE_OF_DATAFILE_IN_MB
---------------- ------------------------------------------------ ------------ ---
------------------ --------------------------
TEMP             /oracle/db/wmsprd/oradata/temp01.dbf                  3000
```

```
YES                 500                         8192
```

ZMIENIA SYSTEM

## Przeniesienie datafile offline: procedura klasyczna

Zanim zaczniemy: ta metoda działa na każdej wersji Oracle i nie wymaga zamykania całej instancji, ale wymaga zdjęcia offline tablespace, do którego przenoszony plik należy — przez czas przenoszenia dane w tym tablespace są niedostępne dla aplikacji. Dysk docelowy musi mieć wolne miejsce co najmniej na rozmiar przenoszonego pliku.

Zdejmujemy tablespace offline:

```
ALTER TABLESPACE NazwaTablespace OFFLINE;
```

Kopiujemy plik w systemie operacyjnym na nową lokalizację — z poziomu powłoki systemowej albo bezpośrednio z SQL*Plus przez `HOST`:

```
HOST cp SciezkaDoPliku SciezkaDoNowegoPliku
```

Informujemy bazę o nowej lokalizacji pliku:

```
ALTER DATABASE RENAME FILE 'SciezkaDoPliku' TO 'SciezkaDoNowegoPliku';
```

Wracamy z tablespace online:

```
ALTER TABLESPACE NazwaTablespace ONLINE;
```

Weryfikujemy, że baza widzi plik pod nową ścieżką:

```
SELECT file_name FROM dba_data_files WHERE tablespace_name = 'NazwaTablespace';
```

Przykład:

```
Tablespace altered.


Database altered.


Tablespace altered.


FILE_NAME
```

```
----------------------------------------------------
/oracle/db/wmsprd/oradata_new/intg07.dbf
```

Dopiero po potwierdzeniu, że baza otworzyła plik pod nowym adresem i tablespace wrócił online bez błędów, usuwamy starą kopię pliku z dysku.

**SQL** = Structured Query Language: język zapytań do relacyjnych baz danych

**ZMIENIA SYSTEM**

### Przeniesienie datafile online od wersji 12c: ALTER DATABASE MOVE DATAFILE

```
ALTER DATABASE MOVE DATAFILE 'SciezkaDoPliku' TO 'SciezkaDoNowegoPliku';
```

Od wersji 12c przenosimy plik bez zdejmowania tablespace offline i bez przerwy w dostępności danych — Oracle kopiuje plik w tle, przełącza się na nową lokalizację i, w zależności od wariantu polecenia, sam usuwa starą kopię. To krótsza i bezpieczniejsza droga niż procedura klasyczna, o ile baza jest już na 12c lub wyżej.

Przykład:

```
ALTER DATABASE MOVE DATAFILE '/oracle/db/19300/dbs/intg07' TO
'/oracle/db/wmsprd/oradata/intg07.dbf';
```

```
Database altered.
```

Weryfikujemy nową ścieżkę tym samym zapytaniem po `dba_data_files` co w klasycznej procedurze.

# Z życia wzięte

**Z ŻYCIA WZIĘTE**

Po migracji dysków pod ora01 na nowy wolumen ścieżki kilku plików danych WMSPRD wciąż wskazują na stary punkt montowania. Tomek sprawdza `dba_data_files` i zauważa, że jeden z plików `intg07.dbf` nie doszedł do nowej lokalizacji razem z resztą. Poleceniem `ALTER DATABASE MOVE DATAFILE` przenosi go bez zatrzymywania instancji, żeby WMS w Swarzędzu nie zauważył przerwy.

# Czym jest datafile i tempfile

> **Pojęcie: datafile.** Fizyczny plik na dysku, w którym leżą dane jednego tablespace — sam podział na tablespace i to, jak ma się on do datafile i segmentu, opisuje strona Tablespace: sprawdzenie, powiększenie, utworzenie, usunięcie. Każdy datafile ma swój numer (`FILE_ID`), ścieżkę, rozmiar i, opcjonalnie, ustawienia autoextend.

> **Pojęcie: tempfile.** Plik fizyczny należący do tablespace typu TEMP. Różni się od zwykłego datafile tym, że nie trzyma trwałych danych — leżą w nim tylko tymczasowe struktury sortowania i operacji, które się w nim nie mieszczą w pamięci. Dlatego tempfile nie jest częścią backupu bazy w takim samym sensie jak datafile — w razie utraty odtwarza się go samym poleceniem tworzącym nowy plik, bez przywracania z kopii zapasowej.

Zmiana nazwy albo lokalizacji datafile jest operacją, po którą sięgamy zwykle w dwóch sytuacjach: przenosimy dane na inny wolumen z powodu miejsca na dysku, albo porządkujemy nazewnictwo po migracji bazy na inny serwer, gdzie pliki wylądowały pod inną ścieżką niż docelowa.

# Częste błędy

**Zapomnienie o zdjęciu tablespace offline przy metodzie klasycznej.** `ALTER DATABASE RENAME FILE` bez wcześniejszego `ALTER TABLESPACE ... OFFLINE` kończy się błędem — baza nie pozwoli podmienić ścieżki pliku, który jest aktualnie otwarty i używany.

**Usunięcie starego pliku z dysku przed weryfikacją.** Dopóki nie potwierdzimy zapytaniem po `dba_data_files`, że baza faktycznie otworzyła plik pod nową ścieżką i tablespace wrócił online bez błędu, stara kopia zostaje na dysku — to jedyna droga odwrotu, gdyby coś poszło nie tak.

**Próba `MOVE DATAFILE` na wersji starszej niż 12c.** Polecenie po prostu nie istnieje w starszych wersjach — jedyną drogą jest tam procedura klasyczna z offline tablespace.

**Powiązane:** Tablespace: sprawdzenie, powiększenie, utworzenie, usunięcie · UNDO i TEMP · Migracja bazy danych na inny serwer

**Źródła:** notatki własne, dokumentacja Oracle Database (Administrator's Guide — Managing Data Files)

---

Revision #43
Created 2026-09-24 23:34:26 UTC by Tomasz Konieczny
Updated 2026-10-26 14:08:57 UTC by Tomasz Konieczny